Univariate Distributional Analysis with L-moment Statistics using R

by

William H. Asquith, B.S., M.S., Ph.D., P.G.

A Dissertation

In

Civil Engineering

Department of Civil and Environmental Engineering

Submitted to the Graduate Faculty
of Texas Tech University in
Partial Fulfillment of
the Requirements for
the Degree of

Doctor of Philosophy

Approved

Theodore G. Cleveland, Ph.D., P.E.
Chair of Committee

Stacey A. Archfield, Ph.D.

Ken Rainwater, Ph.D., P.E., BCEE, D.WRE

David B. Thompson, Ph.D., P.E., D.WRE, CFM

Peggy Gordon Miller
Dean of the Graduate School

May, 2011

Acknowledgments

I am especially grateful to my second Ph.D. (civil engineering) advisor, Theodore Cleveland, for years—that is, many years—of collaboration and friendship. I am especially grateful to George R. Herrmann, Meghan Roussel, David Thompson, and Ken Rainwater. These individuals and friends collectively encouraged and guided my post-2003 Ph.D. (geosciences) research efforts into L-moment applications and also were early users and collective beneficiaries of an originally little L-moment package in R. I also am grateful to Stacey Archfield for being an early user of *lmomco* and for lively discussions about the Kappa distribution, streamflow-duration curves, and all things as a "linear moment."

I am grateful for my U.S. Geological Survey colleagues or supervisors (former, retired, or current) for career opportunities and many adventures: Jim Bartolino, Peter Bush, David Dunn, Franklin Heitmuller, Sonya Jones, Robert Joseph, George Ozuna, Meghan Roussel, Raymond Slade, and Lloyd Woosley. I am especially grateful to Thomas Burley for performing a comprehensive review and becoming an R aficionado in the process.

I am grateful to Cal Barnes and Melanie Barnes for many years of support and for providing a departmental home at Texas Tech University.

I am grateful to John Sharp, my 2003 Ph.D. (geosciences) advisor and his long support of my statistical interests. To this day asks in jest: "What exactly is an L-moment?" My reply, absent the math, is something like, "An average of intra-differences of ordered samples." (He then chuckles . . .)

I am especially grateful to Jonathan Hosking for his considerable breadth of theoretical and applied research that led to the founding of L-moments, he is justifiably labeled as the "father of L-moments" to quote Robert Serfling at the JSM 2008 conference in Denver, Colorado. I must acknowledge Jonathan's generosity in the development of the FORTRAN library and many years of correspondence. The library and many of his publications are

the canonical references on which much of the central or core code of the *lmomco* package is based. I am grateful for early encouragement with R from Juha Karvanen. I am grateful to the community of other researchers of L-moment and related topics whose work I hope is adequately cited throughout this dissertation. Thanks.

I must give a sweeping thanks to the global open-source software community. A myriad of powerful tools for research and communication, of which R is an obvious example, are freely available for multiple computer platforms. Other open-source tools or organizations, which are indispensable to me and my associates, deserving of special recognition are the GNU project, TEX, LATEX, Perl, Linux, the MacOSX development communities, and the contributers and users that support these efforts.

Finally, I am grateful to the loss of words for the unending support of my wife D'Anne, our children Nathaniel and Caroline, and my parents George, Ann, Tom, and Linda. I am proud and relieved to bring years of nights and weekends of reading, coding, writing, reading, coding, writing, and editing to completion.

Table of Contents

Acknowledgments ii

List of Tables xii

List of Figures xxi

Preface xxii

1 Introduction 1

1.1 Distributional Analysis 1

1.2 The R Environment for Statistical Computing 3

1.2.1 Internet Resources for R 7

1.2.2 Traditional Publishers for R 7

1.3 L-moments—A general description 8

1.3.1 L-moments in R 10

1.3.2 Internet Resources for L-moments 11

1.4 Purpose and Organization 11

1.5 Reader Expectations and Advice to the Reader 14

1.6 Types of Data 15

1.7 Visualization of Sample Distributions—Histograms and Box Plots 18

1.7.1 Histograms 18

1.7.2 Box Plots 19

2 Distributional Analysis 23

2.1 A Review of Continuous Random Variables and Distributions 23

2.1.1 Probability Density Functions 24

2.1.2 Cumulative Distribution Functions 25

2.1.3 Hazard Functions 28

2.1.4 Quantile Functions 29

2.1.5 The Algebra of Quantile Functions 33

2.1.6 Exceedance Probability and Recurrence Interval 42

2.2 Basic Summary Statistics and Distributional Analysis 44

2.2.1 Basic Summary Statistics 44

2.2.2 Fitting a Distribution by the Method of Moments 47

2.2.3 Construction of an Empirical Distribution by Plotting Positions 50

2.2.4 Two Demonstrations of Basic Distributional Analysis 55

2.3 Summary 59

3 Order Statistics 61

3.1 Introduction 61

3.1.1 Expectations and Distributions of Order Statistics 63

3.1.2 Distributions of Order Statistic Extrema 66

3.2 L-estimators—Special Statistics Related to L-moments 70

3.2.1 Sen Weighted Mean 71

3.2.2 Gini Mean Difference 73

3.3 Summary 77

4 Product Moments 78

4.1 Introduction 78

4.1.1 Sampling Bias and Sampling Variance 79

4.2 Product Moments—Definitions and Math 83

4.2.1 Theoretical Product Moments 83

4.2.2 Sample Product Moments 84

4.3 Some Sampling Properties of Product Moments 87
4.3.1 The Mean, Standard Deviation, and Coefficient of Variation 87
4.3.2 Bias of Standard Deviation 88
4.3.3 Bias and Boundedness of Coefficient of Variation 90
4.3.4 Bias and Boundedness of Skew 93
4.4 On the Use of Logarithmic Transformation 97
4.5 Summary 98

5 Probability-Weighted Moments 99
5.1 Introduction 99
5.2 Probability-Weighted Moments—Definitions and Math 100
5.2.1 Theoretical Probability-Weighted Moments 101
5.2.2 Sample Probability-Weighted Moments 103
5.3 The Method of Probability-Weighted Moments 109
5.4 Summary 111

6 L-moments 112
6.1 Introduction 112
6.2 L-moments—Definitions and Math 117
6.2.1 Theoretical L-moments 117
6.2.2 Sample L-moments 126
6.2.3 Visualization of L-moment Weight Factors 130
6.2.4 Reference Frame Comparison Between L-moments and Product Moments 133
6.3 The Method of L-moments 135
6.4 TL-moments—Definitions and Math 139
6.4.1 Theoretical TL-moments 139
6.4.2 Sample TL-moments 143
6.5 Some Sampling Properties of L-moments 145
6.5.1 Estimation of Distribution Dispersion 145

6.5.2 Estimation of Distribution Skewness (Symmetry) . . . 145
6.5.3 Estimation of Distribution Kurtosis (Peakedness) . . . 146
6.5.4 Boundedness of Coefficient of Variation Revisited . . . 148
6.5.5 Consistency and the Use of Logarithmic Transformation . . . 150
6.6 Multivariate L-moments . . . 156
6.7 Summary . . . 157

7 L-moments of Univariate Distributions . . . 158
7.1 Introduction . . . 158
7.1.1 Chapter Organization . . . 159
7.1.2 Distributions of the *lmomco* Package . . . 162
7.2 One- and Two-Parameter Distributions of the *lmomco* Package . . . 171
7.2.1 Normal Distribution . . . 173
7.2.2 Exponential Distribution . . . 175
7.2.3 Gamma Distribution . . . 179
7.2.4 Cauchy Distribution . . . 183
7.2.5 Gumbel Distribution . . . 186
7.2.6 Reverse Gumbel Distribution . . . 190
7.2.7 Kumaraswamy Distribution . . . 196
7.2.8 Rayleigh Distribution . . . 198
7.2.9 Rice Distribution . . . 204
7.3 Summary . . . 214

8 L-moments of Three-Parameter Univariate Distributions . . . 216
8.1 Introduction . . . 216
8.2 Three-Parameter Distributions of the *lmomco* Package . . . 217
8.2.1 Generalized Extreme Value Distribution . . . 217
8.2.2 Generalized Logistic Distribution . . . 221
8.2.3 Generalized Normal Distribution . . . 227
8.2.4 Generalized Pareto Distribution . . . 235
8.2.5 Right-Censored Generalized Pareto Distribution . . . 239

8.2.6 Trimmed Generalized Pareto Distribution 240
8.2.7 Pearson Type III Distribution 242
8.2.8 Weibull Distribution 247
8.3 Three-Parameter Distributions not yet supported by the *lmomco* Package . . 251
8.3.1 Polynomial Density-Quantile3 Distribution 252
8.3.2 Polynomial Density-Quantile4 Distribution 253
8.3.3 Student t (3-parameter) Distribution 254
8.4 Summary 256

9 L-moments of Four- and More Parameter Univariate Distributions 259
9.1 Introduction 259
9.2 Four- and More Parameter Distributions of the *lmomco* Package 260
9.2.1 Kappa Distribution 260
9.2.2 Generalized Lambda Distribution 267
9.2.3 Trimmed Generalized Lambda Distribution 275
9.2.4 Wakeby Distribution 280
9.3 Special Demonstration of Distributions 291
9.4 Summary 295

10 L-moment Ratio Diagrams 298
10.1 Introduction 298
10.2 Assessment of Distribution Form using L-moment Ratio Diagrams 299
10.3 Construction of L-moment Ratio Diagrams 305
10.4 Summary 313

11 Short Studies of Statistical Regionalization 314
11.1 Analysis of Annual Maxima for 7-Day Rainfall for North-Central Texas Panhandle 314
11.1.1 Background and Data 315
11.1.2 Distributional Analysis 315
11.2 Peak-Streamflow Distributions for Selected River Basins in the United States 324
11.2.1 Background and Data 324
11.2.2 Distributional Analysis 324
11.3 Summary 336

12 Advanced Topics 337

12.1 Introduction 337

12.2 L-moments from Probability-Weighted Moments for Right-Tail Censored Distributions 338

12.2.1 Theoretical Probability-Weighted Moments for Right-Tail Censored Distributions 339

12.2.2 Sample Probability-Weighted Moments for Right-Tail Censored Data 341

12.3 L-moments from Probability-Weighted Moments for Left-Tail Censored Distributions 343

12.3.1 Theoretical Probability-Weighted Moments for Left-Tail Censored Distributions 344

12.3.2 Sample Probability-Weighted Moments for Left-Tail Censored Data 345

12.4 L-moments of Right-Tail Censored Data by Indicator Variable 347

12.5 L-moments of Left-Tail Censored Data by Indicator Variable 352

12.6 Conditional Adjustment for Zero Values by Blipped-Distribution Modeling 358

12.7 Exploration of Quantile Uncertainty 361

12.7.1 Exploration of Sampling Error for a Single Data Set 362

12.7.2 Exploration of Sampling Error for a Regional Data Set 368

12.7.3 Exploration of Model-Selection Error 369

12.8 Product Moments versus L-moments for Estimation of Pearson Type III Distributions 374

12.8.1 Logarithmic Transformation 375

12.8.2 Simulation Study of Pearson Type III Parameter Estimation 376

12.8.3 Further Evaluation of Pearson Type III Parameter Estimation 378

12.8.4 Thought Experiment—To Product Moment or L-moment and To Transform Data? 382

12.8.5 Some Conclusions 385

12.9 L-comoments—Multivariate Extensions of L-moments 386

12.10 Summary 398

Epilogue 399

Index 417

Index of R Functions 429

List of Tables

1.1 Summary of L-moment related R packages available on the CRAN in order of initial release 11

2.1 Selected plotting-position coefficients for eq. (2.18) 52

5.1 Summary of probability-weighted moment related functions of the *lmomco* package by Asquith (2011) 101

6.1 Summary of L-moment computation and support functions of the *lmomco* package by Asquith (2011) 115

6.2 Summary of L-moment computation functions for probability distributions of the *lmomco* package by Asquith (2011) 116

6.3 Summary of L-moment computation functions of the *Lmoments* package by Karvanen (2009) 116

6.4 Summary of L-moment computation functions for samples and by probability distribution of the *lmom* package by Hosking (2009a) 119

7.1 Summary of distribution functions provided by the *lmom* package by Hosking (2009a) 160

7.2 Summary of L-moment and parameter functions by distribution provided by the *lmom* package by Hosking (2009a) 160

7.3 Summary of distribution functions provided by the *lmomco* package by Asquith (2011) 165

7.4 Summary of L-moment and parameter functions by distribution provided by the *lmomco* package by Asquith (2011) 167

7.5 Summary of convenience functions by distribution provided by the *lmomco* package by Asquith (2011) 168

7.6 Summary of high-level conversion functions provided by the *lmomco* package by Asquith (2011) 170

7.7 Summary high-level distribution functions of *lmomco* package by Asquith (2011) that mimic the nomenclature of R 171

7.8 L-moments of storm depth for storms defined by a minimum interevent time of 72 hours in Texas derived from Asquith and others (2006, table 5) . . 181

7.9 Comparison of computed L-moments for four Gumbel distribution parameter lists from example 7–26 194

8.1 L-moments of wind speed data reported by Hosking and Wallis (1997, table 2.5) 218

8.2 Parameters and corresponding L-moments of Generalized Logistic distribution for 1-hour annual maximum rainfall for Travis County, Texas derived from Asquith (1998) 224

8.3 L-moments of annual peak streamflow data for 05405000 Baraboo River near Baraboo, Wisconsin and parameters for fitted Generalized Normal distribution 233

10.1 Coefficients for polynomial approximations of L-kurtosis as a function of L-skew for selected distributions 307

11.1 Summary of selected U.S. Geological Survey streamflow-gaging stations for distributional analysis using L-moments 325

12.1 L-moments of annual peak streamflows for Llano River at Llano, Texas (1940–2006) and Wakeby distribution parameters 362

12.2 Regional L-moments and equivalent Wakeby parameters for dimensionless distribution of annual peak streamflow in Texas 368

List of Figures

1.1 Histograms of ozone and temperature for data in the *airquality* data frame from example 1–5 20

1.2 Box plots of ozone, solar radiation, temperature, and wind speed data for data in the *airquality* data frame from example 1–6 21

2.1 Probability density function for a Weibull distribution from example 2–1 . . 26

2.2 Cumulative distribution function for standard Normal distribution from example 2–2 28

2.3 Quantile function for an Exponential distribution from example 2–5 31

2.4 Blending two quantile functions to form a third by the Intermediate Rule from example 2–10 36

2.5 Reflection (bottom) of an Exponential distribution (top) about $x = 0$ and $F = 0.5$ using the Reflection Rule from example 2–11 37

2.6 F-transformation function from example 2–15 40

2.7 Comparison of original Generalized Pareto distribution (solid line) and F-transformed Generalized Pareto distribution (dashed line) from example 2–16 41

2.8 Comparison of a parent Normal distribution (thin line) and sample Normal distribution (thick line) for a sample of size 20 from example 2–26 50

2.9 Empirical distribution by plotting position of porosity (fraction of void space) from neutron-density, well log for 5,350–5,400 feet below land surface for Permian Age Clear Fork formation, Ector County, Texas from example 2–29 54

2.10 Empirical distribution of simulated data from specified Weibull distribution and Weibull distribution fit to L-moments of the simulated data from example 2–31 56

2.11 Empirical distribution of annual peak streamflow for U.S. Geological Survey streamflow-gaging station 05405000 Baraboo River near Baraboo, Wisconsin and Normal (solid line) and log-Normal (dashed line) distributions fit by method of moments from example 2–32 59

2.12 Box plot of the distribution of annual peak streamflow for U.S. Geological Survey streamflow-gaging station 05405000 Baraboo River near Baraboo, Wisconsin from example 2–34 60

4.1 Demonstration of upper limit boundedness (dashed line) and bias of $\hat{CV}$ (thick solid and curved line) as computed by 500 simulations for each sample size for a Gamma distribution having $\mu = 3{,}000$ and $CV = 10$ (solid horizontal line) from example 4–8 93

4.2 Parent Pearson Type III distribution of PE3(1000, 500, 5) used to assess bias in product moment skew from example 4–9 95

4.3 Demonstration of upper limit boundedness (dashed line) and bias of $\hat{G}$ (thick solid curved line) as computed by 500 simulations for each sample size for a Pearson Type III distribution of PE3(1000, 500, 5) ($G = 5$ and is the solid horizontal line) from example 4–10 96

6.1 Graphics showing the weight factors of sample L-moment computation for each observation from a $n = 19$ sample on the respective L-moment from example 6–11 132

6.2 Comparison of intra-sample distances (solid circles) and individual trip distance to mean (open circles) for respective L-moment and product moment computation from example 6–13 136

6.3 Bias ratios of product moment kurtosis (dashed line) and L-kurtosis (solid line) as a function of sample size for standard Normal distributed data from example 6–19 147

6.4 Demonstration of boundedness and bias of $\hat{CV}$ and unbiased property of $\hat{\tau}_2$ for a Gamma distribution having $\mu = 3{,}000$ and $CV = 10$ from example 6–22 150

6.5 Empirical distribution of simulated log-Normal data from example 6–23 . . 151

6.6 Relation between $\hat{\tau}_3$ and sample size of simulated log-Normal distribution shown in figure 6.5 from example 6–24 153

6.7 Relation between $\hat{G}$ and sample size of simulated log-Normal distribution shown in figure 6.5 from example 6–24 154

6.8 Relation between $\hat{G}$ of logarithms and sample size of simulated log-Normal distribution shown in figure 6.5 from example 6–24 155

7.1 Example of three distributions, Generalized Extreme Value (thin line), Gumbel (dashed line), and Weibull (thick line) fit to the identical L-moments from example 7–7 172

7.2 Quantile function of the Gamma distribution with $\alpha = 0.633$ and $\beta = 1.46$ from example 7–14 182

7.3 PDF of example Cauchy distribution from example 7–17 185

7.4 Gumbel distribution fit by non-linear optimization to data available from Hershfield (1961) from example 7–21 189

7.5 Gumbel distribution fit by method of percentiles from example 7–22 191

7.6 Comparison five Gumbel-like distributions as sequentially described in example 7–25 195

7.7 Relation between Kumaraswamy distribution parameters and L-skew and L-kurtosis from example 7–29 199

7.8 Comparison two Rayleigh distributions fit as one- or two-parameter versions to L-moments of $\lambda_1 = 0.964$ and $\lambda_2 = 0.581$ for unknown and known ($\xi = 0$) lower bounds from example 7–30 202

7.9 Comparison two Rayleigh distributions (solid line) and Gamma distribution (dashed line) fit to a time to peak (mode) of 5 hours from example 7–31 203

7.10 Example PDF and two computations of CDF of a RICE(20, 40) distribution from example 7–35 209

7.11 Comparison of CDF for signal $\nu = 17$ for a range of signal-to-noise (SNR) ratios for Rice distribution from example 7–37 211

7.12 L-moment ratio diagram showing 500 simulations of $n = 200$ samples for a Rice having $\nu = 5$ and $\alpha = 3$ from example 7–38. The large open circle reprsents the pair-wise means of L-skew and L-kurtosis and large solid circle represents the population values. 213

8.1 Comparison of T-year recurrence interval of individual annual peak streamflow data points estimated by CDF of Generalized Extreme Value distribution and those from Weibull plotting positions for U.S. Geological Survey streamflow-gaging station 08167000 Guadalupe River at Comfort, Texas from example 8–4 221

8.2 Comparison of empirical distribution of annual peak streamflow data (open circles) and fitted Generalized Extreme Value distribution (solid line) for U.S. Geological Survey streamflow-gaging station 08167000 Guadalupe River at Comfort, Texas from example 8–4 222

8.3 CDF and QDF of Generalized Logistic distribution fit to L-moments in table 8.2 from example 8–5 225

8.4 Comparison of QDF for Generalized Extreme Value and Generalized Logistic distributions fit to L-moments of $\lambda_1 = 2000$, $\lambda_2 = 500$, and $\tau_3 = 0$ from example 8–8 227

8.5 Comparison of PDF for Generalized Extreme Value and Generalized Logistic distributions fit to L-moments of $\lambda_1 = 2000$, $\lambda_2 = 500$, and $\tau_3 = 0$ from example 8–9 228

8.6 Probability density functions for three selected Generalized Normal distributions. 232

8.7 Empirical distribution of annual peak streamflow data for U.S. Geological Survey streamflow-gaging station 05405000 Baraboo River near Baraboo, Wisconsin and Generalized Normal (solid line) and log-Normal (dashed line) distributions fit by method of L-moments from example 8–11 233

8.8 Quantile function by 5-percent intervals for a Generalized Normal (dashed line) distribution and several log-Normal3 fits using selected lower limits and fit (red line) treating lower limit as unknown from example 8–15 236

8.9 Time series by day of daily mean streamflow for U.S. Geological Survey streamflow-gaging station 06766000 Platte River at Brady, Nebraska from example 8–19 246

8.10 Flow-duration curve of daily mean streamflow for U.S. Geological Survey streamflow-gaging station 06766000 Platte River at Brady, Nebraska from example 8–20 247

8.11 Comparison of probability density functions for Weibull and Generalized Extreme Value distributions fit to same L-moments of number of Internal Revenue Service refunds by state from example 8–21 250

8.12 Comparison of cumulative probability functions for Weibull (thick line) and Generalized Extreme Value (thin line) distributions fit to same L-moments and empirical distribution of number of Internal Revenue Service refunds by state from example 8–22 251

9.1 Empirical distribution of annual peak streamflow for U.S. Geological Survey streamflow-gaging station 08190000 Nueces River near Laguna, Texas and Kappa distribution fit by the method of L-moments from example 9–1 264

9.2 Empirical distribution of annual peak streamflow for U.S. Geological Survey streamflow-gaging station 08190000 Nueces River near Laguna, Texas and three selected distributions from example 9–2 265

9.3 Empirical distribution of annual peak streamflow for U.S. Geological Survey streamflow-gaging station 08190000 Nueces River near Laguna, Texas and two Generalized Lambda distributions and Kappa fit to sample L-moments from example 9–4 272

9.4 Simulated standard Normal distribution for $n = 500$ and three fitted Generalized Lambda distributions using algorithms of the *GLDEX* and *lmomco* packages from example 9–8 275

9.5 Comparison of PDF of Cauchy and Generalized Lambda distributions fit to 300 random samples of CAU(3000, 40000) by method of TL-moments from example 9–11 280

9.6 Comparison of Wakeby distributions (and Generalized Pareto, if applicable, dashed lines) for λ_1, λ_2, and τ_4 consistent with standard Normal distribution and τ_3 and τ_5 swept through ± 0.7 and ± 0.1, respectively from example 9–13 285

9.7 Comparison of QDF and PDF of Wakeby distribution (or Generalized Pareto, if applicable) for λ_1, λ_2, and τ_4 consistent with standard Normal distribution and $\tau_3 = 0$ and $\tau_5 = 0$ from example 9–15 288

9.8 Comparison of QDF and PDF of Wakeby distribution (or Generalized Pareto, if applicable) for λ_1, λ_2, and τ_4 consistent with standard Normal distribution and $\tau_3 = 0.1$ and $\tau_5 = 0$ from example 9–15 288

9.9 Comparison of QDF and PDF of Wakeby distribution (or Generalized Pareto, if applicable) for λ_1, λ_2, and τ_4 consistent with standard Normal distribution and $\tau_3 = -0.1$ and $\tau_5 = 0$ from example 9–15 289

9.10 Comparison of QDF and PDF of Wakeby distribution (or Generalized Pareto, if applicable) for λ_1, λ_2, and τ_4 consistent with standard Normal distribution and $\tau_3 = 0.1$ and $\tau_5 = 0.1$ from example 9–15 289

9.11 Comparison of QDF and PDF of Wakeby distribution (or Generalized Pareto, if applicable) for λ_1, λ_2, and τ_4 consistent with standard Normal distribution and $\tau_3 = 0.1$ and $\tau_5 = 0.5$ from example 9–15 290

9.12 Comparison of QDF and PDF of Wakeby distribution (or Generalized Pareto, if applicable) for λ_1, λ_2, and τ_4 consistent with standard Normal distribution and $\tau_3 = -0.2$ and $\tau_5 = -0.1$ from example 9–15 290

9.13 Comparison of PDF for twelve distributions fit to L-moments from example 9–17 293

9.14 Alternative depiction of PDF of Kappa distribution shown in figure 9.13 from example 9–18 294

9.15 Comparison of symmetrical PDFs of two Kappa distributions having positive (top) and negative (bottom) τ_4 values of equivalent magnitude from example example 9–19 295

10.1 High-quality L-moment ratio diagram showing L-skew and L-kurtosis of selected distribuions and sample values for storm depth and storm duration from Asquith and others (2006) 301

10.2 L-moment ratio diagram showing 50 sample simulations of L-skew and L-kurtosis for $n = 20$ samples drawn from three distributions from example 10–4 306

10.3 L-moment ratio diagram showing 50 sample simulations of L-skew and L-kurtosis for $n = 200$ samples drawn from three distributions based on examples 10–3 and 10–4 307

10.4 L-moment ratio diagram showing 50 sample simulations of L-skew and L-kurtosis values for $n = 200$ samples drawn from three distributions with superimposed theoretical lines for the Generalized Logistic distribution (thin line) and Pearson Type III distribution (thick line) from examples 10–4 and 10–6 309

10.5 Default L-moment ratio diagram provided by package *lmomco* from example 10–7 310

10.6 More typical L-moment ratio diagram for generally positively skewed phenomena provided by package *lmomco* from example 10–8 311

10.7 L-moment ratio diagram shown distribution of 50 sample simulations of of L-skew and L-kurtosis for $n = 30$ samples drawn from a KAP(10000, 7500, 0.3, 0.2) distribution from example 10–9 312

11.1 Box plots of the distributions of 7-day annual maxima rainfall for seven communities in the north-central Texas Panhandle from example 11–3 317

11.2 Bean and violin plots of the distributions of 7-day annual maxima rainfall for seven communities in the north-central Texas Panhandle from example 11–4 319

11.3 L-moment ratio diagram showing $\hat{\tau}_3$ and $\hat{\tau}_4$ of 7-day annual maximum rainfall for seven communities in Texas Panhandle (open circles) and weighted mean value (filled circle) from example 11–8 322

11.4 Empirical distribution of 7-day annual maxima rainfall for seven communities in the Texas Panhandle 323

11.5 Box plots of empirical distribution of annual peak streamflow for five selected river basins in the United States from example 11–12 326

11.6 L-moment ratio diagram showing $\hat{\tau}_3$ and $\hat{\tau}_4$ of annual peak streamflow for five selected river basins in the United States from example 11–15 328

11.7 Plots of QDF of fitted Wakeby distribution of annual peak streamflow and empirical distribution for five selected river basins in the United States from example 11–18 330

11.8 Plots of PDF of fitted Wakeby distribution of annual peak streamflow for five selected river basins in the United States from example 11–19 332

11.9 Empirical distribution of annual peak streamflow data for U.S. Geological Survey streamflow-gagin station 02366500 Choctawhatchee River near Bruce, Florida and Wakeby and four Generalized Lambda distributions fit by method of L-moments from example 11–22 335

12.1 Comparison of Right-Censored Generalized Pareto distribution fit to right-tail censored probability-weighted moments (thick line) and Generalized Pareto fit to whole sample probability-weighted moments (thin line) from example 12–2 343

12.2 Comparison of three Kappa distribution fits to right-tail censored survival data (dotted lines extended from reported limit) from example 12–5 352

12.3 Empirical survival function (thin line and dashed 95-percent confidence bands) by Kaplan-Meier method from the *NADA* package to left-tail censored arsenic concentration in `Oahu` dataset compared to fit of Generalized Normal distribution (thick line) by flipped and right-censored L-moments by indicator variable from example 12–9 356

12.4 Conditional adjustment for zero values by blipped-distribution modeling of the Generalized Pareto from example 12–14 361

12.5 Time series of annual peak streamflows for Llano River at Llano, Texas (1940–2006) from example 12–15 363

12.6 Empirical distribution and fitted Wakeby distribution to annual peak streamflows for Llano River at Llano, Texas from example 12–17 364

12.7 Study of 100 simulations of sample size of $n = 67$ for indicated Wakeby parent from example 12–18 365

12.8 Study of 100 simulations of sample size $n = 20$ for indicated Wakeby parent from example 12–19 366

12.9 Study of 100 simulations of sample size $n = 200$ for indicated Wakeby parent from example 12–20 367

12.10 Comparison of simulated τ_2 values for 50th (open circles) and 90th (closed circles) percentiles of regional Wakeby parent using 20 simulations for indicated sample size from example 12–25 370

12.11 Comparison of simulated τ_2 values for 50th (open circles) and 90th (closed circles) percentiles of regional Wakeby parent using 2,000 simulations for indicated sample size from repeating of examples 12–23 and 12–25 using `nsim=2000` 371

12.12 Empirical distribution and five fitted distributions to annual peak streamflows for Llano River at Llano, Texas from example 12–26 373

12.13 Bias of sample standard deviation and skew statistics for a Pearson Type III parent and sample size 10 379

12.14 Bias of sample standard deviation and skew statistics for a Pearson Type III parent and sample size 20 379

12.15 Bias of sample standard deviation and skew statistics for a Pearson Type III parent and sample size 40 380

12.16 Bias of sample standard deviation and skew statistics for a Pearson Type III parent and sample size 100 380

12.17 Comparison of product moment and L-moment fits of Pearson Type III and log Pearson Type III to 20 samples drawn from a Pearson Type III parent 382

12.18 Comparison of product moment and L-moment estimation of the 0.99 quantile of Pearson Type III and log-Pearson Type III parents using both nontransformed and $\log_{10}$ transformed data for a sample size of 20 383

12.19 Simulated bivariate data for computation of L-comoments from example 12–29 388

12.20 Simulated bivariate data from Marshall-Olkin copula $\alpha = 0.4$ and $\beta = 0.9$ (open circles) from example 12–37 394

12.21 Simulated bivariate data from Marshall-Olkin copulas with $\alpha = 0.4$ and $\beta = 0.9$ (open circles) and $\alpha = 0.9$ and $\beta = 0.4$ (filled circles) from example 12–38 397

Preface

This dissertation concerns distributional analysis of univariate and continuous data with L-moment statistics using the R environment for statistical computing. The primary audience are practitioners (engineers and scientists) involved in magnitude and frequency analyses. These practitioners might not necessarily consider themselves as statisticians or be extensively educated as such, yet they possess a basic or working knowledge of statistics and have a need to conduct distributional analysis in a computational context that involves the development of empirical fits of probability distributions. It is anticipated that these practitioners are responsible for, or have an interest in, the analysis of data having large range, variation, skewness, or large or small outliers. These data otherwise have long or heavy tails—that is, these data are considerably non-Normal.

As shown herein, L-moment statistics are useful tools for addressing practical problems involving such data. Intended readers are expected to have some statistical education or post-graduate training, but the topic of L-moment statistics very likely is new. Therefore, this dissertation fills a gap in the applied literature and bridges a general gap between statistics and the applied disciplines of science, engineering, finance, and medicine.

Hundreds of examples of R code and ancillary discussion are provided herein and are intended to provide basic functional details of distributional analysis such as computation of statistics, selection of distributions, and distribution fit. The examples also show general use of L-moment-related functions and procedures available in R.

Through the code examples, demonstrations of L-moment statistics in the context of applied circumstances are made, but background statistics, such as the well-known product moments and lesser known probability-weighted moments, also are presented. Demonstrations of the various properties of L-moments also are made along with comparisons to the product moments.

Instead of extensive mathematical derivation, I have chosen to use R to demonstrate a style of distributional analysis developed principally from my experiences and interests over the years. I believe the examples should be especially accessible to readers who do not come from a background of formal statistical education (like myself) and thus such concepts might be new. This dissertation therefore is intended to (1) serve as a general reference about continuous univariate distributions, L-moments, and probability-weighted moments, (2) provide supplementary text for courses in probability and univariate distributions, and (3) provide a primary text for a discipline-specific courses such as hydrologic statistics in the context of a civil, environmental, and hydrologic engineering curriculum. Practitioners in other disciplines, however, should find the material informative.

To describe the origin of this dissertation, some historical background is needed. As I recall, I was introduced to L-moments in Fall of 1995 by Charles Parrett (then with the U.S. Geological Survey in Montana). At the time, I was to conduct a study (Asquith, 1998) of the depth-duration and frequency of annual maximum values of rainfall in Texas, and such data are considerably non-normal. Charles sent a reference to Hosking (1990) and a then current version of the Hosking FORTRAN library (Hosking, 1996b). I acquired detailed knowledge of L-moments during the U.S. Government shut downs in fall 1995—I had some quiet time to work on learning something "new" (Hosking and Wallis, 1993). During this time of disrupted schedule, I began in earnest to digest the literature on L-moments. Although I had operational knowledge of FORTRAN from a groundwater modeling course in graduate school (circa 1993), as I began the rainfall study, I struggled at first to build my own L-moment applications. These were mostly dependent on the FORTRAN library. Eventually I mastered the L-moment library and used it in the ensuing years for Asquith (1998, 2001). Further, my occasional communications with J.R.M. Hosking (IBM), J.R. Wallis (retired IBM, 1996; Yale University, 1996–2010), S. Rocky Durrans (University of Alabama), Mel Schaefer (MGS Engineering Consultants, Inc.), Jurate Landwehr (USGS), and a few others whose names have now slipped my mind (or whose business cards I have lost), were extremely helpful. The L-moment community was (and is) indeed generous.

My first Ph.D. in geosciences (Asquith, 2003) involved the use of L-moments and during the course of that research I had a growing requirement for a suite of tools to support inquires into the L-moments of natural phenomena. I began thinking about a larger system of functions than those then available in the Hosking FORTRAN library. Because of its

marvelous richness relative to FORTRAN, my thoughts during this time were to write an L-moment package (module) for the Perl programming language.

In spring 2004, while delaying before a trip to the airport for a return to Austin from a visit to Texas Tech University in Lubbock, I stopped by the University Bookstore and stumbled onto Peter Dalgaard's "Introductory Statistics with R." On a whim, I purchased the book. I do not remember whether I had even heard of R at that time. (Although by that time, I had received training in the graphical-user interface of S-Plus.) I am a supporter of the world of multi-platform, open-source computing—I am a fan of the Linux operating system and the Perl programming language. It was immediately apparent that R filled a substantial void in my tool chest because R would run on Linux, and at the time I currently lacked an integrated, non-spreadsheet environment for statistical analysis, which would run on that platform.

In the subsequent year or so, I used R extensively for regression analysis and other statistical inquiries. I had written my own high-quality typesetting, data-visualization system in Perl named TKG2, which interfaces with METAPOST and adheres to the refined graphic style of the U.S. Geological Survey, so I had only limited need in production situations for the graphical capabilities of R. (This dissertation significantly follows the style of Hansen (1991) with various adaptions to styles seen in books on R.) During the ensuing years, I continued to acquire other books on R, and often these books described add-on packages. These packages were easily found on the Internet, generally worked as advertised, and represent an impressive feature of R—perhaps the *feature*. Further, books about R had a profound influence on my thoughts about statistical analysis in a practical (workflow, productivity) sense as well as R as a tool for statistical education. From my perspective as an applied researcher, mentor, and occasional educator, the R environment is a fantastic system. In time, I became dependent on several packages, and I began to think about L-moments in R and easing away from FORTRAN-based L-moment analysis.

In June 2005, I began, in my free time, a long process of R-package design and porting of a large portion of the Hosking FORTRAN library to native R. I named the package *lmomco*—a play on "*lmom*ents and *co*mpany" or "*lmom*ents and *co*moments." During and after the initial porting, which often was more or less a syntax re-expression exercise, I refined numerous functions with increasingly more R-like syntax. I make no claim to writing idiomatic R in general. For this dissertation, in particular, I have only used more idiomatic constructs where I believe the context is clear. Several false starts in function nomenclature (dialect) for *lmomco* were made before settling on a style that is seen herein. Further,

I continued to use free time to fill the nascent package with additional developments in L-moment theory such as trimmed L-moments, censored L-moments , L-comoments, and many other computation or convenience functions well beyond the Hosking FORTRAN library.

Near the end of January 2006, I posted the *lmomco* package to the Comprehensive R Archive Network for colleague review by the broader statistical community. This review continues to the present. About that time, Juha Karvanen (the author of the *Lmoments* package) and I had a several month discussion about L-moments, R, and the Generalized Lambda distribution. The results of these discussions then governed further refinements to the *lmomco* package. Following the initial release of *lmomco*, development continued as numerous users from the global R community provided feedback. I am grateful for their support of the *lmomco* package—the bug reports and suggested enhancements and features are welcome and most appreciated. In conclusion, several bugs and many needed enhancements were identified and added during the writing of this dissertation.

Lubbock, Texas *William H. Asquith*
April 18, 2011

Chapter 1
Introduction

1.1 Distributional Analysis

Information is contained in data, data originate from measurements, and measurements arise from intent. Whether the intent is to expand the breadth of human knowledge on the frontiers of physics, protect citizenry from natural hazards, or to provide information for lifetime analysis of a simple automotive switch, measurements are intentionally made and recorded as data.

The collection of data exists within a context, and this context includes a community of individuals and institutions with an interest in the process of data acquisition, analysis, and interpretation, as well as the ultimate implementation of ensuing results. Distributional analysis of data is a fundamental component of that process.

For this dissertation, "distributional analysis" is a term and concept that, for generally univariate data (single variable), encompasses the inherently iterative steps of exploratory data analysis, summarization of samples, selection of distributions, and techniques of fitting distributions to data. A general goal of distributional analysis is to provide parametric models of one or more data sets (samples) under study. Distributional analysis, therefore, can be used to answer questions such as "How frequently is a given value exceeded?" or "What is the magnitude of a value for a given nonexceedance probability or cumulative percentile?"

There is no room unfortunately to weave into the fabric of this dissertation the extensive discussion by Klemeš (2000a,b) on distributional analysis. The discussion by Klemeš however is especially germane to distributional analysis and interpretation of non-Normal data and a review by practitioners is highly recommended.

Perhaps the one universal feature of data is that data are distributed—that is, all data have a **sample distribution**. This distribution is produced by random samples from a typically unknown **parent distribution**. The parent distribution often is a hypothetical model of the population from which the data were drawn. The parent distribution can range from the simple to the complex. Numerous probability distributions are described by Ross (1994), Evans and others (2000), and similar texts cited later and R-oriented texts such as Venables and others (2008) and others cited later. Many of the distributions considered by those authors are described in this dissertation. Although several are complex, the univariate, smoothly-varying, limited-parameter distributions used herein never-the-less represent idealized models of distribution geometry of the population.

It is possible that distributional analysis more often than not represents one of the first forays towards interpretation and understanding of the information represented by the data. Further, this foray likely is made before hypothesis tests, analysis of variance, linear modeling (regression), or other analyses are performed. Distributional analysis also can contribute to exploratory data analysis. Distributional analysis might be used for data screening (such as for the detection of anomalous or erroneous data), or perhaps used for the detection of changes in industrial processes (quality assurance and control purposes), or the analysis might provide a means to an end, such as for the specification of design heights for flood-control levees.

The context surrounding, and the community involved in, distributional analysis can result in specific statistical approaches to become embedded by the colored lens of tradition. Perhaps the community involved with a particular data type deems that preference should be given to a branch of statistics such as nonparametrics, or preference should be given to a log-Normal distribution fit by the method of maximum likelihood, or preference should be given to the method of moments for fitting a Weibull distribution.

Newcomers to a field—be they freshly minted statisticians, mathematicians, scientists, computer scientists, engineers, interdisciplinary specialists, or temporary consultants—can be major contributors when given *freedom* of expression and *freedom* to explore by administrators, managers, and mentors. Sometimes ignorance of the newcomer to community-accepted nuances or *a priori* interpretations of data along with a lack of knowledge of traditional techniques for distributional analysis can be a positive source of change. Newcomers often bring new insights, approaches, and tools to statistical problems. Old-timers can become *newcomers* when experienced practitioners invest in new approaches and tools and can assimilate these into the acquired wisdom of their careers.

1.2 The R Environment for Statistical Computing

A theme now established is *analysis freedom* and the associated use of new approaches and tools for statistical problems. One tool that from its very inception offers analysis freedom is the **R environment** for statistical computing (R Development Core Team, 2010). As shown in this dissertation, the R environment is an exceptionally useful tool for statistical work and, by association, distributional analysis.

Quoting (circa 2011) from the R web site `http://www.r-project.org`, the R environment is

> . . . a language and environment for statistical computing and graphics. It is a GNU project [`http://www.gnu.org`], which is similar to the S language and environment. The S language was developed at Bell Laboratories (formerly AT&T, now Lucent Technologies) by John Chambers and colleagues. R can be considered as a different implementation of S. There are some important differences, but much code written for S runs unaltered under R.
>
> R provides a wide variety of statistical (linear and nonlinear modelling, classical statistical tests, time-series analysis, classification, clustering, . . .) and graphical techniques, and is highly extensible. The S language is often the vehicle of choice for research in statistical methodology, and R provides an open-source route to participation in that activity.
>
> . . .
>
> R is available as Free Software under the terms of the Free Software Foundation's GNU General Public License in source code form. It compiles and runs on a wide variety of UNIX platforms and similar systems (including FreeBSD and Linux), Windows, and MacOSX.
>
> . . .
>
> R is an integrated suite of software facilities for data manipulation, calculation and graphical display. It includes an effective data handling and storage facility, a suite of operators for calculations on arrays, in particular matrices, a large, coherent, integrated collection of intermediate tools for data analysis, graphical facilities for data analysis and display either on-screen or on hardcopy, and a well-developed, simple and effective programming language which includes conditionals, loops, user-defined recursive functions and input and output facilities.
>
> . . .
>
> Many users think of R as a statistics system. [The R Development Core Team] prefer[s] to think of it of an environment within which statistical techniques are

> implemented. R can be extended (easily) via packages. There are about eight packages supplied with the R distribution and many more are available through the [Comprehensive R Archive Network (CRAN)] family of Internet sites covering a very wide range of modern statistics.

This dissertation is oriented around statistical computing using R. The choice of R is made in part because of the open and generous global community involved in the R project, and this dissertation is a way of paying *alms* to the community. The power of the community and the freedom available to members through use of R is made manifest by the vast armada of packages available through the Comprehensive R Archive Network (CRAN), which is accesssible through the R web site or directly at `http://www.cran.r-project.org`. The core development team of R has made custom extensions to the language for specialized applications an economical (time wise as well as intellectually and financially) and straightforward process (R Development Core Team, 2009).

As recently as 2002, Dalgaard (2002) reported that over one hundred R packages were publicly available. At the time of this writing (2011), more than 2,800 packages are available that encompass innumerable topics, techniques, and tools. This phenomenal rise in R packages or extensions to the language is evidence of the growing and global popularity of the R environment. Through the CRAN, by brief connection to the Internet, one can bring new or unfamiliar statistics and computational approaches to their R installations, which might exist on a range of computing platforms. The CRAN also stores many advanced or otherwise discipline-specific packages. Further, the CRAN contains innumerable implementations of classical and well-known statistical concepts. In either case, R packages are readily incorporated or installed by the user. By providing a mature and multi-platform computing environment along with a vast array of extensions and packages to solve problems, the R environment, therefore, provides the analyst with *Freedom*:

- *Freedom* for thought and reflection,
- *Freedom* for exploration and discovery,
- *Freedom* to choose computing platform, and
- *Freedom* to contribute to the R project.

Within this dissertation, a total of 246 examples using about 425 functions in R are presented along with considerable coupling to about 515 numbered equations. The examples demonstrate the freedom provided by R in the broad topic of distributional analysis, in general, and with L-moment statistics (see Section 1.3 and Chapter 6), in particular. The

numerous examples and associated discussion, figures, and tables thus are intimately tied to R. As context or layout requirements (or constraints) facilitated, USING R identifiers—an example is shown below—are placed throughout this dissertation. These identifiers are intended to demark or signify a transition from the general statistical and mathematical narrative to a computational context using R.

USING R ——————— ——————— USING R

Assuming that R has been downloaded, installed, and can be successfully started, one is presented with a textural interface (command line) similar to that shown in example 1–1. For the example, the `help()` function is used to display the documentation for the `mean()` function. The `mean()` function is used to compute the arithmetic mean. The example ends with a termination of the R process by the `q()` function ("quit"). The use of the `q()` function in the example also demonstrates the use of named arguments and how such arguments are passed to R functions. For example 1–1, the named argument `save="no"` tells the exiting sequence to not save the current workspace and to bypass a manual prompt or dialog box requiring action by the user.

1–1

```
R version 2.12.1 (2010-12-16)
Copyright (C) 2010 The R Foundation for Statistical Computing

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

  Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

>help(mean) # calling the help function on function mean
>print("First␣use␣of␣R?"); print("Yep!")
[1] "First␣use␣of␣R?"
[1] "Yep!"
>avariable <- "assignment␣of␣a␣string␣to␣a␣variable"
>bvariable <- 4e-5 # assignment of a small number to a variable
>q(save="no") # quitting the application
```

The example and the associated discussion illustrate that source code herein is typeset in a monospaced font and that matched `()` characters preceded by an alphabetic string are used to signify the name of a function. Code comments, characters not interpreted by R, are preceded by the # character and are typeset herein with an *# obligue* type face. Multiple commands on a line are separated by the `;` (semicolon) as done for the two `print()` commands in the example. Spaces in strings are indicated by the `"␣"` character, but this character is not typed by the user—the space bar suffices. Assignments to variables, scalars in the case of the example, are made by the "`<-`" operator and generally not by the = sign. The example also shows that the "`>`" character is the (default) command prompt. To facilitate cut-and-paste operations from portable document format (PDF) versions and the monolithic LaTeX source files of this dissertation, the prompts are suppressed in all other examples. An additional note concerning the contents in the examples is needed.

Many of the examples have the code *#pdf()* and *#dev.off()* as pairs with graphical plotting functions between the pair. The `pdf()` function is used to generate portable document format (PDF) files and shows the name of the portable document format (PDF) file used in the typesetting of this dissertation; the `dev.off()` function closes the file. These two functions are commented out by the # character. These functions are not intended for end users but are specifically retained to remain available for errata correction in derivatives of this dissertation.

The large number of R examples and their typical length results in formidable challenges in page breaking and therefore layout. As a result, the examples in this dissertation have been permitted to internally break like paragraphs across pages. It is anticipated that this style will not be of major inconvenience to most readers. Virtually all of the figures are generated by the examples, and for layout purposes, the figures for the most part have been permitted to "float" to their present locations by the algorithms of the LaTeX typesetting system. Finally, breaks between topically distinct examples or demonstrations, which typically follow the Using R identifiers, are separated by a right-justified, black triangle. An example is shown to the right. ◀

The intent of the triangles is to help alert the reader to interruptions between narratives describing one or more listings of example code and the example-specific discussion.

1.2.1 Internet Resources for R

Because R is a popular and global project, the Internet naturally provides a myriad of sources related to the use of R. The R project website `http://www.r-project.org` contains numerous links to portable document format (PDF) manuals (R Development Core Team, 2009; Venables and others, 2008), frequently asked questions (FAQs), conferences, newsletter ("R News," 2001–08), dedicated journal ("The R Journal," 2009–present), books, mailing lists, tutorials, and other supporting material. A handy ensemble of R reference cards is provided by Short (2004).

The CRAN provides "home pages" for specific packages. These can be accessed by `http://www.cran.r-project.org/package=NAME`. For example, the *lmomco* package (Asquith, 2011), which is a major component of this dissertation, can be accessed by `http://www.cran.r-project.org/package=lmomco`. Finally, a particularly useful entry point on the topic of distributional support in R is found at `http://www.cran.r-project.org/web/views/Distributions.html`. Readers are strongly encouraged to review this web page for a sweeping review of distributional packages using R; many other details also are provided.

1.2.2 Traditional Publishers for R

In the few years previous to 2011, many books related to R have been published. Notable publishers of R-related books include Cambridge University Press, Chapman & Hall/CRC, John Wiley, McGraw-Hill, O'Reilly Media, Sage Publications, and Springer. The Journal of Statistical Software `http://www.jstatsoft.org` regularly publishes articles on R including R-package introductions.

There are now (2011) many books focused on the R software and statistical analysis using R. There often is substantial overlap between the material in the cited books herein. Some books have both introductory and advanced material. In each case, the authors all have their own unique stories to tell. Books focused on introductory material are Dalgaard (2002), Heiberger and Holland (2004), Braun and Murdoch (2007), and Sawitzki (2009). For an additional introduction to data analysis, R syntax, and techniques, Everitt and Hothorn (2006) and Maindonald and Braun (2003) are recommended. A specialized book on R programming, but more significantly data manipulation, is provided by Spec-

tor (2008). Readers interested in multivariate analysis might consider Everitt (2005). Two outstanding books as references related to linear model building with R are Faraway (2005, 2006). A comprehensive reference on use of R for graphical display is provided by Murrell (2006), and extensive discussion of R graphics in thoroughly documented applications is provided by Keen (2010). Jurečková and Picek (2006) provide a reference on robust statistics with R. A comprehensive quick reference to R and many auxiliary packages is provided by Adler (2010).

Additional references that encompass the use of R in statistics education and applied probability are Rizzo (2008), Verzani (2005), and Ugarte and others (2008). Finally, Baclawski (2008) provides a detailed review of R programming techniques used in practical circumstances. Reimann and others (2008) provide extensive presentation of R in an environmental statistics context; these authors have a different approach relative to the other books in that they do not present line-by-line R programming examples. Qian (2010) provides an outstanding case-study oriented review of basic to advanced statistical inference and modeling techniques associated with environmental, hydrologic, and ecological data. Collectively, the cited books and others therein show that R has earned tremendous popularity across a wide spectrum of disciplines.

1.3 L-moments—A general description

Data have sample distributions. Analysis of these data, univariate in the context here, is a complex subject, which is simultaneously influenced by, and has influence in, many branches of statistics and other disciplines. The literature of distributions is extensive, and disciplines requiring distributions are encompassing. L-moments provide a powerful and easy to use statistical framework for distributional analysis.

The **theory of L-moments** described herein includes definition of L-moments, trimmed L-moments, methods for L-moment computation for distributions and estimation from samples, inclusion of probability-weighted moments, sample properties of both moment types, parameter estimation methods for numerous familiar and not-so-familiar distributions, techniques for discriminating between distributions, and other topics. An outstanding contextual entry point for some of the analytical themes herein is available in Wallis (1988).

Beyond a reasonable sampling of the relevant "journal" literature cited herein, there are several books containing information about L-moments includeing Stedinger and others (1993), Hosking and Wallis (1997), Hosking (1998) (article), Gilchrist (2000), Dingman (2002), and Kottegoda and Rosso (2008). In particular, Stedinger and others (1993) and Hosking and Wallis (1997) are canonical L-moment references as well as the monograph of Hosking (1986) on probability-weighted moments.

What are L-moments? To answer succinctly, although not expected to be clear to most readers at this point, **L-moments** (Hosking, 1990) are defined through linear combinations of the expected values of order statistics. The study of order statistics is a branch of statistics concerned with the statistics of ordered random variables and samples. The familiar minimum, maximum, and median are likely the most familiar order statistics.

L-moments are direct analogs—but not numerically equivalent—to well-known product moments, such as standard deviation or skew. The first L-moment is the arithmetic mean, which should be particularly comforting to readers who are not previously familiar with L-moments. As analogs, L-moments have similar, that is, familiar, interpretations and hence applications as the product moments. L-moments, therefore, are useful and are intellectually accessible to most of the general scientific and engineering community. Accessibility into L-moment theory is greatly enhanced in practical application by the L-moment support available in R as described and demonstrated in this dissertation.

L-moments have many advantages over the product moments including natural unbiasedness, robustness, and often smaller sampling variances than provided by other estimators. These advantages are particularly important with data having large range or variation, large skewness, and heavy tails. The sampling properties of L-moments are central to their attractiveness for distributional analysis of Normal to non-Normal, symmetrical to asymmetrical, and thin to heavy-tailed distributions. The attractive sampling properties in the context of using R are shown by example. In short, L-moments provide comprehensive "drop in" replacements for product moments in many practical situations or at the very least are complementary to the product moments.

L-moments have an exciting extension to multivariate data. These L-moments are known as L-comoments (Serfling and Xiao, 2007). L-comoments can measure asymmetrical relations between variables in multivariate data. Multivariate distributional analysis is generally outside the univariate scope of this dissertation. However, in a circumstance (the terminal section of this dissertation) where it makes sense, L-comoments are included along with copulas, which are convenient mathematical constructs for multivariate work.

1.3.1 L-moments in R

At the time of this writing (2011), three R packages in particular provide generalized support L-moment-based approaches for distributional analysis. The packages are *Lmoments* (*L-moments and Quantile Mixtures*) by Karvanen (2009), *lmomco* (*L-moments, Trimmed L-moments, L-comoments, Censored L-moments, and Many Distributions*) by the author (Asquith, 2011), and *lmom* (*L-moments*) by Hosking (2009a). There also is the more-discipline-specific *lmomRFA* (*Regional Frequency Analysis using L-moments*) package by Hosking (2009b).

Collectively, these packages answer a call by Royston (1992) who states in the abstract that "Indices of distributional shape based on linear combinations of order statistics have recently [1990] been described by [Hosking (1990)]. [The] usefulness [of L-moments] as tools for practical data analysis is examined. [L-moments] are found to have several advantages over the conventional [product moment] indices of [skew] and kurtosis [with] no serious drawbacks." Royston (1992) continues "It is proposed, therefore, that [L-moments] should replace [skew] and [kurtosis] in routine data analysis, [and] to implement this suggestion, **action by the developers of standard statistical software** is needed." (The bold typeface is this author's.)

Other packages, such as the *POT* package (*Generalized Pareto and Peaks over Threshold*) by (Ribatet, 2009) and the *RFA* package (*Regional Frequency Analysis*) by (Ribatet, 2010), provide for computation of L-moments and discipline-specific features. Collectively, the six cited packages appear to currently (2011) cover, albeit with some redundancy, the general gambit of L-moment theory and support from the CRAN. A listing of packages that provide L-moment support,[1] in the order of initial release, is provided in table 1.1. There remains much room for growth in R for packages related to L-moments, and additional discussion is provided in the Epilogue of this dissertation.

[1] Gilleland and others (2010) provide the *extRemes* package related to extreme value analysis that uses some L-moment functions from the *Lmoments* package by Karvanen (2009). Also, Su (2010) provides the *GLDEX* that is focused on the Generalized Lambda distribution (see page 272 for more discussion). The package provides for parameter estimation using Su's own L-moment functions. These functions are credited to Karvanen and thus seem to derive from the *Lmoments* package. The *GLDEX* package also provides many appropriate citations to Asquith (2007).

Table 1.1. Summary of L-moment related R packages available on the CRAN in order of initial release

Package	Citation	Initial release	Current release
POT	Ribatet (2009)	September 6, 2005	October 16, 2009
RFA	Ribatet (2010)	September 14, 2005	January 14, 2010
Lmoments	Karvanen (2009)	October 12, 2005	January 19, 2011
lmomco	Asquith (2011)	January 31, 2006	April 15, 2011
lmom	Hosking (2009a)	July 3, 2008	November 29, 2009
lmomRFA	Hosking (2009b)	March 3, 2009	August 22, 2010

1.3.2 Internet Resources for L-moments

Across the distal reaches of the Internet are several useful resources related to L-moments. Robert Serfling at University of Texas at Dallas provides a central location `http://www.utdallas.edu/~serfling/` that contains useful references and links. J.R.M. Hosking's L-moments page is located at `http://www.research.ibm.com/people/h/hosking/lmoments.html`. A Matlab program by Kobus Bekker is located at `http://www.mathworks.com/matlabcentral/fileexchange/loadAuthor.do?objectType=author&objectId=1094208` and a Stata module by Nicholas J. Cox to generate L-moments and derived statistics is available at `http://ideas.repec.org/c/boc/bocode/s341902.html`.

1.4 Purpose and Organization

There are several interrelated purposes of this dissertation. One purpose is to present a framework by which distributional analysis of univariate data with L-moment statistics using R can be performed by practitioners with a wide variety of skill levels and educational backgrounds. This dissertation is structured to provide a general reference to L-moment and related statistics in which many readers might find useful the ability to browse or use specific pages of the text. The general reference nature also requires the occasional use of both foreshadowing and back referencing by cross reference within the text.

Another purpose of this dissertation is to serve as a supplemental text in courses involving analysis of univariate distributions and samples. Dingman (2002, Appendix C) and also Kottegoda and Rosso (2008) are textbooks oriented towards civil and environmental engineering, and both books provide treatment, albeit brief, of L-moments.[2] To enhance the textbook purpose, vocabulary words at their primary introduction or definition are typeset in bold typeface as are the page numbers in the index.

The purposes of this dissertation are achieved by a balance of mathematical discussion (about 515 numbered equations) and use of L-moments along with related statistics in both theoretical (simulation) and practical (real-world data) circumstances. To achieve this purpose, numerous examples of R code are provided, and the *lmomco*, *lmom*, and *Lmoments* packages are used. The focus here however is near universal on the *lmomco* package, and the author's unique contributions to the field. A major purpose of this dissertation is to further enhance the documentation of the author's *lmomco* package far beyond the scope of the user's manual (Asquith, 2011).

This dissertation generally is organized as follows. This introductory chapter provides (1) background discussion prior to delving into distributional analysis and (2) a small section of basic visualization of sample distributions using R.

Chapter 2 provides an introduction to the concepts of distributional analysis, probability distributions, and discussion of basic summary statistics. Also in Chapter 2, the properties of probability distributions, the technique of fitting a distribution by moments (a generic or conceptual meaning at this and that point in the narrative), and alternative methods for visualization of distributions are described. Ending Chapter 2 is a simple demonstration of distributional analysis for both simulated and real-world data in order to cast appropriate themes for the remainder of this dissertation. To complete the background and setup, Chapter 3 provides an introduction to the order statistics and demonstrates some connections to L-moment theory.

In order to provide a complete narrative and provide for juxtaposition with L-moments, Chapter 4 defines and demonstrates the use of product moments. Some basic sampling properties of product moments are expressed in that chapter through many examples. Chapter 5 defines and demonstrates use of the probability-weighted moments, which were historic predecessors to L-moments. The probability-weighted moments are very

[2] It should be noted that Dingman provides more detailed treatment. The author took a civil engineering course in 1994 in which a "handbook" containing Stedinger and others (1993) was used as a supplemental text. Stedinger and others (1993) provides much detail concerning L-moments.

useful companions to L-moments and are an important component of L-moment theory. Further, the probability-weighted moments facilitate use of L-moments for censored distributions. The L-moments are formally defined in Chapter 6, and the sampling properties of L-moments are numerically explored by example and graphical output in that chapter.

L-moments primarily are used with probability distributions. Therefore, the rather lengthy sequence of Chapters 7–9 summarizes numerous distributions and their respective L-moment statistics as supported by the *lmomco* package and in many cases the *lmom* package as well. Many examples of these three chapters use functions that are otherwise interwoven throughout the tapestry of the greater text. A given reader in time would be expected to make several passes through these three chapters to grasp the entirety of the material.

Distribution selection is an important and complex subject. The theory of L-moments offers a convenient and powerful tool for discriminating between distributional form and *ad hoc* judging of goodness-of-fit through the use of L-moment ratio diagrams. These are described in Chapter 10. Hypothesis testing of goodness-of-fit is outside the scope here; readers are directed[3] to basic statistical texts (Verzani, 2005, Section 9.3) and papers (Vogel, 1986). Sawitzki (2009, pp. 10–34) provides considerably relevant attention to "distribution diagnostics." Finally, the R language has several built-in functions for goodness-of-fit testing, such as `ks.test()` (Kolmogorov-Smirnov test) or `shapiro.test()` (Shapiro-Wilk test for normality).

Short studies and advanced topics are presented in Chapters 11 and 12, respectively. The References section contains all citations used in the text and is followed by an Epilogue that will provide the reader with the author's vision for further expansion of the large body of work herein and L-moment support in R.

A topical "Index" is provided to enhance accessibility and reference component of this dissertation. Finally, the "Index of R Functions" separately lists built-in R functions, other miscellaneous functions created in this dissertation, and functions in the *lmomco*, *Lmoments*, and *lmom* packages as well as selected functions of a few other R packages.

[3] The topic of goodness-of-fit is enormous and the single book and paper cited here are but a trifle of the literature on the subject. Internet searches are suggested: "L-moments goodness-of-fit" will provide hits of particular relevance to this dissertation. Finally, because plotting positions are so common in the hydrologic sciences and this dissertation, the citation to Vogel (1986) is justified.

1.5 Reader Expectations and Advice to the Reader

For this dissertation, it is assumed that the reader already possesses basic understanding of statistics, knowledge of the concepts of probability distributions, and working knowledge of a programming language—not necessarily R. The basic understanding of statistics implies familiarity with distributions and product moments, such as the arithmetic mean and standard deviation, as well as rank-based (order-based) statistics such as the median.

Many readers are advised that a single thorough or meticulous pass through the text might not be as effective as first understanding the general organization and content and then proceeding to search for specific content as needed. The text is large, complex, and multipurpose. Each pass through the text is expected to produce new discoveries to many readers.

This dissertation is not intended as an introduction to R *per se*, and thus an initial tutorial or extensive introduction to R is not explicitly provided. For a broad introduction to R, Dalgaard (2002) provides an outstanding resource. The author also recommends Heiberger and Holland (2004). However, for the code-based examples herein, programmatic operations (assignments, loops, conditionals), which are outside the topic of L-moments and related statistics, rely on built-in and commonly used R functions. Readers possessing some programming aptitude thus should be able to readily grasp, adapt, and extend the examples.

This dissertation presents various elements of problem solving in an algorithmic framework. The examples are purposefully written in a generally verbose style with perhaps excess redundancy of description for the examples spread throughout the text. The author has avoided shortcuts in syntax, which the R language can provide through its own idiomatic constructs. The general avoidance of shortcuts is done so that the programming style remains elementary and more self explanatory. As a result, the concepts and code functionality herein should be readily accessible to the intended audience. Finally, the author has purposefully tried to use as many built-in R functions (about 125) as contextually appropriate so as to also have this dissertation serve as an effective document of algorithmic programming for distributional analysis using R.

For general documentation and educational puposes, the examples often are explained in detail. This dissertation also is supposed to be a reference, and therefore, examples in early portions of this dissertation generally and purposefully are more thoroughly explained than in later portions. This practice is especially evident in Chapters 11 and 12,

which are more advanced and naturally are more dependent on material presented in previous chapters—the elementary portions of the examples are less thoroughly described.

It also is assumed that readers are capable of installing external R packages and have already installed the *lmomco*, *lmom*, and *Lmoments* packages. For virtually all of the examples herein, it is assumed that at least the *lmomco* package has been loaded into the work session to gain access to package functionality.

Example 1–2 demonstrates the package loading mechanism or `library()` function of R. The majority of the examples, however, use (require) the *lmomco* package only. When the *lmom*, *Lmoments*, or other packages are used, it will be made clear to the reader and often made explicitly clear by `library()` calls. The narrative is purposefully written so as to generally not identify the source of the function such as: "the `library()` function of R" or "the `cdfgum()` function of *lmomco*." Such a practice would considerably lengthen the text. The "Index of R Functions" distinguishes between the source of each function presented herein, and readers are explicitly directed there when confusion arises concerning the source package of a function.

1–2

```
# load the three packages
library(lmomco) # only this package is needed for most examples
library(lmom)
library(Lmoments)

help(pmoms) # help for the sample product moments (lmomco)
```

Finally, the installation methods for R packages vary slightly by computer platform and security credentials available to the user. Readers requiring initial instruction or assistance should consult the R website at `http://www.r-project.org` and follow the links such as *Manuals* or *FAQS*. Even brief searches on the Internet with terms such as "installation of R" should find helpful guides and documents on installing R for most computer platforms.

1.6 Types of Data

There are literally an infinite number of univariate data types. One type of data, very loosely defined, is of interest for the distributional analysis described herein; those data types that are non-Normal. In particular, data characterized by, or having a tendency

towards, heavy tails (left, right, or both), asymmetry, and the regular presence of outliers are examples for which the properties L-moments are attractive. Examples of such data types are earthquake (geophysical), floods and droughts (hydrological), and rainfall (meteorological). Hydrological and meteorological data often will be used herein as these data are the most familiar to the author.

Throughout this dissertation, numerous, and generally self-contained, examples are provided. Often these examples use simulation (see Ross, 1994, chap. 10) to generate synthetic data by random drawings from a specified parent distribution. An R-oriented discussion of simulation is found in Rizzo (2008, chap. 3), Verzani (2005, chap. 6), and Qian (2010).

Simulations and simulated data are used herein for at least two purposes. First, generation of simulated data in the examples facilitates the construction of self-contained code and minimizes the presentation "overhead" related to accessing and reading in external data. Second, by explicitly specifying the parent distribution or "truth" in a statistical context, the characteristics or properties of various statistics or distributional form can be explored here and independently by self-study-minded readers. Simulated data removes the constraints of sample size and permits exploration of the effects of sample size on statistical procedures. As will be seen, the R environment is outstanding for statistical experimentation by simulation.

It is assumed that most readers who originate from nonstatistical backgrounds might have limited or perhaps no prior experience with simulation and exploration of sampling properties of statistical estimators. This assumption is made based on the author's experiences with curricula outside of degrees in statistics, and particularly experiences with geoscience and engineering programs, that lack a core statistical component. As a result, many of the examples are intended to provide a sufficient structure to aid adventurous readers in self study. By incorporating simulated data, readers implementing the examples will produce numerical or graphical output that should differ in value or appearance—but the general nature of the results should remain the same. An appropriate balance between real-world data and simulated data hopefully has been achieved.

USING R ——— ——— USING R

Input of external data and output of results to external files is an important feature of R. For some examples, the loading of external data files is needed. Five functions in

particular are useful and are listed at the R prompt by `?read.table`. Those functions are listed in example 1–3.

1–3

```
read.table(file, header = FALSE, sep = "", quote = "'",
           dec = ".", row.names, col.names,
           as.is = !stringsAsFactors,
           na.strings = "NA", colClasses = NA, nrows = -1,
           skip = 0, check.names = TRUE,
           fill = !blank.lines.skip, comment.char = "#",
           strip.white = FALSE, blank.lines.skip = TRUE,
           allowEscapes = FALSE, flush = FALSE,
           stringsAsFactors = default.stringsAsFactors(),
           encoding = "unknown")

read.csv(file, header = TRUE, sep = ",", quote='"', dec=".",
         fill = TRUE, comment.char="", ...)

read.csv2(file, header = TRUE, sep = ";", quote='"', dec=",",
          fill = TRUE, comment.char="", ...)

read.delim(file, header = TRUE, sep = "\t", quote='"', dec=".",
           fill = TRUE, comment.char="", ...)

read.delim2(file, header = TRUE, sep = "\t", quote='"', dec=",",
            fill = TRUE, comment.char="", ...)
```

Following Rizzo (2008, p. 367), the creation and use of a comma separated file or a `*.csv` file is informative. In example 1–4, a data frame is created for some fabricated streamflow data, and these data are written using the `write.table()` function to a file titled "`temp.csv`." In turn, the data are reloaded using the `read.csv()` function. A type of input-output process in R is shown.

1–4

```
# create a "hydrograph" of streamflow in cubic meters per second
streamflow <- c(0, 10, 40, 50, 100, 400, 300, 200, 75, 50)
minutes    <- 1:length(streamflow)*60 # cumulative time
# create the data frame that contains both "columns" of data
d <- data.frame(FLOWcms=streamflow, TIMEmin=minutes)
# write the data frame, temp.csv can be opened by text editor
write.table(d, "temp.csv", sep=",",
            row.names=FALSE, quote=FALSE)
rm(d,streamflow,minutes) # remove the objects
# Now read data back in, but output for read.csv shown only
# read.table(file="temp.csv", sep=",", header=TRUE)
read.csv(file="temp.csv") # same thing
   FLOWcms TIMEmin
```

```
1         0       60
2        10      120
3        40      180
4        50      240
5       100      300
6       400      360
7       300      420
8       200      480
9        75      540
10       50      600
```

For the examples in this dissertation however, the majority of external data have been formatted into the `*.RData` format (see `?save`) and are available from the *lmomco* package. These data are accessed by the `data()` function, which is formally introduced in the next section. ◀

1.7 Visualization of Sample Distributions—Histograms and Box Plots

The visualization of sample distributions is an elementary and informative step of distributional analysis. Sample distributions, sample (fitted) probability distributions, and parent probability distributions are graphically depicted throughout this dissertation. However, two elementary graphical techniques, which will not see extensive use elsewhere in the examples herein, are **histograms** and **box plots**. Those graphics are described by Chambers and others (1983), Helsel and Hirsch (1992; 2002), Murrell (2006), and applicable references therein, and R-oriented treatments can be found in Rizzo (2008, chap. 10) or Ugarte and others (2008, pp. 45–46).

1.7.1 Histograms

To summarize, histograms are *ad hoc* depictions of the frequency or number of occurrences of data points within specified intervals of the data understudy. Using R and the built-in data frame titled *airquality*, two histograms are readily generated in example 1–5. The *airquality* data frame is loaded by the `data()` function as shown in the following example (see `?data.frame`). The `ls()` function lists the contents of the current workspace. The `names()` function has no core use in the example, but is shown to illustrate a feature of R for querying the named contents of a data frame (and other data structures).

First, the `layout()` function is used to specify the plotting layout, which is defined by the `matrix()` function, of future graphic calls. In the example, two vertically stacked plots are setup by the `layout()` function. Subsequent calls to the `hist()` function actually produce the corresponding histograms that are shown in figure 1.1. There are many options available to the user of the `hist()` function but are not explored here.

1–5

```
data(airquality)  # load in the airquality data frame
ls() # list the contents of the workspace
[1] "airquality"

names(airquality) # query the data frame for the named fields
[1] "Ozone"   "Solar.R" "Wind"    "Temp"    "Month"
[6] "Day"

#pdf("hist.pdf")
layout(matrix(1:2, nrow=2)) # two plots, top and bottom
hist(airquality$Ozone) # histogram for the top plot
hist(airquality$Temp)  # histogram for the bottom plot
#dev.off()
```

◀

Although easy to use and common in graphical display of distributions in popular culture, histograms are easily and unfortunately distorted by the size of the bins (intervals on the horizontal axis), and in the author's opinion, histograms generally are of limited usefulness for quantitative distributional analysis. Somewhat more sophisticated graphics and tools are described in later examples. Histograms however do represent real features of the data. The histograms of the previous example show that the ozone data have positive skewness or in other words, skewed to the right (long right tail), and the air temperature data are more symmetrical with a mean value in the upper 70s.

1.7.2 Box Plots

Box plots (Helsel and Hirsch, 1992, pp. 24–26) are another graphical construct for visualizing sample distributions. Example 1–6 produces default box plots for the *airquality* data frame, and the results are shown in figure 1.2. The `boxplot()` function is powerful, and a wide range of options are available to the user. As a reminder, documentation of a function is easily accessed by the user through the `help()` function: `help(boxplot)`.

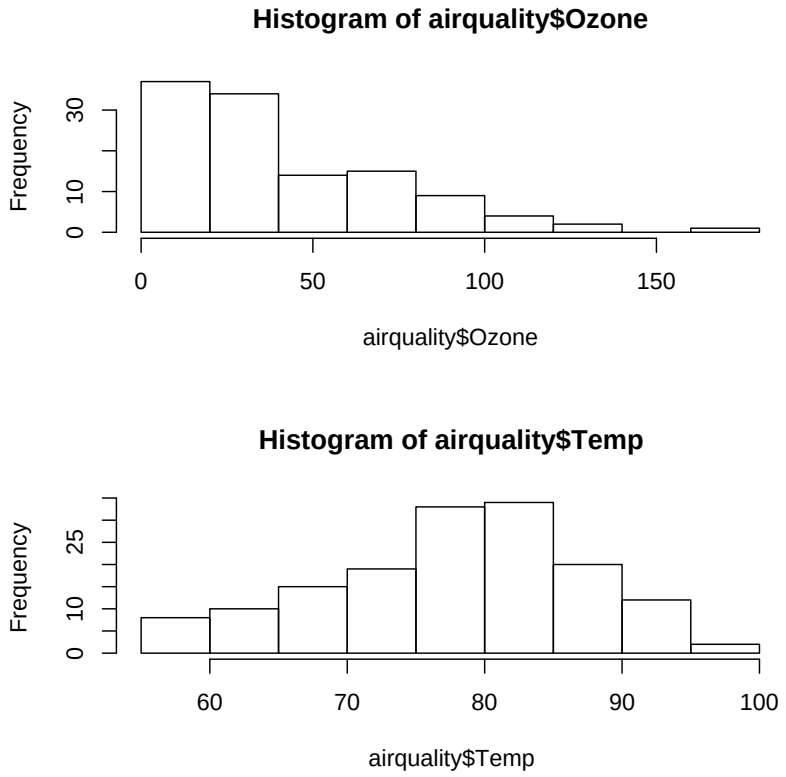

Figure 1.1. Histograms of ozone and temperature for data in the *airquality* data frame from example 1–5

1–6

```
data(airquality); attach(airquality)
#pdf("boxplot.pdf")
boxplot(Ozone/mean(Ozone,     na.rm=TRUE),
        Solar.R/mean(Solar.R, na.rm=TRUE),
        Temp/mean(Temp), Wind/mean(Wind),
        names=c("Ozone", "Solar␣Rad", "Temp", "Wind"),
        ylab="VALUE␣DIVIDED␣BY␣MEAN", range=0)
#dev.off()
```

This example differs from example 1–5 in that the method to access the data of the *airquality* data frame is distinctly different. This example `attach()`es the names into the workspace so that ozone data are accessible as a simple name `Ozone` instead of the longer syntax `airquality$Ozone`, which was shown in the previous example. (The function `detach()` detaches the named contents of a data frame from the current workspace.)

For the ensemble of four box plots in figure 1.2, dimension in the data is removed through division by the respective mean values. The `na.rm=TRUE` is needed for ozone and

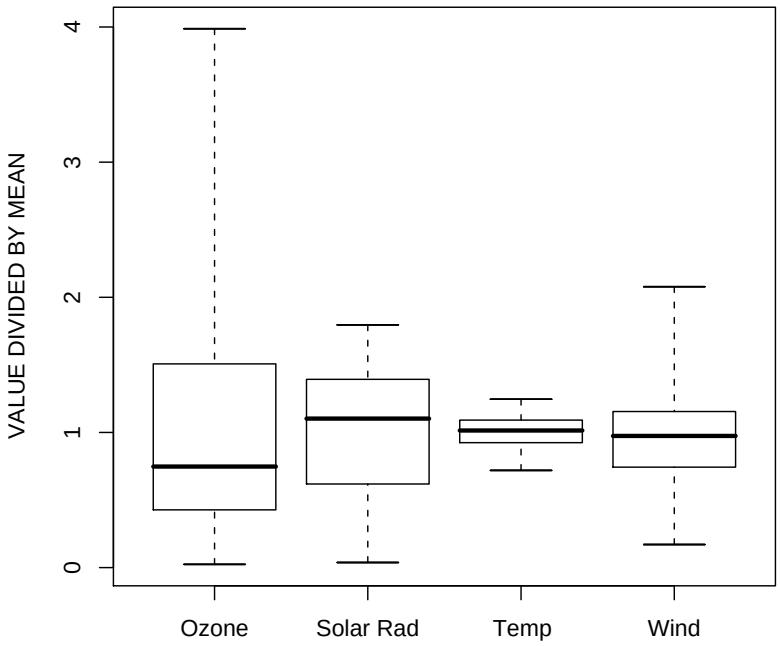

Figure 1.2. Box plots of ozone, solar radiation, temperature, and wind speed data for data in the *airquality* data frame from example 1–6

solar radiation data because missing values are present in each. The `range=0` argument causes the whiskers of the boxes to extend to the minimum and maximum of the data. The bottom and top of the boxes represent the 25th and 75th percentiles, respectively. The heights of the boxes, therefore, represent the interquartile range IQR. The thick horizontal line represents the median or 50th percentile. In general, box plots provide compact and unique visualization of the distribution for a given style of plotting parameters in contrast somewhat to histograms.

Several interpretations of the box plots can be made. For the box plots, ozone has the largest relative variation and appears positively (right) skewed towards large values. Temperature has the smallest relative variation and is nearly symmetrical—perhaps the symmetry is caused in part by the diurnal (daily) heating and subsequent cooling of the land surface. ◀

Because box plots provide a more succinct graphical depiction of the distribution of the data relative to histograms, box plots are highly recommended and are preferable to histograms. Conventional box plots, such as those in figure 1.2, are preferable because:

1. Arbitrary bins are not used—the visual impact is affected by the width of the bins,
2. The plots locate the distribution on the real number line (like a histogram),
3. The plots quantitatively depict the statistics such as the median, mean, and quartiles,
4. The plots often depict the relative lengths of the left and right tails of the distribution with greater visual precision than histograms,
5. The plots can specifically depict minimum and maximum of the sample, and
6. The plots can be configured to represent individual outliers (a feature not used in figure 1.2).

The utility of graphical depiction of distributions—both parent or theoretical and sample—cannot be stressed enough. The R environment provides powerful graphical features and visualization of data to aid in interpretation of phenomena under study. Many plot styles of distributions are illustrated in the figures of this dissertation.

Chapter 2
Distributional Analysis

In this chapter, I present an introduction to distributional analysis by covering what I believe are some of the most isolatable components of the dissertation. These components include definition of distribution functions, basic summary statistics, fitting a distribution to the sample moments, plotting positions, and demonstration of two common distributions to real-world data. Readers are advised to thoroughly understand much of the material from this chapter with the exception of the algebra of quantile functions. Such understanding will fulfill many prerequisites needed to understand this dissertation and perform distributional analysis with L-moment statistics using R.

2.1 A Review of Continuous Random Variables and Distributions

Univariate data typically are generated from measurement methods of finite resolution that are effectively continuous on the real-number line $\mathbb{R}$. For purposes of this dissertation, it is reasonable to treat data as coming from a **continuous random variable** as opposed to a **discrete random variable**. By convention, the symbol X or Q generally is used to denote the random variable, and x will be used in reference to realizations or sample values of the variable. Specific random samples will be identified by notation such as x_i and $x_{i:n}$ for the "ith sample" and "ith-largest sample of size n," respectively.

It is important to remark about the concept of "random." For many data examples, it is sometimes obvious or generally understood that the data do not originate from a purely random process in a statistical sense. However, the tools for distributional analysis never-the-less remain useful. In some circumstances, distributional analysis provides a convenient means to "fit" nonlinear functions (functions that are probability distributions for the context here) to data. An example of data that do not originate from a purely

random process is the sample distribution of air temperature depicted in the box plot in figure 1.2 of the previous chapter. The hour-to-hour, day-to-day changes in air temperature would not be expected to originate from a purely random process. For these data, the air temperature data likely have considerable **serial correlation** in time. The box plot, however, still provides quantitative information about the distribution of air temperature during the monitored time period.

Particular phenomena, such as earthquake magnitude, reside in a strictly positive domain. This fact does not pose an especially complex situation for distributional modeling, but unique problems to such "bounded" data do arise. For now, it is sufficient to understand that awareness of the physical meaning of data can be useful as part of distributional analysis. Some distributions described herein are bounded and can be specified to honor specific numerical bounds such as the Generalized Pareto. Whereas boundedness might seem an appropriate piece of information to bring to problems within the context of distributional analysis, the specific nature of the analysis might advise against the practice of honoring theoretical or physical bounds.

Other phenomena can acquire exactly zero values, such as streamflow for a generally dry wash in the American southwest. Sometimes, special accommodation is needed for zero magnitude values using conditional probability techniques.

Three types of expressions for the distribution of a random variable are common. These are the probability density function, cumulative distribution function, and quantile function (Ross, 1994; Evans and others, 2000; Gilchrist, 2000). These functions are described in sections that follow. A comprehensive summary of built-in R support for probability distributions is found in Venables and others (2008, chap. 8).

2.1.1 Probability Density Functions

The **probability density** is a concept in which the probability $\Pr[\,]$ of any given value of a continuous random variable X is zero. The probability is zero because there are infinitely many numbers infinitely close to the value x. So the probability at a given value x for the variable is specified by the concept of density. The **probability density function** (PDF, $f(x)$) is defined by

$$f(x)\,\mathrm{d}x = \Pr[x \leq X \leq x + \mathrm{d}x] \tag{2.1}$$

Other than depicting qualitative information about the structure of the probability density, PDFs sometimes have more restricted usefulness compared to the two other types of functions described in sections that follow. The usefulness is restricted because the numerical values of cumulative probability or nonexceedance probability are not available (only probability density is), and in practice numerical values of probability often are needed.

USING R ——— ——— USING R

An example PDF for illustration is shown in figure 2.1 for a Weibull distribution with a specified shape parameter. The Weibull PDF in the figure was created by example 2–1.

2–1

```
#pdf("pdf1.pdf")
x <- seq(0,3, by=0.01)
f <- dweibull(x, shape=1.5) # prepended "d" to "weibull" dist.
plot(x,f, type="l")
#dev.off()
```

The example produces a vector of x values from 0 to 3 in increments of 0.01 using the `seq()` function, the vector is passed to the PDF of the Weibull distribution by the built-in `dweibull()` function. In the R language, the `plot()` function produces the graphic shown in figure 2.1. The `type="l"` named argument produces a line plot. The shape parameter of the distribution is set by a named argument `shape=1.5`. The other built-in distributions use a similarly named-argument interface. The letter "`d`" (density) is prepended to the name or an abbreviation of the distribution for at least the distributions built-in to R. ◀

2.1.2 Cumulative Distribution Functions

The **cumulative distribution function** (CDF) is defined as

$$F(x) = \Pr[X \leq x] \tag{2.2}$$

where F is **nonexceedance probability** $0 \leq F \leq 1$ for value x of random variable X. The equation is to read "the probability that random variable X is less than or equal to x." The CDF is a nondecreasing function that defines the relation between F and x. The

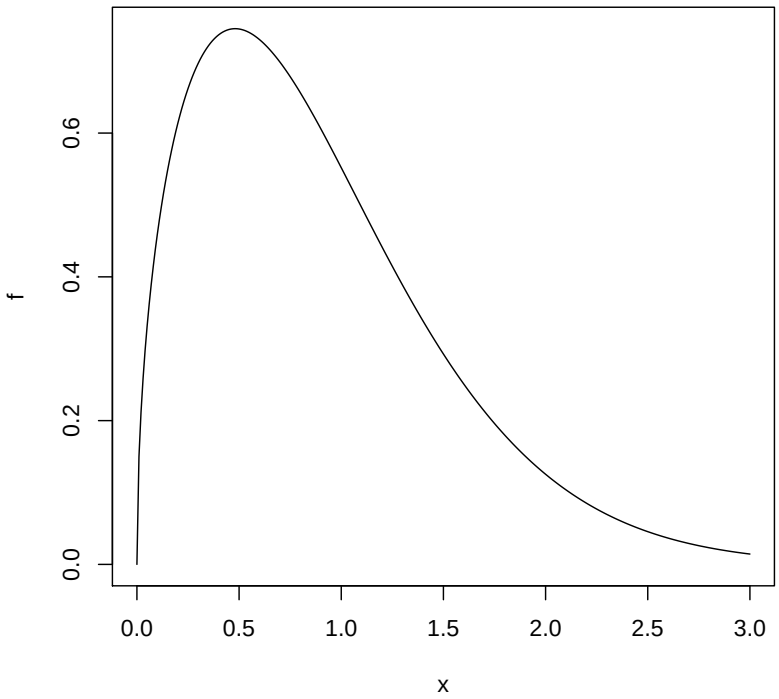

Figure 2.1. Probability density function for a Weibull distribution from example 2–1

derivative of the CDF or $f(x) = \mathrm{d}F/\mathrm{d}x$ is the probability density so the CDF in terms of the PDF is the integral

$$F(x) = \int_{-\infty}^{x} f(t)\,\mathrm{d}t \tag{2.3}$$

CDFs are common in fields for which values for the random variable X are thought of as "independent," and thus graphical depictions of CDFs often have x on the horizontal axis and F on the vertical axis. Practioners frequently "enter" problems from the perspective that measurement of a phenomena has been made and a mapping to the cumulative probability or percentile is needed. For example, a baby boy weighs 20 lbs (9 kg) at four months, is this a large or small baby? The CDF of the weights of baby boys with ages of approximately four months would hold the answer. Partly for convenience, therefore, CDFs are common in some disciplines—meaning in graphical contexts that x conventionally is depicted on the horizontal axis and F is on the vertical axis.

Also, in some disciplines the CDF is replaced by a function known as the **survival function** (other names are: **complementary** CDF, **reliability function**, and **survivor function**) $S(x)$. The function is an expression of **exceedance probability** or

$$S(x) = \Pr[X > x] \tag{2.4}$$

where S denotes exceedance probability. The relation between the $F(x)$ and $S(x)$ is straightforward: $S(x) = 1 - F(x)$. In this dissertation, however, $S(x)$ do not have a central role and purposeful preference to $F(x)$ usually is made.

USING R ——— ——— USING R

An example CDF for illustration is shown in figure 2.2 for standard Normal distribution, which is a Normal distribution that has a mean of zero and a standard deviation of 1. The figure was created by example 2–2. The `pnorm()` function is the CDF of the Normal distribution, which defaults to the standard Normal if no other arguments are provided.

2–2

```
#pdf("cdf1.pdf")
x <- seq(-3,3, by=0.01)
F <- pnorm(x)
plot(x,F, type="l")
#dev.off()
```

◀

A followup to example 2–2 is 2–3 that shows how the mean and standard deviation are set with the `pnorm()` function using the named arguments `mean` and `sd`. The mean is set to -600 and the standard deviation is set to 400. The F value for $x = -300$ is about 0.77 (the 77th percentile).

2–3

```
mu  <- -600 # mean
sig <-  400 # standard deviation
myF <- pnorm(-300, mean=mu, sd=sig)

print(myF)
[1] 0.7733726
```

Readers are asked to note in examples 2–2 and 2–3 the use of the "`p`" (probability or percentile) in the name of the `pnorm()` function to call the respective CDF. The letter "`p`" is prepended to the name or an abbreviation of the distribution for at least the distributions built-in to R. ◀

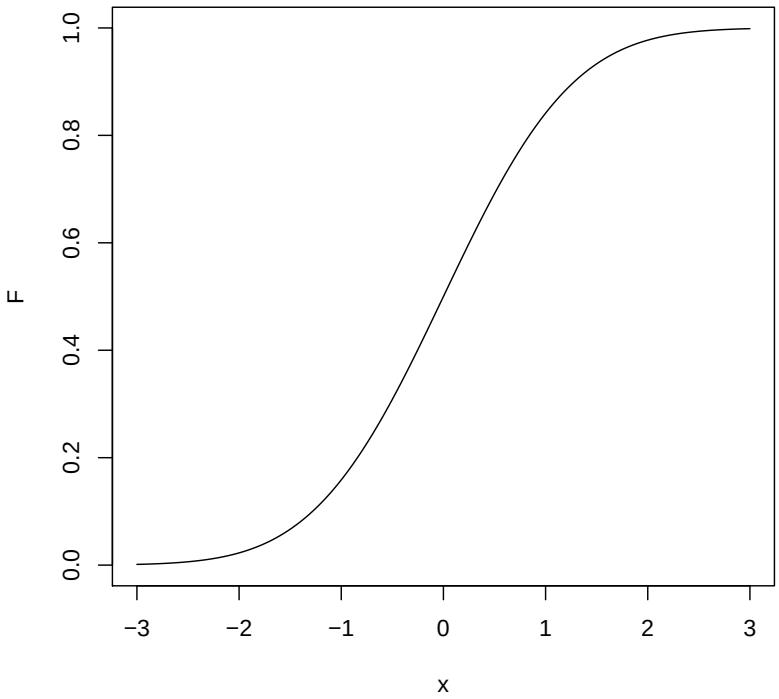

Figure 2.2. Cumulative distribution function for standard Normal distribution from example 2–2

2.1.3 Hazard Functions

A special function related to distributions is the **hazard function**, which is potentially less commonly referred to as the **failure rate function**. Hazard functions are particularly useful in distributional analysis involving life time data, such as the distribution of the life span of a person or a part. The hazard function $h(x)$ can be expressed in terms of the PDF and CDF for random variable X (usually time). The function is defined by

$$h(x) = \frac{f(x)}{1 - F(x)} \tag{2.5}$$

where $f(x)$ is a PDF and $F(x)$ is the CDF. It is important to stress that $h(x)$ is not an expression of probability.

To help with intuitive understanding of what $h(x)$ means (Ugarte and others, 2008, p. 143), let $\mathrm{d}x$ represent a small unit of measurement. The quantity $h(x)\mathrm{d}x$ then can be conceptualized as the approximate probability that random variable X takes on a value in the interval $[x, x+\mathrm{d}x]$ or the approximate probability $\Pr[\,]$

$$h(x)\mathrm{d}x = \frac{f(x)\mathrm{d}x}{1 - F(x)} \approx \Pr[X \in (x, x+\mathrm{d}x) \mid X > x] \tag{2.6}$$

Ugarte and others (2008, p. 144) continue by stating that $h(x)$ represents the instantaneous rate of death or failure at time x, given that survival to time x has occurred ($\mid X > x$). Emphasis is repeated that $h(x)$ is a rate of probability change and not a probability itself.

USING R ——————————————— ——————————————— USING R

The *lmomco* package provides the `hlmomco()` function, which computes eq. (2.5) using the `dlmomco()` (PDF) and `plmomco()` (CDF) functions. Mimicking the example by Ugarte and others (2008, p. 144), the failure rate for an Exponential distribution is a constant as example 2–4 shows. A vector of repeated failure rates equal to 0.01 is shown and when inverted by $1/0.01$, the scale parameter of 100 in `my.lambda` is recovered.

2–4

```
my.lambda <- 100 # scale parameter of Exponential dist.

# set a list of parameters for the Exponential distribution
para <- vec2par(c(0,my.lambda), type="exp") # used by lmomco

x <- 50:60 # sequence of 50 to 60 by increments of 1
hlmomco(x,para) # returns vector of repeated 0.01 values
[1] 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01
```

◀

2.1.4 Quantile Functions

A **quantile distribution function** (QDF, often simply a **quantile function**) provides an alternative means of defining a distribution. Gilchrist (2000) provides a focused and outstanding treatment of QDFs. A QDF is defined as

$$x(F) = x_F = \text{the value } x \text{ for which } \Pr[X \leq x_F] = F \tag{2.7}$$

where x_F could be referred to as the F-quantile of the distribution. The notation $x(F)$ or $Q(F)$ typically will be used to refer a QDF as a whole. Generally, the x_F notation refers to specific **quantiles** such as $x(0.50)$ or $x_{0.50}$. This quantile is the median or 50th percentile. The CDF and QDF are inverses of each other, and in fact within some disciplines, the term

inverse distribution function is the term used when referring to the QDF. This term is not used herein. Notationally the following holds

$$g = F[x(g)] \quad \text{or} \quad x(F) = F^{(-1)}(x) \tag{2.8}$$

for a nonexceedance probability g, the CDF $F(x)$, and the QDF $x(F)$. The superscripted $^{(-1)}$ notation of a QDF (inverse of the CDF) is seen in some publications and only rarely used here.

QDFs are common in fields, such as hydrology, for which values for the random variable X are unknown, but concepts such as risk are thought of as "independent." Thus, graphical depictions of QDFs often have F on the horizontal axis and x on the vertical axis. Practioners frequently enter their problems from the perspective that a cumulative percentile or nonexceedance probability is a known quantity. For example, suppose that a government requires levees to be built for the 99.9th-percentile storm. The QDF of storms for the geographic region under consideration would hold the answer.

In terms of the exploration of distributional properties and broader distributional analysis, working with QDFs generally provides for easier programming because values of F are defined on a precisely constrained interval as $0 \leq F \leq 1$, whereas the range of x is distribution specific and exists in arbitrary portions of (or even the entire, $-\infty < x < \infty$) real-number line $\mathbb{R}$. Emphasis is made, however, that the functionality of R makes working with either PDF, CDF, and QDF operations not particularly burdensome. The analyst, when using R, has freedom to choose the syntax that is most natural for the problem at hand.

The **sample quantile function** $\hat{X}(F)$ can be defined as

$$\hat{X}(F) = x_{\lfloor nF \rfloor : n} \tag{2.9}$$

where F is nonexceedance probability, $\lfloor a \rfloor$ is the floor function and $x_{i:n}$ is the ith sample order statistic (see Chapter 3). The floor function is implemented in R by the `floor()` function.

USING R

An example QDF for illustration is shown in figure 2.3 for a scaled (`2*qexp`) and shifted (`+10`) distribution that is Exponential (note the use of "`q`" for quantile, `qexp()`). The letter "`q`" is prepended to the name or an abbreviation of the distribution for at least

the distributions built-in to R. The figure is created in example 2–5 in which the `seq()` function is used to generate a sequence of F values on a $\mathrm{d}F = 0.01$ interval.

2–5

```
#pdf("qdf1.pdf")
F <- seq(0.01,0.99, by=0.01) # nonexceedance probability
x <- 2*qexp(F) + 10 # exponential distribution quantiles
plot(F,x, type="l")
#dev.off()
```

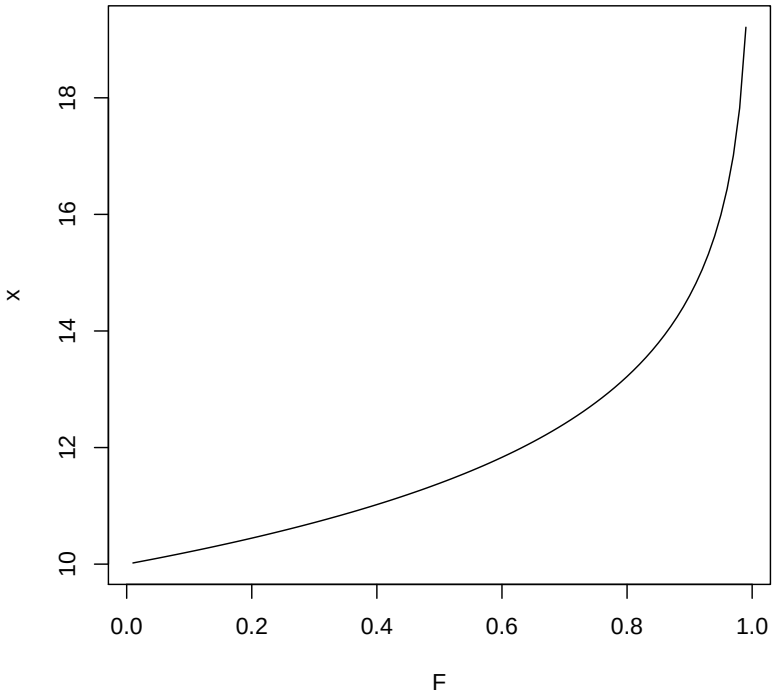

Figure 2.3. Quantile function for an Exponential distribution from example 2–5

◀

Another example is example 2–6, which shows that eq. (2.8) is correct using the choice of the Gamma distribution. The output of the two `print()` functions shows that `g.A` is numerically equivalent to `g.B`. The `qgamma()` and `pgamma()` functions are inverses of each other and represent the **inverse transform method** (Rizzo, 2008, p. 49).

2–6

```
g.A <- 0.76 # nonexceedance probability
x <- qgamma(g.A,4, scale=3) # the quantile at that prob.
print(x) # output the quantile
```

```
[1] 15.55525
g.B <- pgamma(x,4, scale=3) # invert the quantile function
print(g.B) # the result, which is a nonexceedance prob.
[1] 0.76
```

◀

To complete this section concerning QDFs, consider again the definition of probability density

$$f(x) = \lim_{|x_{n+1}-x_n|\to 0} \frac{F(x_{n+1}) - F(x_n)}{x_{n+1} - x_n} = \frac{\mathrm{d}F}{\mathrm{d}x} \tag{2.10}$$

or in other words, the change in probability per unit change in x. This differencing equation will be used in example 2–7, and the utility of using R for statistical computing is further shown.

Equation (2.10) provides a recurrence relation to solve for the QDF, which is a useful construct when the QDF does not have an analytical solution. The recurrence relation is

$$x_{n+1} = x_n + \frac{F(x_{n+1}) - F(x_n)}{f(x_n)} \tag{2.11}$$

where the quantile for a nonexceedance probability F can be computed using the CDF $F(x)$ and PDF $f(x)$, and this is done in example 2–7 for $F = 0.2$ for a Pearson Type III distribution (see page 243).

2–7

```
"qua.by.recursion" <-
function(F, para, x, eps=1e-8, ...) {
  Fx <- plmomco(x, para, ...) # CDF of the lmomco package
  tmp <- F - Fx # compute once, use twice
  if(abs(tmp) < eps) { # very close in probability
    names(x) <- NULL
    return(x) # stop recursion and return
  } else {
    fx <- dlmomco(x, para, ...) # PDF of the lmomco package
    newx <- x + tmp/fx # as seen in the equation
    x.np1 <- qua.by.recursion(F, para, newx, eps=eps, ...)
    return(x.np1)
  }
}
# Set some parameters of the Pearson Type III distribution
# in the fashion of the lmomco package
para <- vec2par(c(1000,900,1.2), type="pe3")
# Compute 20th percentile by guess of 1000 (mean) for F=0.2
qua.by.recursion(0.2, para, 1000)
```

```
[1] 240.6772

# QDF of PE3 distribution, uses qnorm() function
qlmomco(0.2, para)
[1] 240.6772
```

In the example, the `vec2par()` function sets the first three product moments of the Pearson Type III as $\mu = 1000$, $\sigma = 900$, and $\gamma = 1.2$, respectively. A first guess of the solution is made, the guess is $1{,}000$, which is the mean. The output shows that $\text{PE3}(F{=}0.2, 1000, 900, 1.2) = 240.7$ from both functions, although the internal algorithms differ. The "`...`" is a separate argument and represents additional and arbitrary arguments that are to be passed to other functions, which in this case are called inside the `qua.by.recursion()` function. The internally called functions are `plmomco()` and `dlmomco()`. ◀

2.1.5 The Algebra of Quantile Functions

Gilchrist (2000, pp. 62–67) provides a summary of mathematical rules or algebraic operations that can be performed with QDFs. These are summarized and demonstrated in this section. Quantile function algebra facilitates the construction of new distributions from existing distributions, and the algebra is readily implemented using R.

In particular, the *vectorized* arithmetic of R, which is one of its most attractive features, facilitates QDF algebra. With relatively few commands and hence keystrokes, many types of statistical operations can be performed. The compact syntax of R facilitates the use of QDF algebra. For example, several properties of QDFs are easily combined to create new distributions with syntactically clear code—clear and concise code is a characteristic of maintainable software.[1]

The Addition Rule

The distributions $Q_1(F)$ and $Q_2(F)$ can be added:
$Q(F) = Q_1(F) + Q_2(F)$.

[1] The author argues that modularity that enhances development of reusable code units has a higher level of importance for maintaining code, but further discussion is beyond the scope here.

The Addition Rule is easily demonstrated. The $Q(F)$ for the addition of an Exponential and a Normal distribution is produced by example 2–8.

2–8
```
F  <- seq(0.5,0.62, by=0.04) # a narrow range of F values to show
Q1 <- qexp(F, rate=1/30)
Q2 <- qnorm(F, mean=10, sd=100)
Q  <- Q1 + Q2 # Addition Rule

cbind(round(Q1,1), round(Q2,1), round(Q,1))
     [,1] [,2] [,3]
[1,] 20.8 10.0 30.8
[2,] 23.3 20.0 43.3
[3,] 26.0 30.2 56.2
[4,] 29.0 40.5 69.6
```

The `cbind()` function binds a list of vectors into columns and the `round()` function rounds each element of a vector to one digit to the right of the decimal in the example. The example shows the $x(F)$ values as a matrix. The 50th percentile or median of $Q(F)$ or $Q(0.50)$ can be written as $Q_{0.50} = 30.8$.

The Multiplication Rule for strictly positive variables

If each is strictly positive, the distributions $Q_1(F)$ and $Q_2(F)$ can be multiplied: $Q(F) = Q_1(F) \times Q_2(F)$.

The Multiplication Rule also is readily demonstrated and is shown in example 2–9. The product $Q(F)$ of the same two distributions (Q_1 and Q_2) from example 2–8 is

2–9
```
Q <- Q1*Q2 # Multiplication Rule

rbind(round(Q1,1), round(Q2,1), round(Q,1))
      [,1]  [,2]  [,3]   [,4]
[1,]  20.8  23.3  26.0   29.0
[2,]  10.0  20.0  30.2   40.5
[3,] 207.9 466.9 785.7 1177.0
```

where the third row is the product $Q(F)$. The `rbind()` function is used and binds a list of vectors into rows. To clarify the effects of `cbind()` and `rbind()`, readers are asked to compare the orientation of the matrices in examples 2–8 and 2–9.

The Intermediate Rule

The distributions $Q_1(F)$ and $Q_2(F)$ can be mixed:
$Q(F) = wQ_1(F) + (1 - w)Q_2(F)$ for $0 \leq w \leq 1$. As a result,
$Q(F)$ will lie between the two original distributions.

The Intermediate Rule provides for blending of two QDFs together. Such blending could be part of a more intricate model building process. One reason for blending two QDFs together might be to achieve better performance or fit in the far left and right tails than can be achieved by either distribution alone. It might be favorable to have a model that can be tuned somewhat for tail behavior. The Intermediate Rule is demonstrated in example 2–10, and the results are shown in figure 2.4.

2–10

```
F  <- seq(0.001,0.999, by=0.001)
w  <- 0.35 # weighting
Q1 <- qweibull(F, shape=3, scale=100) # Weibull dist. quantiles
Q2 <- 25*qexp(F) # exponential distribution quantiles
Q  <- w*Q1 + (1-w)*Q2 # Intermediate Rule

mylo <- min(Q1, Q2)
myup <- max(Q1, Q2)

#pdf("qdf2.pdf")
plot(F,Q1, type="l", ylim=c(mylo,myup),
     ylab="QUANTILE",
     xlab="NONEXCEEDANCE␣PROBABILITY")
lines(F,Q2, lty=2)
lines(F,Q, lwd=3)
legend(0,150,c("Q1", "Q2", "Q"), lty=c(1,2,1), lwd=c(1,1,3))
#dev.off()
```

In the example, a high-resolution sequence of F values is first produced, and the weight `w` is set to 0.35. Second, `Q1` and `Q2` are created as numerical curves of the two quite different QDFs—a two-parameter Weibull and a one-parameter Exponential distribution. Third, the two curves are blended by the weight `w` to form `Q`. Example 2–10 finally is completed by using three graphic functions (`plot()`, `lines()`, and `legend()`) to generate figure 2.4. The `legend()` function creates the legend or explanation in the plot, and the function can take many arguments to control legend appearance and position.

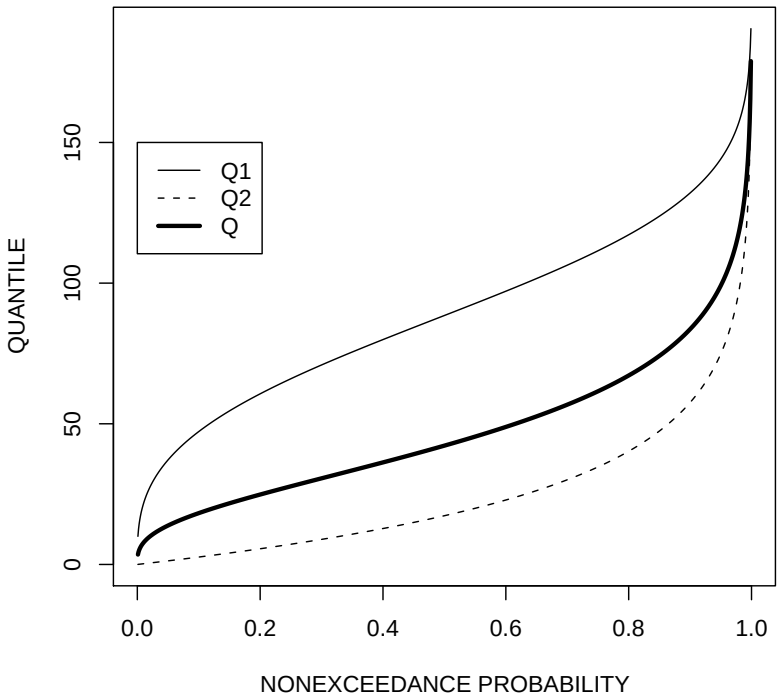

Figure 2.4. Blending two quantile functions to form a third by the Intermediate Rule from example 2–10

The Reflection Rule

The distribution $Q(F)$ can be reflected about $x = 0$ and $F = 0.5$ by $-Q(1 - F)$, which also is the distribution of the random variable $-Q$.

The Exponential distribution is selected for a demonstration of the Reflection Rule in example 2–11. A convenient vector of F values is again generated by the `seq()` function. A variable `myna` is given canonical missing value (`NA`) by using the `as.character()` function. The `myna` variable will be used to "trick" R into "lifting-the-pen" to accommodate a single call to the `plot()` function for convenience. The Q vector contains the unreflected Q_1 distribution and the reflected Q_2 distribution. The two distributions are juxtaposed in figure 2.5.

2–11

```
F  <- seq(0.01,0.99, by=0.01)
myna <- as.character(NA)
Q1 <-  qexp(  F, rate=1/100) # top curve
```

```
Q2 <- -qexp(1-F, rate=1/100) # Reflection Rule, bottom curve
F  <- c( F, myna,  F)
Q  <- c(Q1, myna, Q2)
#pdf("qdf3.pdf")
plot(F, Q, type="l")
#dev.off()
```

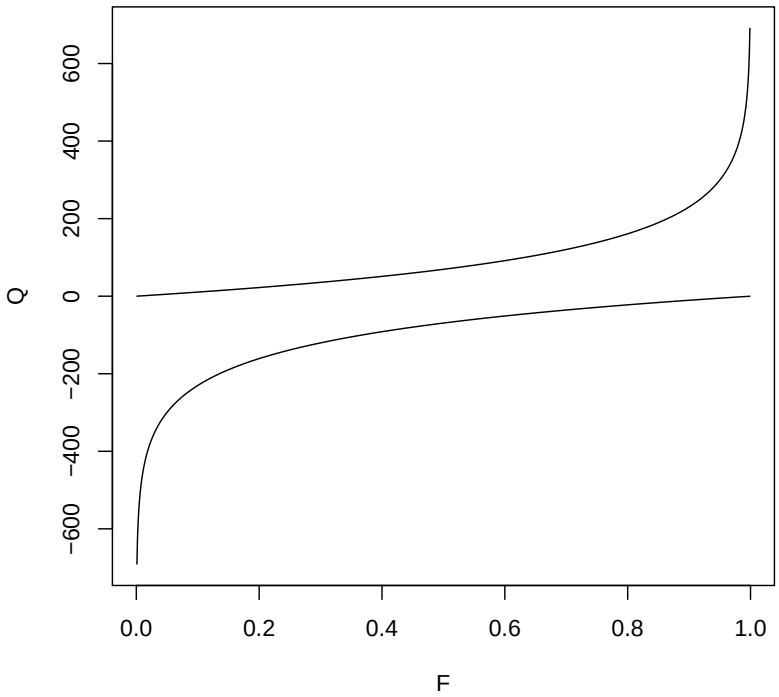

Figure 2.5. Reflection (bottom) of an Exponential distribution (top) about $x = 0$ and $F = 0.5$ using the Reflection Rule from example 2–11

The Reciprocal Rule

The quantile function for the reciprocal $1/Q$ of random variable Q is $1/Q(1 - F)$.

The Reciprocal Rule, through the use of the vectorized arithmetic of R, is easily applied. Example 2–12, without further explanation, would suffice for a quantile function $Q(F)$ and nonexceedance probability F.

2–12

```
newQ <- 1 / Q(1-F) # Reciprocal Rule
```

The F-transformation Rule (transformation of probabilities)

If $H(F)$ is a monotonic increasing function of F, but not necessarily a QDF, but is non-decreasing on the interval $0 \leq F \leq 1$ and is standardized so that $H(0) = 0$ and $H(1) = 1$, then operation $Q[H(F)]$ produces a distribution with the same range as $Q(F)$.

Consider a candidate transform function $H'(F) = F^2 + 2F + 1$ for demonstration of the F-transformation Rule. This function can be standardized to a new function $H(F)$ such that $H(0) = 0$ and $H(1) = 1$ by

$$H(F) = \frac{H'(F) - H'(0)}{H'(1) - H'(0)}$$

or

$$H(F) = \frac{F^2 + 2F + 1 - 1}{4 - 1} = \frac{F^2 + 2F}{3}$$

Some functions for generalized application of the F-transformation Rule are now created. Example 2–13 is used to define a transformation function `HF()` that is structurally similar to $H(F)$ just described.

2–13

```
"HF" <- function(F) {
  return(600*F^2 + 20*F + 100)
}
```

Next in example 2–14, a function is created titled `Ftrans()` for actual implementation of the F-transformation Rule. The function receives a vector of F values and the transformation function as a named argument (`transfunc=NULL`). The `Ftrans()` function uses the `check.fs()` function to verify that the F values are $0 \leq F \leq 1$ and return `FALSE` if they are not. A test is made on the argument `transfunc` by the `is.null()` function and return `FALSE` if the argument is not provided. The standardization of the transform function is set into the variable `nf`. Finally, a check whether `any()` of the `nf` values are less than 0 or greater than 1 is made as a precaution against a poorly specified transformation function.

2–14

```
"Ftrans" <- function(F, transfunc=NULL) {
  if(! check.fs(F)) return(FALSE)
  if(is.null(transfunc)) {
    warning("a␣transformation␣function␣is␣required")
    return(FALSE)
  }
  nf <- transfunc(F) - transfunc(0) / # the standardization
       (transfunc(1) - transfunc(0))  # to a 0 to 1 interval
  if(any(nf < 0)) {
    warning("transformed␣value␣less␣than␣zero---revise␣transform␣
       function")
    return(FALSE)
  }
  if(any(nf > 1)) {
    warning("transformed␣value␣greater␣than␣one---revise␣
       transform␣function")
    return(FALSE)
  }
  return(nf)
}
```

The demonstration of the F-transformation function `Ftrans()` is provided in example 2–15, and the results are graphically depicted in figure 2.6.

2–15

```
F <- nonexceeds() # convenient nonexceedance probabilities
#pdf("FtransC.pdf")
plot(F,Ftrans(F,trans=HF), type="l")
#dev.off()
```

Finally, the F-transformation Rule is demonstrated in example 2–16, in which the parameters for the Generalized Pareto distribution are set by the `vec2par()` function. The `qlmomco()` function provides the QDF of the Generalized Pareto. The `plot()` function plots the distribution without the F-transformation. The example ends with a second plotting of the QDF of the Generalized Pareto by the `lines()` function, but this time the F values are transformed beforehand by the `Ftrans()` function. Both distributions are plotted in figure 2.7.

2–16

```
#pdf("FtransD.pdf")
PARgpa <- vec2par(c(-400,100,-0.2), type="gpa")
plot(F,qlmomco(F,PARgpa), type="l")
nf <- Ftrans(F,trans=HF)
lines(F,qlmomco(nf,PARgpa), lty=2)
#dev.off()
```

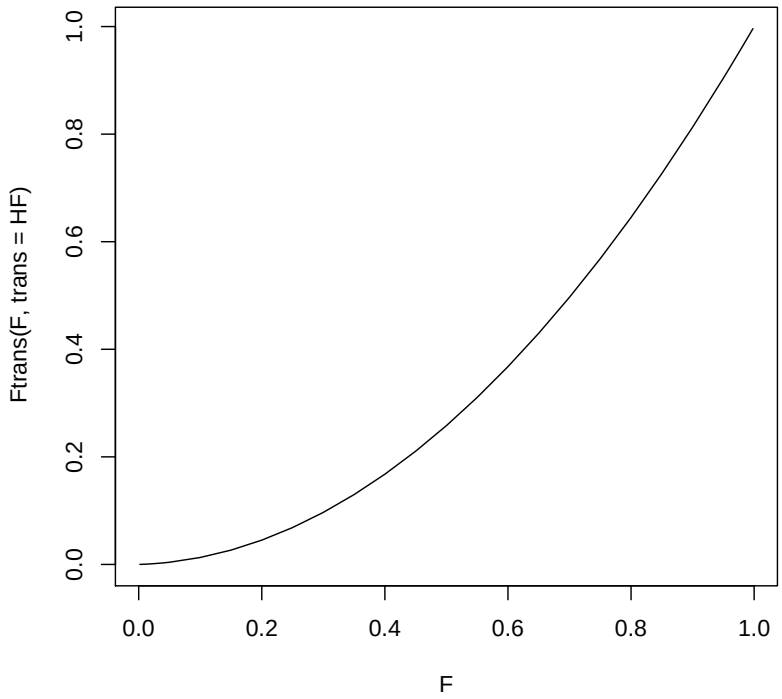

Figure 2.6. F-transformation function from example 2–15

The Q-transformation Rule (transformation of quantiles)

If $z = T[x]$ is monotonic increasing function, then $T[Q(F)]$ is a quantile function.

The Q-transformation Rule is demonstrated in example 2–17. The example uses a transformation function similar in structure to that shown in example 2–13.

2–17

```
"Tx" <- function(x) {
  return(0.1*x^2 + x + 100)
}
F <- nonexceeds()
PARgpa <- vec2par(c(78,100,0.2), type="gpa")

Q1 <-   par2qua(F,PARgpa)  # original GPA distribution
Q2 <- Tx(quagpa(F,PARgpa)) # Tx as just defined

summary(Q1)
  Min.  1st Qu.  Median    Mean 3rd Qu.    Max.
  78.20   93.99  142.70  182.10  235.90  433.70
```

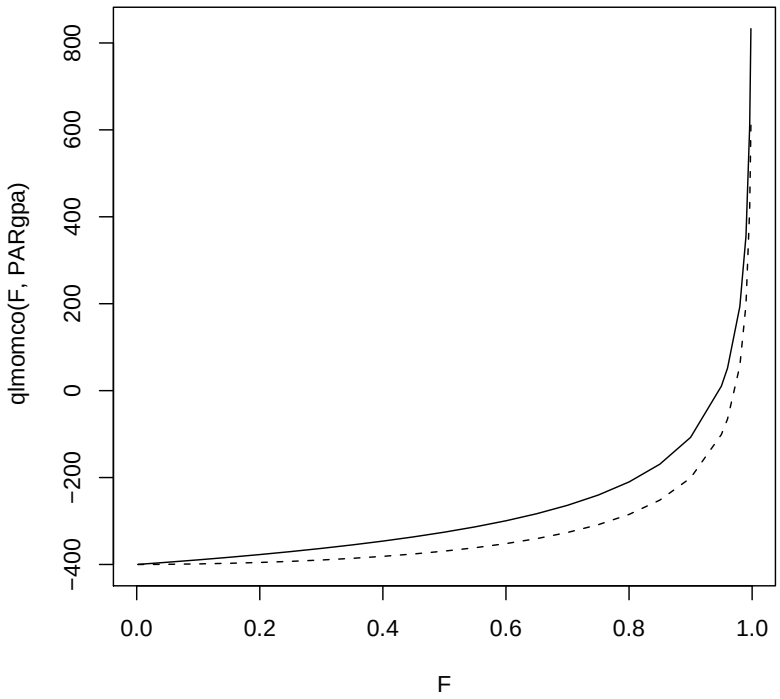

Figure 2.7. Comparison of original Generalized Pareto distribution (solid line) and F-transformed Generalized Pareto distribution (dashed line) from example 2–16

```
summary(Q2)
  Min.  1st Qu.  Median    Mean 3rd Qu.    Max.
  789.7  1077.0  2280.0  4754.0  5899.0 19350.0
```

The Standardization Rule

If a distribution $R(F)$ has a location parameter such as the median or mean equal to zero and some linear measure of dispersion or scale is unity, then the distribution $Q(F) = \eta + \psi R(F)$ has a location parameter η and scale parameter ψ.

The Standardization Rule is straightforward and is shown in example 2–18. In the example, the location and scale parameters of the distribution are set into the variables `eta` and `psi`. A random sample from a standard Normal distribution of $n = 4{,}000$ is drawn and set into `RF`. Conversion of `RF` to `QF` is made, and the mean and standard deviation statistics are computed and shown.

```
eta <- 62; psi <- 56
RF  <- rnorm(4000, mean=0, sd=1) # standard Normal distribution
QF  <- eta + psi*RF # Standardization Rule

mean(QF) # estimate location parameter
[1] 62.19526
sd(QF) # estimate scale parameter
[1] 55.86824
```

2.1.6 Exceedance Probability and Recurrence Interval

It has been tacitly assumed that readers are versed in the basic concepts of probability (Ross, 1994; Baclawski, 2008; Ugarte and others, 2008). By convention for this dissertation, probability is considered in terms of F for consistency. Other practitioners, however, based on personal preference or tradition within a discipline, might use exceedance probability S. The relation between the two is trivial: $S = 1 - F$.

In some disciplines, including those such as hydrology, meteorology, geophysics, volcanology, and the supporting disciplines of engineering such as civil, earthquake, and structural, the concept of probability often is expressed as **recurrence interval** or **return period** in the parlance (Stedinger and others, 1993). Phrases such as the "100-year flood" or "50-year storm surge" are common and have found a firm place in the regulatory context of government and general media. In the author's opinion, events of T-year (x_T) recurrence intervals are more precisely described by their **annual nonexceedance probability**. (Despite misgivings, the author, however, still frequently uses the recurrence interval for much of his professional communication.) The conversions between the two are straightforward and are

$$T = 1/(1 - F) \tag{2.12}$$

$$F = 1 - 1/T \tag{2.13}$$

The discussion of recurrence interval explicitly is made because many examples here involve annually sampled data. These data are derived on an annual basis, that is, data that are measured on annual intervals. Such data might represent the coldest daily temperature for each year or represent the number of frost-free days per year. A T-year recurrence interval is valid as an expression of probability for such data—although confusion

regarding recurrence interval interpretation is common. Conversationally, using the term "T-year recurrence interval" is convenient.

Contemporary understanding of recurrence interval by the general populace (nonscientists, nonengineers, or nonstatisticians) is incomplete at best and misinformed at worst. The T-year recurrence interval often is misunderstood as implying that one and only one T-year event will occur in an interval of T years. Further, large T-year events, such as the 100-year event, often are misunderstood as some sort of physical upper limit—the phenomena cannot be greater than the 100-year event. Unfortunately, the $F = 0.99$ event is seldom near or otherwise should be interpreted as representing an approximate upper bounds for many natural phenomena such as flood magnitude.

There are two correct interpretations (Stedinger and others, 1993, p. 18.3) of recurrence interval if events are independent from year to year. First, the expected number of exceedances of x_T (the T-year event) in a fixed T-year period is equal to 1 or alternatively an event greater than the x_T event will occur once every T years. Second, the distribution of success or failures of exceedance above the x_T threshold is Geometric with mean $\mu = 1/(1 - F)$. The CDF of the Geometric distribution is

$$\Pr[\text{exactly } k \text{ years until } X \geq x_F] = F^{k-1}(1 - F) \tag{2.14}$$

Thus, another interpretation of recurrence interval is that the interval is the average time until x_T is exceeded.

USING R

USING R

The conversion between T-year recurrence interval and F can be performed in R using descriptively named functions as example 2–19 demonstrates. In the example, the `prob2T()` and `T2prob()` functions are used. The example shows that the 75th percentile of annual data is the 4-year recurrence interval and visa versa. The `"\n"` value ("newline" character) is a universal symbol used to create a newline in many programming languages including R.

2–19

```
F <- 0.75 # 75th percentile
T <- prob2T(F); G <- T2prob(T)
cat(c("RI = ", T, " and computed F=", G, "\n"), sep="")
RI =  4  and computed F= 0.75
```

◀

The Geometric distribution is a built-in distribution of R. For example, the CDF of the Geometric is provided by the `pgeom()` function. Example 2–20 shows that the probability of experiencing or witnessing at least one 100-year event ($F = 0.99$ and $S = 1 - 0.99 = 0.01$) in 100 years (an exceptionally long life time) is about 63.8 percent according to the assumptions leading to use of the Geometric distribution and not 100 percent as, by the author's professional experience, the general populace often appears to assume.

2–20

```
pgeom(100,0.01) # nonexceed prob. of one 100-year event in
# exactly 100 years
[1] 0.637628
```

In conclusion, for distributional analysis of rare events in disciplines in which terms such as the T-year event are used, care is suggested when reference to, or expression of, probability as a T-year recurrence interval is made. ◀

2.2 Basic Summary Statistics and Distributional Analysis

The general properties of probability distributions are described at the beginning of this chapter, and the examples shown thus far provide some details of distributional analysis. Further, several useful functions for generic programming operations are used. Preparation for more expansive consideration of distributional analysis has been made. In this section, a segue is made into basic summary statistics and using these in some introductory distributional analysis.

2.2.1 Basic Summary Statistics

Basic summary statistics for a data sample $x_1, x_2, \cdots, x_n$ are straightforward to compute and are shown in examples in this section. Formal definitions for some of these statistics (mean and standard deviation) are deferred until Chapter 4.

Basic summary statistics include the mean μ; median $x_{0.50}$; lower and upper quartiles $x_{0.25}$ and $x_{0.75}$, respectively; and the minimum $x_{\min}$ and maximum and $x_{\max}$, respectively. Another basic summary statistic is the standard deviation σ. Each of these statistics are readily computed using R by the `summary()` and `sd()` functions.

In example 2–21, a simulated data sample of size $n = 100$ is simulated or "drawn" from a standard Normal distribution using the `rnorm()` function.

2–21
```
fake.dat <- rnorm(100) # 100 standard normal deviates
summary(fake.dat) # summary(rnorm(100)) using vectorized notation
    Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
-2.38100 -0.64010 -0.05406 -0.10800  0.52350  1.90100
sd(fake.dat)
[1] 0.917965
```

The summary statistics in the example show that 25 percent of the data are less than about -0.64, whereas about 50 percent of the data are less than about -0.05, and 75 percent are less than about 0.52. The minimum and maximum are self explanatory, but each can be separately computed by the `min()` and `max()` functions, which are shown in the example. The μ and σ statistics are said to be measures of **location** (place on the real-number line) and **scale** (alternatively, dispersion, or spread) of the data. The $x_{0.50}$ (median), like the μ, also is a measure of location.

Several concepts require further discussion. The use of **statistical simulation**, which is shown repeatedly herein, is a powerful technique for exploration of the properties of various sample statistics. For the example, simulation is used to generate "data" for the previous example. In simulation, n pseudo-random numbers of F:$0 \leq F \leq 1$ are mapped through the QDF of the chosen distribution or its numerical equivalent should the QDF of the distribution have no explicit form.

For example, instead of using the `rnorm()` function to generate random values from the standard Normal distribution, the same statistical result for $n = 100$ also could be produced by example 2–22. In the example, the `qnorm()` computes the $x(F)$ or quantiles of the standard Normal. The `runif()` function, by default, provides uniformly-distributed, pseudo-random numbers between 0 and 1.

2–22
```
fake.dat <- qnorm(runif(100)); print(fake.dat) # not shown here
```

Because simulation is used, examples 2–21 or 2–22, if rerun, will produce different numerical values for the outputted results. Thus, the basic summary statistics are to be understood as estimators for which the numerical values are dependent on values from a *finite* sample. The standard Normal distribution by definition has $\mu = 0$ and $\sigma = 1$ (the population values). However, the sample estimates are $\hat{\mu} = -0.108$ and $\hat{\sigma} = 0.918$. The

sample statistics are not equal to the **population statistics** (or values), but in this case they clearly are close. It is desirable for a given statistic that as $n \to \infty$ that the sample statistic approaches or becomes equal to the population value. Unfortunately, in the real world, analysts must contend or be content (often forced) with sample sizes less than ideal or desired. The performance of a statistic as a function of sample size is a common theme in this dissertation.

An important property of any sample statistic is that over repeated samplings for reasonable values of n (reasonably meaning—sample sizes seen in real-world circumstances) that the statistic "on average" attains the population value and that variability of the statistic (sampling variance) is not too large. These and other sampling properties of statistics, primarily product moment and L-moment statistics, will be demonstrated through short simulation examples throughout this dissertation. Formal definitions of some of the more informative sample properties of sampling bias and sampling variability are deferred to Section 4.1.1. ◀

From the perspective of the programming needs for large applications, the access of the summary statistics (and similar data structures in R) is important. These can be accessed by collecting the summary statistics into a variable. Although the function `IQR()` that is built-in to R computes the **interquartile range**, for purposes of illustration, example 2–23 computes the interquartile range of a standard Normal distribution using eq. (2.15) by expression of the difference between the upper and lower quartiles from the results of the `summary()` function.

$$\mathrm{IQR} = X_{0.75} - X_{0.25} \tag{2.15}$$

2–23

```
fake.dat <- qnorm(runif(100)) # generate 100 random samples
thesum   <- summary(fake.dat) # basic summary statistics
dnQ <- thesum[2] # accessing the lower quartile
upQ <- thesum[5] # accessing the upper quartile
IQR <- upQ - dnQ # compute interquartile range
print(IQR)
3rd Qu.
 1.1081
```

The printed value of `IQR` is improperly labeled, through inheritance, with an attribute "`3rd Qu.`"—see the documentation for the `attributes()` and `names()` functions. The attribute of the variable `IQR` is changed by the `names()` function, which changes the name

of the first item of the variable `IQR` to `"IQR"`. This is demonstrated in example 2–24. The `attributes()` function can be used to remove the label as shown.

2–24
```
names(IQR)[1] <- "IQR" # set first label of IQR variable

print(IQR) # show that the setting was made
   IQR
1.1081
# remove the IQR label, the following can be used
attributes(IQR) <- NULL; print(IQR)
[1] 1.1081
```

This use of labels for variables through the `attributes()` function is a powerful method for enhancing the maintainability of software or enhancing the comprehension of end users. Such attributes provide a means for self documentation of code when used effectively. ◀

The **range** is the difference between the maximum and minimum of a distribution and is defined as

$$\mathrm{W} = X_{n:n} - X_{1:n} \tag{2.16}$$

Using the `fake.dat` generated in example 2–23, the range is computed in example 2–25 where the W is understandably larger than the IQR.

2–25
```
W <- diff(range(fake.dat))
print(W)
[1] 4.855895
```

◀

2.2.2 Fitting a Distribution by the Method of Moments

Distributions have parameters and distributions have moments. The statistics μ and σ are known as the first two product moments and are respective measures of location and scale of a distribution. If the parameters of a particular distribution are chosen such that the product moments of a distribution are equated to the sample product moments ($\mu = \hat{\mu}, \ldots$), then a distribution is said to be "fit" to the data. This moment-to-parameter technique is venerable, is known as the **method of moments**, and is in widespread use

(Rizzo, 2008, p. 38). The method of moments term typically is reserved for the context of fitting a distribution by the product moments. However, distributions can be fit using other method-of-moment-like algorithms. Mays (2005, chap. 10) reports "By fitting a distribution to a set of data, a great deal of the probabilistic information in the sample can be compactly summarized in the function and its associated parameters."

Some of the reasons that probability distributions are fit to samples include:

1. A *continuous and portable* model of the distribution of the data is needed so that either F values can be converted to $F \rightarrow x$ using the CDF or $x \rightarrow F$ using the QDF. For example, a manager or regulator of water quality for a river needs an estimate of the streamflow at the 10th percentile (a drought) because water quality can be of concern during periods lacking abundant rainfall;
2. A *parametric* model is needed for extrapolation to quantiles not represented by the data. For example, the estimation of the 99.9th percentile from a small sample $n = 20$ is needed. This extrapolation is of critical interest in the design and management of infrastructure, such as dams or earthquake hazards, in which design against the deleterious effects of large events is paramount; and
3. A *simulation* model is needed to drive further investigation. For example, studies of sample variability or studies involving the consequences of the failure of a part in a larger system are to be made.

The method of moments using the Normal distribution is now demonstrated. Recalling from elementary statistics courses, the PDF of the Normal is

$$f(x) = \frac{\exp(-[(x-\mu)/\sigma]^2/2)}{\sigma\sqrt{2\pi}} \tag{2.17}$$

where μ and σ are parameters and also the first two product moments (mean and standard deviation) of the distribution. Because the product moments are parameters and visa versa, the steps for fitting the Normal distribution are straightforward. To state succinctly, to use the method of moments, first, the sample $\hat{\mu}$ and $\hat{\sigma}$ are computed, and second, these sample values are substituted into eq. (2.17) as values for μ and σ, respectively. The method of moments is thus applied.

From the author's experience, as a periodic educator of both graduate and undergraduate geoscientists and civil or environmental engineers, many of these students have only one or two "statistics" courses. These students might recognize the symbols μ and σ, their

respective meanings, and be familiar with the meaning of each. However, often these students do not recognize that μ and σ in the case of the Normal distribution also represent "model parameters" and not just abstract statistics. When μ and σ are presented or cast as model parameters—a concept familiar to the students from the language of other courses—then greater insight into the Normal distribution and distributions in general is acquired by course participants.

USING R — USING R

The method of moments for the Normal distribution is readily shown in example 2–26 through simulation of a sample of size $n = 20$ from a Normal distribution with parameters set to $\mu = 500$ and $\sigma = 200$. The population statistics are set into variables `pop.mu` and `pop.sd`. The `rnorm()` function returns 20 sample (random) NOR$(500, 200)$ values into the variable `fake.dat`. The sample statistics $\hat{\mu}$ and $\hat{\sigma}$ are computed by `mean()` and `sd()`, respectively. The `max()` function demonstrates the utility of the vectorized arithmetic of R—notice how the two vectors `pop.PDF` and `sam.PDF` are effectively merged, and the global maximum is returned to `myup`.

2–26

```
#pdf("pdf2.pdf")
pop.mu <- 500; pop.sig <- 200; n <- 20
fake.dat <- rnorm(n, mean=pop.mu, sd=pop.sig)
x.bar <- mean(fake.dat); x.sig <- sd(fake.dat)
F <- seq(0.01,0.99, by=0.01)
x <- qnorm(F, mean=pop.mu, sd=pop.sig)
pop.PDF <- dnorm(x, mean=pop.mu, sd=pop.sig) # PDF of population
sam.PDF <- dnorm(x, mean=x.bar,  sd=x.sig)   # PDF of sample
myup <- max(pop.PDF, sam.PDF) # need a global max for plotting
plot(x,pop.PDF, type="l", ylim=c(0, myup),
     ylab="PROBABILITY␣DENSITY") # thin line
lines(x,sam.PDF, lwd=3) # thick line
#dev.off()
```

The results of the example are shown in figure 2.8. In the figure, it is seen that the location and scale of the parent distribution and the sample are similar, but the two curves obviously do not have a one-to-one correspondance. The lack of correspondence exists because the sample $\hat{\mu}$ and $\hat{\sigma}$ values (`x.bar` and `x.sig`) are (expectedly) not numerically equal to the parent μ and σ values. Therefore, the sample PDF (thick line) represents a fit to the parent PDF (thin line). The differences in this case are substantial because of the relatively small sample size of $n = 20$. If the sample size were increased to say

2,000, then the resulting thick line will likely mask or hide the thin line of the parent distribution. Because a parametric distribution is used and the distribution is Normal, it must be stressed, that in both cases, the general shapes (curvatures) of the two fitted distributions are identical. ◀

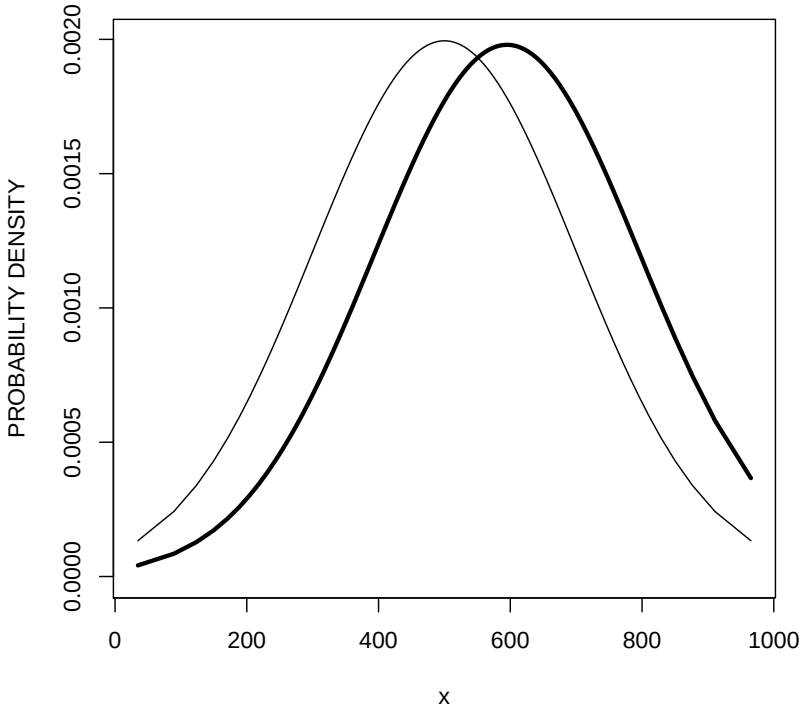

Figure 2.8. Comparison of a parent Normal distribution (thin line) and sample Normal distribution (thick line) for a sample of size 20 from example 2–26

2.2.3 Construction of an Empirical Distribution by Plotting Positions

An important component of distributional analysis is the construction of an empirical distribution. An empirical distribution is a distribution defined by the data values without an explicit acknowledgment or interpretation of a parametric form of the parent distribution. The empirical distribution has many applications including: (1) a technique to estimate quantiles within the range of the sample, (2) a technique for effective visualization of the distribution of the sample, and (3) a distribution to compare to one or more fitted distri-

butions. The second application (visualization) is common in many of the examples and resulting figures in this dissertation.

To construct an empirical distribution, **plotting positions**, which are well described by Helsel and Hirsch (1992, p. 23) and Stedinger and others (1993, chap. 18, pp. 22–26), are used to define the F or cumulative percentages of individual data points within a sample. Plotting positions can provide complementary components for alternative graphics to box plots. Plotting positions also can be used to construct probability graph paper or be used to compare two or more distributions. Plotting positions often are used for graphical display, and this is their primary use within this dissertation. The general formula for computing plotting positions or **plotting-position formula** is

$$F(x) = \frac{i - a}{n + 1 - 2a} \tag{2.18}$$

where i is an ascending rank, a is known as the **plotting-position coefficient**, and n is the sample size. To generate the plotting positions, the following algorithm is followed:

1. Order the data $x_{1:n} \leq x_{2:n} \leq \cdots \leq x_{n:n}$ from smallest to largest to form the sample order statistics (see Chapter 3),
2. Assign ranks to the ordered data $1, 2, \cdots, i, \cdots, n$ (i is rank), and
3. Compute $F(x)$ = nonexceedance probability by the plotting-position formula in eq. (2.18).

The true probability associated with the largest (and smallest) observation is a random variable with mean $1/(n + 1)$ and a standard deviation of nearly $1/(n + 1)$. Hence, all plotting-position formulas give crude estimates of the unknown probabilities associated with largest and smallest events. The plotting-position coefficient can be set to several different values. However, for general purposes the coefficient is $a = 0$, and thus, the so-called Weibull plotting positions are accessed. The reader can convince themselves that, as n becomes large, the choice of a becomes relatively unimportant. A listing of some potential plotting-position coefficients is provided in table 2.1. For the examples here, $a = 0$ or Weibull plotting positions will see near universal use in this dissertation. For general quantile estimation, the Cunnane plotting positions are recommended (Cunnane, 1989; Helsel and Hirsch, 1992, 2002).

There exists use of i/n as a plotting position estimator, which is called the California plotting position. This form is not recommended and not further considered here.

Table 2.1. Selected plotting-position coefficients for eq. (2.18)

Name	a	Motivation
Weibull	$0.$	Unbiased exceedance probabilities for all distributions
Median	$.3175$	Median exceedance probabilities for all distributions
APL	$\approx.35$	Useful with probability-weighted moments
Blom	$.375$	Nearly unbiased quantiles for Normal distribution
Cunnane	$.40$	Approximately quantile unbiased
Gringorten	$.44$	Optimized for Gumbel distribution
Hazen	$.50$	A traditional choice

USING R ——————— ——————— USING R

The `quantile()` function returns estimates of underlying distribution quantiles based on one or two order statistics from the supplied elements in a vector at specified nonexceedance probabilities. An R-oriented discussion is provided by Ugarte and others (2008, pp. 42–43). The inverse of the `quantile()` function is the `ecdf()` function, which represents the **empirical cumulative distribution function**. The `quantile()` function supports each of nine quantile algorithms. A discussion of sample quantiles from statistical packages is provided by Hyndman and Fan (1996). For the `quantile()` function the sample quantiles of type i are defined by

$$Q_{[i]}(F) = (1 - \psi)\, x_{j:n} + \psi\, x_{j+1:n} \tag{2.19}$$

where $1 \leq i \leq 9$, $(j - m)/n \leq F < (j - m + 1)/n$, and $x_{j:n}$ is the jth sample order statistic (see Chapter 3), n is the sample size, and m is a constant determined by the sample quantile type controlled by the value for i. Here ψ depends on the fractional part of $g = nF + m - j$.

For the continuous sample quantile types ($4 \leq i \leq 9$), the sample quantiles can be obtained by linear interpolation between the kth order statistic and $F(k)$ or

$$F(k) = \frac{k - A}{n - A - B + 1} \tag{2.20}$$

where A and B are constants determined by the specified type by i. Further, $m = A + F(1 - A - B)$, and $\psi = g$.

Two example applications of the `quantile()` function are shown in example 2–27 in which 999 standard Normal random samples are drawn by the `rnorm()` function.

2–27

```
x <- rnorm(999)

quantile(x) # Extremes and quartiles by default
         0%         25%         50%         75%        100%
-4.75557859 -0.62393911  0.09476024  0.73220817  3.24781578

quantile(x,  probs=c(0.1,5,10,50,NA)/100)
       0.1%          5%         10%         50%
-3.40851783 -1.58113414 -1.26403948  0.09476024          NA
```

◀

The *lmomco* package provides specific support for computation of plotting positions by the `pp()` function. Uses of the `pp()` function as well as the Weibull, Cunnane, and Hazen plotting positions are now demonstrated.

In example 2–28, some porosity (fraction of void space) data from an oil well in Texas are available in the file `clearforkporosity.csv`, which is located along the *lmomco* path `lmomco/data/clearforkporosity.csv`. The data from this file can be loaded by the `read.csv()` function or, for purposes of this dissertation, by the `data()` function because the data is distributed with *lmomco* and resides in the `data` subdirectory (R Development Core Team, 2009). In the example, the data are loaded, the respective plotting positions computed, and set into `PPw`, `PPc`, and `PPh`.

2–28

```
data(clearforkporosity)  # file extension is not needed
                         # from the lmomco package
names(clearforkporosity) # named contents of the data frame
[1] "POROSITY"

attach(clearforkporosity)
PHI <- POROSITY; PHI <- sort(PHI)
PPw <- pp(PHI)         # Weibull plotting position
PPc <- pp(PHI, a=0.4) # Cunnane plotting position
PPh <- pp(PHI, a=0.5) # Hazen plotting position
```

The `pp()` function demonstration continues in example 2–29 by plotting the data, and the effects of the choice of plotting-position formula on the tails of the empirical distribution are seen in the resulting figure 2–28.

2–29

```
#pdf("clearforkPP.pdf")
plot(qnorm(PPw),PHI, cex=3, pch=16, col=8, xlim=c(-2.5,2.5),
     xlab="STANDARD␣NORMAL␣DEVIATE",
     ylab="POROSITY")
points(qnorm(PPc),PHI, cex=2)
points(qnorm(PPh),PHI, cex=0.5, pch=16)
#dev.off()
```

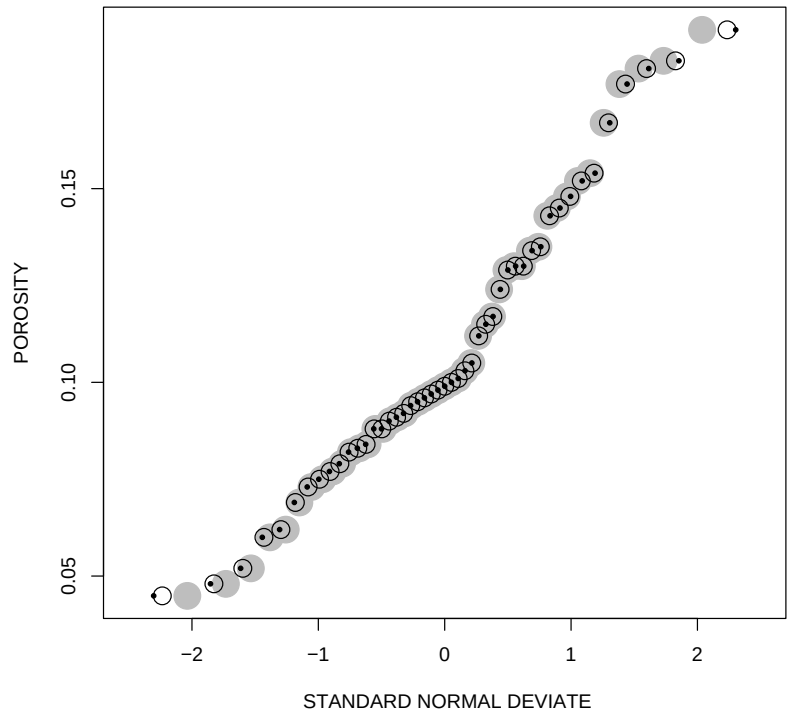

Figure 2.9. Empirical distribution by plotting position of porosity (fraction of void space) from neutron-density, well log for 5,350–5,400 feet below land surface for Permian Age Clear Fork formation, Ector County, Texas from example 2–29. The grey circles are Weibull positions, the open circles are Cunnane positions, and the black dots are Hazen positions.

◀

A comparison between the `pp()` and `quantile()` functions is made in example 2–30. For the example, the porosity data in variable `PHI` from example 2–28 are used.

2–30

```
PHI[1:3] # extract first three values of PHI
[1] 0.0449 0.0480 0.0520
# reversing Hazen method
```

```
round(quantile(PHI, probs=PPh, type=5)[1:3], digits=4)
 1.06383% 3.191489% 5.319149%
   0.0449    0.0480    0.0520
# reversing through Weibull method
round(quantile(PHI, probs=PPw, type=6)[1:3], digits=4)
 2.083333% 4.166667%     6.25%
   0.0449    0.0480    0.0520
```

In the example, the first three smallest values: 0.0449, 0.0480, and 0.0520 are printed. Second, using appropriate rounding, the Hazen plotting-position formula is used through the `type=5` argument for the Hazen plotting positions in variable `PPh` from example 2–28 and extract the quantiles. Third, the Weibull plotting-position formula is used by setting the `type=6` argument to the `quantile()` function. For the Hazen and Weibull cases the percentages change, but the two quantile ensembles are equivalent as shown by the first three values of `PHI`. ◀

2.2.4 Two Demonstrations of Basic Distributional Analysis

Two demonstrations of basic distributional analysis, which also include use of plotting positions, are provided in this section. The purposes of this section are (1) to provide a glimpse forward to more formal and thorough distributional analyses described in later chapters and (2) to establish the theme of the remainder of this dissertation. At the cost of getting ahead in, but also foreshadowing, the narrative, plotting positions and distribution fit are now demonstrated using the *lmomco* package.

USING R ——————————— ——————————— USING R

The first demonstration is example 2–31, which simulates $n = 30$ values from a two-parameter Weibull distribution using the built-in `rweibull()` function. The `lmom.ub()` function computes a sample L-moments of the simulated data, and the `parwei()` function computes the parameters for a three-parameter Weibull distribution from the L-moments. The `pp()` function implements eq. (2.18) with a default to the Weibull plotting positions. (The `pp()` function is used in many examples as a precursor to graphical operations.) The empirical distribution finally is plotted with F on the horizontal axis and the `sort()`ed data on the vertical. The quantiles for the plotting-position values of F are drawn as a line by the `lines()` function, which makes use of the `quawei()` function for the QDF of the Weibull. The output from the example is shown in figure 2.10.

2–31

```
#pdf("pp1.pdf")
fake.dat <- rweibull(30,1.4, scale=400) # selected parameter vals
WEI <- parwei(lmom.ub(fake.dat)) # compute Weibull parameters
                                 # from sample L-moments
PP  <- pp(fake.dat) # plotting positions
plot(PP, sort(fake.dat), xlab="NONEXCEEDANCE␣PROBABILITY",
                         ylab="QUANTILE")
lines(PP,quawei(PP,WEI))
#dev.off()
```

◀

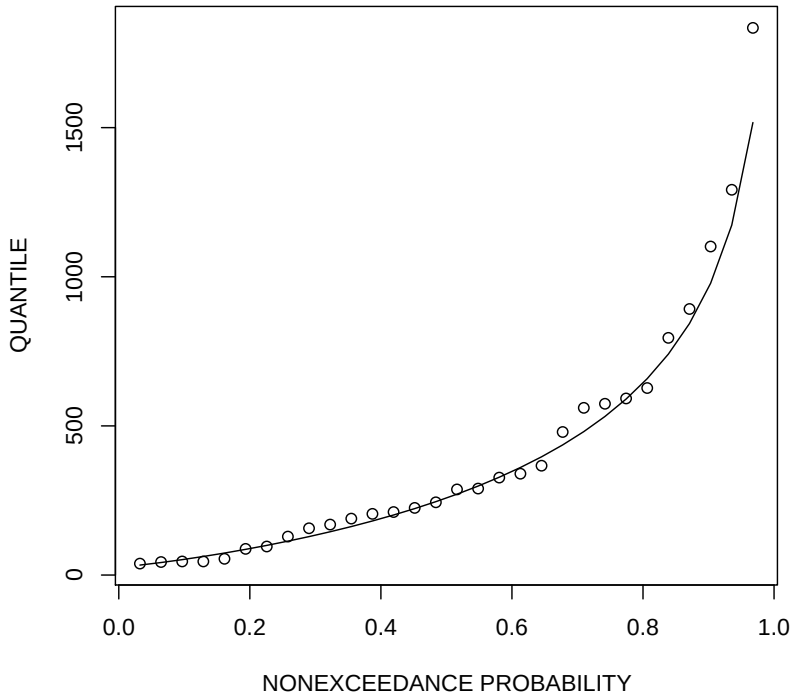

Figure 2.10. Empirical distribution of simulated data from specified Weibull distribution and Weibull distribution fit to L-moments of the simulated data from example 2–31

Instead of using simulated data to provide a second demonstration, the observed annual peak streamflow data for a selected streamflow-gaging station operated by the U.S. Geological Survey are used. An annual peak streamflow is the largest instantaneous volumetric rate of flow in a stream for a given year. Such data provide the backbone for statistical analyses that govern the management of flood plains and design of drainage infrastructure such as bridges.

The selected streamflow-gaging station is 05405000 Baraboo River near Baraboo, Wisconsin. The annual peak streamflow data were acquired at `http://nwis.waterdata.usgs.gov/nwis/peak/?site_no=05405000&`. This station has $n = 73$ years of record between 1914–2006, and the data are available as a `*.RData` file provided in the *lmomco* package. The data are titled `lmomco/data/USGSsta05405000peaks.RData`.

In example 2–32, the Baraboo River data are loaded by the `data()` function and `attach()`ed to the workspace for convenient access to the annual peak data. Readers can find more details for the `data.frame()`, `attach()`, and `detach()` functions in the R documentation: `help(data.frame)`. The data of interest are in the column labeled `peak_va` of the data frame. These data are `sort()`ed into the variable `Q`. The Weibull plotting positions are computed using the `pp()` function. To demonstrate the fit of the Normal distribution by the method of moments, the $\hat{\mu}$ and $\hat{\sigma}$ sample statistics of the data are set equal to the variables `mu` and `sig`, respectively. The data are strictly positive and heavy tailed; therefore, the log-Normal distribution (a Normal distribution of logarithms of the data) should also be considered. The $\hat{\mu}$ and $\hat{\sigma}$ sample statistics of the `log10(Q)` values are set equal to the variables `mu.lg` and `sig.lg`, respectively. The values are shown in the example.[2]

2–32

```
data(USGSsta05405000peaks) # load the *.RData file
attach(USGSsta05405000peaks)
Q   <- sort(peak_va) # sort the annual peak streamflow values
PP  <- pp(Q) # compute Weibull plotting positions
mu  <- mean(Q)
sig <- sd(Q)
mu.lg <- mean(log10(Q)); sig.lg <- sd(log10(Q))
cat(c("#", round(mu.lg, digits=3), round(sig.lg, digits=4),"\n"))
# 3.438 0.2326
#pdf("pp2.pdf")
plot(qnorm(PP),Q,
     xlab="STANDARD␣NORMAL␣DEVIATE",
     ylab="STREAMFLOW,␣IN␣FT^3/S")
lines(qnorm(PP),
      qnorm(PP, mean=mu, sd=sig)) # plot normal distribution
lines(qnorm(PP),
      10^qnorm(PP, mean=mu.lg, sd=sig.lg),
      lty=2) # plot lognormal distribution as dashed line
#dev.off()
```

[2] This example is treated in further detail in Chapter 8. The values in the example are repeated in table 8.3 on page 233.

The example continues by `plot()`ting the empirical distribution of the data in which, unlike example 2–31, the horizontal axis is shown in standard normal deviates. Such an axis is obtained by the `qnorm()` function. Data that are normally distributed will plot as a straight line with a `qnorm()`-transformed horizontal axis and a linear vertical axis. The `qnorm()` function, therefore, can be used to construct a "normal probability axis" of standard normal deviates. Finally, the Normal distribution is plotted by the quantile function `qnorm()` with $\mu = \hat{\mu}$ and $\sigma = \hat{\sigma}$, which in code is `qnorm(PP, mean=mu, sd=sig)` and represents the method of moments. The log-Normal distribution also is plotted by suitable argument substitution and transformation. The results are shown in figure 2.11. For completeness, example 2–33 computes basic summary statistics and $\hat{\sigma}$ by the `summary()` and `sd()` functions, respectively.

2–33

```
summary(Q) # Q is defined in the previous example
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
    710    1950    3000    3135    4150    7900
sd(Q) # compute the standard deviation
[1] 1602.115
```

By elementary interpretation of the data points and the two fitted distributions in figure 2.11, the author concludes that these annual peak streamflow data are not normally distributed. This conclusion is made because of the curvature of the data points relative to the straight line of the Normal distribution. Visually, the log-Normal distribution provides a much more reasonable model of the distribution of annual peak streamflow for these data, but even this distribution appears to curve too much and away from the data in the far tails.[3] ◀

The basic distributional analysis of the Baraboo River annual peak streamflow is completed by creation of a box plot of the annual peak streamflows so that a juxtaposition of the empirical distribution shown in figure 2.11 can be made. The code listed in example 2–34 suffices, and the box plot is shown in figure 2.12. The whiskers extend to the most extreme data points, which for this particular box plot are no more than 1.5 times the IQR (interquartile range in eq. (2.15)) from the box. (The `IQR()` function computes the IQR.) The lone open circle represents the largest, maximum, or $x_{n:n}$ value, which is an order statistic maxima.

[3] These annual peak streamflows also are used in the context of L-moment statistics in Section 8.2.3 in search of a better fit.

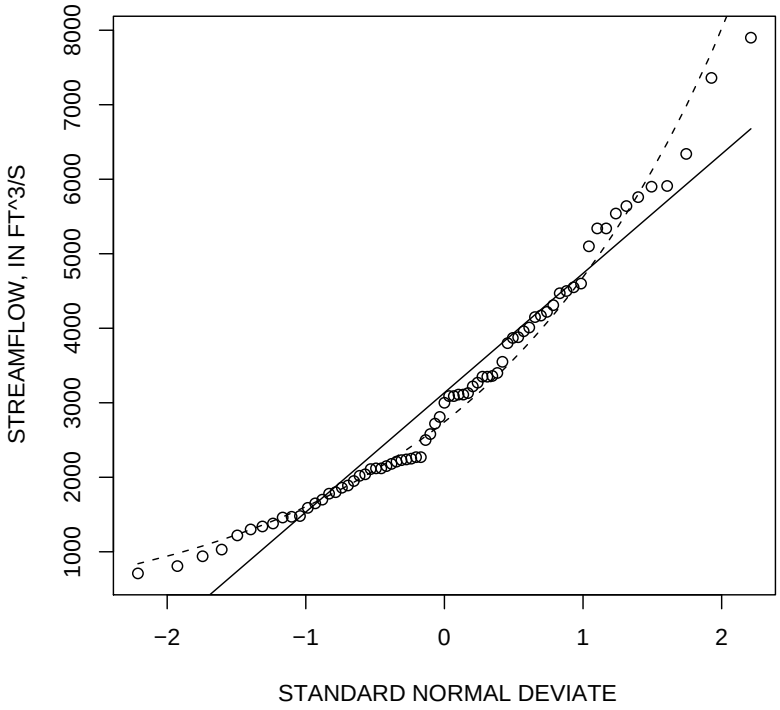

Figure 2.11. Empirical distribution of annual peak streamflow for U.S. Geological Survey streamflow-gaging station 05405000 Baraboo River near Baraboo, Wisconsin and Normal (solid line) and log-Normal (dashed line) distributions fit by method of moments from example 2–32

2–34

```
#pdf("pp2boxplot.pdf")
boxplot(Q)
mtext("BARABOO␣RIVER", side=1)
#dev.off()
```

◀

2.3 Summary

In this chapter, the concept of distributional analysis is expanded on relative to that provided in Chapter 1, and 34 examples are provided. This chapter reviewed continuous random variables and the mathematics of probability density functions, cumulative distribution functions, hazard functions, and and quantile functions. The quantile function discussion extends into the algebra of quantile functions, and examples of how this algebra can be used as a model building tool to create alternative distributions are provided.

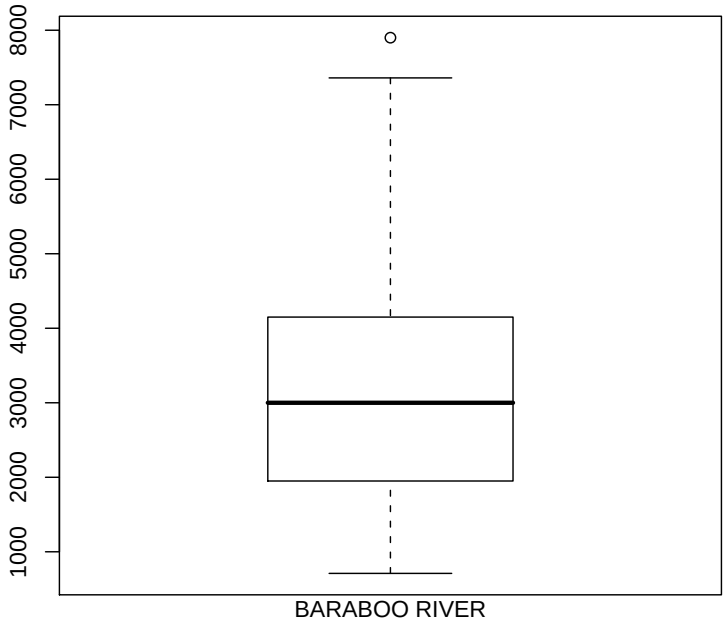

Figure 2.12. Box plot of the distribution of annual peak streamflow for U.S. Geological Survey streamflow-gaging station 05405000 Baraboo River near Baraboo, Wisconsin from example 2–34

A conceptual review of the relations between nonexceedance and exceedance probabilities and annual recurrence intervals is made.

A review of the basic summary statistics, namely the product moments of the mean and standard deviation, sets up more thorough discussion of product moments in Chapter 4. In the summary statistics context, the concept of generation of random variables is shown. The concept that distributions are simultaneously characterized by moments and parameters is introduced. If the parameters are chosen so that the product moments of the distribution match or are equated to the sample product moments, the distribution is fit by the method of moments. The method of moments is demonstrated by example. An important component of distributional analysis is visualization of the data, as shown by empirical distributions developed by plotting positions. Finally, the basic steps of distributional analysis are demonstrated by examples using simulation and observed annual peak streamflow data involving the method of moments.

Chapter 3
Order Statistics

In this chapter, I provide discussion concerning a branch of statistics known as order statistics in which the sorting (ranking) of a random variable or sorting of a sample is of primary importance. Common statistics, such as the minimum, maximum, and median, result from ranking from smallest to largest. The study of order statistics however also includes study of other characteristics of ordered random variables. For purposes here, the expectation of an order statistic is of special importance because L-moments are fundamentally defined and conceptualized as linear combinations of order statistic expectations. Further, special statistics such as the Sen weighted mean and Gini mean difference are based on ordering and are of historical interest to L-moments. This chapter appears early in the dissertation because foundational material is provided concerning the mathematics of L-moments, but readers are not required to thoroughly understand this chapter in order to perform distributional analysis with L-moment statistics using R.

3.1 Introduction

As mentioned in Section 1.3, a branch of statistics known as **order statistics** plays a prominent role in L-moment theory. The study of order statistics is the study of the statistics of ordered (sorted) random variables and samples. This chapter presents a very brief introduction of order statistics to provide a foundation for later chapters. A comprehensive exposition on order statistics is provided by David (1981), and an R-oriented approach is described in various contexts by Baclawski (2008).

The random variable X for a sample of size n when sorted creates the order statistics of X: $X_{1:n} \leq X_{2:n} \leq \cdots \leq X_{n:n}$. The **sample order statistics** from a random sample are created by sorting the sample into ascending order: $x_{1:n} \leq x_{2:n} \leq \cdots \leq x_{n:n}$. As will be

seen, the concept and use of order statistics take into account both the value (magnitude) and the relative relation (order) to other observations. Barnett (2004, p. 23) reports that

> …the effects of ordering can be impressive in terms of both what aspects of sample behavior can be usefully employed and the effectiveness and efficiency of resulting inferences.

and that

> …linear combinations of all ordered samples values can provide efficient estimators.

This dissertation will show that the L-moments, which are based on linear combinations (recalling page 9) of order statistics, do in fact provide effective and efficient estimators of distributional geometry.

In general, order statistics are already a part of the basic summary statistic repertoire possessed by most individuals, including nonscientists and nonstatisticians. The **minimum** and **maximum** are examples of **extreme value order statistics** and are defined by the following notation:

$$\min\{X_n\} = X_{1:n} \tag{3.1}$$

$$\max\{X_n\} = X_{n:n} \tag{3.2}$$

The familiar **median** $X_{0.50}$ by convention is

$$X_{0.50} = \begin{cases} (X_{[n/2]:n} + X_{[(n/2)+1]:n})/2 & \text{if } n \text{ is even} \\ X_{[(n+1)/2]:n} & \text{if } n \text{ is odd} \end{cases} \tag{3.3}$$

and thus clearly is defined in terms of one order statistic in the case of odd sample size or a linear combination of two order statistics in the case of even sample sizes.

Other order statistics exist and several important interpretations towards the purpose of this dissertation can be made. Concerning L-moments discussed in Chapter 6, Hosking (1990, p. 109) and Hosking and Wallis (1997, p. 21) provide an "intuitive" justification for L-moments and by association the probability-weighted moments (see Chapter 5). The justification is founded on order statistics:

- The order statistic $X_{1:1}$ (a single observation) contains information about the location of the distribution on the real-number line $\mathbb{R}$;

- For a sample of $n = 2$, the order statistics are $X_{1:2}$ (smallest) and $X_{2:2}$ (largest). For a highly dispersed distribution, the expected difference between $X_{2:2} - X_{1:2}$ would be large, whereas for a tightly dispersed distribution, the difference would be small. The expected differences between order statistics of an $n = 2$ sample hence can be used to express the variability or scale of a distribution; and
- For a sample of $n = 3$, the order statistics are $X_{1:3}$ (smallest), $X_{2:3}$ (median), and $X_{3:3}$ (largest). For a negatively skewed distribution, the difference $X_{2:3} - X_{1:3}$ would be larger (more data to the left) than $X_{3:3} - X_{2:3}$. The opposite (more data to the right) would occur if a distribution were positively skewed.

These interpretations hint towards expression of distribution geometry by select use of intra-sample differences. In fact, various intra-sample differences can be formulated to express fundamental and interpretable measures of distribution geometry. Intra-sample differences are an important link to L-moments, and the link justifies exposition of order statistics in a stand-alone chapter. Kaigh and Driscoll (1987, p. 25) defined O-statistics as "smoothed generalizations of order statistics" and provide hints (Kaigh and Driscoll, 1987, eq. 2.4, p. 26) towards L-moments by suggesting that linear combinations of the order statistics in the previous list and others not listed provide for location, scale, and "scale-invariant" skewness and kurtosis estimation.

3.1.1 Expectations and Distributions of Order Statistics

A definition regarding order statistics, which will be critically important in the computation of L-moments in Chapter 6 and probability-weighted moments in Chapter 5, is the **expectation of an order statistic**. The expectation is defined in terms of the QDF. The expectation of an order statistic for the jth largest of r values is defined (David, 1981, p. 33) in terms of the QDF $x(F)$ as

$$\mathrm{E}[X_{j:n}] = \frac{n!}{(j-1)!(n-j)!} \int_0^1 x(F) \times F^{j-1} \times (1-F)^{n-j} \, \mathrm{d}F \tag{3.4}$$

where the quantity to the left of the integral is

$$\frac{n!}{(j-1)!(n-j)!} = n \binom{n-1}{j-1} \tag{3.5}$$

where the $\binom{a}{b}$ notation represents binomial coefficients. The $\binom{a}{b}$ notation is defined as

$$\binom{a}{b} = \frac{a!}{(a-b)!\,b!} \quad \text{for } b \leq a \tag{3.6}$$

and by convention $0! = 1$; eq. (3.6) is an expression for the number of possible combinations of a items taken b at a time. Factorials $u!$ are defined as

$$u! = u(u-1)(u-2)\ldots(u-u+2)(u-u+1) = \Gamma(u+1) \tag{3.7}$$

For both integer and non-integer u, the factorial can be computed using $\Gamma(u+1)$ where $\Gamma(\,)$ is the complete gamma function defined in eq. (8.85) on page 244. In R, the `gamma()` function is $\Gamma(\,)$ and the `lgamma()` function is the natural logarithm of $\Gamma(\,)$. The later is the most often used version in general programming because ratios of factorials of large u are regularly needed such as in eq. (3.4) or more specifically

$$\frac{n!}{(j-1)!(n-j)!} = \exp[\log\Gamma(n+1) - \log\Gamma(j) - \log\Gamma(n-j+1)] \tag{3.8}$$

The PDF of the $X_{j:n}$ (jth order statistic of a sample of size n) for a random variable having CDF $F(x)$ and PDF $f(x)$ is defined (David, 1981, p. 9) as

$$f_{j:n}(x) = \frac{[F(x)]^{j-1}[1-F(x)]^{n-j}f(x)}{\mathrm{B}(j, n-j+1)} \tag{3.9}$$

where $\mathrm{B}(a,b)$ is the **beta function** or **complete beta function**. The beta function is defined for $a, b > 0$ as

$$\mathrm{B}(a,b) = \int_0^1 F^{a-1}(1-F)^{b-1}\,\mathrm{d}F = \int_0^\infty \frac{x^{a-1}}{(1+x)^{a+b}}\,\mathrm{d}x = \frac{\Gamma(a)\Gamma(b)}{\Gamma(a+b)} \tag{3.10}$$

where $\Gamma(u)$ is the complete gamma function. A useful relation is

$$\binom{n}{k}^{-1} = (n+k)\;\mathrm{B}(n-k+1, k+1) \tag{3.11}$$

The CDF of $X_{j:n}$ for a random variable having CDF $F(x)$ and PDF $f(x)$ is defined (David, 1981, p. 8) as

$$F_{j:n}(x) = \sum_{i=j}^{n} \binom{n}{i} [F(x)]^i [1-F(x)]^{n-i} \tag{3.12}$$

It follows that the expectation of an order statistic is defined respectively in terms of the CDF and PDF as

$$\mathrm{E}[X_{j:n}] = \int_{-\infty}^{\infty} x f_{j:n}(x)\ \mathrm{d}x \tag{3.13}$$

or

$$\mathrm{E}[X_{j:n}] = \frac{\int_{-\infty}^{\infty} [F(x)]^{j-1}[1-F(x)]^{n-j} x\, f(x)\ \mathrm{d}x}{\mathrm{B}(j, n-j+1)} \tag{3.14}$$

The expectation of an order statistic for a sample of size $n = 1$ is especially important because

$$\mathrm{E}[X_{1:1}] = \int_0^1 x(F)\ \mathrm{d}F = \mu = \text{arithmetic mean} \tag{3.15}$$

Therefore, the familiar mean can be interpreted thus: The mean is the expected value of a single observation if one and only one sample is drawn from the distribution.

Hosking (2006) reports from references cited therein that "the expectations of extreme order statistics characterize a distribution." In particular, if the expectation of a random variable X is finite, then the set $\{\mathrm{E}[X_{1:n}{:}n{=}1, 2, \cdots]\}$ or $\{\mathrm{E}[X_{n:n}{:}n{=}1, 2, \cdots]\}$ uniquely determine the distribution. Hosking (2006) reasons that such sets of expectations contain redundant information. Technically a subset of expectations therefore can be dropped, and the smaller set is still sufficient to characterize the distribution. This feature of extreme order statistics is further considered in Chapter 6 on page 122 in the context of distribution characterization by L-moments.

USING R

Using eq. (3.4), the expected value of the 123rd-ordered (increasing) value of a sample of size $n = 300$ is computed for an Exponential distribution in example 3–1. The ratio of factorial functions in eq. (3.4) is difficult to compute for large values—judicious use of the fact that $n! = \Gamma(n+1)$ and use of logarithms of the complete Gamma function $\Gamma(a)$ suffices. The results of the integration using the Exponential QDF by the `qexp()` function and stochastic computation using random variates of the Exponential by the `rexp()` function for $\mathrm{E}[X_{123:300}]$ are very similar.[1]

[1] The first and second values (the third is from simulation) should seemingly be the same, but a bug in logic has not been found. The following example that uses `j=300` shows that the first and second values are identical.

3–1

```
nsim <- 10000; n <- 300; j <- 123; set.seed(10)
int <- integrate(function(f,n=NULL,j=NULL) {
                   exp(lgamma(n+1) - lgamma(j) - lgamma(n-j+1)) *
                   qexp(f) * f^(j-1) * (1-f)^(n-j)
                 }, lower=0, upper=1, n=n, j=j)

E_integrated <- int$value
E_byfunc     <- expect.max.ostat(300, para=vec2par(c(0,1),
                        type="exp"), pdf=pdfexp, cdf=cdfexp, j=j)
   # This function can be used for non maximum too if j provided.
E_stochastic <- mean(replicate(nsim, sort(rexp(n))[j]))

cat(c("RESULTS:", round(E_integrated, digits=3),
      "and",      round(E_byfunc,     digits=3),
      "and",      round(E_stochastic, digits=3), "\n"))
RESULTS: 0.526 and 0.543 and 0.526
```

Finally, changing `j=123` in example 3–1 to `j=300` for the maximum order statistic, produces `RESULTS: 6.283 and 6.283 and 6.297`. These values are also similar. ◀

3.1.2 Distributions of Order Statistic Extrema

The extrema $X_{1:n}$ and $X_{n:n}$ are of special interest in many practical problems of distributional analysis. Consider the sample maximum of random variable X having CDF of $F(x) = \Pr[X_{n:n} \leq x]$, if $X_{n:n} \leq x$, then all $x_i \leq x$ for $i = 1, 2, \cdots, n$, it can be shown for the sample maximum that

$$F_n(x) = \Pr[X \leq x]^n = [F(x)]^n \tag{3.16}$$

Similarly, it can be shown for the sample minimum that

$$F_1(x) = \Pr[X > x]^n = [1 - F(x)]^n \tag{3.17}$$

Durrans (1992) considers eq. (3.16) in more detail by exploring the possibility of fractional order of the exponent by suggesting the substitution of n (integer) for γ, which is real-valued ($\gamma > 0$). Durrans (1992, p. 1650) comments that "an attractive feature of distributions of fractional order statistics is the thickening and thinning of the [distribution]

tails as the parameter γ is varied." Further consideration of fractional order statistics is not made in this dissertation.

Using the arguments producing eqs. (3.16) and (3.17) with a focus on the QDF, Gilchrist (2000, p. 85) provides

$$x_{n:n}(F) = x(F^{1/n}) \tag{3.18}$$

$$x_{1:n}(F) = x(1 - (1 - F)^{1/n}) \tag{3.19}$$

for the QDF of the maximum and minimum, respectively. Gilchrist (2000, p. 85) comments that, at least for $x_{n:n}$, that "the quantile function of the largest observation is thus found from the original quantile function in the simplest of calculations."

For the general computation of the distribution of non-extrema order statistics, the computations are more difficult. Gilchrist (2000, p. 86) shows that the QDF of the distribution of the jth order statistic of a sample of size n is

$$x_{j:n}(F) = x[\mathrm{B}^{(-1)}(F,\ j,\ n - j + 1)] \tag{3.20}$$

where $x_{j:n}(F)$ is to be read as "the QDF of the jth order statistic for a sample of size n given by nonexceedance probability F." The function $\mathrm{B}^{(-1)}(F, a, b)$ is the QDF of the **Beta distribution**—the $^{(-1)}$ notation represents the inverse of the CDF, which is of course a QDF. The PDF of the Beta distribution is

$$f(x) = \frac{x^{\alpha-1}(1-x)^{\beta-1}}{\mathrm{B}(\alpha, \beta)} \tag{3.21}$$

where α and β are parameters and $\mathrm{B}(\alpha, \beta)$ is the beta function of eq. (3.10). The first two product moments (mean and variance) of the Beta distribution are

$$\mu = \frac{\alpha}{\alpha + \beta} \tag{3.22}$$

$$\sigma^2 = \frac{\alpha\beta}{\alpha + \beta + 1} \tag{3.23}$$

It follows that the QDF for an F and sample size of n of the order statistic extrema are

$$x_{1:n}(F) = x[\mathrm{B}^{(-1)}(F,\ 1,\ n)] \quad \text{and} \quad x_{n:n}(F) = x[\mathrm{B}^{(-1)}(F,\ n,\ 1)] \tag{3.24}$$

for the minimum $X_{1:n}$ and maximum $X_{n:n}$, respectively.

In the context of eqs. (3.16) and (3.17), the expectations of extrema for the Exponential distribution are stochastically computed in example 3–2 using the `min()` and `max()` functions. The random variates from the Exponential are computed by the `rexp()` function. The example begins by setting the sample size $n = 4$, the size of a simulation run in `nsim`, and finally, the scale parameter (note that R uses a rate expression for the dispersion parameter) of the Exponential distribution is set to $1{,}000$. (A location parameter of 0 is implied.) The example reports 1000, 1500, and 500 for the respective mean and expectations of the maximum and minimum. (It is known, as shown in Section 7.2.2 in eq. (7.17), that the mean of this Exponential distribution is $1{,}000$.)

3–2

```
n <- 4; nsim <- 200000
s <- 1/1000 # inverse of scale parameter = 1000
# Expectation of Expectation of Exponential Distribution
mean(replicate(nsim, mean(rexp(n, rate=s))))
[1] 1000.262

# Expectation of Maximum from Exponential Distribution
mean(replicate(nsim, max(rexp(n, rate=s))))
[1] 1504.404

# Expectation of Minimum from Exponential Distribution
mean(replicate(nsim, min(rexp(n, rate=s))))
[1] 499.6178
```

The demonstration continues in example 3–3 with the stochastic computation of the expected values of the maximum and minimum by eqs. (3.16) and (3.17). An interesting consideration of these equations is that sorting a vector of extrema distributed values as for the maximum and minimum computation is not needed. (The quantiles of the Exponential are computed by the `qexp()` function; whereas, Uniform variates are computed by the `runif()` function.) The output of examples 3–2 and 3–3 are consistent with each other.

3–3

```
# Expectation of Maximum from Exponential Distribution
mean(qexp(runif(nsim)^(1/n), rate=s))
[1] 1497.001
# Expectation of Minimum from Exponential Distribution
mean(qexp(1 - runif(nsim)^(1/n), rate=s))
[1] 501.1628
```

◀

The two previous examples imply that eqs. (3.16) and (3.17) provide a more efficient means of computing the distribution of extrema because sorting is computationally expensive. The `system.time()` function in example 3–4 measures the relative time to compute the expectation of a minimum value of a sample of size $n = 4$. The example shows that use of eq. (3.17) is more than 35 times faster for the author's computer.

3–4

```
system.time(mean(replicate(nsim, min(qexp(runif(n), rate=s)))))
   user  system elapsed
  3.337   0.047   3.502

system.time(mean(qexp(1 - runif(nsim)^(1/n), rate=s)))
   user  system elapsed
  0.059   0.006   0.064
```

◀

The distributions of individual order statistics in eq. (3.20) are easily demonstrated. Example 3–5 defines the `qua.ostat()` function to compute the quantiles for a given order statistic. The arguments `f` and `para` to the function are the $F_{j:n}$ and *lmomco* parameter list (see page 163 and ex. 7–1), respectively. The parameter list is a data structure specific to the *lmomco* package. The other two arguments are self explanatory. The `qbeta()` function is used to compute quantiles of the Beta distribution. Finally, the `par2qua()` function dispatches the `para` parameter list to the appropriate distribution with $F =$`betainv.F`.

3–5

```
"qua.ostat" <-
function(f,j,n,para) {
  betainv.F <- qbeta(f,j,n-j+1) # compute nonexceedance prob.
  return(par2qua(betainv.F,para))
}
# Now demonstrate usage of the qua.ostat() function
PARgpa <- vec2par(c(100,500,0.5), type="gpa") # make parameters
n <- 20; j <- 15; F <- 0.5 # sample size, rank, and nonexceedance
ostat <- qua.ostat(F,j,n,PARgpa)
print(ostat)
[1] 571.9805
```

After defining the `qua.ostat()` function by the `function()` "function," the example continues by specifying an *lmomco* parameter list for the Generalized Pareto distribution into variable `PARgpa` using `vec2par()` through the `type="gpa"` argument. A sample size of $n = 20$ is set, and the median of the distribution of the 15th-order statistic for such a

sample is computed. The example reports $x_{15:20}(0.5) = 572$ or the "50th percentile of the 15th value of a sample of size 20." The `qua.ostat()` function actually is incorporated into the *lmomco* package. The function is shown here as an example of syntax brevity by which eq. (3.20) can be implemented using the vectorized nature of the R language. ◀

3.2 L-estimators—Special Statistics Related to L-moments

Jurečková and Picek (2006, pp. 63–70) summarize linear statistical estimators known as **L-estimators** and Serfling (1980, pp. 262–291) considers the asymptotic (very large sample) properties of L-estimators. L-estimators T_n for sample of size n are based on the order statistics and are expressed in a general form as

$$T_n = \sum_{i=1}^{n} c_{i:n} h(X_{i:n}) + \sum_{i=1}^{n} d_j h^\star(X_{[np_j+1]:n}) \tag{3.25}$$

where $X_{i:n}$ are the order statistics, $c_{1:n}, \cdots, c_{n:n}$ and $d_1, \cdots, d_n$ are given coefficients or weight factors, $0 < p_1 < \cdots < p_k < 1$, and $h(a)$ and $h^\star(a)$ are given functions for argument a. The coefficients $c_{i:n}$ for $1 \leq i \leq n$ are generated by a bounded weight function $J(a)$ with a domain $[0, 1]$ with a range of the real-number line $\mathbb{R}$ by either

$$c_{i:n} = \int_{(i-1)/n}^{i/n} J(s)\ \mathrm{d}s \tag{3.26}$$

or approximately

$$c_{i:n} = \frac{J(i/[n+1])}{n} \tag{3.27}$$

The quantity to the left of the $+$ in eq. (3.25) uses all of the order statistics whereas the quantity to the right of the $+$ is a linear combination of a finite number of order statistics (quantiles). L-estimators generally have the form of either quantity, but not both. Estimators defined by the left quantity are known as type I and those of the right are known as type II. L-estimators of type I are discussed by Huber (1981, p. 55) and Barnett and Lewis (1995, p. 146).

The simplest example suggested by Jurečková and Picek (2006, p. 64) of an L-estimator of distribution location are the sample median and the **midrange**, in which the later is defined as

$$\mathrm{T}_n = \frac{X_{1:n} + X_{n:n}}{2} \tag{3.28}$$

A simple L-estimator of distribution scale is the **range** or

$$\mathrm{R}_n = X_{n:n} - X_{1:n} = \text{largest} - \text{smallest} \tag{3.29}$$

Two particularly interesting L-estimators that have immediate connection to L-moments are the Sen weighted mean and Gini mean difference statistics. These two statistics are described in the sections that follow.

3.2.1 Sen Weighted Mean

A special L-estimator of distribution location that is based on order statistics is the **Sen weighted mean** (Sen, 1964) or the quantity $\mathcal{S}_{n,k}$. The $\mathcal{S}_{n,k}$ is a robust estimator (Jurečková and Picek, 2006, p. 69) of the mean of a distribution and is defined as

$$\mathcal{S}_{n,k} = \binom{n}{2k+1}^{-1} \sum_{i=1}^{n} \binom{i-1}{k} \binom{n-i}{k} X_{i:n} \tag{3.30}$$

where $X_{i:n}$ are the order statistics and k is a weighting or trimming parameter. A sample version $\hat{\mathcal{S}}_{n,k}$ results when $X_{i:n}$ are replaced by their sample counterparts $x_{i:n}$. Readers are asked to note that $\mathcal{S}_{n,0} = \mu = \overline{X}_n$ or the arithmetic mean, and $\mathcal{S}_{n,k}$ is the median if either n is even and $k = (n/2) - 1$ or n is odd and $k = (n-1)/2$.

USING R

The *lmomco* package provides support for $\hat{\mathcal{S}}_{n,k}$ through the `sen.mean()` function, which is demonstrated in example 3–6. In the example, some fake data are set into `fake.dat`, and a "Sen" object `sen` is created. A list `sen` is returned by the `sen.mean()` function.

3–6

```
fake.dat <- c(123, 34, 4, 654, 37, 78)

# PART 1, sample means
sen <- sen.mean(fake.dat)
print(sen)
$sen
[1] 155
```

```
$source
[1] "sen.mean"

mean(fake.dat)
[1] 155

# PART 2, sample medians
sen <- sen.mean(fake.dat, k=(length(fake.dat)/2) - 1)
print(sen)
$sen
[1] 57.5
$source
[1] "sen.mean"

median(fake.dat)
[1] 57.5
```

The first part of the example shows that by default $\hat{\mathcal{S}}_{n,0} = \mu$, which is 155 for the example. The second part shows that k can be chosen to yield the median, which is 57.5 for the example. ◀

Finally, $\mathcal{S}_{n,k}$ is equivalent to the first symmetrically trimmed TL-moment (that will be formally introduced as $\lambda_1^{(k)}$ in Section 6.4). The numerical equivalency $\mathcal{S}_{n,k} = \lambda_1^{(k)}$ is demonstrated in example 3–7 by computing a two sample (two data point) trimming from each tail (side) of a Normal distribution having a $\mu = 100$ and $\sigma = 1$ or in moment-order listing: NOR(100, 1). The magnitude of the difference between $\hat{\mathcal{S}}_{n,k}$ and the first TL-moment for symmetrical trimming k should be zero and is shown in the last line.

3–7

```
fake.dat <- rnorm(20, mean=100) # generate a random sample
lmr <- TLmoms(fake.dat, trim=2) # compute trimmed L-moments
sen <- sen.mean(fake.dat, k=2)  # compute Sen mean
the.diff <- abs(lmr$lambdas[1] - sen$sen)
print(the.diff) # should be zero
[1] 0
```

◀

Foreshadowing Section 4.1.1, but here providing an informative example in the context of the trimmed mean, in example 3–8, the mean square errors (MSE) of the `sen.mean()`, `trim.mean()` (Rizzo, 2008, p. 156), and `median()` estimators are computed and compared the three errors to those reported by Rizzo (2008, pp. 156–157). The example begins by defining a `trim.mean()` function and using the same sample size $n = 20$ as used by Rizzo. For this particular example, the `set.seed()` function is used to set a seed for the

random number generator in current use by R. By setting the seed, users for this example should precisely reproduce the output shown.[2]

3–8

```
"trim.mean" <- function(x) { # mimicking Rizzo (2008)
  x <- sort(x); n <- length(x)
  return(sum(x[2:(n-1)])/(n-2))
}
n <- 20; nsim <- 75000
set.seed(1000) # set the seed for the random number generator

S1 <- replicate(nsim, sen.mean(rnorm(n))$sen)
sam.biasvar(S1,0, verbose=FALSE)$mse
[1] 0.04990509

# Sampling statistics of the trim.mean()
# Rizzo (2008) p.156 reports mse=0.0518
S2 <- replicate(nsim, trim.mean(rnorm(n)))
sam.biasvar(S2,0, verbose=FALSE)$mse
[1] 0.05124172

# Rizzo (2008) p.157 reports mse=0.0748
S3 <- replicate(nsim, median(rnorm(n)))
sam.biasvar(S3,0, verbose=FALSE)$mse
[1] 0.07363024
```

The example continues using the `sam.biasvar()` function, which is created in example 4–1, to perform `nsim` simulations of the `sen.mean()`, `trim.mean()`, and `median()` estimates of the standard Normal distribution. The results in example 3–8 show numerical equivalency between the values reported by Rizzo. Further, the results show that the equivalent algorithms for `sen.mean()` and `trim.mean()` have smaller mean square errors than the familiar median. This is a natural consequence of the median using far less numerical information contained in the sample than used by the trimmed mean. ◀

3.2.2 Gini Mean Difference

Another special L-estimator of distribution scale (dispersion, spread, variability) that is based on order statistics is the **Gini mean difference** (Gini, 1912), which is closely related

[2] Note that the general practice in this dissertation is to be independent of specific seeds so users should expect numerically different, but stochastically similar results for other examples herein.

to the second L-moment λ_2. The Gini mean difference $\mathcal{G}$ (Serfling, 1980, p. 263) is a robust estimator (Jurečková and Picek, 2006, p. 64) is defined as respective population $\mathcal{G}$ and sample $\hat{\mathcal{G}}$ statistics as

$$\mathcal{G} = \mathrm{E}[X_{2:2} - X_{1:2}] = \mathrm{E}[X_{2:2}] - \mathrm{E}[X_{1:2}] \tag{3.31}$$

and

$$\hat{\mathcal{G}} = \frac{2}{n(n-1)} \sum_{i=1}^{n} (2i - n - 1)\, x_{i:n} \tag{3.32}$$

where $X_{i:n}$ are the order statistics, $x_{i:n}$ are the sample order statistics, and $n \geq 2$. The statistic $\mathcal{G}$ is a measure of the expected difference between two randomly drawn values from a distribution. Hence, the statistic is a measure of distribution scale or spread (see second justification in the list starting on page 63).

The Gini mean difference is considered by Barnett and Lewis (1995, p. 168). However, David (1981, p. 192) considers $\mathcal{G}$ in more detail and reports that, although the statistic is named after Gini (1912), $\mathcal{G}$ was "already studied by Helmert in 1876 [(Helmert, 1876)[3]] and not brand new then!" (Exclamation point is David's.) Hald (1998, p. 644) provides historical discussion of Helmert's article.

USING R

The *lmomco* package provides support for $\hat{\mathcal{G}}$ through the `gini.mean.diff()` function, which is demonstrated in example 3–9. In the example, a fake data set is set into `fake.dat`, a "Gini" object is created, and assigned to variable `gini`. A list `gini` is returned. The $\hat{\mathcal{G}}$ statistic is listed in `gini$gini`, and the second sample L-moment ($\hat{\lambda}_2$, see Chapter 6) is listed in `gini$L2`. Thus, $\hat{\mathcal{G}} = 237$.

3–9

```
fake.dat <- c(123,34,4,654,37,78) # fake data
gini <- gini.mean.diff(fake.dat)  # from lmomco
str(gini) # output the list structure
List of 3
 $ gini  : num 237
 $ L2    : num 119
 $ source: chr "gini.mean.diff"
```

◀

[3] Astronomische Nachrichten is the oldest astronomical journal of the world that is still being published (`http://www.aip.de/AN/`).

By definition, $\mathcal{G} = 2\lambda_2$ where λ_2 is the second L-moment. Example 3–10 computes the sample L-moments using the `lmoms()` function of *lmomco* and demonstrates the numerical equivalency of $\mathcal{G} = 2\lambda_2$ by the `print()` function outputting zero in the last line of the example.

3–10

```
lmr <- lmoms(fake.dat) # compute L-moments from lmomco
print(abs(gini$gini/2 - lmr$lambdas[2])) # should be zero
[1] 0
```

◀

After reporting, within discussion of order-based inference, that "linear functions of the ordered sample values can form not only useful estimators but even optimal ones," Barnett (2004, p. 27) goes on to report that the quantity

$$V = \frac{1.7725}{n(n-1)} \sum_{i=1}^{n} (2i - n - 1) X_{i:n} = \hat{\mathcal{G}} \tag{3.33}$$

is "more easily calculated than the unbiased sample variance [$\hat{\sigma}^2$], and for normal X it is about 98 [percent] efficient relative to [$\hat{\sigma}^2$] for all sample sizes." Barnett apparently has made a mistake on the units—the units of V are not squared like those of variance. Therefore, a conclusion is made that $V^2 = \hat{\mathcal{G}}^2$ is what Barnett means. Emphasis is needed that these two statistics are both variance estimators. (The concept of efficiency is formally described in Section 4.1.1, and the sample variance $\hat{\sigma}^2$ is defined in eq. (4.18).)

There are many specific connections of eq. (3.33) to this dissertation, which are particularly interesting to document, because Barnett (2004) makes no reference to L-moments, no reference to the Gini mean difference, and a solitary reference to L-estimators (Barnett, 2004, p. 122). The connections are:

- Eq. (3.33) is very similar to eq. (3.32): $1.7725 \times \hat{\mathcal{G}}/2 = V$;
- The Gini mean difference is related to the second L-moment λ_2 by $\mathcal{G} = 2\lambda_2$. Thus, λ_2 is related to V;
- The sample standard deviation is $\hat{\sigma} = \sqrt{\hat{\sigma}^2}$;
- In terms of L-moments, the standard deviation of the Normal distribution is $\sigma = \lambda_2\sqrt{\pi}$ by eq. (7.10); and
- The value $\sqrt{\pi} = 1.772454\ldots$, which has an obvious connection to eq. (3.33).

Barnett (2004) asserts that the "efficiency" of V is "about 98 percent" for all sample sizes. Assuming that relative efficiency[4] RE is meant, R is used to test this claim. In example 3–11, the variance of V and the familiar definition $\hat{\sigma}^2$ by the `var()` function are computed for a large sample size of $n = 2{,}000$ for a very large number of simulations.

3–11

```
n    <- 2000   # sample size
nsim <- 200000 # no. of simultions
"Barnett" <- function(n) {
    gini <- gini.mean.diff(rnorm(n))$gini
    return((sqrt(pi)*gini/2)^2)
}
GiniVar    <- var(replicate(nsim, Barnett(n)   ))
ClassicVar <- var(replicate(nsim, var(rnorm(n))))
RE <- ClassicVar/GiniVar # relative efficiency

print(RE)
[1] 0.9738433
# Barnett (2004, p. 27) reports 98 percent.
```

The example estimates that $\mathrm{RE} \approx 0.97$, which is acceptably close to the "about 98 percent" value reported by Barnett. Therefore, the computed value in example 3–11 is consistent with Barnett's value. Barnett also states that this RE holds for all sample sizes. This conclusion is tested in example 3–12 for a sample size of $n = 10$.

3–12

```
n <- 10
GiniVar    <- var( replicate(nsim, Barnett(n)    ))
ClassicVar <- var( replicate(nsim, var(rnorm(n)) ))
RE <- ClassicVar/GiniVar # relative efficiency

print(RE)
[1] 0.8752343
```

Example 3–12 estimates $\mathrm{RE} \approx 0.88$ for $n = 10$, which is clearly at odds with Barnett's statement—RE is in fact substantially related to sample size. Another experiment shows that $\mathrm{RE} \approx 0.93$ for $n = 20$. Finally, the performance (bias) of the Gini mean difference (equivalently, the second L-moment) compared to the sample standard deviation is explored in Section 7.2.1. ◀

[4] Relatively efficiency RE is formally defined in eq. (4.7).

3.3 Summary

In this chapter, order statistics are formally introduced, and 12 examples are provided. The order statistics are based on ordering or sorting the random variable or the sample data. The order statistics are a fascinating class of statistics, which are relatively obscure to nonstatisticians, yet ironically are within the natural experience of virtually all persons—for example the minimum and maximum and to a lesser degree the median. The primary results shown in the chapter are the expression for the expectation of an order statistic, the Sen weighted mean, and the Gini mean difference. The expectation of an order statistic has great importance for the remainder of this dissertation. Foreshadowing, the L-moments and TL-moments of Chapter 6, the theoretical and numerical connections between these and both the Sen weighted mean and Gini mean difference are shown.

Chapter 4
Product Moments

In this chapter, I present generally salient background context for the remainder of the dissertation. The chapter primarily focuses on the definitions and sample counterparts of the product moments. Because the product moments are expected to be familiar to many readers, this chapter serves as a relatively independent component of the larger dissertation and establishes a basic structure for the parallelism of the two chapters on probability-weighted moments and L-moments. The topic of sampling bias and sampling variance very likely is new material to readers lacking a statistical background, but the topics are important to understand for the discussions that justify the author's preference towards use of L-moment statistics. Additionally, the discussion of bias and boundedness as a function of sample size of some product moments is particularly influential albeit not well known. Direct use of the results in this chapter is not expected for purposes of distributional analysis with L-moment statistics using R.

4.1 Introduction

Data are distributed, and data are acquired through sampling (ideally substantial sampling) of a random variable. One of the challenges before the practitioner of distributional analysis is the reduction of a sample of many numbers to geometric characterization of a distribution by a few "more salient" numbers. This reduction can be made by computing percentiles such as $x_{0.10}$, $x_{0.25}$, $x_{0.50}$, $x_{0.75}$, and $x_{0.90}$ for the 10th, 25th, 50th, 75th, and 90th percentiles, respectively; by computing other distribution metrics such as the sample range; or by computing the **statistical moments** (a generic meaning of the term). The moments of a distribution are particularly useful because specific mathematical opera-

tions are readily performed to compute moments on either distribution functions or their samples.

Moments are statistics that quantify different components or geometric characteristics of a distribution. For example, the arithmetic mean locates the distribution on the real-number line $\mathbb{R}$ and therefore is an expression of central tendency, and the standard deviation describes the variability or spread along $\mathbb{R}$. These are but two well-known examples of a moment type known as the **product moments**. There are, however, many different ways that moments can be defined and computed. As seen throughout this dissertation, there also are probability-weighted moments, L-moments, trimmed L-moments, and other variations.

The product moments such as the mean, standard deviation, skew, and kurtosis are familiar statistics—the others listed at the end of the previous paragraph are less so. The product moments are used in elementary examples in Chapter 2. In contrast, formal definitions and some experiments with their sampling properties are provided in this chapter. Before product moments are introduced, a review of some statistical concepts and terminology is needed. The review provides background for some of the examples used in this chapter and elsewhere in this dissertation.

4.1.1 Sampling Bias and Sampling Variance

The concepts of **sampling bias** and **sampling variance** (Stedinger and others, 1993, chap. 18, p. 10) involve the **accuracy** and **precision** of statistical estimation. Because distributional analysis inherently involves finite samples, the concepts of sampling bias and variance are important. R-oriented treatments of these and related concepts are provided by Rizzo (2008, pp. 37–38) and Ugarte and others (2008, pp. 245–255). For a given circumstance perhaps statistics such as moments, percentiles, or distribution parameters are to be estimated. Whichever is the case, consider the estimated statistic $\hat{\Theta}$ as a random variable with a true value that is denoted as Θ. Values for $\hat{\Theta}$ are dependent on the sampled data. The **bias** in the estimation of $\hat{\Theta}$ is defined as the difference between the expectation of the estimate minus the true value or

$$\text{Bias}[\hat{\Theta}] = \text{E}[\hat{\Theta}] - \Theta \tag{4.1}$$

The sample-to-sample variability (or sampling variance) of a statistic is expressed by **root mean square error**, which is defined as

$$\mathrm{RMSE}[\hat{\Theta}] = \sqrt{\mathrm{E}[(\hat{\Theta} - \Theta)^2]} \tag{4.2}$$

and upon expansion the error is split into two parts

$$\mathrm{RMSE}[\hat{\Theta}] = \sqrt{\mathrm{Bias}[\hat{\Theta}]^2 + \mathrm{E}[(\hat{\Theta} - \mathrm{E}[\hat{\Theta}])^2]} \tag{4.3}$$

or

$$\mathrm{RMSE}[\hat{\Theta}] = \sqrt{\mathrm{Bias}[\hat{\Theta}]^2 + \mathrm{Var}(\hat{\Theta})} \tag{4.4}$$

The square of the RMSE is known as the **mean square error** (MSE). Rizzo (2008, p. 155) reports for MSE, but shown here as RMSE, that

$$\mathrm{RMSE}[\hat{\Theta}] = \sqrt{\frac{1}{m}\sum_{j=1}^{m}(\hat{\Theta}^{(j)} - \Theta)^2} \tag{4.5}$$

where $\Theta^{(j)}$ is the estimator for the jth sample of size n and m is the number of simulation runs of samples of size n.

$\mathrm{Bias}[\hat{\Theta}]$, $\mathrm{Var}[\hat{\Theta}]$, and $\mathrm{RMSE}[\hat{\Theta}]$ are useful measures of **statistical performance**. They are performance measures because the sampling bias and sampling variance describe the accuracy and precision, respectively, of the given estimator.

If $\mathrm{Bias}[\hat{\Theta}] = 0$, then the estimator is said to be **unbiased**. For an unbiased estimator, the sampling variance will be equal to the variance or $\mathrm{Var}(\hat{\Theta})$ of the statistic. These two measures of statistical performance can exhibit considerable dependency on sample size n.

Amongst an ensemble of estimators, the estimator with the smallest $\mathrm{RMSE}[\hat{\Theta}]$ or $\mathrm{MSE}[\hat{\Theta}]$ is said to be the most **statistically efficient**. If an estimator is resistant to large changes because of the presence of outliers or otherwise influential data values, then the estimator is said to be **robust**. The **relative efficiency** of two estimators is

$$\mathrm{RE}[\hat{\Theta}_1, \hat{\Theta}_2] = \frac{\mathrm{MSE}[\hat{\Theta}_2]}{\mathrm{MSE}[\hat{\Theta}_1]} \tag{4.6}$$

and when two estimators are unbiased, then the relative efficiency can be defined as

$$\mathrm{RE}[\hat{\Theta}_1, \hat{\Theta}_2] = \frac{\mathrm{Var}[\hat{\Theta}_2]}{\mathrm{Var}[\hat{\Theta}_1]} \tag{4.7}$$

Relative efficiency is important in assessing or otherwise comparing the performance of two estimators. Relative efficiency saturates the literature of statisticians exploring the performance of estimators but has limited role in this dissertation.

USING R ——— ——— USING R

Sampling bias and sampling variance are used as metrics to evaluate and compare the properties of product moments, L-moments, and other statistics. For the sake of brevity, the R functions `mean()`, `sd()`, and occasionally `summary()` will be used to compute statistics of the difference $\hat{\Theta} - \Theta$. However, an opportunity is taken in this USING R to delve into statistics of the difference $\hat{\Theta} - \Theta$ in more detail.

In example 4–1, the function `afunc()` is defined as a high-level interface to the distribution of choice. For the example, the random variates for the standard Normal distribution are accessed using the `rnorm()` function. This style of programming is shown in order to make extension to non-standard R distibutions easier, and such a programming practice is known as abstraction. The function `sam.biasvar()` is defined next to compute eqs. (4.1) and (4.4) as well as $\mathrm{Var}[\hat{\Theta}]$.

4–1

```
MN <- 0; SD <- 1 # parameters of standard normal
# Define a separate function to implement a distribution
"afunc" <- function(n,mean,sd) {
   return(rnorm(n, mean=mean, sd=sd))
}
nsim <- 100000; n <- 10 # no. simulations and sample size to sim.

# Define function to compute sampling statistics
"sam.biasvar" <- function(h,s, verbose=TRUE, digits=5) {
   b <- mean(h) - s     # solve for the bias

   mse  <- mean((h - s)^2) # mean square error
   rmse <- sqrt(mse)       # root MSE

   vh   <- sqrt(mean((h - mean(h))^2)) # sqrt(variance
   # of the statistic), which lacks a n-1 division

   nv   <- sqrt(rmse^2 - b^2) # alternative estimation
```

```
   if(verbose) {
    cat(c("Bias␣(B)␣␣␣␣␣␣␣␣␣=␣",round(b,   digits=digits), "\n",
          "MSE(h,s)␣␣␣␣␣␣␣␣␣=␣",round(mse, digits=digits), "\n",
          "RMSE(h,s)␣␣␣␣␣␣␣␣=␣",round(rmse,digits=digits), "\n",
          "sqrt(Var(h))␣␣␣␣␣=␣",round(vh,  digits=digits), "\n",
          "sqrt(RMSE^2-B^2)␣=␣",round(nv,  digits=digits), "\n"),
        sep="")
   }
   return(list(bias=b, mse=mse, rmse=rmse, sd=vh))
}
```

The `sam.biasvar()` function is demonstrated in example 4–2 for a sample of size $n = 10$ for a large simulation size `nsim=100000`. First, the `Rmean` list is generated to hold the sampling statistics of the `mean()` function, and second, the `Rmedn` list is generated to hold the sampling statistics of the `median` function. The reported biases are near zero because the mean and median are both unbiased estimators.

4–2

```
# Sampling statistics of the mean()
Rmean <- sam.biasvar(replicate(nsim,mean(afunc(n,MN,SD))),MN)
Bias (B)          = -0.00158
MSE(h,s)          = 0.10058
RMSE(h,s)         = 0.31714
sqrt(Var(h))      = 0.31713
sqrt(RMSE^2-B^2)  = 0.31713
# Report the theoretical to show equivalence
cat(c("Theoretical␣=␣",
      round(SD/sqrt(n), digits=3), "\n"), sep="")
Theoretical = 0.316

# Sampling statistics of the median()
Rmedn <- sam.biasvar(replicate(nsim,median(afunc(n,MN,SD))),MN)
Bias (B)          = 0.00132
MSE(h,s)          = 0.13717
RMSE(h,s)         = 0.37036
sqrt(Var(h))      = 0.37036
sqrt(RMSE^2-B^2)  = 0.37036

RE <- (Rmean$sd/Rmedn$sd)^2 # RE^{mean}_{median} in LaTeX
cat(c("Relative␣efficiency␣=␣",
      round(RE,digits=3), "\n"), sep="")
Relative efficiency = 0.733
```

A natural followup question concerning the mean and the median is asked. Which has the smaller sampling variance? The end of example 4–2 reports that the $\mathrm{RE}[\mathrm{mean}, \mathrm{median}] \approx 0.73$, which is less than unity so the conclusion is that the arithmetic mean has a smaller

sampling variance than the median for at least the Normal distribution as used here. Finally, a previous demonstration of MSE computation is made for a trimmed mean and the median using `sam.biasvar()` in example 3–8. ◀

4.2 Product Moments—Definitions and Math

The product moments are formally defined in this section and are separately introduced as theoretical and sample counterparts.

4.2.1 Theoretical Product Moments

The **theoretical product moments** of random variable X are defined by centering differences on the mean μ. These product moments also have been historically termed the **central product moments** because second- and higher-order product moments are based on differences from the mean. The first product moment is the **mean**, and as previously stated, the mean measures the location of the distribution on the real-number line $\mathbb{R}$. The mean is defined as

$$\mu = \mathrm{E}[X] \tag{4.8}$$

where $\mathrm{E}[X]$ is the expectation of X having PDF $f(x)$ or

$$\mathrm{E}[X] = \int_{-\infty}^{\infty} x\, f(x)\, \mathrm{d}x \tag{4.9}$$

Higher-order product moments are defined in terms of expectations of powers of differences from μ

$$M_r = \mathrm{E}[(X - \mu)^r] \quad \text{for } r \geq 2 \tag{4.10}$$

and in integral form

$$\mathrm{E}[(X - \mu)^r] = \int_{-\infty}^{\infty} (x - \mu)^r\, f(x)\, \mathrm{d}x \tag{4.11}$$

The quantity M_2 is known as the **variance** of the distribution, which is familiarly written as σ^2. An often useful measure is the **standard deviation** σ or

$$\sigma = \sqrt{\sigma^2} \tag{4.12}$$

because σ has the same units as μ. The σ also is useful because the magnitude of the number is more similar to the μ than is σ^2 and similar scientific notation can be used in written communication when needed.

It is often convenient to remove dimension from the higher product moments for $r \geq 2$ and form the **product moment ratios**. In particular, the common ratios are coefficient of variation CV, skew G, and kurtosis K of a distribution and are defined as the three dimensionless quantities

$$CV = \sigma/\mu = \textbf{coefficient of variation} \tag{4.13}$$

$$G = M_3/M_2^{3/2} = \textbf{skew} \tag{4.14}$$

$$K = M_4/M_2^2 = \textbf{kurtosis} \tag{4.15}$$

It is typical for the term "ratio" to be dropped in reference to CV, G, and K, and refer to these three statistics as product moments. This practice will generally be adhered to here.

4.2.2 Sample Product Moments

The **sample product moments** for a random sample $x_1, x_2, \cdots, x_n$ are ubiquitous sample statistics throughout all branches of statistics, science and engineering, and society. The **sample mean** $\hat{\mu}$ is by far the most common and is taught to students even before adulthood and is computed by

$$\hat{\mu} = \frac{1}{n}\sum_{i=1}^{n} x_i \tag{4.16}$$

and the higher product moments are computed by

$$\overline{M}_r = \frac{1}{n}\sum_{i=1}^{n}(x_i - \hat{\mu})^r \quad \text{for } r \geq 2 \tag{4.17}$$

It is important to emphasize that these two statistics are only estimates of the true underlying and generally unknown values μ and M_r.

The $\overline{M}_r$ unfortunately are biased and in practice so-called unbiased estimators are used instead. An unbiased estimator of the **sample variance** $\hat{\sigma}^2$ is

$$\hat{\sigma}^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \hat{\mu})^2 \tag{4.18}$$

and note the division by $n-1$ instead of n as seen for the theoretical definition ($\overline{M_2}$). By convention, the **sample standard deviation** is

$$\hat{\sigma} = \sqrt{\hat{\sigma}^2} \tag{4.19}$$

and the **sample coefficient of variation** is $\hat{CV} = \hat{\sigma}/\hat{\mu}$.

Although $\hat{\sigma}^2$ is an unbiased estimator of variance, $\sqrt{\hat{\sigma}^2}$, as demonstrated in this chapter, paradoxically it is not an unbiased estimator of σ. However, $\sqrt{\hat{\sigma}^2}$ is in common use as exemplified by the `sd()` function of R and perhaps all other widely available statistical and spreadsheet programs. An unbiased estimator of σ is slightly more complicated. The uniformly-minimum variance unbiased estimator of σ (David, 1981, p. 185) is

$$\hat{\sigma}' = \frac{\Gamma[(n-1)/2]}{\Gamma(n/2)\sqrt{2}}\sqrt{\sum_{i=1}^{n}(x_i - \hat{\mu})^2} \tag{4.20}$$

where $\Gamma(a)$ is the complete gamma function that is shown in eq. (8.85) and is obtained by the `gamma()` function.[1] The $\hat{\sigma}'$ estimator of σ will be compared to $\hat{\sigma}$ by example in this chapter.

The sample variance can be written in an unusual way based on sample order statistics. As given by Jones (2004, p. 99) after Yatracos (1998), two expressions for the sample variance are

$$\tilde{\sigma}^2 = \frac{2}{n^2}\sum_{i=1}^{n-1} i(n-i)(\overline{x_{\{i,i+1\}}} - \overline{x_{\{1,i\}}})(x_{[i+1:n]} - x_{[i:n]}) \tag{4.21}$$

$$\tilde{\sigma}^2 = \frac{2}{n^2}\sum_{i=1}^{n-1} i(n-i)(\overline{x_{\{i+1,n\}}} - \overline{x_{\{i,i+1\}}})(x_{[i+1:n]} - x_{[i:n]}) \tag{4.22}$$

where $\tilde{\sigma}^2$ is the sample variance, $x_{[i:n]}$ are the usual sample order statistics, and $\overline{x_{\{j,k\}}}$ for $j \le k$ is the mean of $x_{[j:n]}, \ldots, x_{[k:n]}$. Numerical experiments, which are not reported here,

[1] Actually the computation of the `gamma()` function for large arguments—sample sizes in the case here—is problematic. The computationally preferred technique is to use logarithms: `exp(lgamma((n-1)/2) - lgamma(n/2))` (see eq. (3.8)). This technique is used in the `pmoms()` function of *lmomco*.

indicate that the two expressions yield numerically equivalent values for $\tilde{\sigma}^2$. A function[2] implementing the first expression for $\tilde{\sigma}$ (standard deviation) is shown in example 4–3. Numerical experiments suggest that $\tilde{\sigma}$ is about -1 in the *fifth* significant figure less than $\hat{\sigma}$ of eq. (4.19).

4–3

```
"ostat.sd" <- function(x) {
   x <- sort(x); n <- length(x)
   tmp <- sapply(1:(n-1),
                 function(i) { ip1 <- i + 1
         return(i*(n-i)*(mean(x[i:ip1]) - mean(x[1:i]))*
                  (x[ip1] - x[i])) })
   return(sqrt(2/n^2*sum(tmp)))
}
```

Continuing with the higher product moments, a nearly unbiased estimator of **sample skew** $\hat{G}$ is

$$\hat{G} = \frac{\overline{M}_3}{\hat{\sigma}^3} \times \frac{n^2}{(n-1)(n-2)} \tag{4.23}$$

A nearly unbiased estimator of **sample kurtosis** $\hat{K}$ is

$$\hat{K} = \frac{1}{\hat{\sigma}^4} \times \frac{n^2}{(n-2)(n-3)} \times \left[\left(\frac{n+1}{n-1} \right) \overline{M}_4 - 3\overline{M_2}^2 \right] + 3 \tag{4.24}$$

and care should be exercised with $\hat{K}$ because its definition can vary between software packages—consult program documentation for further details.

As discussed throughout Hosking and Wallis (1997) and numerous other authors, as well as generally well known among advanced practitioners, the estimators of the product moments have many undesirable properties (Gilchrist, 2000, p. 197) under departure from distribution symmetry.

For example, Hosking and Wallis (1997, p. 18) conclude that inferences based on sample product moments from skewed "distributions are likely to be very unreliable," and argue that L-moments provide "a more satisfactory" means of distribution characterization. In particular, L-moments might be preferable to the product moments for characterization of distribution shape as advocated by Hosking (1992) and Royston (1992). A particularly influential, yet succinct discussion, of the weaknesses of sample product moments in a hydrologic context is provided by Wallis and others (1974). A guide to some sampling

[2] The function (or method) is not an efficient means to compute the standard deviation.

properties of product moments through numerical experiments is provided in the next section.

4.3 Some Sampling Properties of Product Moments

In this section, a topical exploration of the product moments and some of their sampling properties is made using built-in R functions, direct computation, and the `pmoms()` function. The `pmoms()` function was explicitly written and included in the *lmomco* package to facilitate comparisons between product and L-moments. Examples for computation of the sample product moments are shown and are presented in several similar constructs (code parallelism) to demonstrate graphically or numerically the sampling properties of bias and boundedness.

4.3.1 The Mean, Standard Deviation, and Coefficient of Variation

The mean μ and standard deviation σ are for good reason perhaps the most popular statistics of samples. The μ is a measure of the location of the data, and σ is a measure of the variation of the data about μ. The dimensionless CV can be useful in some applications because it is an expression of relative variability or variation that often is independent of the scale of many phenomena.

USING R ——— ——— USING R

Example 4–4 demonstrates the computation of $\hat{\mu}$, $\hat{\sigma}$, and $\hat{CV}$ for a small, hand-made data set in `fake.dat` using the `mean()` and `sd()` functions. The `cat()` function concatenates and prints each element of a list. The `c()` function is used to build this list. For more attractive output, the `round()` function is used to round to selected decimal places.

4–4

```
fake.dat <- c(123, 546, 345.2, 12, 875, 321, 90, 800)
mu <- mean(fake.dat); sig <- sd(fake.dat); cv <- sig/mu
cat(c(round(mu, digits=2), round(sig, digits=2),
      round(cv, digits=3), "\n"))
389.02 324.83 0.835
```

The example reports all three values in the indicated order: $\hat{\mu} = 389.02$, $\hat{\sigma} = 324.83$, and $\hat{CV} = 0.835$. ◀

4.3.2 Bias of Standard Deviation

The sample estimator of standard deviation $\hat{\sigma}$ computationally is simple. The division by n in the computation of the $\hat{\mu}$ seems intuitively reasonable, but why is there the $(n-1)$ term in the computation of $\hat{\sigma}^2$ and what is the purpose of the term?

Does the $(n-1)$ term mean that we do not compute the average (straight division by n) square deviation? Yes, it does. Speaking frankly, in the author's first college statistics course as a student (an introductory undergraduate course in Mechanical Engineering), the students were simply told something like "you give up a degree of freedom because the mean itself requires estimation," and no other discussion is recalled. Ok—but what does "degree of freedom" mean?

The author was unsatisfied with the paraphrased answer. Many years after that, during the study (by necessity) of L-moments, the concept of sample statistics as *estimators* of unknown population values was made manifest. This dissertation is a result of a legacy of deep reflection and insatiable curiosity resulting from that first statistics course.

The message to convey is that individual estimators have their own unique statistical properties. With a simple n term in the denominator, σ^2 is on average underestimated and division by a "corrected" sample size compensates. In distributional analysis, interests often are in the expression of variability in the same units as the mean. As a result, interest commonly involves estimation of σ, and a simple square-rooting of the sample variance ($\sqrt{\hat{\sigma}^2}$) might not be sufficient.

What does degree of freedom mean? Spatz (1996, p. 188) states "the 'freedom' in *degrees of freedom* [Spatz's italics] refers to the freedom of a number to have any possible value." Spatz (1996) continues with further detailed description and attributes an explanatory quotation to Walker (1940) who states, "A universal rule holds—The number of degrees of freedom is always equal to the number of observations minus the number of necessary relations obtaining among these observations."

USING R ——— ——— USING R

Example 4–5 concretely demonstrates, through numerical output, the bias inherent in the standard deviation. The example involves the idea of statistical simulation, sampling error, and statistics of statistics. Suffice to say that when the example is executed, the reader can confirm that the $(n-1)$ definition of standard deviation, which in fact is used in the `sd()` function, provides a closer estimate to `sd=10000`.

4–5

```
# two vectors to hold sample estimates of standard deviation
bias.tmp <- unbias.tmp <- vector(mode="numeric")
n <- 30; nsim <- 1000 # sample size and no. of simulations
for(i in seq(1,nsim)) {
   # sim. Normal dist. with large standard deviation
   fake.dat <- rnorm(n, mean=0, sd=10000)
   # compute the sample mean of the count-th simulation
   mu <- mean(fake.dat)
   # theoretical definition of standard deviation
   bias.tmp[i]   <- sqrt(sum((fake.dat-mu)^2)/n)
   unbias.tmp[i] <- sd(fake.dat) # unbiased sigma^2 estimate
}
# compute summary of each vector of simulated standard devs
summary(bias.tmp)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   6263    8855    9670    9700   10520   13760

summary(unbias.tmp)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   6370    9007    9835    9865   10690   14000
```

The example reports by two calls to the `summary()` function that $\sqrt{\overline{M_2}} = 9{,}700$ and $\sqrt{\hat{\sigma}^2} = 9{,}865$, and the latter being closer to $10{,}000$ is obviously the preferable estimator of the two. ◀

The minimum variance unbiased estimator of σ was defined earlier as $\hat{\sigma}'$ in eq. (4.20). How does $\hat{\sigma}'$ measure up against $\hat{\sigma}$? Example 4–6, which uses an order of magnitude more simulations than used in example 4–5 and for the same population moments ($\mu = 0$ and $\sigma = 10{,}000$), compares the three estimators. For the example, the sample size has been reduced by a third to $n = 10$. The example shows that the $\hat{\sigma}' = 9{,}987$ indeed provides less biased estimates of σ than $\hat{\sigma}$ and that $\hat{\sigma}'$ is consistently closer to $10{,}000$ (see `summary(umvubias.tmp)`). This particular example uses the `pmoms()` function. The `pmoms()` function simultaneously supports M_2, $\hat{\sigma}$, and $\hat{\sigma}'$ computations of σ. The

`pmoms()` function returns an R list that is referred to as an "***lmomco* product moment list**."

4–6

```
# two vectors to hold sample estimates of standard deviation
bias.tmp   <- vector(mode="numeric")
unbias.tmp <- umvubias.tmp <- bias.tmp
n <- 10; nsim <- 10000 # sample size and no. of simulations
for(i in seq(1,nsim)) {
   fake.dat <- rnorm(n, mean=0, sd=10000)
          # a large standard deviation?
   pm <- pmoms(fake.dat) # returns lmomco product moment list

   bias.tmp[i]     <- pm$classic.sd  # square root of M2
   unbias.tmp[i]   <- pm$sd          # sigma hat
   umvubias.tmp[i] <- pm$umvu.sd     # sigma hat prime
}
# compute summary of each vector of simulated standard dev
summary(bias.tmp)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   1553    7697    9111    9215   10640   20060

summary(unbias.tmp)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   1637    8114    9603    9714   11220   21150

summary(umvubias.tmp)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   1683    8342    9873    9987   11530   21740
```

◀

4.3.3 Bias and Boundedness of Coefficient of Variation

The statistic $\hat{CV}$ is biased. The sample estimator $\hat{CV}$ underestimates CV in part because $\hat{\sigma}$ underestimates σ. The ratio σ/μ is on average too small.

Dalen (1987, p. 329) reports that "it is an established but not well-known fact that many types of sample statistics are algebraically bounded by some function of sample size." Concerning this dissertation, the bounds of $\hat{CV}$ and $\hat{G}$ (see Section 4.3.4) are of interest. Kirby (1974) provides applicable discussion of the sample size boundedness of $\hat{CV}$ as well as $\hat{G}$ and other statistics. Further discussion is provided by Wallis and others (1974).

For a strictly positive distribution, the $\hat{CV}$ ($\hat{CV} = \hat{\sigma}/\overline{\mu}$) is bounded (Kirby, 1974) according to sample size n. Specifically, $\hat{CV}$ can attain values no larger than

$$\hat{CV} \leq \sqrt{(n-1)} \tag{4.25}$$

regardless of how large the CV is of the distribution from which the sample was drawn. The property of $\hat{CV}$ boundedness can be especially disturbing for distributions possessing large relative variability—that is, CV values near the upper bounds for the sample size of a given data set. The product moments, as a result, could be considered as unsatisfactory estimators of relative variability for highly-dispersed samples.

USING R

For a demonstration, which will be returned to later in Chapter 6 in the context of sample L-moments as estimators, a Gamma distribution having $\mu = 3{,}000$ (`True.MU=3000`) and $CV = 10$ (`True.CV`) is defined in example 4–7. These statistics result in $\sigma = 30{,}000$ or `True.SD=30000`. The `help()` function for a random Gamma variate (the `rgamma()` function) reports the relation between the product moments of the distribution and the shape (`a`) and scale (`s`) parameters. The algebra is shown in the last line of example 4–7.

4–7

```
True.MU <- 3000; True.CV <- 10 # population statistics
True.SD <- True.MU*True.CV; True.VAR <- True.SD^2
help(rgamma) # use to lookup equations for parameters
s <- True.VAR/True.MU; a <- True.MU/s # the parameters
```

The demonstration continues in example 4–8 by the creation of a vector `nsam` of sample sizes. A portable document format (PDF) graphics device at `version="1.4"` (or better) is initiated by `pdf()` because the feature of transparency provided by the `rgb()` color function will be used.[3]

Continuing, the number of simulations `nsim` is set at 500. The `plot()` function immediately follows to initiate the graphics. Next, an outer `for()` loop is initiated to loop through the samples sizes in `nsam`. The inner `for()` loop runs the simulations on samples drawn from the Gamma distribution with the `rgamma()` built-in R function. The $\hat{CV}$ values are computed by `sd(x)/mean(x)` and stored in `cvtmp`. The estimate of $\hat{CV}$ for the each sample size is computed by `mean(cvtmp)` and stored in the variable `cv`. The

[3] Transparency is supported in portable document format (PDF) version greater than or equal to `"1.4"`, and transparency is not supported by all graphics devices supported by R.

`points()` function, with each operation, plots a single semi-transparent red filled circle. The results of the simulation are shown in figure 4.1.

4–8

```
nsim <- 500
nsam <- c(5,   8, 10,  14,  16,  20,  25,  30,  40, 50,
          60, 70, 80, 100, 120, 140, 160, 180, 200)
#pdf("cv.pdf", version="1.4")
plot(c(0), c(0), type="n", xlab="SAMPLE␣SIZE", ylab="CV",
     xlim=range(nsam),   ylim=c(1,1.5*True.CV))

counter <- 0
cv <- cvtmp <- vector(mode="numeric")
for(n in nsam) {
   counter <- counter + 1
   for(i in seq(1,nsim)) {
      x <- rgamma(n, shape=a, scale=s) # GAM(a,s)
      tmp <- sd(x)/mean(x); cvtmp[i] <- tmp
      points(n,tmp, pch=16, col=rgb(0.5,0,0,0.05))
   }
   cv[counter] <- mean(cvtmp)
}
lines(nsam,cv, lwd=3) # solid thick line
lines(nsam, sqrt(nsam-1), lty=2) # dashed line (bounds)
abline(True.CV,0) # line of true value
#dev.off()
```

In the figure, the dashed line represents the $\sqrt{(n-1)}$ upper limit. The true $CV = 10$ is shown by the solid horizontal line. The thick line represents the mean of 100 simulated sample values for each sample size. (There are 100 symbols within each vertical strip.) For very small samples sizes, it is seen that the sample estimate of CV generally is severely limited because of the $\sqrt{(n-1)}$ bounds and as sample size increases to $n = 200$, the expected value of $\hat{CV}$ is about 7.8. The $\hat{CV}$ is biased low because of the underestimation sampling property of $\hat{\sigma}$. The figure is but one example that could be constructed for different parent distributions. Figures such as 4.1 show that the product moments can have considerable limitations for distributions having large relative variation. This example is considered again in the context of L-moments in Chapter 6 and specifically in Section 6.5.4. ◀

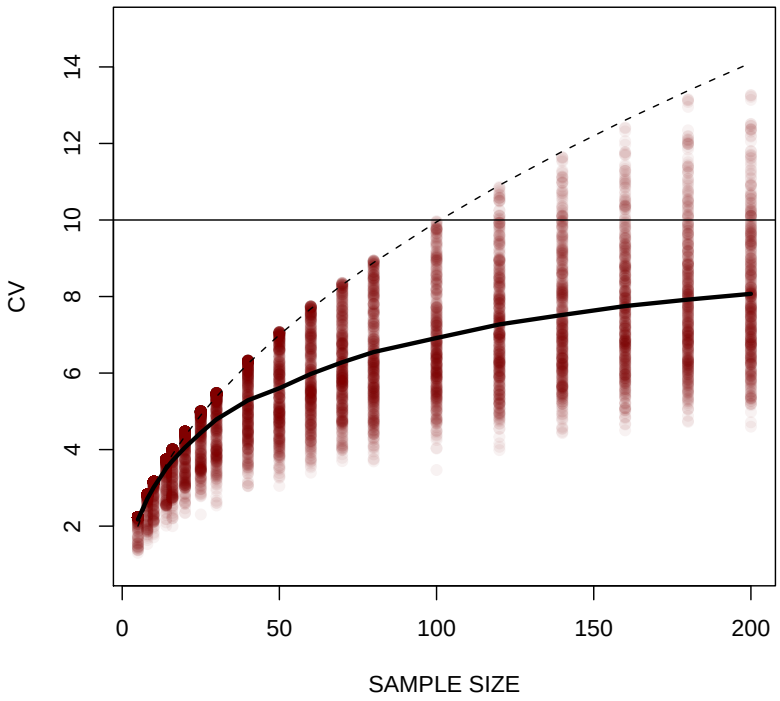

Figure 4.1. Demonstration of upper limit boundedness (dashed line) and bias of $\hat{CV}$ (thick solid and curved line) as computed by 500 simulations for each sample size for a Gamma distribution having $\mu = 3{,}000$ and $CV = 10$ (solid horizontal line) from example 4–8

4.3.4 Bias and Boundedness of Skew

In small samples, $\hat{G}$ is a severely biased and bounded (Kirby, 1974) statistic; the $\hat{G}$ underestimates the population G. Specifically, the magnitude of $\hat{G}$ can attain values no larger than

$$|\hat{G}| \leq \frac{n-1}{\sqrt{n-2}} \tag{4.26}$$

where n is sample size. Dingman (2002, p. 559, eq. CB2-10) reports eq. (4.26) as $|g_x| = (n-2)/\sqrt{n-1}$, which is in obvious algebraic conflict. However, it is shown in figure 4.3 through simulation that eq. (4.26) is in fact correct.

The bias and bounds of $\hat{G}$ are so severe that $\hat{G}$ could be rendered essentially useless for the analysis of highly skewed data. Highly skewed data are particularly common in hydrometeorological or other earth-system data sets that could have orders of magnitude of range and great asymmetry. The $\hat{G}$ should be used with some caution; although $\hat{G}$ might remain a reasonable expression of the direction of distribution asymmetry.

The effects of the boundedness and negative bias (underestimation) of $\hat{G}$ are readily demonstrated by statistical simulation using the *lmomco* package. The Pearson Type III distribution is selected. The Pearson Type III distribution is particularly interesting to study using product moments because the parameters of the distribution are the first three product moments in a similar fashion as the first two product moments are parameters of the Normal distribution. Therefore, comparisons of skewness estimators using the Pearson Type III distribution are readily made.

For a demonstration that begins in example 4–9, a Pearson Type III distribution with parameters $\mu = 1000$, $\sigma = 500$, and $G = 5$ or PE3(1000, 500, 5) is specified using the `vec2par()` (vector to parameters) function. The `nonexceeds()` function returns a useful vector of F values and `quape3()` function returns the quantiles of the distribution as set by the `pe3` parameters. This Pearson Type III distribution is shown in figure 4.2.

4–9

```
#pdf("pe3experimentA.pdf")
True.Skew <- 5
pe3 <- vec2par(c(1000,500,True.Skew), type="pe3")
F   <- nonexceeds(); Q <- quape3(F,pe3)
plot(F,Q, type="l")
#dev.off()
```

The demonstration continues in example 4–10. The example sets up of the number of simulation runs `nsim` to perform for each of several selected sample sizes `nsam`. The vector `G` stores the mean values of $\hat{G}$ for each of the sample sizes. The `rlmomco()` function is used to generate random variables of sample size n from the Pearson Type III parent. Specifically, the `rlmomco()` function returns simulated values by dispatching to the QDF of the Pearson Type III distribution. Similar random variable generation was performed in example 4–9 using the `quape3()` function. The `rlmomco()` function actually dispatches to the `quape3()` function. The correct dispatch is made because the content in the `type` field of the `pe3` *lmomco* parameter list declares the distribution as Pearson Type III (see example 7–1 on page 163). The `pmoms()` function computes the product moments of the simulated sample. For this particular study, interest is in the $\hat{G}$ returned by `pmoms()`, and therefore, $\hat{G}$ for each simulation run is stored in the vector `sG`. Finally, the example ends by plotting the results of the experiment in figure 4.3. The solid line in the figure is the upper bounds of G that is set by the sample size.

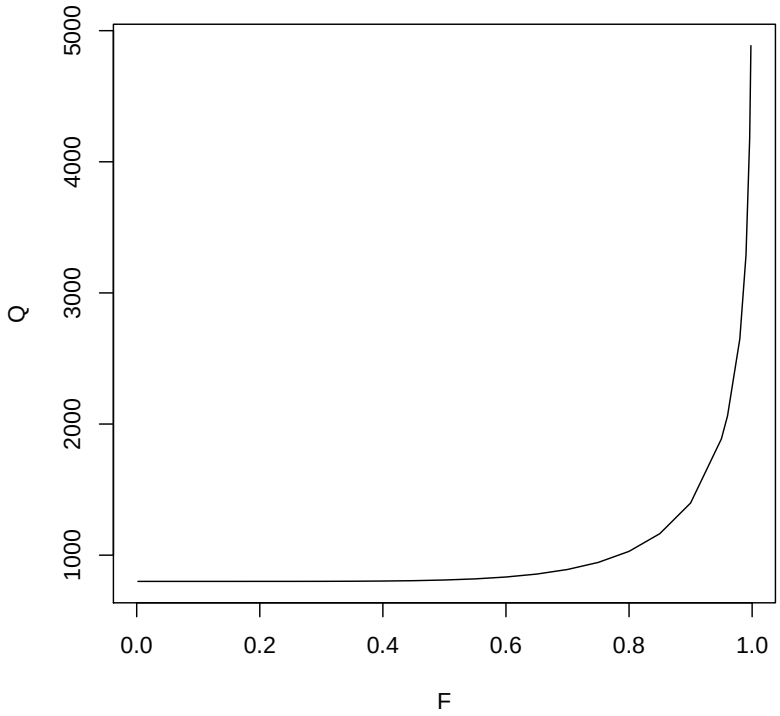

Figure 4.2. Parent Pearson Type III distribution of PE3(1000, 500, 5) used to assess bias in product moment skew from example 4–9

4–10

```
nsim <- 500
nsam <- c(6, 8, 10, 12, 15, 20, 25, 30)
counter <- 0
G <- sG <- vector(mode = "numeric")

#pdf("pe3experimentB.pdf", version="1.4")
plot(c(), c(), type="b",
     xlim=range(nsam), ylim=c(0,1.25*True.Skew),
     xlab="SAMPLE␣SIZE", ylab="PRODUCT␣MOMENT␣SKEW")
for(n in nsam) {
   for(i in seq(1,nsim)) {
      D   <- rlmomco(n,pe3)
      PM  <- pmoms(D)
      myG <- PM$ratios[3]
      sG[i] <- myG
      points(n,myG, pch=16, col=rgb(0.5,0,0,0.05))
   }
   counter <- counter + 1
   G[counter] <- mean(sG)
}
```

```
lines(nsam, G, lwd=3) # solid thick line
lines(nsam,(nsam-1)/sqrt(nsam-2), lty=2) # dashed line (bounded)
abline(True.Skew,0) # line of true value
#dev.off()
```

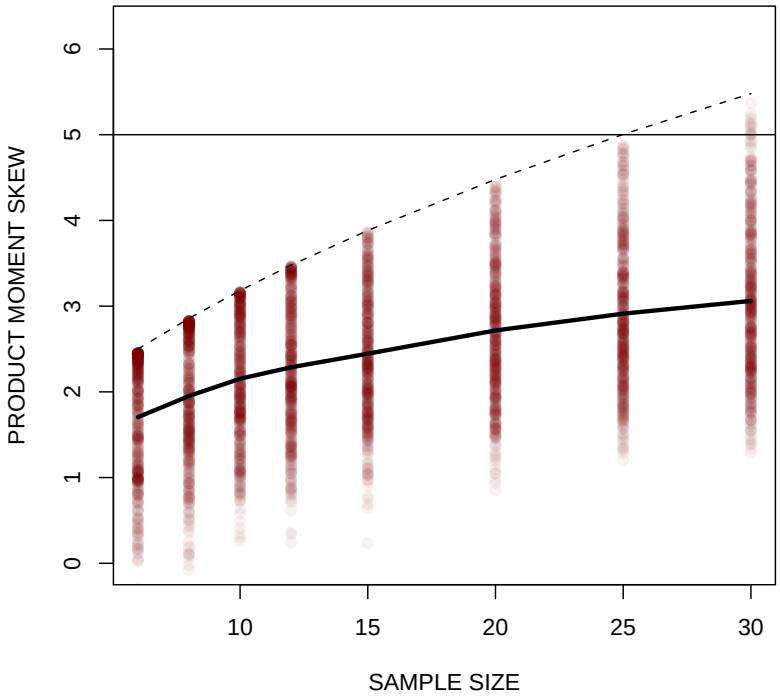

Figure 4.3. Demonstration of upper limit boundedness (dashed line) and bias of $\hat{G}$ (thick solid curved line) as computed by 500 simulations for each sample size for a Pearson Type III distribution of PE3(1000, 500, 5) ($G = 5$ and is the solid horizontal line) from example 4–10

The results of figure 4.3 demonstrate that the bias of $\hat{G}$ for a substantially asymmetrical distribution is considerable and in fact is alarming for general application of $\hat{G}$ for highly-skewed data. Further, the bias reduces slowly as sample size increases. The results also show that the boundedness of $\hat{G}$ for small sample sizes so greatly affects the estimate that it is very unlikely that any secure inference of third-order (and higher) distributional shape could be made with $\hat{G}$ for highly skewed distributions. The ramifications of figure 4.3 are substantial and far reaching. The estimator $\hat{G}$ cannot acquire G for reasonable sample sizes for the specified parent distribution for conditions of large skewness.

These shortcomings of $\hat{G}$ are reasonably known among statisticians and many other practitioners, and hence, the use of transformation of X often is recommended to reduce

skewness by increasing distribution symmetry. Logarithmic transformation is common in many disciplines and analysis of heavy-tailed distributions and the subject of the next section, and Section 12.8 also provides extensive additional discussion and computational example. ◀

4.4 On the Use of Logarithmic Transformation

It has long been understood that the sample product moments have limitations when applied to samples having large variation, heavy-tails, or substantial departures from symmetry. To mitigate for the limitations, data often are transformed into log-space. Such transformation serves the purpose of reducing variance and skewness. The reduction in skewness mitigates for the rather poor sampling properties of the product moments for highly skewed data. Additional discussion of logarithmic transformation is found in Section 6.5.5.

It is conventional in many disciplines to compute logarithms and subsequently compute the sample product moments. A chosen distribution might encompass the log-Normal for which Jensen and others (1997, p. 87) comment "Log-normal distributions appear to be at least as common in nature as normal distributions," and Qian (2010) concludes similarly. The log-Pearson Type III distribution is another example and is widely used in hydrology (U.S. Water Resources Council, 1981), and this distribution is extensively considered in Section 12.8.

It can be said that analysts employing logarithmic transformation can exchange one problem for another. The influence of low magnitude values (low outliers) now have the capacity, depending on their values and other factors, to greatly influence the computation of the sample product moments.[4] The application of logarithms in the context of L-moment capabilities is further discussed beginning on page 150. The basic message to convey for the present is that use of L-moments removes the requirement that logarithms be used—distributional analysis in real-space for heavy-tailed and non-normal data is possible. Throughout this dissertation, it is seen that reliable distributional analysis to such

[4] Treatment for low outliers is particularly important in analysis of annual peak streamflow in semiarid to arid regions like Texas. Asquith and Roussel (2009, p. 19) provide salient discussion. The low outlier problem in Texas flood hydrology, as encountered by the author circa 1995, has had a profound philosophical impact on the author's policies towards analysis of hydrometeorological data in Texas and the greater American Southwest.

data is possible without a need to apply logarithmic transformation when L-moments or probability-weighted moments are used to fit probability distributions.

4.5 Summary

In this chapter, the product moments are described, and both the theoretical and sample product moments are named and mathematically described. Principally, these are the product moments of mean, standard deviation, variance, coefficient of variation, skew, and kurtosis. The 22 examples in the chapter demonstrate the computation of these statistics and many of their properties. Among these properties are the concepts of bias and sampling variance, which also are introduced in the chapter, and how each reflects the properties of a statistical estimator is discussed. Several examples are provided and built-in R functions that are demonstrated include `mean()`, `median()`, and `sd()`. The `pmoms()` function of the *lmomco* package is used, and this function returns the first four product moments and alternative definitions of standard deviation. The bias of the standard deviation is demonstrated as is the boundedness of the coefficient of variation. The bias and boundedness of the skew also is demonstrated. Finally, a discussion of logarithmic transformation, which often is used to mitigate for the sampling properties of the product moments, is provided.

Chapter 5
Probability-Weighted Moments

In this chapter, I present a nearly exclusive discussion of the probability-weighted moments. The L-moments are simply linear combinations of these, but understanding of probability-weighted moments provides an additional prerequisite needed for accessibility into this dissertation. The probability-weighted moments are convenient in support of L-moments for censored data, but I have purposefully placed censored probability-weighted moments in a later chapter. This chapter presents the defining mathematics and sample counterparts of probability-weighted moments along with application of the two for fitting of a distribution. Although some practitioners might need few of the results herein, this chapter never-the-less is important to distributional analysis with L-moment statistics using R.

5.1 Introduction

The **probability-weighted moments** (Greenwood and others, 1979) are an alternative statistical "moment" that like the product moments, characterize the geometry of distributions and are useful for parameter estimation. The probability-weighted moments emerged in the late 1970s generally for the purposes of parameter estimation for distributions having only a QDF form. In particular, the five-parameter Wakeby distribution of Section 9.2.4 was the subject of many of the early studies. At the time, the Wakeby distribution (Landwehr and others, 1979a) seems to have been of particular discipline-specific interest for flood hydrology. However, the theory of probability-weighted moments (Hosking, 1986) and their appearance as a new tool in the statistician's tool box garnered additional interest (Landwehr and others, 1979b, 1980; Hosking and others, 1985; Ding and Yang, 1988).

The probability-weighted moments are well suited, and generally superior, to the product moments for parameter estimation for distributions of data having large skew, heavy or long tails, or outliers. Although powerful for parameter estimation, the probability-weighted moments unfortunately are difficult to individually interpret as measures of distribution geometry. For example, Ulrych and others (2000, p. 53) remark that probability-weighted moments "obscure the intuitive understanding of L-moments."[1] By the mid 1980s, the probability-weighted moments were reformulated into the L-moments, which were unified by Hosking (1990) and are formally described in Chapter 6.

The L-moments are readily interpreted in similar fashions as the product moments in Chapter 4. The probability-weighted moments and L-moments are linear combinations of each other. Computation of one therefore yields the other; so inferences based on either are identical. The choice of between probability-weighted moments and L-moments can be influenced by simple mathematical convenience.

Variations on probability-weighted moments exist. For example, they are amenable to situations of data censoring, and the definitions and applications of probability-weighted moments for some types of censored data are deferred to Sections 12.2 and Section 12.4 in the context of advanced topics of distributional analysis. Another variant of probability-weighted moments has been developed (Haktanir, 1997) called **self-determined probability-weighted moments**, which increases statistical performance by "utilizing mathematical properties of the underlying probability distribution" (Whalen and others, 2002, p. 177). This variant is not considered in this dissertation.

The *lmomco* package provides probability-weighted moment support, and the functions are listed in table 5.1. These functions support both theoretical and sample computations. The distinctions between the two computation types are discussed in the next section. The listed functions are thoroughly demonstrated following the Using R identifiers in this chapter and elsewhere in this dissertation.

5.2 Probability-Weighted Moments—Definitions and Math

The probability-weighted moments are formally defined in this section and separately introduced as theoretical and sample counterparts.

[1] See Section 3.1 and particularly page 62 of this dissertation related to L-moment interpretation in terms of order statistics.

Table 5.1. Summary of probability-weighted moment related functions of the *lmomco* package by Asquith (2011)

Function	Purpose
`theopwms()`	Compute theoretical probability-weighted moments by distribution
`pwm()`	Compute unbiased sample probability-weighted moments
`pwm.ub()`	Compute unbiased sample probability-weighted moments by dispatch to `pwm()`
`pwm.gev()`	Compute sample probability-weighted moments that are optimized for the Generalized Extreme Value distribution
`pwm.pp()`	Compute sample probability-weighted moments by plotting positions
`vec2pwm()`	Convert a vector to probability-weighted moments
`pwm2vec()`	Convert probability-weighted moments to a vector

5.2.1 Theoretical Probability-Weighted Moments

The **theoretical probability-weighted moments** of random variable X with a CDF of $F(x)$ and QDF of $x(F)$ are defined by the expectations

$$M_{p,r,s} = \mathrm{E}[x(F)^p \; F(x)^r \; (1 - F(x))^s] \tag{5.1}$$

where p, r, and s are integers. By historical convention, the most common probability-weighted moments β_r are

$$\beta_r = M_{1,r,0} = \mathrm{E}[x(F) \; F^r] \tag{5.2}$$

and so for a QDF $x(F)$, the β_r for $r \geq 0$ are

$$\beta_r = \int_0^1 x(F) \; F^r \; \mathrm{d}F \tag{5.3}$$

At this point, it is informative to juxtapose the definition of β_r to the product moments (noncentral, no offset of the mean μ for $r \geq 2$) and consider the mathematical similarities and differences. Noncentral product moments are the expectations

$$\mathrm{E}[X^r] = \int_0^1 [x(F)]^r \; \mathrm{d}F \tag{5.4}$$

Readers are asked to juxtapose the quantities being raised to the power r in eqs. (5.3) and eq. (5.4). In the case of product moments, the quantities x are raised to r. Whereas, for the probability-weighted moments, the nonexceedance probability values $0 \leq F \leq 1$ are raised to r. In other words, each x is weighted by a power of F, hence, the descriptive name of *probability-weighted* moment.

This subtle mathematical adjustment makes substantial changes and specific improvements to the sampling properties of the probability-weighted moments relative to the product moments. As values for the differences $x - \mu$ become large in the computation of sample product moments, these large differences have an increasingly larger influence on the estimation of the moment. In other words, relatively more weight is contributed by large differences to the computation of the moment in the product moment case. This increased proportionality of more weight does not occur with the weighting by powers of F for the probability-weighted moments in part simply because of the constraint that F is on the interval $0 \leq F \leq 1$. Additionally, the problem of disproportionally larger influence for large differences from the mean (consider the case of outliers) is made much worse by taking powers of 2, 3, 4, and larger. In the numerical summations approximating the integral, large differences are increasingly magnified as r increases beyond $r \geq 2$.

USING R

The `theopwms()` function, which implements eq. (5.3) by using the `integrate()` function, the theoretical probability-weighted moments for the standard Normal distribution are computed in example 5–1. In the example, the *lmomco* parameter list (see page 163 and ex. 7–1) for the distribution is set by the `vec2par()` function and the theoretical probability-weighted moments are set into `NORpwm`. The first two β_r are set into `B0` and `B1` by definition unique to this distribution (see Section 7.2.1). The `deltaMEAN` and `deltaSIGMA` are the respective differences, and the output by the `cat()` function shows numerical equivalency ("`# OUTPUT: 0 0`"). The first β_r is the mean or $\beta_0 = \mu$.

5–1

```
mu  <- 0 # set respective mean
sig <- 1 # and standard deviation
NORpar <- vec2par(c(0,1), type="nor")
          # standard Normal using lmomco nomenclature
NORpwm <- theopwms(NORpar) # PWMs of standard Normal

B0 <- mu # by definition
B1 <- 0.5 * ( (sig/sqrt(pi)) + mu ) # by definition
```

```
deltaMEAN  <- B0 - NORpwm$betas[1]  # difference between the
deltaSIGMA <- B1 - NORpwm$betas[2]  # two PWM computations
cat(c("#␣␣OUTPUT:", round(deltaMEAN), round(deltaSIGMA)))
#  OUTPUT: 0 0
```

◀

5.2.2 Sample Probability-Weighted Moments

The **sample probability-weighted moments** are computed for a sample from the sample order statistics $x_{1:n} \leq x_{2:n} \leq \cdots \leq x_{n:n}$. Unbiased estimators of β_r (Hosking and Wallis, 1997, p. 26) are computed by

$$\hat{\beta}_r = \frac{1}{n}\binom{n-1}{r}^{-1} \sum_{j=1}^{n} \binom{j-1}{r} x_{j:n} \tag{5.5}$$

and the so-called plotting-position estimators of β_r are computed by

$$\tilde{\beta}_r = \frac{1}{n} \sum_{j=1}^{n} \left(\frac{j+A}{n+B} \right)^r x_{j:n} \tag{5.6}$$

where A and B are plotting-position coefficients $A > B > -1$. Hosking (1986, pp. 32–33) reports that "there is no general reason to estimate [probability-weighted moments] by any particular unbiased or plotting-position estimator." The presentation of either estimator occurs throughout probability-weighted moment (and L-moment) literature; in particular, $\hat{\beta}_r$ is particularly common: (Landwehr and others, 1979b; Hosking, 1990, 1995; Zafirakou-Koulouris and others, 1998). The $j = 1$ term in eq. (5.5) is sometimes written as $j = r + 1$ with no numerical change in results because of zero multipliers on the first r terms.[2] The following example demonstrates

5–2

```
n <- 10; r <- 3
sapply(1:n,   function(j) choose(j-1,3))
 [1]  0  0  0  1  4 10 20 35 56 84
```

[2] This variation in $\hat{\beta}_r$ definition can lead to considerable frustration to early students on the subject. The author prefers the $j = 1$ notation as it is parallel with the idea of iterating through the entire sample at the minor cost of a few extra steps.

```
sapply(r+1:n, function(j) choose(j-1,3))
 [1]   1   4  10  20  35  56  84 120 165 220
```

The $\hat{\beta}_r$ are used in general practice, but in special circumstances in which a parent distribution is known, optimal values of A and B might exist. For the vast majority of applications, $\hat{\beta}_r$ are sufficient. Hosking and Wallis (1997, pp. 33–34) provide succinct discussion and pertinent literature citations. Chen and Balakrishnan (1995) make comparisons of the two probability-weighted moment estimators for the Generalized Extreme Value, Generalized Logistic, and Generalized Pareto distributions in the context of the "infeasibility" problem.

Chen and Balakrishnan defined the infeasibility problem as a situation in which the upper limit of a distribution is less than one or more of the largest sample values or in which the lower limit is greater than one or more of the smallest sample values. The authors consider, using simulation and a range of sample sizes and shape parameters, the percent of time that $\hat{\beta}_r$ and $\tilde{\beta}_r$ produce infeasible distribution parameters. The authors conclude that $\tilde{\beta}_r$ can measurably reduce the probability of infeasible parameters for certain conditions of sample size and shape. Chen and Balakrishnan (1995, p. 569) recommend that a "routine check be carried out to see whether the problem of infeasible parameter estimates occurs, and use modified probability-weighted moment estimators if the problem does occur."

Modified probability-weighted moment estimators are described on various pages by Hosking (1986) to mitigate for infeasible parameter estimates, and the algorithm is summarized by Chen and Balakrishnan (1995, p. 568) in a near verbatim quote:[3]

> Let x denote $x_{1:n}$ or $x_{n:n}$, if the boundary condition [limit] is found to be violated by the [probability-weighted moment] estimators [applicability for either unbiased and plotting-position seems apparent to this author (Asquith)] of the parameters, equate x to $\xi + \alpha/\kappa$ and solve for κ. This leads to
>
> $\kappa = -\log[(2\beta_1 - x)/(\beta_0 - x)]/\log(2)$ for the Generalized Extreme Value,
>
> $\kappa = (2\beta_1 - \beta_0)/(x - \beta_0)$ for the Generalized Logistic,
>
> $\kappa = \beta_0/(x - \beta_0)$ for the Generalized Pareto, and
>
> the other parameters are estimated as before.

[3] This algorithm is much more meaningful after reviewing the Generalized Extreme Value, Generalized Logistic, and Generalized Pareto distributions in Chapter 8.

Further commentary, based on the author's experiences (Asquith and Roussel, 2009) with large numbers of sample probability-weighted moment computations with real-world data using small sample sizes, is needed. In rare circumstances, values for sample probability-weighted moments computed by $\hat{\beta}_r$, when converted to L-moments, violate (barely) the theoretical bounds or constraints discussed in Chapter 6. Because probability-weighted moments and L-moments are linear combinations of each other, having one is the same as having the other. Therefore, when the L-moment constraints are violated, the author suggests that plotting-position estimators can be used as a fall-back method of computation.[4]

Special classes of data could have prior estimates of F; in other words, probability is known *a priori*. An example are distributions of grainsize, in which the fraction of the sample passing specific diameters (sieve size) is recorded. The random variable in this situation is seemingly the fraction passing and not the diameter—the diameter being fixed by the measurement apparatus. Asquith (2003, chap. 4) suggests that a probability-weighted moment type referred to as **prior probability-weighted moments** can be used for a numerical approximation to eq. (5.3). Prior probability-weighted moments are not considered in this dissertation.

USING R

The sample probability-weighted moments are shown mathematically in eq. (5.5) with notation $\binom{a}{b}$. The $\binom{a}{b}$ notation is defined as

$$\binom{a}{b} = \frac{a!}{(a-b)!\,b!} \quad \text{for } b \leq a \tag{5.7}$$

and by convention $0! = 1$; eq. (5.7) is an expression for the number of possible combinations of a items taken b at a time.

The computation of combinations is trivial in R with the `choose()` function. The function is demonstrated in example 5–3 for the problem of solving for the number of combinations of a committee of 3 from a group of 20 people. This example is adapted from Ross (1994, example 4a). The result is that there are $1{,}140$ possible combinations.

[4] The specific algorithm is discussed in Chapter 6 in the context of example 6–4.

5–3

```
choose(20,3) # built-in to R
[1] 1140
```

Combinatorial theory and notation important for order statistics and other statistics based on order. Readers should note that the use of the `choose()` function is important because of the ratio in eq. (5.7) by direct computation by use of three `factorial()` functions is not always feasible for large a and b because of inherent numerical limitations of the computer. Finally, the terms returned by $\binom{a}{b}$ are known as **binomial coefficients**. ◀

The **probability mass function** of the Binomial distribution, not a PDF because the distribution is discrete (not continuous), is available as the `pbinom()` function and is defined as

$$P(i) = \binom{n}{i} p^i (1-p)^{n-1} \quad \text{for } i = 0, 1, \cdots, n \tag{5.8}$$

where $P(i)$ is the probability of i successes in n attempts, and p is the probability of success, and $1 - p$ is the probability of failure.

To demonstrate, suppose that a coin is flipped 5 times and let "heads" be a success ($p = 0.5$). What is the probability that exactly 3 heads will be observed for 5 flips? The solution (Ross, 1994, example 7a) is shown in example 5–4 and is 0.3125 or $10/32$.

5–4

```
dbinom(3,5,0.5) # probability density of Binomial from R
[1] 0.3125
```

◀

Returning to the sample probability-weighted moments, the unbiased $\hat{\beta}_r$ are readily computed in example 5–5 with the `pwm.ub()` function. In the example, the Normal distribution is sampled for $n = 100$, which has $\mu = 100$ and $\sigma = 50$, and the sample is placed into `fake.dat`. The unbiased $\hat{\beta}_r$ values of the sample finally are computed by `pwm.ub()` on the `fake.dat` vector.

5–5

```
fake.dat <- rnorm(100, mean=100, sd=50) # random sample
betas <- pwm.ub(fake.dat); print(betas) # compute betas and print
$betas
[1] 106.36847  67.13285  49.15510  38.93202  32.33183
$source
[1] "pwm.ub"
```

◀

For further demonstration of sample probability-weighted moments, a custom function is created for computation of an arbitrary number of $\tilde{\beta}_r$ using the plotting-position formula. The `test.pwm.pp()` function is defined in example 5–6.

5–6

```
"test.pwm.pp" <- function(x, nmom=5, A=-0.35, B=0) {
  n <- length(x); x <- sort(x)
  betas <- rep(0,nmom)
  for(r in 0:(nmom-1)) {
     beta <- 0
     for(j in 1:n) {
        beta <- beta + ((j+A)/(n+B))^r * x[j]
     }
     betas[r+1] <- beta
  }
  return(list(betas=betas/n, source="test.pwm.pp"))
}
```

The `test.pwm.pp()` function subsequently is used in example 5–7. The example simulates a Normal distribution for a sample size of $n = 10{,}000$. The first five $\tilde{\beta}_r$ are computed by the `test.pwm.pp()` function. The default plotting-positions coefficients of $A = -0.35$ and $B = 0$ are favorable for the Generalized Extreme Value distribution. An equivalent is provided in the *lmomco* package as the `pwm.gev()` function. Consistent with the name, the `pwm.gev()` function uses the optimal coefficients for the Generalized Extreme Value distribution, and the output of `pwm.gev()` is shown as well in the example. The two lists of $\tilde{\beta}_r$ are identical and are shown following the `$betas` attribute in the example.

5–7

```
fake.dat <- rnorm(10000, mean=100, sd=50)
test.pwm.pp(fake.dat)
$betas
[1] 99.60012 63.92522 47.33939 37.80697 31.59183
$source
[1] "test.pwm.pp"

pwm.gev(fake.dat)
$betas
[1] 99.60012 63.92522 47.33939 37.80697 31.59183
$source
[1] "pwm.gev"
$A
[1] -0.35
$B
[1] 0
```

The `$source` attribute in the output shown in the example from the `test.pwm.pp()` and `pwm.gev()` functions identifies the calling function. The `$A` and `$B` variables of the list returned by the `pwm.gev()` function store the `A` and `B` argument values for later reference if needed. ◀

Finally, it is informative to finish this USING R with a formal presentation of the probability-weighted moment data structures of *lmomco* with commentary. This data structure is known as the "***lmomco* probability-weighted moment list**." To demonstrate, an *lmomco* probability-weighted moment list for $\beta_0 = 450$, $\beta_1 = -214$, $\beta_2 = -139$, and $\beta_3 = -102$ in example 5–8 is created and displayed from the `PWM` variable using the `str()` function.

5–8

```
PWM <- vec2pwm(c(450, -214, -139, -102), as.list=TRUE)
str(PWM)
List of 5
 $ BETA0: num 450
 $ BETA1: num -214
 $ BETA2: num -139
 $ BETA3: num -102
 $ BETA4: num NA
 $ source: chr "vec2pwm"
```

The example shows that only the first five (only four are computable for the example, so `NA` [not applicable] is returned for β_5) are supported by the function for `as.list=TRUE` and are available as the `PWM$BETAr` values for $0 \leq r \leq 4$. An alternative probability-weighted moment list structure also is used in *lmomco* and is shown in example 5–9. In the example, the previous variable `PWM` of example 5–8 is converted to L-moments and back to probability-weighted moments. The L-moments are not shown in the example in order to maintain the focus on probability-weighted moments. The β_r are stored in a vector named `$betas`. The name of the generating function of the values in the vector is stored in the `$source` string. The `$source` variable is used in many list structures by *lmomco* to cast heredity of the numerical results.

5–9

```
lmom2pwm(pwm2lmom(PWM))
$betas
[1]  450 -214 -139 -102   NA

$source
[1] "lmom2pwm"
```

That there are two data structures in *lmomco*, which represent probability-weighted moments, is a historical artifact. (Technically, there are variations of the theme such as shown in example 5–7.) The fact that there are two primary structures containing probability-weighted moments is partly a reflection of changing design ideas and decisions by the author. The form seen in example 5–9 is preferable because of the vector form of the β_r, which can grow to arbitrary length, can readily be queried to extract specific β_r in a programming context. ◀

5.3 The Method of Probability-Weighted Moments

The **method of probability-weighted moments** is a method of parameter estimation in which the parameters of a distribution are chosen so as to equate the theoretical probability-weighted moments of the distribution to the sample probability-weighted moments. For example, the parameters Θ are chosen such that $\beta_r = \hat{\beta}_r$ (or $\beta_r = \tilde{\beta}_r$ for the quantity $r - 1$ equal to the number of parameters. The method is demonstrated in this section.

USING R ——————————— ——————————— USING R

The Gamma distribution is described in Section 7.2.3, and from that section, the relations between β_0 and β_1 and the parameters α and β are

$$\beta_0 = \alpha\beta \tag{5.9}$$

$$2\beta_1 = \frac{\beta}{\sqrt{\pi}} \exp(\log[\Gamma(\alpha + 0.5)] - \log[\Gamma(\alpha)]) - \beta_0 \tag{5.10}$$

where $\Gamma(\alpha)$ is the complete gamma function.[5] The relations between the product moments and the parameters are more straightforward and are

$$\alpha = \mu/\beta \tag{5.11}$$

$$\beta = \sigma^2/\mu \tag{5.12}$$

In example 5–10, the method of probability-weighted moments and general parameter estimation abilities of the sample probability-weighted moments are demonstrated for

[5] The complete gamma function is shown in eq. (8.85).

a GAM$(2, 3)$ (`shape=2` and `scale=3` in the R parlance) distribution using an $n = 20$ sample for $10{,}000$ simulations. For the example, the *lmomco* style of distribution specification is used through the `vec2par()` function. The `rlmomco()` function returns random values from the Gamma distribution because this distribution is identified by the `type` attribution in the list `PAR.gam`. The probability-weighted moments of the sample `X` are computed using the `pwm()` function. These values are converted to L-moments using the `pwm2lmom()` function. The conversion is needed because the `pargam()` function is setup to operate on L-moments and not probability-weighted moments. Finally, the results of the simulation are output at the end of the example and are $\hat{\alpha} = 2.15$ and $\hat{\beta} = 3.13$ from probability-weighted moments and $\hat{\alpha} = 2.33$ and $\hat{\beta} = 2.94$ from product moments.

5–10

```
n <- 20; nsim <- 10000 # sample size and number of simulations
# set the gamma distribution according to lmomco style
Alp <- 2; Beta <- 3
PAR.gam <- vec2par(c(Alp,Beta), type="gam")
# Alp is SHAPE and Beta is SCALE.
# create some vectors
Alp.PWM <- vector(mode="numeric")
Alp.PM  <- Beta.PM <- Beta.PWM <- Alp.PWM

# the simulation loop
for(i in 1:nsim) {
  X <- rlmomco(n,PAR.gam) # random samples
  sampar <- pargam(pwm2lmom(pwm(X)))
  Alp.PWM[i]  <- sampar$para[1]
  Beta.PWM[i] <- sampar$para[2]
  sampms <- pmoms(X)
  samMU  <- sampms$moments[1]
  samSD  <- sampms$moments[2]
  tmpB <- samSD^2/samMU
  Alp.PM[i]  <- samMU/tmpB
  Beta.PM[i] <- tmpB
}
results <- c(mean(Alp.PWM), mean(Beta.PWM),
             mean(Alp.PM),  mean(Beta.PM))
results <- sapply(results, round, digits=3)

cat(c("PWM␣Results:␣alpha=",results[1],"␣beta=",results[2],"\n"))
PWM Results: alpha= 2.154  beta= 3.134

cat(c("PM␣Results:␣␣alpha=",results[3],"␣beta=",results[4],"\n"))
PM Results:  alpha= 2.327  beta= 2.942
```

The results reported at the end of the example show that the probability-weighted moments provide a closer estimate to the true shape of the distribution $\alpha = 2$ and that the product moments provide a closer estimate to the true scale of the distribution $\beta = 3$. Different results would occur for different values of α, β, and sample size. In general, the probability-weighted moments will remain a competitive tool for parameter estimation, and as the magnitude of the scale and/or shape increases they will often be superior to product moments. ◀

5.4 Summary

In this chapter, the probability-weighted moments are described. A brief historical context of the moments and their heredity to L-moments is provided. The mathematics of both the theoretical and sample probability-weighted moments then followed. The sample probability-weighted moments can be computed either by unbiased estimators or by plotting-position estimators, and both techniques are described. A total of 10 examples are provided, and probability-weighted moment related functions of the *lmomco* package that were demonstrated include `pwm.gev()`, `pwm.ub()`, `theopwms()`, `pwm2lmom()`, and `lmom2pwm()`. The *lmomco* probability-weighted moment list is discussed to enhance the understanding of the probability-weighted moment implementation of the *lmomco* package. Finally, a short discussion of some sampling properties of probability-weighted moments in the context of the method of probability-weighted moments for the Gamma distribution is made. Because probability-weighted moments and L-moments are linear combinations of each other, a complementary discussion of sampling properties of probability-weighted moments is indirectly provided in Chapter 6 and specifically in Section 6.5.

Chapter 6
L-moments

In this chapter, I present a comprehensive introduction to L-moments an ancillary discussion. Understanding of the L-moments, but not the entire chapter, provides a critical prerequisite needed for this dissertation. I have purposefully placed both censored and multivariate L-moments in a later chapter. This chapter presents the defining mathematics and sample counterparts of L-moments along with a step-by-step presentation of distribution fit by L-moments. Secondarily important components of this chapter are the visualization of L-moment weight factors, a reference frame perspective between L-moments and product moments, and TL-moments (defining mathematics and sample counterparts). The discussion of the sampling properties of L-moments is to be juxtaposed with similar discussion of the product moments in an earlier chapter. The sampling properties provide an important justification for distributional analysis with L-moment statistics using R.

6.1 Introduction

As with the probability-weighted moments, L-moments (Hosking, 1990) are an "attractive alternative system of moment-like quantities" (Jones, 2004, p. 98) and thus are an alternative to product moments. Like other statistical moments, L-moments characterize the geometry of distributions and summarize samples. L-moments are directly analogous to—that is, have similar interpretations as—the product moments. This makes L-moments conceptually accessible to many potential users.

L-moments are based on linear combinations of differences of the expectations of order statistics (see Section 3.1) as opposed to the product moments, which are based on powers (exponents) of differences (see eq. (4.10)). For example, the product moment definition

of skew (based on differences to a third power, see eq. (4.14)), results in extremely poor sampling performance for distributions characterized by heavy tails, asymmetry, and outliers. The performance of kurtosis, which is based on differences to the fourth power (see eq. (4.15)), is even worse. In part because of favorable sampling performance, Hosking (1992) concludes that "L-moments can provide good summary measures of distributional shape and may be preferable to [product] moments for this purpose."

Data that frequently contain outliers and heavy tails are endemic in the earth-system sciences. The distribution of flood magnitude is one such example and earthquake damages are another. The history of L-moments could be considered as beginning with the statistical needs of researchers of surface-water hydrology (Landwehr and others, 1979a,b, 1980) with interests in floods, extreme rainfall hydrology, and ancillary topics in the mid 1970s through the later parts of the 20th century. However, Hosking (1990) traces statistical connections to L-moments back to the 19th century. Historically, L-moments were developed from probability-weighted moments (see Chapter 5) but were "adumbrated[1] earlier" (Hosking, 1999, p. 1) such as by Kaigh and Driscoll (1987) or Sillitto (1951, 1969). The core theory of L-moments for univariate applications was unified by about the late 1980s to early 1990s. Hosking (1990) provides a canonical reference along with the general historical context and placement of L-moments in the broader statistical literature.

Since that time, the L-moment and probability-weighted moment literature continues to develop and expand (Delicado and Goria, 2008; Elamir and Seheult, 2003, 2004; Haktanir, 1997; Hosking, 1995, 2000, 2006, 2007a,b,c; Jones, 2004; Karvanen, 2006; Kliche and others, 2008; Kroll and Stedinger, 1996; Liou and others, 2008; Royston, 1992; Serfling and Xiao, 2007; Ulrych and others, 2000; Unnikrishnan and Vineshkumar, 2010; Wang and others, 2010; Whalen and others, 2002). Interest in L-moments is not limited to the statistical profession and those interested in distributions of earth-system phenomena, but interest exists within financial (Hosking, 1999; Hosking and others, 2000; Jurczenko and others, 2008) and reliability disciplines (Unnikrishnan and Vineshkumar, 2010) as well.

A summary and a then contemporary statement (early 1990s) concerning the excitement that L-moments caused is informative. Vogel (1995) states that

> The challenges posed by extreme hydrological events continue to vex hydrologists. The introduction of the theory of L-moments (Hosking, 1990) is probably the single most significant recent advance relating to our understanding of extreme events. Generally, L-moments are linear combinations of ordered

[1] Adumbrate—indicate faintly, foreshadow.

> observations, which are unbiased regardless of the parent population, hence L-moments allow us to discriminate the behavior of skewed hydrologic data which was difficult or impossible only a few years ago.

Expanding on these statements, Ulrych and others (2000) conclude from their numerical experiments that "L-moments are superior estimates to those obtained using C-moments [product moments] and the principle of maximum entropy." Finally, Ulrych and others (2000, p. 52) comment forcefully on Hosking (1990) by stating that this Hosking paper is "a beautiful paper indeed [and] has had an explosive effect in some fields."

In another broadly sweeping paper on the general topic of extreme-value analysis in hydrology, Katz and others (2002, pp. 1287–1288) acknowledge the contributions of probability-weighted moments and L-moments. The authors state

> Probability-weighted moments (or L-moments) are more popular than [maximum likelihood] in applications to hydrologic extremes, both because of their computational simplicity and because of their good performance for small samples. . . . [L-moments can serve] as a good choice of starting values for the iterative numerical procedure required to obtain [maximum likelihood] estimates (Hosking, 1985).

L-moments are useful because they are easily used to fit common (Normal, Gamma) and not so common probability distributions (such as the Generalized Logistic or Wakeby) to data sets. L-moments have powerful features for discriminating between distribution types. L-moments are approximately unbiased (unlike the higher product moments), robust, and consistent. Further, L-moments provide more secure inference of distribution shape than the product moments. Because of inherently attractive sampling characteristics, L-moments are regarded as highly reliable summary statistics, and through the use of R, the L-moments are straightforward to incorporate into practical problems. Therefore, the primal objective of this dissertation is to advocate for the use, or at least consideration, of L-moments because

L-moments can be "drop-in" replacements to the product moments.

The *lmomco* package provides substantial L-moment support. The primary L-moment functions from the package are listed in table 6.1, and distribution-specific L-moment functions are listed in table 6.2. These functions support both theoretical and sample computations. The distinctions between the two computation types are discussed in the next section. The listed functions are thoroughly demonstrated following the Using R identifiers in this chapter and elsewhere in this dissertation. A similar summary of functions

from the *Lmoments* package is listed in table 6.3, and those functions of the *lmom* package are listed in table 6.4. Many of the functions listed in the four tables are used in examples in this chapter. The functions in these tables and others are answers to the call for L-moment support by developers of statistical packages made by Royston (1992) that is summarized on page 10 of this dissertation.

Table 6.1. Summary of L-moment computation and support functions of the *lmomco* package by Asquith (2011)

Function	Purpose
`are.lmom.valid()`	Check theoretical bounds of L-moments
`lmorph()`	Morphs between two styles of L-moment lists
`theoLmoms()`	Compute theoretical L-moments of a distribution
`theoTLmoms()`	Compute theoretical TL-moments of a distribution
`theoLmoms.max.ostat()`	Compute theoretical L-moments by maximum order statistics
`lmoms()`	Compute an unbiased sample L-moments by dispatch to `TLmoms()`
`lmoms.ub()`	Compute unbiased sample L-moments by `lmoms()`
`lmomRCmark()`	Compute a right-censored sample L-moment by indicator variable
`lmomsRCmark()`	Compute right-censored sample L-moments by indicator variable
`TLmom()`	Compute an unbiased sample TL-moment
`TLmoms()`	Compute unbiased sample TL-moments by dispatch to `TLmom()`
`pwm2lmom()`	Convert probability-weighted moments to L-moments
`lmom2pwm()`	Convert L-moments to probability-weighted moments
`vec2lmom()`	Convert a vector to L-moments
`vec2TLmom()`	Convert a vector to TL-moments
`lmom2vec()`	Convert L-moments to a vector

Table 6.2. Summary of L-moment computation functions for probability distributions of the *lmomco* package by Asquith (2011)

Distribution	L-moments
Cauchy	`lmomcau()`
Exponential	`lmomexp()`
Gamma	`lmomgam()`
Generalized Extreme Value	`lmomgev()`
Generalized Lambda	`lmomgld()`
Generalized Logistic	`lmomglo()`
Generalized Normal	`lmomgno()`
Generalized Pareto	`lmomgpa()`
Gumbel	`lmomgum()`
Kappa	`lmomkap()`
Kumaraswamy	`lmomkur()`
log-Normal3	`lmomln3()`
Normal	`lmomnor()`
Pearson Type III	`lmompe3()`
Rayleigh	`lmomray()`
Reverse Gumbel	`lmomrevgum()`
Rice	`lmomrice()`
Wakeby	`lmomwak()`
Weibull	`lmomwei()`
Right-Censored Generalized Pareto	`lmomgpaRC()`
Trimmed Generalized Lambda	`lmomTLgld()`
Trimmed Generalized Pareto	`lmomTLgpa()`

Table 6.3. Summary of L-moment computation functions of the *Lmoments* package by Karvanen (2009)

Function	Purpose
`Lmoments()`	Compute unbiased sample L-moments
`t1moments()`	Compute unbiased sample TL-moments with trim $t = 1$

6.2 L-moments—Definitions and Math

The L-moments are formally defined in this section and separately introduced as theoretical and sample versions.

6.2.1 Theoretical L-moments

The **theoretical L-moments** for a real-valued random variable X with a QDF of $x(F)$ are defined from the expectations of order statistics. The order statistics of X for a sample of size n are formed by the ascending order $X_{1:n} \leq X_{2:n} \leq \cdots \leq X_{n:n}$. The theoretical L-moments for $r \geq 1$ are defined by

$$\lambda_r = \frac{1}{r}\sum_{k=0}^{r-1}(-1)^k \binom{r-1}{k} \mathrm{E}[X_{r-k:r}] \tag{6.1}$$

where r is the integer order of the L-moment, and $\mathrm{E}[X_{r-k:r}]$ is the expectation of the $r-k$ order statistic of a sample of size r, and this equation commonly is expressed in terms of the QDF as described presently. L-moments also are commonly formulated (Hosking, 1990) from **rth-shifted Legendre polynomials** $P^*_r(F)$, which are defined as

$$P^*_r(F) = \sum_{k=0}^{r}(-1)^{r-k}\binom{r}{k}\binom{r+k}{k}F^k \tag{6.2}$$

from which the L-moments are

$$\lambda_r = \int_0^1 x(F)P^*_{r-1}(F)\,\mathrm{d}F \tag{6.3}$$

The first four theoretical L-moments in terms of the order statistic expectations follow from eq. (6.1) and are

$$\lambda_1 = \mathrm{E}[X_{1:1}] \tag{6.4}$$

$$\lambda_2 = \tfrac{1}{2}(\mathrm{E}[X_{2:2}] - \mathrm{E}[X_{1:2}]) \tag{6.5}$$

$$\lambda_3 = \tfrac{1}{3}(\mathrm{E}[X_{3:3}] - 2\mathrm{E}[X_{2:3}] + \mathrm{E}[X_{1:3}]) \tag{6.6}$$

$$\lambda_4 = \tfrac{1}{4}(\mathrm{E}[X_{4:4}] - 3\mathrm{E}[X_{3:4}] + 3\mathrm{E}[X_{2:4}] - \mathrm{E}[X_{1:4}]) \tag{6.7}$$

and arguments justifying their interpretations as respective measures of location, scale or variability or dispersion, skew, and kurtosis are provided in Section 3.1. It is noteworthy to compare the similarities of eq. (6.5) and eq. (3.31) (Gini mean difference, $\mathcal{G}$) to see the source of relation between λ_2 and $\mathcal{G}$ considered in Chapter 3. The system of equations in eq. (6.7) are virtually identical to those shown by Kaigh and Driscoll (1987, eq. 2.4, p. 26).

An expression, based on proof by Hosking (1986, 1996a) and alternative proof by Jones (2004), for λ_r for $r \geq 2$ in terms of the CDF is

$$\lambda_r = \frac{1}{r}\sum_{j=0}^{r-2}(-1)^j \binom{r-2}{j}\binom{r}{j+1}\int_{-\infty}^{\infty}[F(x)]^{r-j-1} \times [1-F(x)]^{j+1}\,\mathrm{d}x \tag{6.8}$$

or

$$\lambda_r = \int_{-\infty}^{\infty} F(x) \times [1-F(x)] \times L_r(F(x))\,\mathrm{d}x \tag{6.9}$$

where

$$L_r(u) = \frac{1}{1-r}\sum_{j=0}^{r-2}(-1)^j \binom{r-1}{j}\binom{r-1}{j+1}u^{r-2-j}(1-u)^j \tag{6.10}$$

The first four theoretical L-moments in terms of the QDF using eqs. (3.4) and (6.1) are

$$\lambda_1 = \int_0^1 x(F)\,\mathrm{d}F \tag{6.11}$$

$$\lambda_2 = \int_0^1 x(F) \times (2F-1)\,\mathrm{d}F \tag{6.12}$$

$$\lambda_3 = \int_0^1 x(F) \times (6F^2-6F+1)\,\mathrm{d}F \tag{6.13}$$

$$\lambda_4 = \int_0^1 x(F) \times (20F^3-30F^2+12F-1)\,\mathrm{d}F \tag{6.14}$$

The theoretical L-moments can be written in terms of the derivatives of the QDF (notationally $x^{(r)}(F)$; $x^{(0)}(F)$ is the usual QDF, $x^{(1)}(F)$ is the first derivative, ...). This "particularly striking result" (Hosking, 2007b, p. 3027) is

$$\lambda_{r+1} = \frac{1}{r!}\int_0^1 F^r(1-F)^r \times x^{(r)}(F)\,\mathrm{d}F \tag{6.15}$$

This equation (derived from eq. (6.66) for $k = r$) is particularly useful in interpretation of λ_2 (**L-scale**), which is a measure of distribution variability or spread. The spread of

Table 6.4. Summary of L-moment computation functions for samples and by probability distribution of the *lmom* package by Hosking (2009a)

Function	Purpose
`samlmu()`	Compute unbiased sample L-moments
`lmrexp()`	Compute L-moments of Exponential distribution
`lmrgam()`	Compute L-moments of Gamma distribution
`lmrgev()`	Compute L-moments of Generalized Extreme-Value distribution
`lmrglo()`	Compute L-moments of Generalized Logistic distribution
`lmrgpa()`	Compute L-moments of Generalized Pareto distribution
`lmrgno()`	Compute L-moments of Generalized Normal (lognormal) distribution
`lmrgum()`	Compute L-moments of Gumbel (Extreme-Value Type I) distribution
`lmrkap()`	Compute L-moments of Kappa distribution
`lmrln3()`	Compute L-moments of Log-Normal (3 parameter) distribution
`lmrnor()`	Compute L-moments of Normal distribution
`lmrpe3()`	Compute L-moments of Pearson Type III distribution
`lmrwak()`	Compute L-moments of Wakeby distribution
`lmrwei()`	Compute L-moments of the Weibull distribution

the distribution is proportional to the rate of change (the first derivative) of the QDF. The greater the rate of change, the larger distance between successively ordered samples.

All theoretical L-moments can be expressed by the first derivative of a QDF. Hosking (2007a, p. 2877) shows these to be

$$\lambda_1 - L = \int_0^1 (1-F) \times x^{(1)}(F)\, \mathrm{d}F \quad \text{when the lower bound } L \text{ is finite} \tag{6.16}$$

$$\lambda_2 = \int_0^1 F(1-F) \times x^{(1)}(F)\, \mathrm{d}F \tag{6.17}$$

$$\lambda_3 = \int_0^1 F(1-F)(2F-1) \times x^{(1)}(F)\, \mathrm{d}F \tag{6.18}$$

$$\lambda_4 = \int_0^1 F(1-F)(5F^2-5F+1) \times x^{(1)}(F)\, \mathrm{d}F \tag{6.19}$$

and in general

$$\lambda_r = \int_0^1 Z_r(F) \times x^{(1)}(F)\, \mathrm{d}F \quad \text{for } r \geq 2 \tag{6.20}$$

where the polynomial $Z_r(F)$ of degree r in terms of eq. (6.2) is

$$Z_r(F) = \int_F^1 P^*_{r-1}(v)\ \mathrm{d}v \tag{6.21}$$

Useful distributions have non-zero variability, and therefore, a restriction on λ_2 is that

$$\lambda_2 > 0 \tag{6.22}$$

and continuing with the bounds of the L-moments, the **theoretical L-moment ratios** are the dimensionless quantities

$$\tau_2 = \lambda_2/\lambda_1 = \textbf{coefficient of L-variation} \tag{6.23}$$

$$\tau_3 = \lambda_3/\lambda_2 = \textbf{L-skew} \tag{6.24}$$

$$\tau_4 = \lambda_4/\lambda_2 = \textbf{L-kurtosis} \tag{6.25}$$

and for $r \geq 5$, which are unnamed, are

$$\tau_r = \lambda_r/\lambda_2 \tag{6.26}$$

The quantity τ_2 is meaningful for positive random variables ($X \geq 0$) and is $0 < \tau_2 < 1$.

Other authors (most notably J.R.M. Hosking) lack the subscripted 2 on τ_2, but the subscript explicitly is used here and preferred by the author to draw a connection to the second element of a vector of L-moment ratios.[2] As seen in many examples herein, the `lmoms()` and `vec2lmom()` functions (along with many more) return an L-moment ratio vector in the `$ratios` attribute. By definition for symmetrical distributions, it can be shown that

$$\tau_r = 0 \quad \text{for odd } r \tag{6.27}$$

Several L-moments, unlike the theoretically unbounded[3] product moments G (skew) and K (kurtosis) for $n \to \infty$, are bounded (Hosking, 1990, Theorem 2). Two useful examples of boundedness for L-moment ratios are

$$-1 < \tau_r < 1 \text{ for } r \geq 3 \tag{6.28}$$

[2] Another reason advocated by the author for τ_2 is that the symbol τ remains available to refer to the more venerable Kendall's Tau statistic (Hollander and Wolfe, 1973, chap. 8) in investigative settings involving L-moments and correlation (independence) tests by Kendall's Tau.

[3] The irony is noted that these product moments have no theoretical upper limit of magnitude, yet suffer from algebraic bounds based on sample size as discussed in Chapter 4.

$$\tfrac{1}{4}(5\tau_3^2 - 1) \leq \tau_4 < 1 \tag{6.29}$$

which in the later inequality τ_4 must also satisfy

$$\tau_4 \geq -1/4 \tag{6.30}$$

These bounds are useful and are philosophically attractive because the magnitudes of τ_3 and τ_4 are much more constrained than are G and K, and more importantly, these bounds are not a function of sample size, unlike the algebraic sample-size bounds for G and K. Hence, relative comparisons of the quantification of the concepts of skewness and kurtosis for samples and distributions are more informative using L-moments. Additional intra-moment constraints of L-moment ratios exist, Jones (2004) shows that $\tau_6 \geq -1/6$ and a lower bound for τ_6 of

$$\tfrac{1}{25}(42{\tau_4}^2 - 14\tau_4 - 3) < \tau_6 \tag{6.31}$$

which in lower bound, Hosking (1996a) provides further refinement.

The system of linear equations relating L-moments λ_r to probability-weighted moments β_r of Chapter 5 can be obtained by

$$\lambda_{r+1} = \sum_{k=0}^{r} (-1)^{r-k} \binom{r}{k} \binom{r+k}{k} \beta_k \quad \text{for } r \geq 0 \tag{6.32}$$

from which the first five[4] L-moments in terms of probability-weighted moments are

$$\lambda_1 = \quad \beta_0 \tag{6.33}$$

$$\lambda_2 = \quad 2\beta_1 - \quad \beta_0 \tag{6.34}$$

$$\lambda_3 = \quad 6\beta_2 - \quad 6\beta_1 + \quad \beta_0 \tag{6.35}$$

$$\lambda_4 = 20\beta_3 - \quad 30\beta_2 + 12\beta_1 - \quad \beta_0 \tag{6.36}$$

$$\lambda_5 = 70\beta_4 - 140\beta_3 + 90\beta_2 - 20\beta_1 + \beta_0 \tag{6.37}$$

If $x(F)$ is a valid QDF, then the L-moments can be computed directly by numerical integration to either bypass or otherwise verify the algorithms of many functions in Chapters 7–9 that convert distributions set by known parameters into L-moments. The general

[4] Five are shown in the system of equations here instead of the four in parallel constructs in this dissertation because of the τ_5 expression for the Kumaraswamy distribution.

equation derived from eqs. (3.4) and (6.1) for computing L-moments given a QDF is

$$\lambda_r = \frac{1}{r}\sum_{k=0}^{r-1}(-1)^k\binom{r-1}{k}\frac{r!}{(r-k-1)!\,k!} \times \int_0^1 x(F)\times F^{r-k-1}\times(1-F)^k\,\mathrm{d}F \tag{6.38}$$

Hosking (2006) comments that each λ_r can be written as the expectations of extreme order statistics such as by

$$\lambda_r = \sum_{k=1}^{r}(-1)^{r-k}k^{-1}\binom{r-1}{k-1}\binom{r+k-2}{k-1}\mathrm{E}[X_{k:k}] \tag{6.39}$$

in terms of maxima order statistics. The set of λ_r in terms of extreme (minima and maxima) order statistics therefore also characterize a distribution. However, the extreme order statistics do so with redundancy (see Chapter 3, page 65). Hosking (2006, p. 193) shows that a "wide range of distributions can be characterized by their [λ_r] with no redundancy." In other words, the "characterization by [λ_r] is nonredundant, in that if even one [λ_r] is dropped from the set the remaining [λ_r] no longer [uniquely] suffice to determine the distribution" (Hosking, 2006, p. 194). As a result, Hosking (2006, p. 198) suggests that the distribution information contained in λ_r is maximally independent of information contained by the remaining λ_{r-1} in the set. By Hosking's logic and remark, L-moments are "particularly suitable as summary statistics of a distribution."

Expansion of eq. (6.39) results in the following system of equations for the first four λ_r in terms of the largest order statistics

$$\lambda_1 = \mathrm{E}[X_{1:1}] \tag{6.40}$$
$$\lambda_2 = \mathrm{E}[X_{2:2}] - \mathrm{E}[X_{1:1}] \tag{6.41}$$
$$\lambda_3 = 2\mathrm{E}[X_{3:3}] - 3\mathrm{E}[X_{2:2}] + \mathrm{E}[X_{1:1}] \tag{6.42}$$
$$\lambda_4 = 5\mathrm{E}[X_{4:4}] - 10\mathrm{E}[X_{3:3}] + 6\mathrm{E}[X_{2:2}] - \mathrm{E}[X_{1:1}] \tag{6.43}$$

This system of equations is demonstrated later in this chapter (see example 6–10).

USING R ——— ——— USING R

The `theoLmoms()` provides numerical integration by eq. (6.38) for an arbitrary moment order, and the function is used in example 6–1. In the example, a standard Normal distribution is parameterized in the *lmomco* fashion by the `vec2par()` function and set into `NO1`. The `NO1` parameter list, which is a type of *lmomco* parameter list (see page 163 and ex. 7–1), is passed to the `theoLmoms()` with a request to compute the first `nmom` L-moments. The eight L-moments, which are computed by numerical integration, are shown.

6–1

```
N01 <- vec2par(c(0,1), type="nor") # standard normal distribution
theoLmoms(N01, nmom=4) # compute the first nmom L-moments
$lambdas
[1] -4.360355e-17  5.641895e-01 -7.401487e-17  6.917017e-02
$ratios
[1]            NA -1.293907e+16 -1.311880e-16  1.226009e-01
$trim
[1] 0
$source
[1] "theoLmoms"
```

As the output shows, the λ_r for odd r are effectively zero because the Normal distribution is symmetric. The example demonstrates that odd-order L-moment ratios are consistent with the observation that the odd-order ratios measure distribution asymmetry. Specifically, each odd-order ratio provides for a progressively higher measure of distribution asymmetry. The `theoLmoms()` can be used to compute L-moments and L-moment ratios for QDFs for which analytical or numerical solutions have not been developed. The `theoLmoms()` function is useful to verify the computations of other algorithms in several examples in this dissertation. ◀

L-moments and probability-weighted moments are linear combinations of each other. Conversion between the two moment types is readily made using the `lmom2pwm()` function as example 6–2 demonstrates for $\lambda_1 = 100$, $\tau_2 = 0.45$, $\tau_3 = -0.3$, and $\tau_4 = 0.4$.

6–2

```
lmr <- vec2lmom(c(100,0.45,-0.3,0.4), lscale=FALSE)
lmom2pwm(lmr)
$betas
[1] 100.00000  72.50000  53.58333  42.77500
$source
[1] "lmom2pwm"
pwm2lmom(lmom2pwm(lmr))
```

```
$lambdas
[1] 100.0  45.0 -13.5  18.0
$ratios
[1]    NA  0.45 -0.30  0.40

$source
[1] "pwm2lmom"
```

In the example, the numerical equivalency of the L-moments in variable `lmr` to those in `$lambda` and `$ratios` of the terminating output is evident. Readers are asked to note in example 6–2 that the use of the `vec2lmom()` function differs from previous demonstrations because the coefficient of L-variation τ_2 is used instead of λ_2. The argument `lscale=FALSE`, thus, is needed in the example. ◀

The validity of L-moments are readily verified by the `are.lmom.valid()` function. The bounds of the L-moments supported by the function are shown in eqs. (6.22), (6.28), and (6.29). Example 6–3 demonstrates use of the function.

6–3

```
lmr <- list(lambdas=c(100,-20), ratios=c(NA,NA))
are.lmom.valid(lmr) # fails on L2 > 0
[1] FALSE

# The following fails on abs(T3) <= 1
are.lmom.valid(list(lambdas=c(100,   20, -80),
                    ratios=c( NA, 0.20,  -4)))
[1] FALSE

are.lmom.valid(list(L1=1, L2=2, TAU3=0.4, TAU4=-0.04)) # works
[1] TRUE
```

The third and terminal use of the `are.lmom.valid()` function in example 6–3 has a different list style passed into it compared to the other two. The `lmorph()` function is used for internal conversion. Thus, the different implementation styles of L-moments within the *lmomco* package also are shown. The styles are discussed in more detail in Section 6.2.2. ◀

Finally, the author suggests that the following algorithm be considered in circumstances in which the sample L-moments by unbiased estimators are invalid. Such a circumstance might occur in large data mining operations in which the sample L-moments (next section) of hundreds or thousands of observed data sets are computed. It is possible that in a few samples, typically very small, that invalid L-moments would be computed. The unbiased sample L-moments are computed with `lmoms()` and then tested by `are.lmom.valid()`.

If the unbiased sample L-moments are not, then L-moments are computed through the sample probability-weighted moments that are based on plotting positions of the Generalized Extreme Value distribution.

6–4

```
fake.dat <- rnorm(10) # generate some fake data
lmr <- lmoms(fake.dat) # compute L-moments
if(! are.lmom.valid(lmr)) {
   lmr <- pwm2lmom(pwm.gev(fake.dat))
}
```

◀

Eqs. (6.8) and (6.9) both provide expressions for λ_r in terms of the CDF. At first review, both equations appear not too difficult to implement in R; however, eq. (6.9) is less compatible with vectorization of R as provided by the `integrate()` function.[5] Example 6–5 implements eq. (6.8) instead of eq. (6.9) because of the much greater algorithmic burden of placing the series of $L_r(u)$ of eq. (6.10) inside the integral. The code example 6–5 provides an excellent example of the congruent use of function within function (and within function) development, numerical integration, and series solution. The algorithmic flexibility of R is shown.

6–5

```
"lambda.by.cdf" <-
function(r, para, cdf=NULL, lower=-Inf, upper=Inf) {
  sfunc <- function(j) {
        tmpA <- (-1)^j * choose(r-2,j) * choose(r, j+1)
        RspaceIntegral <- function(x, j) {
              Fx <- cdf(x, para)
              return( Fx^(r-j-1) * (1-Fx)^(j+1) )
        }
        tmpB <- integrate(RspaceIntegral, lower, upper, j=j)
        tmpB <- tmpB$value
        return(tmpA*tmpB)
  }
  tmp <- sum(sapply(0:(r-2), sfunc))/r
  return(tmp)
}
```

The function `lambda.by.cdf()` is demonstrated in example 6–6 for the standard Normal distribution by comparison of select λ_r from the `theoLmoms()` function, which

[5] The author initially tried to implement eq. (6.9) as this equation seemed somehow easier than eq. (6.8)—the author failed after considerable and frustrating efforts. However, success was found for eq. (6.8) and is shown in this dissertation.

uses the QDF of the Normal distribution, to those from `lambda.by.cdf()`, which uses the CDF of the distribution.

6–6

```
NORpar <- vec2par(c(0,1), type="nor")
lmr.by.QF <- theoLmoms(NORpar, nmom=8)
print(lmr.by.QF$lambdas[c(2,4,8)])
[1] 0.56418953 0.06917017 0.01232133

L2 <- lambda.by.cdf(2, NORpar, cdf=cdfnor)
L4 <- lambda.by.cdf(4, NORpar, cdf=cdfnor)
L8 <- lambda.by.cdf(8, NORpar, cdf=cdfnor)
print(c(L2,L4,L8))
[1] 0.56418958 0.06917061 0.01232370
```

Example 6–6 shows that λ_2, λ_4, and λ_8 are all equivalent. (Odd order λ_r are not shown as these are zero for the Normal distribution.) The results demonstrate the reliability of the `lambda.by.cdf()` function. ◀

6.2.2 Sample L-moments

The **sample L-moments** are computed for a sample from the sample order statistics $x_{1:n} \leq x_{2:n} \leq \cdots \leq x_{n:n}$. The sample order statistics thus are estimated by simply sorting the data in ascending order. The sample L-moments are

$$\hat{\lambda}_r = \frac{1}{r}\binom{n}{r}^{-1} \sum_{i=1}^{n} \left[\sum_{j=0}^{r-1} (-1)^j \binom{r-1}{j} \binom{i-1}{r-1-j} \binom{n-i}{j} \right] x_{i:n} \tag{6.44}$$

The **sample L-moment ratios** are

$$\hat{\tau}_2 = \hat{\lambda}_2/\hat{\lambda}_1 = \textbf{sample coefficient of L-variation} \tag{6.45}$$

$$\hat{\tau}_3 = \hat{\lambda}_3/\hat{\lambda}_2 = \textbf{sample L-skew} \tag{6.46}$$

$$\hat{\tau}_4 = \hat{\lambda}_4/\hat{\lambda}_2 = \textbf{sample L-kurtosis} \tag{6.47}$$

and for $r \geq 5$, which are unnamed, are

$$\hat{\tau}_r = \hat{\lambda}_r/\hat{\lambda}_2 \tag{6.48}$$

The author recognizes that the sample L-moments (and sample probability-weighted moments) in *lmomco* are not computationally efficient. For efficiency, Wang (1996b) describes direct sample estimators of $\hat{\tau}_r$ for $r \leq 4$ and provides FORTRAN source code based on expansion of eq. (6.44). The FORTRAN algorithm is fast. However, the *lmomco* package uses the `choose()` function of R for the binomial coefficients $\binom{a}{b}$ to support an arbitrary order r of L-moments. Binomial coefficient computation is seen example 5–4 on page 106.

USING R ——————— ——————— USING R

The sample L-moments are readily computed as shown in example 6–7. The output of the `lmoms()` (*lmomco*), `samlmu()` (*lmom*), and `Lmoments()` (*Lmoments*) functions show that the respective package authors (Asquith, 2011; Hosking, 2009a; Karvanen, 2009) have differing implementation ideas for an "L-moment" object. For the *lmomco* package, this data structure is known as the "***lmomco* L-moment list**." In general, these L-moment objects interact in package-specific ways with other functions available in the two packages and are thus evidently intended somewhat for intra-package purposes.

6–7

```
data <- rnorm(30)       # 30 standard normal samples
lmr1 <- lmoms(data)     # from package lmomco
lmr2 <- samlmu(data)    # from package lmom
lmr3 <- Lmoments(data) # from package Lmoments
print(lmr1) # L-moments (and ratios) from the lmomco package
$lambdas
[1] -0.14923254  0.50167639  0.01407900  0.09680829 -0.03395663
$ratios
[1]          NA -3.36170910  0.02806390  0.19296960 -0.06768633
$trim
[1] 0
$leftrim
NULL
$rightrim
NULL
print(lmr2) # L-moments (and ratios) from the lmom package
       l_1         l_2         t_3         t_4
-0.1492325  0.5016764  0.0280639  0.1929696

print(lmr3) # L-moments from the Lmoments package
             L1        L2         L3        L4
[1,] -0.1492325 0.5016764 0.01407900 0.0968083
```

◀

It is informative to present the L-moment list with discussion. An *lmomco* L-moment list is created in example 6–8 and displayed by the `str()` function.

6–8

```
LMR <- vec2lmom(c(-450, 23, -0.1, 0.3))
str(LMR)
List of 9
 $ L1  : num -450
 $ L2  : num 23
 $ TAU3: num -0.1
 $ TAU4: num 0.3
 $ TAU5: NULL
 $ LCV : num -0.0511
 $ L3  : num -2.3
 $ L4  : num 6.9
 $ L5  : NULL
```

As shown in the output of example 6–8, the L-moments `L`x and L-moment ratios `LCV` and `TAU`x self-document or label the values ($\lambda_2 = 23$ or $\tau_3 = -0.1$). This nomenclature style for an *lmomco* L-moment list, however, is restrictive. The nomenclature would rapidly become burdensome as the number of L-moments increases. An alternative data structure is produced in example 6–9.

6–9

```
lmorph(LMR) # make the list conversion
$lambdas
[1] -450.0   23.0   -2.3    6.9
$ratios
[1]          NA -0.05111111 -0.10000000  0.30000000
$trim
[1] 0
$leftrim
NULL
$rightrim
NULL
$source
[1] "lmorph"
```

It is seen in the morphed `LMR` list that the values have been vectorized in `$lamdas` and `$ratios`—the greater programming flexibility of using vectors hopefully is self evident. The `lmorph()` function thus converts (and *visa versa*) the L-moment objects into differing data structures. The structure shown is useful because other L-moment types, such as the TL-moments can be supported. These L-moment types require additional documentation concerning the trimming of the sample. Finally, the `$source` attribute, as seen in other special *lmomco* lists, identifies the name of the called function.

There are two L-moment data structures in *lmomco*, and this is a historical artifact. The fact that there are two primary structures is partly a reflection of changing design ideas and decisions by the author. The form seen in example 6–9 is preferable because of the vector forms of λ_r and τ_r, which can grow to arbitrary length, can readily be queried to extract specific λ_r or τ_r in a programming context. ◀

Consideration of eq. (6.39) results in an another method to compute sample L-moments. The equation shows that each λ_r can be written and computed as a linear combination of maxima order statistics. Concerning expressions of L-moments in terms of maxima order statistics, it is informative to remark that the numerical representation of values less than the mean ($\mathrm{E}[X_{1:1}]$) are still required. However, these representations are tacitly not used in the computation of the L-moments, which is demonstrated in example 6–10.

In the example, the `maxOstat.system()` function is created to compute the coefficients on the linear system of equations by eq. (6.39) and used for $r \leq 4$ to set the respective `coes` variables. The example sets the values $\lambda_1 = 1200$, $\lambda_2 = 500$, and $\tau_3 = 0.3$ in the `lmr` variable. The parameters of the Generalized Extreme Value distribution as computed from these L-moments and set into `GEVpar`.

6–10

```
"maxOstat.system" <-
function(r=1) {
   sapply(1:r, function(k,r) { (-1)^(r-k)/k * choose(r-1,k-1) *
                               choose(r+k-2,k-1) }, r=r)
}
coes1 <- maxOstat.system(1); coes2 <- maxOstat.system(2)
coes3 <- maxOstat.system(3); coes4 <- maxOstat.system(4)
lmr <- vec2lmom(c(1200, 500, 0.3)) # set first three L-moments
GEVpar <- pargev(lmr) # perform parameter estimation for GEV
# Perform large samplings of samples the four
# sample sizes, extract the maximum each time and finally compute
# the mean of each.
x <- rlmomco(2000, GEVpar) # simulate 2000 values for resampling
samlmr <- lmoms(x) # compute sample estimates in typical fashion
E11 <- mean(replicate(100000, max(sample(x, 1, replace=TRUE))))
E22 <- mean(replicate(100000, max(sample(x, 2, replace=TRUE))))
E33 <- mean(replicate(100000, max(sample(x, 3, replace=TRUE))))
E44 <- mean(replicate(100000, max(sample(x, 4, replace=TRUE))))
lam1 <- E11*coes1
lam2 <- E22*coes2[2] + E11*coes2[1]
lam3 <- E33*coes3[3] + E22*coes3[2] + E11*coes3[1]
lam4 <- E44*coes4[4] + E33*coes4[3] + E22*coes4[2]+ E11*coes4[1]
t3 <- lam3/lam2; t4 <- lam4/lam2
cat(c("# By maxima:",
```

```
      round(c(lam1,lam2,t3,t4), digits=3),"\n"))
# By maxima: 1200.245 493.06 0.301 0.281

cat(c("# By lmoms():",
       round(c(samlmr$lambdas[1:2],
               samlmr$ratios[3:4]), digits=3),"\n"))
# By lmoms(): 1198 497.731 0.294 0.206
```

Subsequent operations in example 6–10 demonstrate the viability of sample L-moment computation via eq. (6.39) and substitution of $\mathrm{E}[X_{k:k}]$ with a sample counter part. A random sample of $n = 2{,}000$ values is created in variable `x` by the `rlmomco()` function. The following four operations, which set the `E`xx variables, compute the expected values of the first four maxima order statistics based on bootstrapping by the `sample()` function. The sample L-moments are computed from these expectations and the coefficients and set into the respective `lam`x variables. Finally, the two `cat()` functions output the results. The results are similar and demonstrate the validity of eq. (6.39). ◀

6.2.3 Visualization of L-moment Weight Factors

The relative contribution of individual data values on the computation of $\hat{\lambda}_r$ can be depicted by visualization of L-moment weight factors. To begin, the sample L-moments $\hat{\lambda}_r$ are defined as **weighted-linear combinations** of the sample values. In particular, $\hat{\lambda}_r$ can be shown to be linear combinations of the ordered sample ($x_{1:n} \leq x_{2:n} \leq \cdots \leq x_{n:n}$) and a weight factor $w_{j:n}^{(r)}$. The equation is

$$\hat{\lambda}_r = \frac{1}{n} \sum_{j=1}^{n} w_{j:n}^{(r)} \, x_{j:n} \tag{6.49}$$

where the weights are computed by

$$w_{j:n}^{(k)} = \sum_{i=0}^{\min\{j-1,\,k-1\}} (-1)^{k-1-i} \binom{k-1}{i} \binom{k-1+i}{i} \binom{j-1}{i} \Big/ \binom{n-1}{i} \tag{6.50}$$

The $w_{j:n}^{(r)}$ (weights), when graphically depicted, visually show the relative contribution of each data value on the value for $\hat{\lambda}_r$. Readers should note that the quantity $1/n$ could be combined with $w_{j:n}^{(k)}$ for an alternative form—attention is needed in the form of the weight factor when comparing L-moment computations. When the weight factors are in the form

$w_{j:n}^{(k)}/n$ (note the $1/n$), then the weights express, with regard to sign, the relative amount that each sample order statistic contributes to a given λ_r.

USING R ——— ——— USING R

The $w_{j:n}^{(r)}$ for a sample $n = 19$ are shown in figure 6.1. The figure shows the relative contribution of each ordered observation on the summation for the L-moment. The plots were generated by example 6–11 and are based on the `Lcomoment.Wk()` function. This example reproduces the weight factor distributions as shown in a figure by Hosking and Wallis (1997, fig. 2.6)—the τ_6 has been added for this dissertation.

6–11

```
n <- 19; k <- seq(1,n); Wk1 <- vector(mode = "numeric")
Wk2 <- Wk3 <- Wk4 <- Wk5 <- Wk6 <- Wk1
lab <- "RANK␣OF␣DATA␣VALUE,␣k"
# define pending graphics to two columns of three plots
#pdf("lmomWK.pdf")
layout(matrix(1:6, ncol=2))

for(r in k) Wk1[r] <- Lcomoment.Wk(1,r,n)
plot(k,Wk1, type="h", ylim=c(-1,1), xlab=lab,
                      ylab="Wk1,␣MEAN")
points(k,Wk1,pch=16); abline(0,0)
for(r in k) Wk2[r] <- Lcomoment.Wk(2,r,n)
plot(k,Wk2, type="h", ylim=c(-1,1), xlab=lab,
                      ylab="Wk2,␣L-SCALE")
points(k,Wk2,pch=16); abline(0,0)
for(r in k) Wk3[r] <- Lcomoment.Wk(3,r,n)
plot(k,Wk3, type="h", ylim=c(-1,1), xlab=lab,
                      ylab="Wk3,␣L-SKEW")
points(k,Wk3,pch=16); abline(0,0)
for(r in k) Wk4[r] <- Lcomoment.Wk(4,r,n)
plot(k,Wk4, type="h", ylim=c(-1,1), xlab=lab,
                      ylab="Wk4,␣L-KURTOSIS")
points(k,Wk4,pch=16); abline(0,0)
for(r in k) Wk5[r] <- Lcomoment.Wk(5,r,n)
plot(k,Wk5, type="h", ylim=c(-1,1), xlab=lab,
                      ylab="Wk5,␣TAU5")
points(k,Wk5,pch=16); abline(0,0)
for(r in k) Wk6[r] <- Lcomoment.Wk(6,r,n)
plot(k,Wk6, type="h", ylim=c(-1,1), xlab=lab,
                      ylab="Wk6,␣TAU6")
points(k,Wk6,pch=16); abline(0,0)
#dev.off()
```

◀

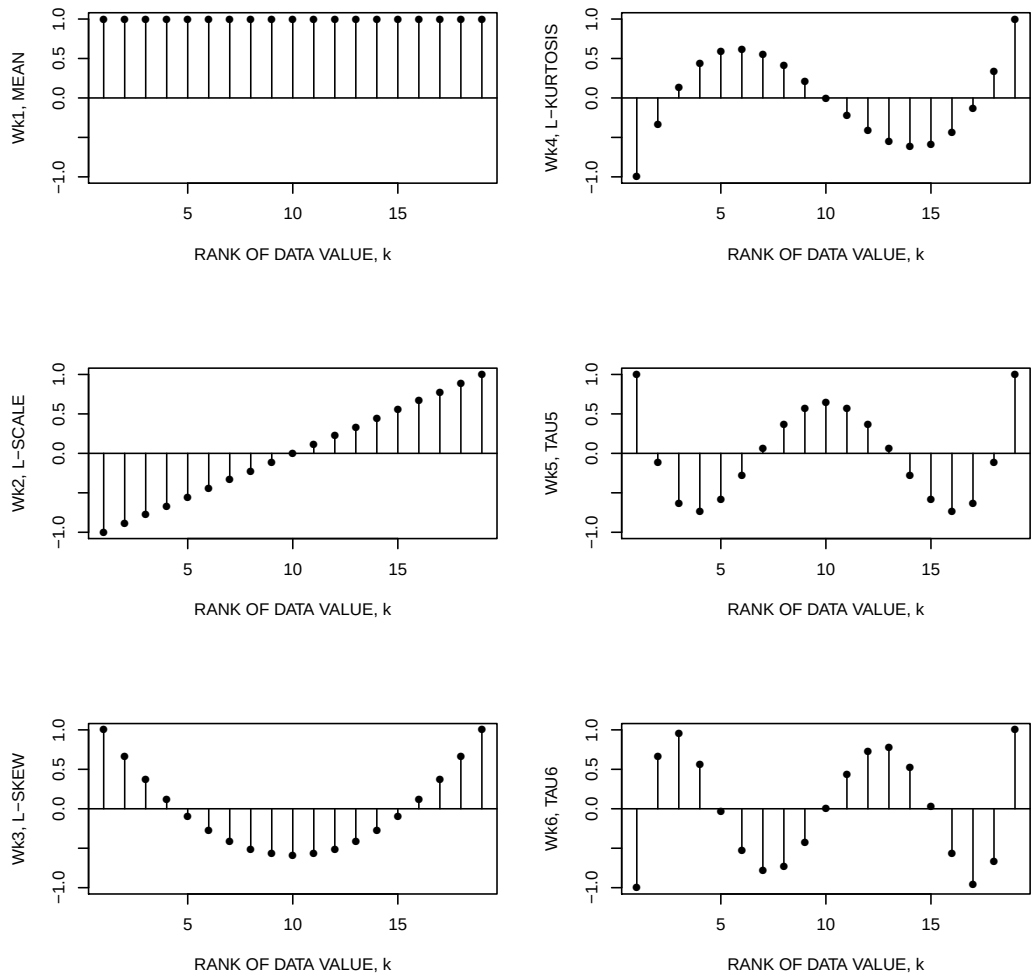

Figure 6.1. Graphics showing the weight factors of sample L-moment computation for each observation from a $n = 19$ sample on the respective L-moment from example 6–11

To conclude this section, it is informative to show an example of the L-moment weight factors for proportional computation of the L-moments from a sample. In example 6–12, it is shown for a sample $n = 4$ that each value contributes 0.25, whereas for the $\hat{\lambda}_2$, the order statistic $x_{2:4} = 20$ contributes -0.0833. Finally, the last two lines of output shows L-moment equivalence—note that `lmoms()` does not use the `Lcomoment.Wk()` function. Therefore, a double check of sorts is provided.

6–12

```
fakedat <- sort(c(-10, 20, 30, 40))
n <- length(fakedat)
Wk1 <- Wk2 <- Wk3 <- Wk4 <- vector(mode="numeric", length=n);
for(i in 1:n) {
```

```
   Wk1[i] <- Lcomoment.Wk(1,i,n)/n
   Wk2[i] <- Lcomoment.Wk(2,i,n)/n
   Wk3[i] <- Lcomoment.Wk(3,i,n)/n
   Wk4[i] <- Lcomoment.Wk(4,i,n)/n
}
cat(c("# Weights for mean",         round(Wk1, digits=4), "\n"))
# Weights for mean 0.25 0.25 0.25 0.25
cat(c("# Weights for L-scale",      round(Wk2, digits=4), "\n"))
# Weights for L-scale -0.25 -0.0833 0.0833 0.25
cat(c("# Weights for 3rd L-moment", round(Wk3, digits=4), "\n"))
# Weights for 3rd L-moment 0.25 -0.25 -0.25 0.25
cat(c("# Weights for 4th L-moment", round(Wk4, digits=4), "\n"))
# Weights for 4th L-moment -0.25 0.75 -0.75 0.25

my.lams <- c(sum(fakedat*Wk1), sum(fakedat*Wk2),
             sum(fakedat*Wk3), sum(fakedat*Wk4))
cat(c("# Manual L-moments:", my.lams, "\n"))
# Manual L-moments: 20 13.333 -5 5
cat(c("# lmomco L-moments:",lmoms(fakedat, nmom=4)$lambdas,"\n"))
# lmomco L-moments: 20 13.333 -5 5
```

◀

6.2.4 Reference Frame Comparison Between L-moments and Product Moments

The large conceptual leap of order-based statistics, such as the L-moments, is that distributional information contained in a sample is contained *both* in the spaces between observations and in the distances that observations are from the center of the distribution. Recognition of this duality is important as the duality might make the concepts and interpretations of L-moments compared to product moments easier to understand.

The author suggests that L-moments and product moments should be considered not just analogous but conceptually identical measures of the same geometric properties of a distribution, but moment-specific measurements differ according to the frame of reference. The reference frame concept is familiar to students of engineering and physics and therefore expository discussion is needed. One moment-type (L-moments) corresponds to the Lagrangian view, the other (product moments) corresponds to the Eulerian view of fluid movement.[6]

[6] The author acknowledges the "reference frame" idea and written suggestions from George "Rudy" Herrmann in fall of 2008.

The Lagrangian view is where the reference frame moves with the fluid, and the Eulerian reference system is where the reference frame is fixed and fluid moves past the reference frame. It can be conceptualized that L-moments often perform better than product moments because the overall set of measures has a narrower range of variation (is more compact), which allows for greater relative precision.

The Lagrangian-Eulerian comparison might be arcane, so consider the reference-frame comparison a little further:

- With L-moments, one traverses an ordered sample by traveling from one point to the next, the length of each leg of the trip is recorded, and various quantities based on these lengths are computed.
- With product moments, one traverses a random sample by traveling from the mean to each individual point in succession with no regard to order (data magnitude), the length of each individual and non-interacting trip (the Eulerian view) is recorded, and various quantities based on powers of the lengths are computed.

The total travel distance for the information content of the sample is greater in the Eulerian view and the average travel distance is greater as well as variously exponentiated. Hence, travel with this view is much less efficient (not the statistical meaning of efficient).

In an effort use other language for description, L-moments are "anchored" to the reference scale differently through ordering and intra-sample computations. Whereas, product moments are explicitly anchored to the reference scale by the mean and order is unimportant. Finally to conclude this discussion, it can be considered that

L-moments are statistics of "jumps" between the ordered sample values,

whereas

Product moments are statistics of "moment arms" about the mean.

Hopefully, this conceptualization and distinct will aid some readers in understanding the differences between the two moment definitions.

The comparison of the reference frame of L-moments and product moments is enhanced upon visualization of the differences in travel distances. In example 6–13, a random sample of $n = 100$ is drawn from an Exponential distribution. The absolute values of the trip distance from each observation to the mean are computed and set in the `PM` variable. The $n - 1$ intra-sample distances are computed and set in the `LM` variable. The values of the two variables are shown. Clearly, the travel distances for the computation of product moments (`PM`) are greater than those for the computation of L-moments (`LM`). The example concludes by plotting the results in figure 6.2. The figure shows that the intra-sample distances and individual trip distances to the mean are considerably smaller in magnitude and have smaller variation—hence, a source of the desirable sampling properties of L-moments that are described in Section 6.5.

6–13

```
n <- 100; fake.dat <- sort(rexp(n))    # n vals from Exponential
LM <- fake.dat[2:n] - fake.dat[1:n-1] # each intra-sample length
PM <- abs(fake.dat - mean(fake.dat))  # each trip from mean
cat(c(     "Total␣LM␣length␣=", round(sum(LM), digits=2),
      "␣and␣Total␣PM␣length␣=", round(sum(PM), digits=2),"\n\n"))
Total LM length = 5.53  and Total PM length = 79.1

#pdf("refframe.pdf")
plot(PM, ylab="INTRA-SAMPLE␣OR␣TRIP␣DISTANCE") # open circles
points(LM, pch=16) # solid circles
#dev.off()
```

◀

6.3 The Method of L-moments

The method of L-moments, as the name suggests, is a parameter estimation technique that is conceptually the same as the methods of product or probability-weighted moments already described. Specifically, the **method of L-moments** is a method of parameter estimation in which the parameters of a distribution are chosen so as to equate the theoretical L-moments of the distribution to the sample L-moments, or in other words, the parameters Θ are chosen such that $\lambda_r = \hat{\lambda}_r$ for the r number of parameters. The method is demonstrated by analytical derivation and then numerical example in this section—the method is used throughout this dissertation.

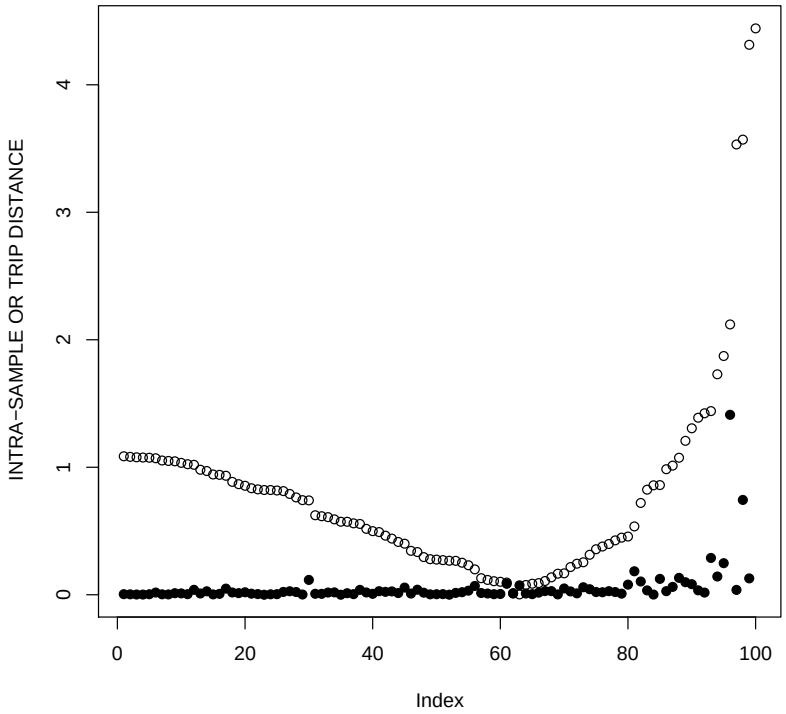

Figure 6.2. Comparison of intra-sample distances (solid circles) and individual trip distance to mean (open circles) for respective L-moment and product moment computation from example 6–13

The analytical mathematics to illustrate the method of L-moments are sufficiently described by derivation of the first two L-moments λ_1 and λ_2 of the **Uniform distribution**. The Uniform distribution is very simple. The QDF of the distribution with parameters α and β for nonexceedance probability F is

$$x(F) = \alpha + (\beta - \alpha)F \tag{6.51}$$

Now using eq. (6.11), λ_1 is defined by

$$\lambda_1 = \int_0^1 x(F)\ \mathrm{d}F \tag{6.52}$$

where upon substitution and expansion

$$\begin{aligned}\lambda_1 &= \alpha \int_0^1 \mathrm{d}F + \beta \int_0^1 F \; \mathrm{d}F - \alpha \int_0^1 F \; \mathrm{d}F \\ &= \alpha F \Big|_{F=0}^{F=1} + \frac{\beta}{2} F^2 \Big|_{F=0}^{F=1} - \frac{\alpha}{2} F^2 \Big|_{F=0}^{F=1} \\ &= \alpha + \frac{\beta}{2} - \frac{\alpha}{2} \end{aligned} \tag{6.53}$$

yields with simplification

$$\lambda_1 = \tfrac{1}{2}(\alpha + \beta) \tag{6.54}$$

And now using eq. (6.12), λ_2 is defined by

$$\lambda_2 = \int_0^1 x(F) \times (2F - 1) \; \mathrm{d}F \tag{6.55}$$

where upon substitution and expansion

$$\begin{aligned}\lambda_2 &= 2\alpha \int_0^1 F \; \mathrm{d}F + 2\beta \int_0^1 F^2 \; \mathrm{d}F - 2\alpha \int_0^1 F^2 \; \mathrm{d}F - \lambda_1 \\ &= \frac{2\alpha}{2} F^2 \Big|_{F=0}^{F=1} + \frac{2\beta}{3} F^3 \Big|_{F=0}^{F=1} - \frac{2\alpha}{3} F^3 \Big|_{F=0}^{F=1} - \lambda_1 \\ &= \alpha + \frac{2\beta}{3} - \frac{2\alpha}{3} - \frac{\beta}{2} - \frac{\alpha}{2} \end{aligned} \tag{6.56}$$

yields with simplification

$$\lambda_2 = \tfrac{1}{6}(\beta - \alpha) \tag{6.57}$$

Thus, the first two L-moments of the Uniform distribution are

$$\lambda_1 = \tfrac{1}{2}(\alpha + \beta) \tag{6.58}$$

$$\lambda_2 = \tfrac{1}{6}(\beta - \alpha) \tag{6.59}$$

The Uniform distribution is of limited interest in distributional analysis with the obvious and considerable exception of $\mathrm{UNI}(\alpha{=}0, \beta{=}1)$, which is equivalent to the default of the R function `runif()`. The Uniform distribution and the `runif()` function are critical for simulation of random variables. The mean (and median) nonexceedance probability of the $\mathrm{UNI}(\alpha{=}0, \beta{=}1)$ distribution is 0.5, which clearly is $(0 + 1)/2$ by eq. (6.58). Likewise, it follows that the λ_2 of the distribution by eq. (6.59) is $1/6$.

Suppose that the sample values $\hat{\lambda}_1$ and $\hat{\lambda}_2$ are 1 and 3, respectively, an equivalent Uniform distribution fit by the method of L-moments is established by

$$\hat{\lambda}_1 = 1 = \tfrac{1}{2}(\alpha + \beta) \Longrightarrow \alpha = 2 - \beta \tag{6.60}$$

$$\hat{\lambda}_2 = 3 = \tfrac{1}{6}(\beta - \alpha) \Longrightarrow \beta = 18 + \alpha \tag{6.61}$$

which upon further simplification yields $\mathrm{UNI}(\alpha{=}{-}8, \beta{=}10)$. For the example, the sample L-moments are thus equated to the theoretical L-moments of the distribution by adjusting (well directly solving for in this situation) the parameters—the method of L-moments is demonstrated. Some distributions are so complex that numerical methods must be employed to perform the method of L-moments. (Numerical methods also are common with use of product moments.)

USING R

The method of L-moments is further demonstrated in example 6–14. In the example, $n = 10{,}000$ values from a Gamma distribution having respective scale and shape parameters $\alpha = 3$ and $\beta = 4$ are simulated using the `rgamma()` function. The sample L-moments are computed by the `lmoms()` function. Next, the parameters are estimated by the `lmom2par()` function and an $n = 10{,}000$ sample is simulated using the `rlmomco()` function instead of the `rgamma()` function. The function concludes with a report of four (`2:5`) of the seven summary statistics returned by the `summary()` function.

6–14

```
n <- 10000 # simulated ten thousand samples
fake1.dat <- rgamma(n, scale=3, shape=4) # simulated values
lmr <- lmoms(fake1.dat) # compute the sample L-moments

# Solve for the parameters such that the theoretical L-moments
# of the distribution are set equal to the sample L-moments
# in the variable lmr.
PARgam <- lmom2par(lmr, type="gam") # L-moments --> parameters
fake2.dat <- rlmomco(n, PARgam) # simula. from gamma in lmomco
Rsum1  <- summary(fake1.dat, digits=5) # store the basic summary
Rsum2  <- summary(fake2.dat, digits=5) # stats in many variables
cat(c( names(Rsum1[2:5]),"\n", Rsum1[2:5],"\n", Rsum2[2:5],"\n"))
1st Qu. Median Mean 3rd Qu.
 7.5774 11.054 12.019 15.388
 7.5626 10.983 11.994 15.255
```

The summary statistics (minus the minimum and maximum) values are shown at the end of the example. The two rows of summary statistics are effectively identical. Many variations (and admittedly copies) of the algorithmic theme of the example are used throughout this dissertation. ◀

6.4 TL-moments—Definitions and Math

A special class of L-moments are **trimmed L-moments** (TL-moments). Elamir and Seheult (2003) describe TL-moments, which are based on trimming of the t_1-smallest and t_2-largest order statistics of a distribution or values from a sample. The TL-moments can be useful as they can extend L-moment-based statistics into difficult to work with distributions such as the Cauchy, which has infinite expectations of extreme value statistics, or increase the viable parameter space of a distribution such as that of the Generalized Lambda distribution. The TL-moments can provide further robustness relative to the L-moments because they provide various levels of symmetrical or asymmetrical trimming. However, this robustness comes at the cost of reducing the "information content" of the sample. However, for extremely heavy-tailed distributions TL-moments are useful in practice (Karvanen, 2006; Hosking, 2007a; Ahmad and others, 2011).

6.4.1 Theoretical TL-moments

The **theoretical TL-moments** for a real-valued random variable X with a QDF $x(F)$ are defined as

$$\lambda_r^{(t_1,t_2)} = \frac{1}{r}\sum_{k=0}^{r-1}(-1)^k \binom{r-1}{k} \mathrm{E}[X_{r+t_1-k:r+t_1+t_2}] \tag{6.62}$$

and can be computed by

$$\lambda_r^{(t_1,t_2)} = \underbrace{\frac{1}{r}}_{\substack{\text{average}\\ \text{of terms}}} \sum_{k=0}^{r-1} \overbrace{(-1)^k}^{\text{differences}} \underbrace{\binom{r-1}{k}}_{\text{combinations}} \frac{\overbrace{(r+t_1+t_2)!}^{\text{sample size}}\, I_{r,k}^{(t_1,t_2)}}{\underbrace{(r+t_1-k-1)!}_{\text{left tail}}\underbrace{(t_2+k)!}_{\text{right tail}}} \tag{6.63}$$

where

$$I_{r,k}^{(t_1,t_2)} = \int_0^1 \underbrace{x(F)}_{\substack{\text{quantile} \\ \text{function}}} \times \overbrace{F^{r+t_1-k-1}}^{\text{left tail}} \times \overbrace{(1-F)^{t_2+k}}^{\text{right tail}} \, \mathrm{d}F \tag{6.64}$$

where t_1 represents the trimming level of the t_1-smallest, t_2 represents the trimming level of the t_2-largest values, r represents the order of the TL-moments. The overbraces and annotations are added to this particular definition of an L-moment to conceptualize how the mathematics interact. For the condition $t_1 = t_2 = 0$, then eq. (6.38) is recovered.

Additional formulations of the theoretical TL-moments exist. Letting $P_r^{*(t_1,t_2)}(F)$ denote **shifted Jacobi polynomials** as

$$P_r^{*(t_1,t_2)}(F) = \sum_{j=0}^{r} (-1)^{r-j} \binom{r+t_2}{j} \binom{r+t_1}{r-j} F^j (1-F)^{r-j} \tag{6.65}$$

Hosking (2007b) shows that the TL-moments (and L-moments by $t_1 = t_2 = 0$) can be expressed in terms of the kth derivative for $k = 0, 1, 2, \ldots, r$ of the QDF $x^{(k)}(F)$ as

$$\begin{aligned} \lambda_{r+1}^{(t_1,t_2)} = {} & \frac{(r-k)!(r+t_1+t_2+1)!}{(r+1)!(r+t_1)!(r+t_2)!} \\ & \times \int_0^1 F^{t_1+k}(1-F)^{t_2+k} \, P_{r-k}^{*(t_1,t_2)}(F) \, x^{(k)}(F) \, \mathrm{d}F \end{aligned} \tag{6.66}$$

and in terms of the CDF $F(x)$, if the derivatives of the QDF do not exist, as

$$\begin{aligned} \lambda_{r+1}^{(t_1,t_2)} = {} & \frac{(r-1)!(r+t_1+t_2+1)!}{(r+1)!(r+t_1)!(r+t_2)!} \\ & \times \int_{-\infty}^{\infty} [F(x)]^{t_1+1} [(1-F(x))]^{t_2+1} \, P_{r-1}^{*(t_1+1,t_2+1)} \, F(x) \, \mathrm{d}x \end{aligned} \tag{6.67}$$

The TL-moments are logically extended to TL-moment ratios by

$$\tau_2^{(t_1,t_2)} = \lambda_2^{(t_1,t_2)} / \lambda_1^{(t_1,t_2)} \quad \text{for } X \geq 0 \tag{6.68}$$

and

$$\tau_r^{(t_1,t_2)} = \lambda_r^{(t_1,t_2)} / \lambda_2^{(t_1,t_2)} \quad \text{for } r > 2 \tag{6.69}$$

Hosking (2007b) shows that the TL-moment ratios, unlike the L-moment ratios, have bounds greater than 1 in absolute value for all $r \geq 2$, and these bounds increase as r

increases. The bounds for $r > 2$ are

$$|\tau_r^{(t_1,t_2)}| \leq \frac{2(m+1)!(r+t_1+t_2)!}{r(m+r-1)!(2+t_1+t_2)!} \quad \text{for } m = \min(t_1, t_2) \tag{6.70}$$

and when $t_1 = t_2 = 0$, eq. (6.70) reduces to eq. (6.28).

The TL-moments for arbitrary trimming levels are related by the following recurrence relations by Hosking (2007b)

$$(2r+t_1+t_2-1)\lambda_r^{(t_1,t_2)} = (r+t_1+t_2)\lambda_r^{(t_1,t_2-1)} - \frac{1}{r}(r+1)(r+t_1)\lambda_{r+1}^{(t_1,t_2-1)} \tag{6.71}$$

$$(2r+t_1+t_2-1)\lambda_r^{(t_1,t_2)} = (r+t_1+t_2)\lambda_r^{(t_1-1,t_2)} + \frac{1}{r}(r+1)(r+t_2)\lambda_{r+1}^{(t_1-1,t_2)} \tag{6.72}$$

Hosking (2007b, p. 3027) remarks that these relations "are of mostly mathematical interest," but does suggest that they might be useful for $\hat{\tau}_r$ for $r \geq 3$ near their theoretical bounds. For example, manipulation of the relations provides ${\tau_3}^{(0,1)} = (\tau_3 - \tau_4)/[2(1-\tau_3)]$, which could be used to estimate $\hat{\tau}_4$ from two different measures of L-skewness.

To conclude this section, Hosking (2007b, pp. 3034–3035) introduces yet another type of theoretical L-moment called **alternative trimmed L-moments**. These particular versions are analogous to, but are numerically distinct from, the "TL-moments" in this dissertation, which are specifically defined by eq. (6.62). These alternative trimmed L-moments are attractive because these and their respective L-moment ratios attain the "same range of feasible values as [the usual] L-moments." The alternative trimmed L-moments $\tilde{\lambda}_r^{(t_1,t_2)}$ in terms of the 1st derivative of the QDF $x^{(1)}(F)$ for $r \geq 2$ are

$$\begin{aligned}\tilde{\lambda}_{r+1}^{(t_1,t_2)} = {} & \frac{(t_1+t_2+1)!}{(r-1)!t_1!(t_2+1)!} \\ & \times \int_0^1 F^{t_1+1}(1-F)^{t_2+1}\, P_{r-1}^{*(1,1)}(F)\, x^{(1)}(F)\, \mathrm{d}F\end{aligned} \tag{6.73}$$

and the $\tilde{\lambda}_1^{(t_1,t_2)}$ (trimmed mean) is

$$\tilde{\lambda}_1^{(t_1,t_2)} = \mathrm{E}[X_{t_1+1:t_1+t_2+1}] \tag{6.74}$$

which can be expanded using eq. (3.4). Finally, Hosking (2007b) ends with an expression for the alternative trimmed L-moments in terms of order statistic expectations. They are the quantities

$$\tilde{\lambda}_{r+1}^{(t_1,t_2)} = \frac{(r-2)!}{(r+t_1+t_2)!}\sum_{k=0}^{r-1}(-1)^k\binom{r-1}{k}\frac{(t_2+k)!(r+t_1+t_2-k-1)!}{(k+1)!(r-k)!}$$
$$\times\,[k(k{+}1)(t_1{+}r{-}k)+(r{-}k)(r{-}k{-}1)(t_2{+}k{+}1)]\times \mathrm{E}[X_{r+t_1-k:r+t_1+t_2}] \quad (6.75)$$

It is obvious from eq. (6.75) that eq. (6.62) is much easier to handle, but by using R such complexity could readily hidden from the user.

USING R ——— ——— USING R

The theoretical TL-moments of a distribution can be computed by numerical integration with the `theoTLmoms()` function. For the following example 6–15, a Generalized Pareto distribution having parameters $(\xi, \alpha, \kappa) = (10, 5, 0.5)$ is specified by the `vec2par()` function to make an *lmomco* parameter list (see page 163 and ex. 7–1) in the variable `PARgpa`. The `theoTLmoms()` function computes the symmetrical ($t = t_1 = t_2 = 1$) TL-moments or $\lambda_r^{(1,1)} = \lambda_r^{(1)}$. The notation $t = \textit{integer}$ signifies symmetrical trimming. So for the example, the smallest and largest values are to be trimmed.

6–15

```
PARgpa <- vec2par(c(10,5,0.5), type="gpa")
lmr <- theoTLmoms(PARgpa, trim=1)

print(lmr)
$lambdas
[1] 13.14285731  0.76190493  0.07696026  0.01998022  0.00745943
$ratios
[1]         NA 0.05797103 0.10101032 0.02622404 0.00979050
$trim
[1] 1
$leftrim
NULL
$rightrim
NULL
$source
[1] "theoTLmoms"
```

By analytical solution $\tau_3^{(1)} = 10(1-\kappa)/[9(\kappa+5)]$ (see Section 8.2.6) and because $\kappa = 0.5$ in the example, $\tau_3^{(1)} = 0.1010101$ as the output from the `theoTLmoms()` function shows. The attributes `$trim`, `$leftrim`, and `$rightrim` of the *lmomco* L-moment list (see page 127 and exs. 6–7–6–9) in `lmr` summarize the t, t_1, and t_2 settings, respectively, for the call made to the `theoTLmoms()` function. The `$source` attribute, as seen in other special *lmomco* lists, such as *lmomco* probability-weighted moment list (see page 108

and examples 5–8 and 5–9) or *lmomco* parameter list (see page 163 and ex. 7–1), identifies the name of the called function. The ***lmomco* TL-moment list** in example 6–15 does not structurally differ from the other *lmomco* L-moment lists presented in USING R on page 127. ◀

6.4.2 Sample TL-moments

The **sample TL-moments** are computed from a sample using the sample order statistics $x_{1:n} \leq x_{2:n} \leq \cdots \leq x_{n:n}$ by

$$\hat{\lambda}_r^{(t_1,t_2)} = \frac{1}{r} \sum_{i=t_1+1}^{n-t_2} \left[\frac{\sum_{k=0}^{r-1} (-1)^k \binom{r-1}{k} \binom{i-1}{r+t_1-k-1} \binom{n-i}{t_2+k}}{\binom{n}{r+t_1+t_2}} \right] x_{i:n} \tag{6.76}$$

where t represents the trimming level of the t_1-smallest or t_2-largest values, r represents the order of the TL-moments.[7] If $t_1 = t_2 = 1$ and $r = 1$ for a TL-mean in eq. (6.76), then the Sen weighted mean of eq. (3.30) results.

USING R ——— ——— USING R

The sample TL-moments with symmetrical $t_1 = t_2 = 1$ trimming can be computed by the `TLmoms()` (*lmomco* package) and `t1lmoments()` (*Lmoments* package) functions. (The *lmom* package by Hosking (2009a) does not currently (2011) support TL-moments.) Example 6–16 demonstrates both functions for a sample drawn from an Exponential distribution. The example clearly shows that the *Lmoments* package has a considerably more curt data structure. However, the *lmomco* package provides for asymmetrical trimming.

6–16

```
fake.dat <- rexp(30) # 30 standard exponential samples
lmr1 <- TLmoms(fake.data, trim=1) # from package lmomco
lmr2 <- t1lmoments(fake.data)     # from package Lmoments
print(lmr1)           # L-moments from the lmomco package
$lambdas
```

[7] The denominator in eq. (6.76) is a constant and should be pulled out to the left and at the level of $1/r$. However, the constraints of typesetting require the quantity to be typeset where shown.

```
[1]  0.7308525656  0.2237977582  0.0581312744  0.0184304696
   -0.0009437471
$ratios
[1]           NA  0.306214644  0.259749136  0.082353236
   -0.004216964
$trim
[1] 1
$leftrim
NULL
$rightrim
NULL
$source
[1] "TLmoms"
print(lmr2) # L-moments from the Lmoments package
[1] 0.73085257 0.22379776 0.05813127 0.01843047
```

◀

The sample TL-moments with at least $t_1 = t_2 = 1$ permit estimation for a distribution such as the Cauchy, which has infinite extreme order statistics. Example 6–17 for sample sizes of $n = 10{,}000$ for 10 simulations shows the individual estimates of the usual (whole sample) sample mean, which does not exist for the distribution, and also shows the TL-mean for symmetrical trimming of the two smallest and two largest values. The results demonstrate that the sample mean is unstable and that the TL-mean ($t_1 = t_2 = 2$) is much more stable and more reliably shows that the central location of the symmetrical Cauchy is zero.

6–17

```
n <- 10000; nsim <- 10
trim <- 2 # symmetrical trimming of two values
for(i in seq(1,nsim)) {
  data   <- rcauchy(n)
  xbar   <- round(mean(data),3)
  lmr    <- TLmoms(data, trim=2)
  xbarTL <- round(lmr$lambdas[1],3)
  cat(    c("Mean␣(unstable)=",xbar,
        "␣␣␣TL-mean␣(trim=2)=",xbarTL, "\n"))
}
Mean (unstable)= -2.743    TL-mean (trim=2)= 0.014
Mean (unstable)= -0.699    TL-mean (trim=2)= -0.01
Mean (unstable)= 0.202    TL-mean (trim=2)= 0.002
Mean (unstable)= -0.006    TL-mean (trim=2)= 0.001
Mean (unstable)= 27.913    TL-mean (trim=2)= -0.048
Mean (unstable)= 0.055    TL-mean (trim=2)= 0.014
Mean (unstable)= 2.185    TL-mean (trim=2)= -0.009
Mean (unstable)= 33.053    TL-mean (trim=2)= 0.016
```

```
Mean (unstable)= 1.375    TL-mean (trim=2)= -0.022
Mean (unstable)= -0.48    TL-mean (trim=2)= 0.006
```

◀

6.5 Some Sampling Properties of L-moments

A feature of the sample L-moments $\hat{\lambda}_r$ is that they are unbiased by definition, and the sample L-moment ratios $\hat{\tau}_r$ are nearly so (Hosking and Wallis, 1995, p. 2021). As a result, specific corrections for sample size biases as seen in the sample product moments, such as the $n - 1$ and $n - 2$ terms, are not present. Further the $|\hat{\tau}_r|$ for $r \geq 3$ have defined bounds regardless of sample size in contrast to the product moments (see Sections 4.3.3 and 4.3.4).

6.5.1 Estimation of Distribution Dispersion

To demonstrate the estimation of distribution dispersion, unbiased estimation of distribution dispersion through $\hat{\lambda}_2$ is shown in example 7–8 on page 175 within ancillary context of the Normal distribution. Readers are left to generalize example 7–8 for other distributions of interest. ◀

6.5.2 Estimation of Distribution Skewness (Symmetry)

To demonstrate the estimation of distribution skewness, a Pearson Type III distribution is defined as PE3(100, 500, 3) in example 6–18. The L-moments of this distribution are computed by the `par2lmom()` function, and this distribution has $\tau_3 = 0.49$. A utility function `afunc()` is created to perform a single simulation of the defined sample size, compute $\hat{\tau}_3$, and return the difference between $\hat{\tau}_3$ (sampled value) and τ_3 (true value). The `replicate()` function is used to execute the full simulation run, and the `summary()` function is used to compute basic summary statistics of the differences in `deltaTau3`.

6–18

```
nsam   <- 20; nsim <- 2000
SHAPE  <- 3
PE3PAR <- vec2par(c(100,500,3), type="pe3")
lmr    <- par2lmom(PE3PAR) # compute L-moments
TAU3   <- lmr$TAU3 # value is 0.4889 with SHAPE <- 3

"afunc" <- function(nsam, para, U) {
   X <- rlmomco(nsam,para)    # draw random samples
   tlmr <- lmoms(X)           # compute L-moments
   return(tlmr$ratios[3] - U) # return the difference
}

deltaTau3 <- replicate(nsim, mean(afunc(nsam, PE3PAR, TAU3)))
summary(deltaTau3)
    Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
-0.46580 -0.09443 -0.01224 -0.01468  0.06798  0.37000
```

The summary statistics show that the mean difference is near zero, so a conclusion could be made that $\hat{\tau}_3$ is effectively an unbiased estimator of τ_3 even for a comparatively small sample size of $n = 20$. ◀

6.5.3 Estimation of Distribution Kurtosis (Peakedness)

To demonstrate the estimation of distribution kurtosis, let a bias ratio be the ratio of the bias [(sample statistic minus value for population) divided by the population statistic]. The bias ratios for the product moment $\hat{K}$ and L-moment $\hat{\tau}_4$ measures of distribution kurtosis are compared to the standard Normal distribution in example 6–19. The results are shown in figure 6.3. The figure shows that $\hat{\tau}_4$ is much more stable or less affected by sample size than $\hat{K}$. In both cases, the statistics over estimate kurtosis (emphasis that term is conceptual), and this over estimation decreases with increasing sample size. In fact by $n \approx 40$ and greater, $\hat{\tau}_4$ appears essentially unbiased. However, $\hat{K}$ is much more severely biased than $\hat{\tau}_4$ and especially so for small (less than about $n = 30$) sample sizes. Therefore, $\hat{\tau}_4$ clearly is a preferable estimator of distribution kurtosis.

6–19

```
nsam <- seq(5,100, by=5)
nsim <- seq(1,10000)
MU <- 0; SIG <- 1; T4 <- 0.1226
THEpar <- vec2par(c(MU,SIG), type="nor") # control dist. here
lme <- pmbias <- lmbias <- pme <- vector(mode = "numeric")
```

```
j   <- 0
for(n in nsam) {
   j <- j + 1; print(j)
   for(i in nsim) {
      pm  <- pmoms(rlmomco(n,THEpar))
      lmr <- lmoms(rlmomco(n,THEpar))
      pme[i] <- (pm$kurt - 3)/3
      lme[i] <- (lmr$ratios[4] - T4)/T4
   }
   pmbias[j] <- mean(pme)
   lmbias[j] <- mean(lme)
}
#pdf("unbias1.pdf")
plot(nsam,pmbias, type="l", lty=2, lwd=2, ylim=c(-0.05,0.40),
     xlab="SAMPLE␣SIZE", ylab="BIAS␣RATIO␣OF␣KURTOSIS")
lines(nsam,lmbias, lwd=3); abline(0,0)
#dev.off()
```

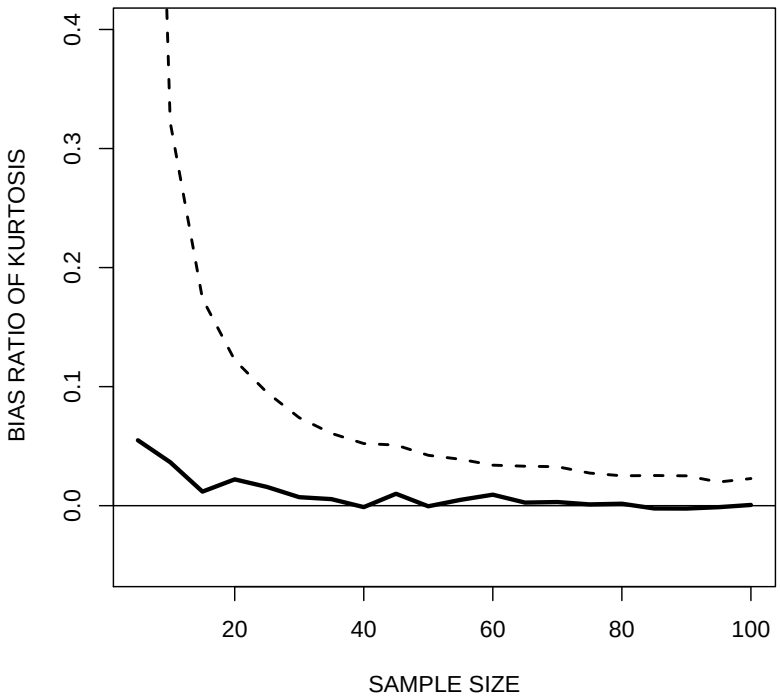

Figure 6.3. Bias ratios of product moment kurtosis (dashed line) and L-kurtosis (solid line) as a function of sample size for standard Normal distributed data from example 6–19

Example 6–19 is based on simulations of a standard Normal distribution. The results suggest an important interpretation—$\hat{\tau}_4$ is a superior estimator even on the "home turf"

(the Normal distribution) of the product moments. Emphasis is needed that the Normal is *not* a distribution associated with particularly heavy tails. Readers are encouraged to experiment with other distribution types and parameter combinations. Readers can rerun the example by changing the distribution type on the line commented as "`# control dist. here`." (See documentation `?dist.list` for the `dist.list()` function for a list of distribution abbreviations used by the *lmomco* package.) ◀

6.5.4 Boundedness of Coefficient of Variation Revisited

In Section 4.3.3, it is graphically demonstrated that the sample coefficient of variation $\hat{CV}$ is a bounded statistic by eq. (4.25) and that the CV will be underestimated by $\hat{CV}$. Succinctly, $\hat{CV}$ is a biased statistic. Although lacking numerical equivalency, the performance of $\hat{\tau}_2$ and $\hat{CV}$ as measures of relative variability can be loosely compared because $\hat{\tau}_2$ expresses relatively variability in the same fashion as $\hat{CV}$.

A comparison of relative variability is now made using the Gamma distribution. To begin, example 6–20 sets the true $\mu = 3{,}000$ and $CV = 0.9$ of the distribution in `True.MU` and `True.CV`, respectively. The standard deviation σ and variance σ^2 are computed. The shape and scale parameters of the Gamma (see Section 7.2.3 for definitions) are computed and set into the variables `s` and `a`, respectively.

Example 6–20 continues by converting a vector of the parameters into an *lmomco* parameter list (see page 163 and ex. 7–1) by the `vec2par()` function and in turn computing the true L-moments using the `lmomgam()` function. The L-moments are set into `True.LMR`. The true τ_2 is extracted from this list by `True.LMR$LCV` and subsequently output. The value is $\tau_2 = 0.460$; this is the relative L-variation of the defined Gamma distribution.

6–20

```
True.MU  <- 3000; True.CV  <- 0.9
True.SD  <- True.MU*True.CV
True.VAR <- True.SD^2
s <- True.VAR/True.MU
a <- True.MU/s # product moments of gamma
True.LMR <- lmomgam(vec2par(c(a,s), type="gam"))
True.LCV <- True.LMR$LCV # extract coe. of L-variation
print(True.LCV)
[1] 0.4599689
```

◀

Following example 6–20 and using the parameters `a` and `s` for selected sample sizes, a simulation study is performed in example 6–21. The results are plotted using example 6–22 and are shown in figure 6.4.

6–21

```
nsam <- c( 5,  8, 10,  14,  16,  20,  25,  30,  40, 50,
          60, 70, 80, 100, 120, 140, 160, 180, 200)
nsim <- 100
counter <- 0
cv  <- vector(mode="numeric")
lcv <- cvtmp <- lcvtmp <- cv
for(n in nsam) {
   counter <- counter + 1
   for(i in seq(1,nsim)) {
      x   <- rgamma(n, shape=a, scale=s)
      lmr <- lmoms(x)
      cvtmp[i]  <- sd(x)/mean(x) # CV hat
      lcvtmp[i] <- lmr$ratios[2] # Tau2 hat or LCV
   }
   cv[counter]  <- mean(cvtmp)
   lcv[counter] <- mean(lcvtmp)
}
```

It is seen in the figure that the bias ratio of $\hat{\tau}_2$ is much closer to unity and even is near unity at small sample sizes. The $\hat{CV}$ is substantially underestimating the population value and is still about 20 percent too low for $n \approx 200$. The utility of L-moments for estimation of relative variability of a distribution is evident.

6–22

```
#pdf("cvlcv.pdf")
plot(nsam, cv/True.CV, type="l",
     ylim=c(0.2, 1.1),
     xlab="SAMPLE␣SIZE",
     ylab="CV/(True␣CV)␣or␣L-CV/(True␣L-CV)")

lines(nsam, lcv/True.LCV, lty=2)

legend(50,0.4,
       c("PRODUCT␣MOMENT␣CV",
         "COE.␣OF␣L-VARIATION␣(L-CV)"),
       lty=c(1,2,3))
#dev.off()
```

◀

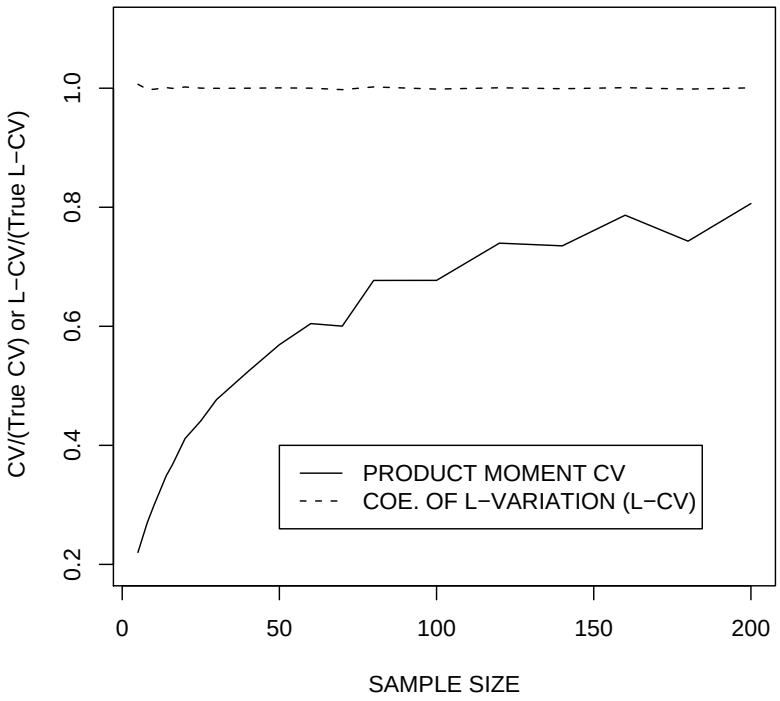

Figure 6.4. Demonstration of boundedness and bias of $\hat{CV}$ and unbiased property of $\hat{\tau}_2$ for a Gamma distribution having $\mu = 3{,}000$ and $CV = 10$ from example 6–22

6.5.5 Consistency and the Use of Logarithmic Transformation

This section is inspired by Hosking and Wallis (1997, pp. 39–40), in general, and their figure 15, in particular. Their figure depicts the performance of $\hat{G}$ and $\hat{\tau}_3$ with increasing sample size, in the context of the effects of a single high outlier on the estimation of $\hat{G}$ and $\hat{\tau}_3$.

An estimator is said to be **consistent** (Ugarte and others, 2008, pp. 252–254), if paraphrasing Ugarte and others, "the variance of a consistent estimator decreases as n increases and that the expected value [of the estimator] tends to [the true value] as n increases." The consistency of $\hat{G}$ in eq. (4.23) and $\hat{\tau}_3$ is explored in eq. (6.46) in the context of the log-Normal distribution. This distribution is positively skewed and hence right-tail heavy. However, further dilation of the right tail is made by contamination so that the robustness of the two estimators also can be compared.

A sampled log-Normal distribution is created for a sample of $n = 100$ in example 6–23 and set into the `fake.dat` vector. The example also produces the plot of the empirical

distribution seen in figure 6.5. The variable `zout` holds the value of the single-value contamination, which is appended to the `fake.dat` vector.

6–23

```
fake.dat <- 10^rnorm(99, mean=2, sd=0.5)
zout <- 7000 # a static value to increase right-tail weight
fake.dat <- c(fake.dat,zout) # add value to the vector
ef <- pp(fake.dat) # Weibull plotting positions
T <- prob2T(pnorm(log10(zout), mean=2, sd=0.5))
print(T) # equivalent recurrence interval
[1] 8925.334

#pdf("consist1.pdf")
plot(qnorm(ef), log10(sort(fake.dat)), type="b",
     xlab="STANDARD␣NORMAL␣DEVIATE")
#dev.off()
```

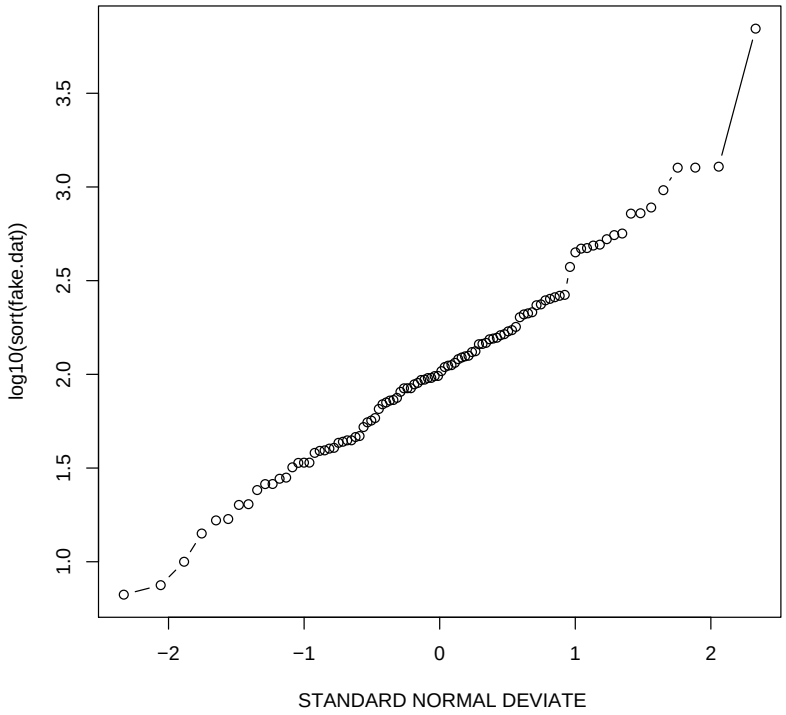

Figure 6.5. Empirical distribution of simulated log-Normal data from example 6–23

It should be pointed out that the value $X = 7{,}000$ (the high outlier and contamination) corresponds to $F = 0.999888$, and if the data were annual maxima, this is nearly the $9{,}000$-year event as shown in example 6–23.

To continue, the exploration of statistical consistency is made in example 6–24 for a range of sample sizes in variable `sams`. In the example, the `sample()` function is used to **bootstrap**[8] samples of size `n` with replacement from the data set. The number of occurrences of `zout` in the bootstrapped sample are set in variable `numzout`. Values in `numzout` are used to dynamically change the plotted symbol type and size. The example ends with the computation of the L-moments, product moments, and the product moments of the base-10 logarithms of each bootstrapped sample. The values for skewness are finally retained in the vectors `lskew`, `skew`, and `logskew`.

6–24

```
sams  <- seq(10,200)
lskew <- vector(mode = "numeric")
skew  <- logskew <- sym <- siz <- lskew
for(n in sams) {
   i <- n - 9
   sim <- sample(fake.dat,n, replace=TRUE) # bootstrap
   sym[i] <- 1; siz[i] <- 1 # reset symbol type and size
   numzout <- length(sim[sim == zout]) # count of outliers
   if(numzout > 0) sym[i] <- 16; siz[i] <- numzout
   lmr <- lmoms(sim) # compute L-moments
   pmr <- pmoms(sim) # compute product moments
   logpmr <- pmoms(log10(sim)) # compute pmoms of log10s

   lskew[i]   <- lmr$ratios[3] # save L-skew
    skew[i]   <- pmr$ratios[3] # save Skew (product moment)
   logskew[i] <- logpmr$ratios[3] # save Skew of log10s
}
```

The values for skewness ($\hat{\tau}_3$ and $\hat{G}$) for each sample size computed in example 6–24 are plotted by example 6–25 and shown in figures 6.6–6.8. In the figures, an effective use of combined symbol size, coloring, and transparency is seen that depicts the effect of the presence of the `zout` values on the random samples.

6–25

```
#pdf("consist2.pdf", version="1.4")
plot(sams, lskew, xlab="SAMPLE␣SIZE", ylab="L-SKEW",
                  pch=sym, cex=siz,
```

[8] Ugarte and others (2008, p. 469) report that "bootstrap" is an allusion to a German legend about a Baron Münchhausen, who was able to lift himself out of a swamp by pulling himself up by his own hair. The author had previously understood this legend to be the source of bootstrap, but does not recall the other source(s). In the 1988 movie *The Adventures of Baron Munchausen* (note spelling difference) or *Abenteuer des Baron von Münchhausen, Die* (Germany), the Baron character played by John Neville pulls himself out of the sea and not a swamp.

```
                    col=rgb(0,0,0,0.5))
#dev.off()
#pdf("consist3.pdf", version="1.4")
plot(sams, skew, xlab="SAMPLE␣SIZE", ylab="SKEW",
                  pch=sym, cex=siz,
                  col=rgb(0,0,0,0.5))
#dev.off()
#pdf("consist4.pdf", version="1.4")
plot(sams, logskew, xlab="SAMPLE␣SIZE",
                    ylab="SKEW␣OF␣LOGARITHMS",
                    pch=sym, cex=siz,
                    col=rgb(0,0,0,0.5))
#dev.off()
```

In figures 6.6–6.8, the open circles represent samples in which `zout` high outlier was not drawn ("drawn" as picked by `sample()` function) and conversely, the grey circles represent samples in which one or more values of `zout` were drawn. The size of the grey circles are successively increased according to the number of `zout` values that were generated by the simulation.

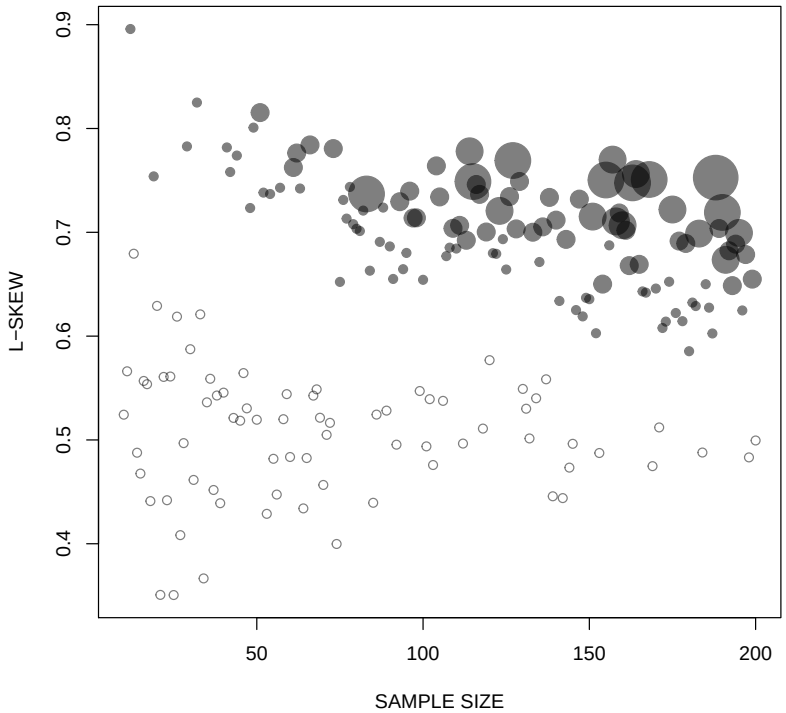

Figure 6.6. Relation between $\hat{\tau}_3$ and sample size of simulated log-Normal distribution shown in figure 6.5 from example 6–24

Figures 6.6–6.8 show that the probability of one or more drawings of `zout` increases with increasing sample size. This conclusion is made because there is increasing density and often size of grey circles as $n \rightarrow 200$. It also is seen in the figures, in particular figure 6.6, that two general states of sample skewness estimation exist. In general, but not exclusively, the sample values of $\hat{\tau}_3$ and $\hat{G}$ become more positive as sample size increases.

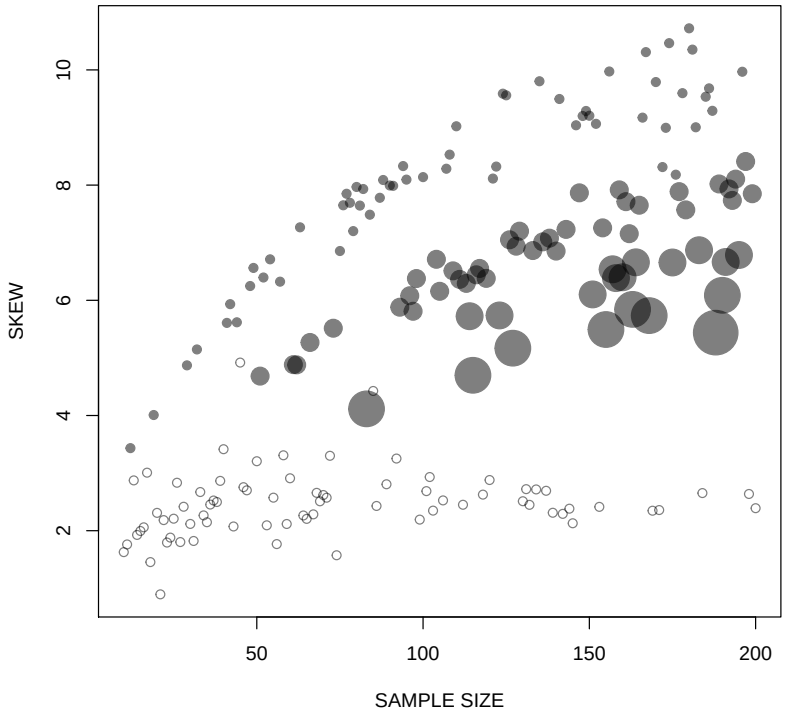

Figure 6.7. Relation between $\hat{G}$ and sample size of simulated log-Normal distribution shown in figure 6.5 from example 6–24

Considering first figure 6.6 and the open circles, these symbols form a "mirrored parabola" shape with the tapered or diminishing end toward the right. This shape shows the reduction in sampling variance as n increases and the tapered end is trending towards $\tau_3 \approx 0.52$, which is about `lmoms(10^rnorm(100000,mean=2,sd=0.5))`. Considering the grey circles, a similar pattern also is seen when `zout` values are included in the samples, but the $\hat{\tau}_3$ values are about 1.4 times larger—the effect of `zout` is thus to increase distribution skewness (not skewness as measured by G) to the right as expected. To clarify, the sample values of $\hat{\tau}_3$ increase as the number of `zout` values in the sample increases.

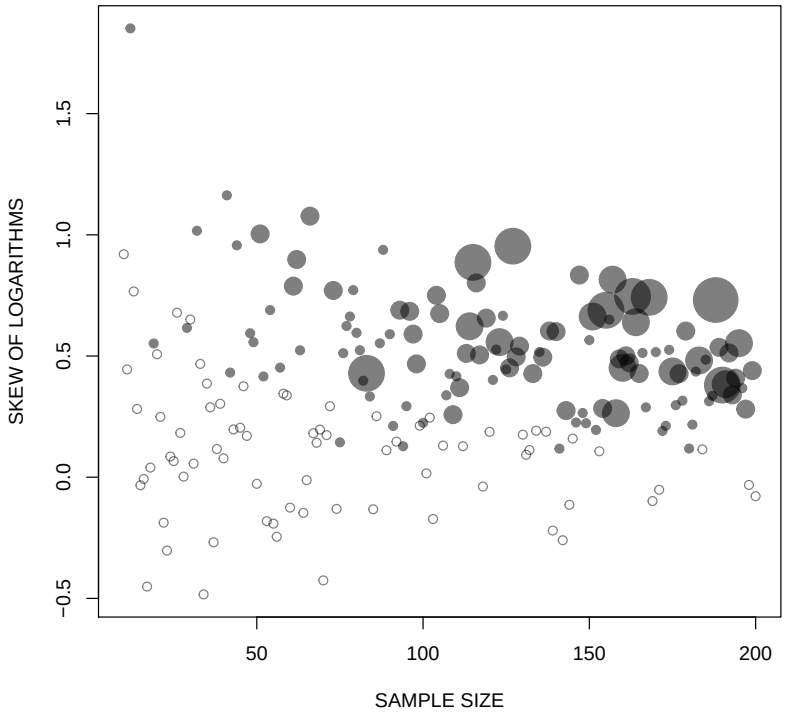

Figure 6.8. Relation between $\hat{G}$ of logarithms and sample size of simulated log-Normal distribution shown in figure 6.5 from example 6–24

Drawing attention from $\hat{\tau}_3$ to $\hat{G}$ in figure 6.7, it is again seen that the open circles form a mirrored parabola with the tapering-end toward the right. Because the open circles taper to the right, consistency for the estimator $\hat{G}$ is suggested. An evaluation of this observation is made in example 6–26.

6–26

```
n <- 100000 # sample size
x <- replicate(20, mean(pmoms(10^rnorm(n,mean=2,sd=0.5))$skew))
summary(x) # summary of 20 replicates
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  6.586   7.483   8.399   8.342   9.147  10.350
```

As seen in the example, the `summary()` function shows statistical values that strongly suggest that $\hat{G}$ is an inconsistent estimator for a heavy-tailed distribution even without contamination. The statistics in example 6–26 have grown without bounds—this is the inconsistency.

Consider now the contamination by `zout`, the grey circles in figure 6.7 plot apparently further from the open circles than seen in figure 6.6, but much more alarming is that the

trend of the $\hat{G}$ values is to expand in variance as n increases. Hence, the inconsistency of $\hat{G}$ is further demonstrated. Values for $\hat{G}$ actually decrease as the number of `zout` values increases. The sample distribution appears less right-tail heavy as more `zout` are encountered, which is a contradiction to, that is, inconsistent with, the anticipated result of the contamination.

Finally, the use of the $\hat{G}$ of the logarithms is considered in figure 6.8. The $\hat{G}$ values now show a tapering-to-right mirrored parabola shape as seen for $\hat{\tau}_3$ in figure 6.6. A consistent estimator of the skew of the logarithms thus is suggested. In fact, notice that the open circles trend towards $G = 0$, which is the product moment skew of the Normal distribution, and the Normal is of course the distribution of the $\log_{10}$ of the sample values drawn from a log-Normal distribution. It can be concluded that if a distribution is Normal or not too far from Normal, the estimator $\hat{G}$ is consistent.

This discussion illustrates why product moments of logarithms are frequently used in disciplines with substantially skewed data. The logarithms of the data frequently reduce data skewness. However, the author argues that the analyst ends up then analyzing the logarithms of the data and not the data in their native unit system—not always a philosophically attractive situation. The use of the L-moments generally avoids the need for logarithmic transformation (or any other) and avoids resultant concerns of transformation and re-transformation bias. ◀

6.6 Multivariate L-moments

This dissertation is focused on univariate distributional analysis using L-moments. L-moments, however, are extendable into multivariate space, and this exciting extension of L-moment theory is described in the final section of this dissertation (Section 12.9 on page 386).

6.7 Summary

In this chapter, historical background of L-moments is presented, and both the theoretical and sample L-moments were described. Principally, these are the L-moments of mean, L-scale, L-skew, and L-kurtosis. The boundedness of the L-moments is presented and compared to the unfavorable sample size boundedness of the sample product moments. The interrelations between L-moments and probability-weighted moments are shown along with visualization of the relative contribution (weight factors) of specific order statistics to the computation of the first five L-moments. The chapter continues with a theoretical and sample description of the TL-moments (the L-moments of trimmed samples). The sampling properties of L-moments are considered and closing discussion of logarithmic transformation, which began in Chapter 4, is completed. Finally, the 26 examples in the chapter demonstrated computations of these statistics using numerous L-moment-related functions such as `lmoms()`, `TLmoms()`, `lmom2pwm()`, and `samlmu()`.

Chapter 7
L-moments of Univariate Distributions

In this chapter, I present an introductory discussion of distribution support by L-moments in several R packages, but focus clearly is on the *lmomco* package. The introductory discussion applies to the following two chapters as well and applies to many examples already seen and others in the remainder of this dissertation. The chapter also, and more importantly, provides a distribution-by-distribution discussion of mathematics, features, parameters, and L-moments of two-parameter distributions. Readers possessing considerable familiarity with statistics and R are likely to generally browse as needed through the distributions. Other readers are encouraged to at least review this chapter with the mindset that periodic return likely will be made. This chapter is central to distributional analysis with L-moment statistics using R.

7.1 Introduction

Probability distributions are obvious and important concepts for distributional analysis. Distributions are chosen and parameters fit to data for various reasons and purposes. Numerous continuous distributions in the context of L-moment theory are available to the analyst, and many are considered in this dissertation. Some distributions such as the Normal or Gamma are well known across disciplines; whereas, others such as the Kumaraswamy or Kappa are not. This chapter, in conjunction with Chapters 8 and 9, represents a major reference component of this dissertation: L-moments and parameters of univariate distributions and using R to perform analysis with these distributions.

The preceding chapters provide background, definitions, general mathematics, and methods for computation of L-moments, probability-weighted moments, and related statistics. As required by the nature of the prior discussion and examples, L-moments

occasionally are used to compute distribution parameters from sample data, and parameters often are used to specify parent distributions in support of simulation experiments or graphical presentation. Many of the preceding examples also used simulation to study the sampling properties of L-moments and, by association, probability-weighted moments. Furthermore, comparisons of the sampling properties of L-moments to those of product moments also are made. Those earlier examples have used several of the distributions that are described in detail in either this chapter or Chapters 8 and 9.

It can be concluded from the preceding discussion that many aspects of this chapter, thus, have been foreshadowed in narrative and example. However, this chapter and Chapters 8 and 9 systematically treat with mathematical exposition the 22 distributions[1] supported by the *lmomco* package and in many cases by the *lmom* package as well.

The author acknowledges the semantic similarity between the distribution functions of the *lmom* and the *lmomco* packages. When both packages (*lmomco* and *lmom*) are loaded, a listing of the object(s) masked by one library over the other is provided to the user. For example, among about two dozen other naming conflicts,[2] both packages import a `cdfgev()` function. This function in both packages provides the CDF of the Generalized Extreme Value distribution.

Like all other chapters of this dissertation, Chapters 7–9 are heavily oriented towards the *lmomco* package. However, because of some functional similarity with *lmomco* and the enormous respect the author has for J.R.M. Hosking, the author has explicitly chosen to first list Jonathan's contributions of functions in the *lmom* package in tables 7.1 and 7.2 for the distribution functions and L-moment and parameter functions, respectively.

7.1.1 Chapter Organization

Although this chapter is focused on one- and two-parameter distributions, a synopsis of distributions and the presentation structure (layout) in this chapter and the following two chapters is appropriate.

[1] The log-Normal3 distribution is a special case of the Generalized Normal so the distribution is not separately counted although separate functions are provided by the *lmomco* package.

[2] The conflicts or "object masks" as of 2011 are: `cdfexp`, `cdfgam`, `cdfgev`, `cdfglo`, `cdfgno`, `cdfgpa`, `cdfgum`, `cdfkap`, `cdfln3`, `cdfnor`, `cdfpe3`, `cdfwak`, `cdfwei`, `quaexp`, `quagam`, `quagev`, `quaglo`, `quagno`, `quagpa`, `quagum`, `quakap`, `qualn3`, `quanor`, `quape3`, `quawak`, and `quawei`.

Table 7.1. Summary of distribution functions provided by the *lmom* package by Hosking (2009a)

Distribution	PDF	CDF	QDF
Exponential	--	`cdfexp()`	`quaexp()`
Gamma	--	`cdfgam()`	`quagam()`
Generalized Extreme Value	--	`cdfgev()`	`quagev()`
Generalized Logistic	--	`cdfglo()`	`quaglo()`
Generalized Normal	--	`cdfgno()`	`quagno()`
Generalized Pareto	--	`cdfgpa()`	`quagpa()`
Gumbel	--	`cdfgum()`	`quagum()`
Kappa	--	`cdfkap()`	`quakap()`
log-Normal3	--	`cdfln3()`	`qualn3()`
Normal	--	`cdfnor()`	`quanor()`
Pearson Type III	--	`cdfpe3()`	`quape3()`
Wakeby	--	`cdfwak()`	`quawak()`
Weibull	--	`cdfwei()`	`quawei()`

Table 7.2. Summary of L-moment and parameter functions by distribution provided by the *lmom* package by Hosking (2009a)

Distribution	L-moments	Parameters
Exponential	`lmrexp()`	`pelexp()`
Gamma	`lmrgam()`	`pelgam()`
Generalized Extreme Value	`lmrgev()`	`pelgev()`
Generalized Logistic	`lmrglo()`	`pelglo()`
Generalized Normal	`lmrgno()`	`pelgno()`
Generalized Pareto	`lmrgpa()`	`pelgpa()`
Gumbel	`lmrgum()`	`pelgum()`
Kappa	`lmrkap()`	`pelkap()`
log-Normal3	`lmrln3()`	`pelln3()`
Normal	`lmrnor()`	`pelnor()`
Pearson Type III	`lmrpe3()`	`pelpe3()`
Wakeby	`lmrwak()`	`pelwak()`
Weibull	`lmrwei()`	`pelwei()`

- Section 7.1.2 provides, without regard to the number of parameters, an overview of the distributions supported and ancillary functions provided by the *lmomco* package. Tabulated suites of conceptually similar functions also are provided. The tables are intended to provide a semantic perspective of, and reference for, the nomenclature of the *lmomco* package.
- Section 7.2 provides details of one- or two-parameter distributions. Three-parameter and four- and more parameter distributions are similarly detailed in Chapters 8 and 9, respectively.

Internally, Chapters 7–9 are similarly organized. The introductory commentary for each distribution provides some measure of context or common application of the respective distribution. For each distribution, the DISTRIBUTION FUNCTIONS headings are mathematically oriented and provide the PDF, CDF, and QDF of the distribution if respective analytical expressions exist. The names and constraints of the parameters are identified, and the ranges or limits of the distribution are shown. The narrative also presents the relations between the L-moments and the parameters for the respective distribution.

The USING R identifiers, which follow the mathematics of each distribution, generally provide discussion and examples of the salient functions supporting the distribution and provide comparisons to built-in R functions as appropriate. The examples also vary by the types of distribution-specific functions that are demonstrated. To mitigate against intra-chapter redundancy and promote broad-scoped discussion of package-specific features across Chapters 7–9, the examples also vary considerably by scope and complexity. Finally, the USING R are written in a style intended to be suitable for readers to browse from distribution to distribution. As opportunity allowed or otherwise seemed appropriate, additional mathematical details are provided in the individual USING R narratives.

The selection of one or more distributions and evaluation of their general applicability is an important subject. Although the examples in this chapter and those in Chapters 8 and 9 provide many comparisons between distributions, neither this chapter or Chapters 8 and 9 specifically address the topic of distribution discrimination and selection. Distribution discrimination and selection is described in Chapter 10.

Final notes about the source of material, in particular, the mathematics of the two-parameter distributions, are needed. Unless otherwise stated the material is heavily based on the distribution-by-distribution summaries of Evans and others (2000), Hosking

(1996b), Hosking and Wallis (1997), and Stedinger and others (1993). These and additional citations are provided as needed on a distribution-specific basis.

7.1.2 Distributions of the *lmomco* Package

The *lmomco* package provides a myriad of functions for general distribution operations as well as distribution-specific functions of the properties of supported distributions. Many of these functions and respective features are demonstrated in this chapter and also in Chapters 8 and 9. As a beginning, several important functions and general conceptual design of function naming convention for the *lmomco* package need formal identification and discussion.

Distribution Functions of *lmomco*

For several distributions, such as the Normal, Exponential, Gamma, and others, R has built-in support, and the distribution functions are descriptively named (see Sections 2.1.1–2.1.4). The *lmomco* package, however, provides an alternative naming convention and parameter argument implementation.

For example of the *lmomco* naming convention, the PDF of the Normal distribution is the `dnorm()` function of R, which is implemented in *lmomco* as `pdfnor()` or in shorthand: `dnorm()`→`pdfnor()`. The CDF and QDF are `pnorm()`→`cdfnor()` and `qnorm()`→`quanor()`, respectively. Following this style, the PDFs are provided by functions titled `pdfXXX()`, where `XXX` is replaced by an abbreviation for the distribution. The CDFs of *lmomco* are provided by functions titled `cdfXXX()`, and the QDFs of *lmomco* are provided by functions titled `quaXXX()`. Distribution functions of *lmomco* for the PDF, CDF, and QDFs are listed in table 7.3.

The "distribution functions" listed in table 7.3 show that the *lmomco* package breaks considerably from R tradition in the naming of functions related to distributions. The nomenclature of R is fine, but the nomenclature can be restrictive if one has a requirement or need for shifting between (or experimenting with) different distributions as part of distributional analysis. The R nomenclature lacks some parallelism. However, mimicking the R tradition, *lmomco* has the following functions, which provide an alternative means of calling distributions by the `dlmomco()`, `plmomco()`, `qlmomco()`, and `rlmomco()` functions. This dialect simultaneously makes *lmomco* distribution support "more familiar"

to users already accustomed to R and provides a singularly unique and package-specific interface.

The *lmomco* package provides a specific style of parameter argument implementation. The foremost difference in style from that of the R language is that *lmomco* provides functions that rely on the *lmomco* parameter list and the `$type` attribute of that list for proper routing. For example, 7–1 and the associated discussion that preceeds formally present the "***lmomco* parameter list**." This list is used by many functions of *lmomco* that need parameters. For the example, a Generalized Normal distribution parameter list `GNOpar` is constructed in which the three parameters are $\xi = -228$, $\alpha = 330$, and $\kappa = 0.413$ (see Section 8.2.3). These parameters are stored in the `$para` attribute of the list. The `$type` attribute has been tagged as `"gno"` (Generalized Normal). The `$source` attribute simply lists the name of the function that generated the list. This attribute is not used for internal operations of *lmomco*, but it is provided for user reference and unforeseen application needs.

Further discussion about the parameter vector in `GNO$para` is needed. The vector `$para` stores the parameters in "moment order," which also is the order shown in the first sentence under the DISTRIBUTION FUNCTIONS headings of this and Chapters 8 and 9. For the example distribution, the moment-order listing for the Generalized Normal distribution in the previous paragraph is GNO(-228, 330, 0.413).

7–1

```
GNOpar <- vec2par(c(-228, 330, 0.413), type="gno")
str(GNOpar)
List of 3
 $ type  : chr "gno"
 $ para  : num [1:3] -228 330 0.413
 $ source: chr "vec2par"
```

◀

Concluding commentary is needed. The R environment is built around the design ideal that distribution functions receive some—that is, not necessarily all—parameters through named arguments to the function. Whereas, *lmomco* has more compartmentalized design ideals in which a data structure represents the single parameter argument to the distribution functions. Example 7–2 provides a comparison of implementation styles for reporting the upper quartile $X_{0.75}$ of the Normal distribution. Four different approaches are used in the example, and the output is shown on the last line of the example: $X_{0.75} = 1{,}067$.

7–2

```
up.qrt.R <- qnorm(0.75, mean=1000, sd=100)    # built-in R
NORpar   <- vec2par(c(1000,100), type="nor") # lmomco
up.qrt.lmomco1 <-  quanor(0.75,NORpar)       # lmomco
up.qrt.lmomco2 <- par2qua(0.75,NORpar)       # lmomco
up.qrt.lmomco3 <- qlmomco(0.75,NORpar)       # lmomco
my75 <- c(up.qrt.R,
          up.qrt.lmomco1, up.qrt.lmomco2, up.qrt.lmomco3)
my75 <- sapply(my75,round)
cat(c(my75,"\n")) # results
1067 1067 1067 1067
```

For its distribution functions, the *lmom* package consistently uses a simple vector of parameter values. This style is an intermediate between the *lmomco* parameter list and the general, but not universal, named argument style of R. In the example, the differences in argument passage are contrasted for the Normal distribution. ◀

Conversion of Vectors to L-moments and Parameters using Functions of *lmomco*

Two commonly used convenience functions in the examples in this chapter and already seen elsewhere in this dissertation are `vec2lmom()` and `vec2par()`. These two functions and three others, which are conceptually related, are summarized in this section. The `vec2lmom()` function converts a vector of L-moments into an *lmomco* L-moment list (see page 127 and exs. 6–7 – 6–9). The list is used by many functions within the *lmomco* package that need L-moments. The list can be reverted to a vector by the `lmom2vec()` function. The `vec2par()` function converts a vector of parameters into an *lmomco* parameter list, which is shown and described in example 7–1 in the previous section. The opposite conversion is supported by the `par2vec()` function. The `vec2pwm()` function converts a vector of parameters into an *lmomco* probability-weighted moment list (see page 108 and examples 5–8 and 5–9). The list is used by many functions of *lmomco* that need probability-weighted moments. The list can be reverted to a vector by the `pwm2vec()` function. The five functions listed in this paragraph also are considered with other "high-level conversion" functions on page 169 and also listed in table 7.6.

Distribution Parameter Functions of *lmomco*

The parameters of a distribution are computed by the method of L-moments using functions that are titled by the following pattern `parXXX()`, where `XXX` is replaced by an abbreviation for the distribution. The function `dist.list()` provides a list of these

Table 7.3. Summary of distribution functions provided by the *lmomco* package by Asquith (2011)

Distribution	PDF	CDF	QDF
Cauchy	`pdfcau()`	`cdfcau()`	`quacau()`
Exponential	`pdfexp()`	`cdfexp()`	`quaexp()`
Gamma	`pdfgam()`	`cdfgam()`	`quagam()`
Generalized Extreme Value	`pdfgev()`	`cdfgev()`	`quagev()`
Generalized Lambda	`pdfgld()`	`cdfgld()`	`quagld()`
Generalized Logistic	`pdfglo()`	`cdfglo()`	`quaglo()`
Generalized Normal	`pdfgno()`	`cdfgno()`	`quagno()`
Generalized Pareto	`pdfgpa()`	`cdfgpa()`	`quagpa()`
Gumbel	`pdfgum()`	`cdfgum()`	`quagum()`
Kappa	`pdfkap()`	`cdfkap()`	`quakap()`
Kumaraswamy	`pdfkur()`	`cdfkur()`	`quakur()`
log-Normal3	`pdfln3()`	`cdfln3()`	`qualn3()`
Normal	`pdfnor()`	`cdfnor()`	`quanor()`
Pearson Type III	`pdfpe3()`	`cdfpe3()`	`quape3()`
Rayleigh	`pdfray()`	`cdfray()`	`quaray()`
Reverse Gumbel	`pdfrevgum()`	`cdfrevgum()`	`quarevgum()`
Rice	`pdfrice()`	`cdfrice()`	`quarice()`
Wakeby	`pdfwak()`	`cdfwak()`	`quawak()`
Weibull	`pdfwei()`	`cdfwei()`	`quawei()`
Right-Censored Generalized Pareto	`pdfgpa()`	`cdfgpa()`	`quagpa()`
Trimmed Generalized Lambda	`pdfgld()`	`cdfgld()`	`quagld()`
Trimmed Generalized Pareto	`pdfgpa()`	`cdfgpa()`	`quagpa()`

abbreviations, but the pattern should be evident from the tables in this section. For example, the parameters for the Normal distribution are computed by the `parnor()` function. Functions for the parameters in terms of L-moments for the distributions in table 7.3 are listed in table 7.4.

Distribution L-moment Functions of *lmomco*

The L-moments of a distribution are computed from the parameters using functions titled according to the following pattern `lmomXXX()`, where `XXX` is replaced by an abbrevi-

ation for the distribution. For example, the L-moments of the Normal distribution are computed by the `lmomnor()` function. Functions for the L-moments in terms of the parameters by distribution for the same distributions in table 7.3 are listed in table 7.4.

The `theoLmoms()` function computes the L-moments of distributions supported by *lmomco*. The function uses numerical integration and therefore bypasses analytical or quasi-analytical solutions shown in this chapter and Chapters 8 and 9. The algorithms in the `theoLmoms()` function are distinct from those in the `lmomXXX()` functions; the `lmomXXX()` functions, when possible, are based on analytical expressions or solutions with numerical approximations.

The author created the `theoLmoms()` function initially to have a development tool to test or otherwise validate the `lmomXXX()` functions. However, the function also can be used to compute L-moments of alternative distributions specified by parameters and R code not specifically provided by *lmomco*. For trimmed distributions, the `theoTLmoms()` function provides a similar role as `theoLmoms()` does for the non-trimmed distributions.

For example, the L-moments of the standard Normal distribution are computed in example 7–3. The results show that the mean $\mu = \lambda_1 = 0$ and $\lambda_2 = 1/\sqrt{\pi} \approx 0.564$ by definition for the standard Normal distribution and that $\tau_3 = 0$ and $\tau_4 \approx 0.123$ (see Section 7.2.1).

7–3

```
NORlmoms <- theoLmoms(vec2par(c(0,1), type="nor"))
str(NORlmoms)
List of 4
 $ lambdas: num [1:5] -4.36e-17  5.64e-01 -7.40e-17  6.92e-02
    -1.33e-16
 $ ratios : num [1:5]        NA -1.29e+16 -1.31e-16  1.23e-01
    -2.36e-16
 $ trim   : num 0
 $ source : chr "theoLmoms"
```

Distribution-Specific Convenience Functions of *lmomco*

The *lmomco* package provides numerous "distribution-specific convenience functions." The functions are listed in table 7.5. The convenience functions primarily are used as internal checks on the type (`is.XXX()`) of the *lmomco* parameter list for a given distribution and whether the parameters within the parameter list are valid (`are.parXXX.valid()`). These functions are provided at the user level so that developers could build higher-level interfaces in which such operations might be useful. The `are.par.valid()` function

Table 7.4. Summary of L-moment and parameter functions by distribution provided by the *lmomco* package by Asquith (2011)

Distribution	L-moments	Parameters
Cauchy	`lmomcau()`	`parcau()`
Exponential	`lmomexp()`	`parexp()`
Gamma	`lmomgam()`	`pargam()`
Generalized Extreme Value	`lmomgev()`	`pargev()`
Generalized Lambda	`lmomgld()`	`pargld()`
Generalized Logistic	`lmomglo()`	`parglo()`
Generalized Normal	`lmomgno()`	`pargno()`
Generalized Pareto	`lmomgpa()`	`pargpa()`
Gumbel	`lmomgum()`	`pargum()`
Kappa	`lmomkap()`	`parkap()`
Kumaraswamy	`lmomkur()`	`parkur()`
log-Normal3	`lmomln3()`	`parln3()`
Normal	`lmomnor()`	`parnor()`
Pearson Type III	`lmompe3()`	`parpe3()`
Rayleigh	`lmomray()`	`parray()`
Reverse Gumbel	`lmomrevgum()`	`parrevgum()`
Rice	`lmomrice()`	`parrice()`
Wakeby	`lmomwak()`	`parwak()`
Weibull	`lmomwei()`	`parwei()`
Right-Censored Generalized Pareto	`lmomgpaRC()`	`pargpaRC()`
Trimmed Generalized Lambda	`lmomTLgld()`	`parTLgld()`
Trimmed Generalized Pareto	`lmomTLgpa()`	`parTLgpa()`

provides a single and alternative interface, if not more convenient for the user, to the `are.parXXX.valid()` functions.

The following two examples in 7–4 and 7–5 demonstrate the use of the "parameter validation function" `are.parXXX.valid()`, and the use of the "distribution type function" `is.XXX()`. Example 7–4 sets the parameters of a Gumbel distribution fit to the L-moments of a fake data set into the `para` variable. Subsequently, the quantile `Q` for the median ($F = 0.5$) of the distribution is computed by the `quagum()` if the parameters in `para` are valid Gumbel parameters. An attempt to compute the median of the Exponential distribution follows; however, the attempt fails because the `type` of the `para`

Table 7.5. Summary of convenience functions by distribution provided by the *lmomco* package by Asquith (2011)

Distribution	Parameter validation	Distribution type
--	`are.par.valid()`	--
Cauchy	`are.parcau.valid()`	`is.cau()`
Exponential	`are.parexp.valid()`	`is.exp()`
Gamma	`are.pargam.valid()`	`is.gam()`
Generalized Extreme Value	`are.pargev.valid()`	`is.gev()`
Generalized Lambda	`are.pargld.valid()`	`is.gld()`
Generalized Logistic	`are.parglo.valid()`	`is.glo()`
Generalized Normal	`are.pargno.valid()`	`is.gno()`
Generalized Pareto	`are.pargpa.valid()`	`is.gpa()`
Gumbel	`are.pargum.valid()`	`is.gum()`
Kappa	`are.parkap.valid()`	`is.kap()`
Kumaraswamy	`are.parkur.valid()`	`is.kur()`
log-Normal3	`are.parln3.valid()`	`is.ln3()`
Normal	`are.parnor.valid()`	`is.nor()`
Pearson Type III	`are.parpe3.valid()`	`is.pe3()`
Rayleigh	`are.parray.valid()`	`is.ray()`
Reverse Gumbel	`are.parrevgum.valid()`	`is.revgum()`
Rice	`are.parrice.valid()`	`is.rice()`
Wakeby	`are.parwak.valid()`	`is.wak()`
Weibull	`are.parwei.valid()`	`is.wei()`
Right-Censored Generalized Pareto	`are.pargpa.valid()`	`is.gpa()`
Trimmed Generalized Lambda	`are.parTLgld.valid()`	`is.TLgld()`
Trimmed Generalized Pareto	`are.parTLgpa.valid()`	`is.TLgpa()`

list is not `"gum"`. The parameter validation functions internally call the distribution type tests by `is.XXX()` and check whether values of the parameters meet distribution-specific constraints.

7–4

```
para <- pargum( lmom.ub( c(123,34,4,654,37,78) ) )
if(are.pargum.valid(para)) Qgum <- quagum(0.5, para)
if(are.parexp.valid(para)) Qexp <- quaexp(0.5, para)
      # error message triggered because para is "gum"bel
Warning message:
In is.exp(para) : Parameters are not exponential parameters
```

◀

The followup to example 7–4 is 7–5 that shows use of the `is.glo()` function for a Generalized Logistic distribution that is fit to the sample L-moments by the `parglo()` function. The example does not verify whether the parameters are consistent with the indicated distribution—they would be in the example because the `parglo()` function returns valid parameters for the distribution for the sample data provided.

7–5

```
para <- parglo( lmom.ub( c(123,34,4,654,37,78) ) )
if(is.glo(para) == TRUE) {
  Q <- quaglo(0.5,para) # compute the median
  print(Q) # print the value, which is shown below
}
[1] 82.21451
```

◀

High-Level Conversion Functions of *lmomco*

The *lmomco* package provides several "high-level conversion" functions. These functions are listed in table 7.6. The functions require the L-moment and parameter lists and dispatch these lists to the respective distribution-specific functions (see tables 7.3 and 7.4). These functions collectively are a visible manifestation of the considerable differences in implementation philosophies for distribution functions built-in to R to those within the *lmomco* package.

Along with the high-level conversion functions listed in table 7.6, four even more general or high-level distribution functions are available. The four are listed in table 7.7 and have a naming convention that mimics the built-in distributions of R. Example 7–6 demonstrates and juxtaposes use of the `plmomco()` function between alternative methods of the *lmomco* package and those of R.

Table 7.6. Summary of high-level conversion functions provided by the *lmomco* package by Asquith (2011)

Function name	Action
`vec2lmom()`	Convert vector to L-moments
`lmom2vec()`	Convert L-moments to a vector
`vec2par()`	Convert vector to parameters
`par2vec()`	Convert parameters to a vector
`vec2pwm()`	Convert vector to probability-weighted moments
`pwm2vec()`	Convert probability-weighted moments to a vector
`par2lmom()`	Convert parameters to L-moments
`lmom2par()`	Convert L-moments to parameters
`par2pdf()`	Convert parameters to the PDF
`par2cdf()`	Convert parameters to the CDF
`par2qua()`	Convert parameters to the QDF
`are.lmom.valid()`	Check theoretical bounds of L-moments
`are.par.valid()`	Check parameters consistency for indicated distribution

7–6

```
the.shape <- 300; the.scale <- 500; my.x <- 150000
PARgam <- vec2par(c(the.shape,the.scale), type="gam")
plmomco(my.x,PARgam) # lmomco
[1] 0.5076778
cdfgam(my.x,PARgam)  # lmomco
[1] 0.5076778
par2cdf(my.x,PARgam) # lmomco
[1] 0.5076778
pgamma(my.x, shape=the.shape, scale=the.scale) # built-in R
[1] 0.5076778
```

◀

Example 7–7 demonstrates the utility of the *lmomco* parameter list. Using the given L-moments set by the `vec2lmom()` function into `lmr`, the parameters for Generalized Extreme Value, Gumbel, and Weibull distributions are computed, and 400 random values produced from each distribution. The empirical distribution of each distribution is developed by the plotting positions (`pp()` function) and the `sort()`ing of the values. The example completes by plotting the distributions. The three empirical distributions (Generalized Extreme Value, thin line; Gumbel, dashed line; and Weibull, thick line) are shown in figure 7.1.

Table 7.7. Summary high-level distribution functions of *lmomco* package by Asquith (2011) that mimic the nomenclature of R

Function name	Action
`dlmomco()`	Probability density functions (see Section 2.1.1)
`plmomco()`	Cumulative probability functions (see Section 2.1.2)
`hlmomco()`	Hazard functions (see Section 2.1.3)
`qlmomco()`	Quantile distribution functions (see Section 2.1.4)
`rlmomco()`	Random variates (random values)

7–7

```
lmr <- vec2lmom(c(100,200,-0.3,0.1)); n <- 400
PP  <- pp(1:n) # pp values of 1 through n
GEV <- rlmomco(n,lmom2par(lmr, type="gev"))
GUM <- rlmomco(n,lmom2par(lmr, type="gum"))
WEI <- rlmomco(n,lmom2par(lmr, type="wei"))
#pdf("rlmomcoA.pdf")
plot(PP,sort(GEV), type="l",
     xlab="NONEXCEEDANCE␣PROBABILITY",
     ylim=c(-1000,1000), ylab="QUANTILE")
lines(PP,sort(GUM), lty=2) # dashed line
lines(PP,sort(WEI), lwd=3) # thick line
#dev.off()
```

The basic algorithm in example 7–7 is simple and syntactically parallel—only the argument `type` to the `lmom2par()` function requires adjustment to change to another distribution. The dashed line of the Gumbel distribution is specified by the `lty=2` (line type) argument to `lines()`, and the thick line of the Weibull is specified by the `lwd=3` (line width) argument. ◀

7.2 One- and Two-Parameter Distributions of the *lmomco* Package

One- and two-parameter distributions are the simplest probability distributions in terms of fitting and interpretation. Such distributions generally are fit only to the first or only to the first (mean) and second moments (standard deviation or L-scale) of the data. In this chapter, unless otherwise stated, such fitting is understood as implying that the distributions are fit by the method of L-moments and specifically fit to the sample L-moments $\hat{\lambda}_1$ (mean) and $\hat{\lambda}_2$ (L-scale).

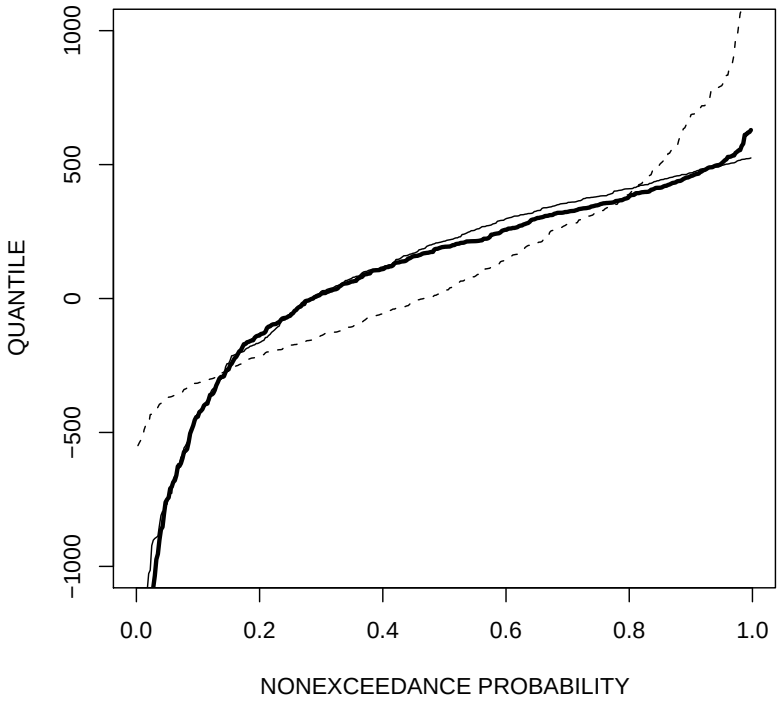

Figure 7.1. Example of three distributions, Generalized Extreme Value (thin line), Gumbel (dashed line), and Weibull (thick line) fit to the identical L-moments from example 7–7

For most one- and two-parameter distributions, the first parameter is known as the location parameter and the second parameter, if present, is known as the scale parameter. Following the lead of J.R.M. Hosking in his written works (see the References section that begins on page 402) and his FORTRAN library (Hosking, 1996b), *lmomco* implements the scale parameter as a true scale—meaning that the scale parameter has the same units as the location parameter. This philosophy is applied to scale parameters for distributions having three or more parameters. The author agrees with this philosophy and explicitly does not support in package-level code the inversion of scale parameters to "rate parameters" as R does—the Gamma distribution as implemented by R is an example.[3]

Occasionally, a two-parameter distribution can be reformulated as a one parameter if the location parameter simply provides a translation on the real-number line $\mathbb{R}$—such a

[3] The author historically has found the inconsistency of presentation of scale or rate parameters in distributions amongst various literature or software sources confusing. A likely source of the confusion is a background in a discipline (civil engineering and geosciences) that does not typically involve survival analysis. For survival analysis, the *rate* of events or death seems to be the more natural perspective/interpretation of distribution dispersion.

distribution is the Exponential distribution, use `help(rexp)` for the `rexp` function for details. Some distributions, such as the two-parameter Gamma distribution, lack a location parameter, but have the addition of a shape parameter along with the scale parameter. Distributions with three or more parameters as a rule have at least one shape parameter and are covered in later chapters.

7.2.1 Normal Distribution

The Normal distribution is well known and an extremely important distribution throughout all branches of statistics. The Normal distribution is a two-parameter distribution in which the parameters are conveniently the first two product moments: mean and standard deviation.

DISTRIBUTION FUNCTIONS

The distribution functions of the Normal having parameters μ (mean, location) and σ (standard deviation, scale, $\sigma > 0$) are

$$f(x) = \varphi(z) \text{ (a symbol by general convention, see below)} \tag{7.1}$$

$$F(x) = \Phi(z) \text{ (no explicit form, but a symbol by general convention)} \tag{7.2}$$

$$x(F) \text{ has no explicit analytical form}$$

where $z = (x - \mu)/\sigma$, $\varphi(a)$ is the PDF, and $\Phi(a)$ is the CDF of the standard Normal distribution, respectively. The QDF has no explicit analytical form, but a standard Normal can be approximated by eq. (7.12). The value z will occasionally be termed "standard normal deviate" for $\mu = 0$ and $\sigma = 1$ for many examples here and will be shown primarily on the horizontal axis of plots. The PDF and CDF are

$$\varphi(x) = \frac{\exp(-x^2/2)}{\sigma\sqrt{2\pi}} \tag{7.3}$$

$$\Phi(x) = \int_{-\infty}^{x} \varphi(x)\,\mathrm{d}t \tag{7.4}$$

The range of the distribution is $-\infty < x < \infty$.

The L-moments are

$$\lambda_1 = \mu \tag{7.5}$$
$$\lambda_2 = \sigma\pi^{-1/2} \tag{7.6}$$
$$\tau_3 = 0 \quad \text{(symmetrical)} \tag{7.7}$$
$$\tau_4 = 30\pi^{-1}\arctan(\sqrt{2}) - 9 = 0.1226 \tag{7.8}$$

The parameters are

$$\mu = \lambda_1 \tag{7.9}$$
$$\sigma = \lambda_2\sqrt{\pi} \tag{7.10}$$

Finally, the CDF and QDF of the standard Normal distribution can be respectively approximated (Stedinger and others, 1993, chap. 18, p. 11) by

$$F(z) = 1 - 0.5\exp\left[-\frac{(83z + 351)z + 562}{703/z + 165}\right] \quad \text{for } 0 < z \leq 5 \tag{7.11}$$

and

$$z(F) = 5.063[F^{0.135} - (1 - F)^{0.135}] \tag{7.12}$$

Readers are encouraged to compare eq. (7.12) to the QDF of the Generalized Lambda distribution in eq. (9.13) and see that eq. (7.12) is in the form of a Generalized Lambda.

A Normal distribution having $\mu = 0$ and $\sigma = 1$ is known as the **standard Normal distribution**. Finally, the **log-Normal distribution** is a Normal fit to the logarithms of a random variable.

USING R ——————————————— USING R

An investigation of the bias of $\hat{\sigma}$ compared to the bias of the product $(\hat{\lambda}_2\sqrt{\pi})$ as distinct estimators of σ when the parent is Normal follows. In example 7–8, a NOR($\mu = 10000$, $\sigma = 6000$) is specified. The vectors `e1` and `e2` will record the individual biases (errors) of each simulated value—the difference between the estimate and true value σ. As commonly done in this dissertation, the `rnorm()` function is used to generate simulated data. The $\hat{\sigma}$ is computed through eq. (4.19) and saved by `simsig <-sd(sim.dat)`, and $\hat{\lambda}_2$ is computed by the `lmoms()` function. The example ends with a report of the results.

7–8

```
mu <- 10000; sig <- 6000; n <- 20; nsim <- 10000
e1 <- vector(mode = "numeric"); e2 <- e1
for(i in seq(1,nsim)) {
  fake.dat <- rnorm(n, mean=mu, sd=sig)
  sim.sig <- sd(fake.dat)  # usual standard deviation
  lmr <- lmoms(fake.dat); siml2 <- lmr$lambdas[2]
  e1[i] <- sig - sim.sig; e2[i] <- sig - sqrt(pi)*siml2
}

cat(c("BIAS␣SD=",          round(mean(e1),2),
      "␣␣BIAS␣SD.via.L2=", round(mean(e2),2), "\n"))
BIAS SD= 57.09   BIAS SD.via.L2= -23.36
```

This particular example shows that $\hat{\lambda}_2$ has less bias ($|-23.36| < |57.09|$) than the familiar $\hat{\sigma}$ for a NOR(10000, 6000) with a small sample size of 20. The numerical results will vary and the sign on the estimated L-moment bias might change from time to time, but the conclusion will generally remain the same for this sample size ($n = 20$). The use of L-moments as potential drop-in-replacements for the product moments is partly demonstrated. Simply stated, the biases reported in example 7–8 show that "on average" for samples of $n = 20$ the estimation of σ using $\hat{\lambda}_2\sqrt{\pi}$ is less biased than $\hat{\sigma}$ when the parent is Normal.

Should L-moments, therefore, be used to estimate σ? Using R, simulation can be readily conducted for other sample sizes and by small modification to other distributions, the reader can judge for themselves. If the parent distribution is Normal, it seems L-moments might be preferred relative to the product moments to estimate the parameter σ for a sample of $n = 20$. However, $\lambda_2\sqrt{\pi}$ will not always be a preferable estimator of σ for other distributions such as for the Gamma distribution. ◀

7.2.2 Exponential Distribution

The Exponential distribution is a relatively simple distribution and is useful in applications involving constant failure rates. Many natural phenomena have, or approximately have, constant arrival (occurrence, failure) rates. The Exponential distribution, therefore, is frequently a first choice for distributional analysis for the aforementioned phenomena. As a result, the Exponential distribution works well for modeling the inter-arrival times. Phenomena involving the Exponential distribution can include arrival of precipitation

(storms), cosmic rays, customers, and wear out of parts. The Exponential as implemented by *lmomco* is a two-parameter version, whereas, the built-in version to R has one parameter.

DISTRIBUTION FUNCTIONS

The distribution functions of the Exponential having parameters ξ (location, lower bounds) and α (scale, $\alpha > 0$) are

$$f(x) = \alpha^{-1} \exp(-Y) \tag{7.13}$$

$$F(x) = 1 - \exp(-Y) \tag{7.14}$$

$$x(F) = \xi - \alpha \log(1 - F) \tag{7.15}$$

where

$$Y = (x - \xi)/\alpha \tag{7.16}$$

The range of the distribution is $\xi \leq x < \infty$.
The L-moments are

$$\lambda_1 = \xi + \alpha \tag{7.17}$$

$$\lambda_2 = \alpha/2 \tag{7.18}$$

$$\tau_3 = 1/3 \tag{7.19}$$

$$\tau_4 = 1/6 \tag{7.20}$$

The α parameter for a known ξ is

$$\alpha = \lambda_1 - \xi \tag{7.21}$$

and the parameters for an unknown ξ are

$$\alpha = 2\lambda_2 \tag{7.22}$$

$$\xi = \lambda_1 - \alpha \tag{7.23}$$

An extended form of the Exponential distribution exists, which is known as the **stretched Exponential** distribution or Kohlrausch function, has a PDF defined by

$$f(x) = \alpha^{-1} \exp(-[(x - \xi)/\alpha]^{\delta}) \tag{7.24}$$

where δ is a shape parameter. This distribution is also the survival function of the Weibull distribution and hence separate implementation in R is not needed.

USING R ——— ——— USING R

The single example 7–8 for the Normal distribution was comparatively complex. The code in that example is substantially simplified for the Exponential distribution to demonstrate the `parXXX()`, `lmomXXX()`, `quaXXX()`, and `cdfXXX()` functions using the Exponential distribution. (A demonstration of the `pdfXXX()` functions is shown for the Cauchy distribution in example 7–17 on page 184.)

The Exponential distribution is fit to some data in example 7–9 by the `parexp()` function. The returned *lmomco* parameter list (see page 163 and ex. 7–1) is labeled as `PARexp`. This list obviously is displayed by the `print()` function, and the output is shown in the example.

7–9

```
fake.dat <- c(1542, 1291, 578, 860, 968, 405, 326, 493, 829, 423)
lmr <- lmoms(fake.dat); PARexp <- parexp(lmr)

print(PARexp) # print the lmomco parameter list
$type
[1] "exp"
$para
      xi    alpha
299.8778 471.6222
$source
[1] "parexp"
```

◀

The L-moments of the fitted Exponential from example 7–9, or more generally any parameterized Exponential, are readily computed by the `lmomexp()` function as shown in example 7–10. The example also compares the fitted L-moments to the sample L-moments of the data. The `cat()` function and respective ensembles of output provide for a comparison between the L-moments—the ensembles are the same only through the second L-moment (λ_1=771.5, λ_2=235.8) and not for higher orders (τ_3=0.249, τ_3^{exp}=0.333).

7–10

```
LMRexp <- lmorph(lmomexp(PARexp))
cat(c(lmr$lambdas[1], lmr$lambdas[2],
      lmr$ratios[3],  lmr$ratios[4],"\n"))
771.5 235.811111111111 0.248928049757339 0.0525440728051105
```

```
cat(c(LMRexp$lambdas[1], LMRexp$lambdas[2],
       LMRexp$ratios[3],  LMRexp$ratios[4],"\n"))
771.5 235.811111111111 0.333333333333333 0.166666666666667
```

For this particular example, a conversion ("morphing") by the `lmorph()` function of the L-moment list is needed in order to acquire the appropriate list structure to make parallelism to the coding style (see documentation, `help(lmorph)`, and examples 6–8 and 6–9). ◀

The lack of rounding of the results shown in example 7–10 is unsightly. The output in example 7–11 is cleaner for the contents of the `lmr` variable originating from example 7–9. The `sapply()` and `round()` functions are used. The output is rounded to three digits by `digits=3`. The example shows how features of R can be used in compact and nested operations.

7–11

```
sapply(c(lmr$lambdas[1:2],
          lmr$ratios[3:4]), round, digits=3)
[1] 771.500 235.811   0.249   0.053
```

◀

The distribution functions of the Exponential are readily accessible. Assuming that the parameters from examples 7–9 and 7–10 are available, the median of the distribution (F=0.5) and the equivalent F value for 999 units of x are computed, respectively, by example 7–12. The $x_{0.50}$ is about 599 and $F(999) = 0.79$.

7–12

```
PARexp <- parexp(lmoms(c(1291, 578, 860, 968, 405, 326)))

quaexp(0.5,PARexp) # quantile function of exponential
[1] 598.9752

cdfexp(999,PARexp) # cdf of exponential
[1] 0.7932147
```

◀

The R environment has built-in functions for the Exponential distribution. For example, the QDF of the distribution is `qexp()`. The R implementation of the Exponential lacks the location parameter, which is provided by the *lmomco* package. A comparison of 75th-percentile computation is informative. Letting $\xi = 0$ and $\alpha = 200$, the parameters are set using the `vec2par()` function and proceed to compute quantiles from both the `quaexp()` and `qexp()` functions in example 7–13.

```
alpha <- 200
F <- 0.75
PARexp <- vec2par(c(0,alpha), type="exp")
built.in    <- qexp(F, rate=1/alpha)
from.lmomco <- quaexp(F,PARexp)

cat(c("#␣␣OUTPUT:",
      "qexp=", built.in, "␣␣",
      "quaexp=", from.lmomco,"\n"))
#  OUTPUT: qexp= 277.258872223978   quaexp= 277.258872223978
```

The two values are identical as anticipated. The `quaexp()` function provides more parallel syntax to other distributions within the *lmomco* package. The *lmomco* package provides more flexibility by implementing a two-parameter version of the Exponential distribution instead of a one-parameter version as is standard with R. The personal preference of the analyst obviously influences the choice of function to use. ◀

7.2.3 Gamma Distribution

The Gamma distribution is a two-parameter distribution that has a flexible width (scale) and shape. Because the distribution starts at the origin, the Gamma distribution can be useful for modeling of some phenomena that also are bounded below by zero. An interesting use of the Gamma distribution in a quasi-probabilistic application is shown in Asquith and Roussel (2007), in which the Gamma provides a structural form of a "unit hydrograph" representation of a streamflow hydrograph. Kliche and others (2008) provides a comparison of Gamma fitting to raindrop size using product moments, L-moments, and maximum likelihood and conclude that "[L-moments] outperform [product moments] and [maximum likelihood] for all [complete samples] studied" (Kliche and others, 2008, p. 3128).

DISTRIBUTION FUNCTIONS

The distribution functions of the Gamma having parameters α (shape, $\alpha > 0$) and β (scale, $\beta > 0$) are

$$f(x) = \frac{(x/\beta)^{\alpha-1} \exp(-x/\beta)}{\beta^{\alpha}\,\Gamma(\alpha)} \tag{7.25}$$

$$F(x) = \frac{1}{\beta^{\alpha}\,\Gamma(\alpha)} \int_0^x t^{\alpha-1} \exp(-t/\beta)\,\mathrm{d}F \tag{7.26}$$

$x(F)$ has no explicit analytical form

where $\Gamma(\alpha)$ is the complete gamma function that is shown in eq. (8.85).
The range of the distribution is $0 \leq x < \infty$.
The first two L-moments are

$$\lambda_1 = \alpha\beta \tag{7.27}$$

$$\lambda_2 = \frac{\beta}{\sqrt{\pi}} \exp(\log[\Gamma(\alpha + 0.5)] - \log[\Gamma(\alpha)]) \tag{7.28}$$

and the higher order L-moments are complex. Hosking (1996b) provides an algorithm using rational-function approximations for τ_3 and τ_4. The parameters in terms of the L-moments are complex. Hosking (1996b) provides minimax approximations for parameter estimation from the L-moments.

The **mode** statistic is the most frequently occurring value, and in continuous variables, the mode is the peak of the PDF. The mode of the distribution is $\mathrm{Mode}^{\mathrm{gam}} = \beta(\alpha - 1)$ for $\alpha \geq 1$. If $\alpha < 1$, then the PDF of the Gamma acquires a decaying shape towards the right in a similar fashion as the Exponential distribution. The mode can be used for parameter estimation if the mode of the distribution is known or otherwise needs to be locked-in at a given position. This application of the mode is of interest in use of the Gamma distribution for streamflow hydrograph modeling in which the peak streamflow corresponds to the mode of the distribution (Asquith and Roussel, 2007, appendix 4).

Unlike those for the L-moments, the relations between the product moments and the parameters are more straightforward and are

$$\alpha = \mu/\beta \tag{7.29}$$

$$\beta = \sigma^2/\mu \tag{7.30}$$

USING R ——————————

—————————— USING R

The Gamma distribution is demonstrated using some L-moments derived from a previous study. The L-moments listed in table 7.8 are derived from Asquith and others (2006)

and represent the first three L-moments of storm depth (depth of rainfall). Rainfall depth is a strictly positive phenomena and as a result positive skewness generally is present. These L-moments are based on real values—that is, not $\log_{10}$-transformed values—therefore application of a log-Normal distribution is not immediately feasible. However, the Gamma distribution has a zero lower bounds.

Table 7.8. L-moments of storm depth for storms defined by a minimum interevent time of 72 hours in Texas derived from Asquith and others (2006, table 5)

$\hat{\lambda}_1$ (inches)	$\hat{\tau}_2$	$\hat{\tau}_3$
0.964	0.581	0.452

Continuing the discussion with the code in example 7–14, the L-moments are set by the `vec2lmom()` function with the `lscale=TRUE` option being set because τ_2 is provided and not λ_2 as in virtually all other examples herein. The `pargam()` function estimates the Gamma distribution parameters from the L-moments, and the parameters are shown by the `str()` function. The QDF of the distribution for selected F values from the `nonexceeds()` function is generated by `quagam()`. The resulting plot is shown in figure 7.2. Example 7–14 shows that the $\tau_3^{\text{gam}} = 0.407$ of the fitted distribution by `lmomgam()` is close as well but is less than the $\hat{\tau}_3 = 0.452$ provided in table 7.8.

7–14

```
lmr <- vec2lmom(c(0.964,0.581,0.452), lscale=FALSE)
PARgam <- pargam(lmr); F <- nonexceeds()
print(PARgam)
$type
[1] "gam"
$para
    alpha      beta
0.6626539 1.4547565
$source
[1] "pargam"

#pdf("gammadistribution.pdf")
plot(F,quagam(F,PARgam), type="l")
LMRgam <- lmomgam(PARgam)
T3lmr <- lmr$TAU3; T3gam <- round(LMRgam$TAU3,3)
cat(c("TRUE␣L-skew=",T3lmr,"␣␣L-skew␣of␣Gamma=",T3gam,"\n"))
TRUE L-skew= 0.452   L-skew of Gamma= 0.407
#dev.off()
```

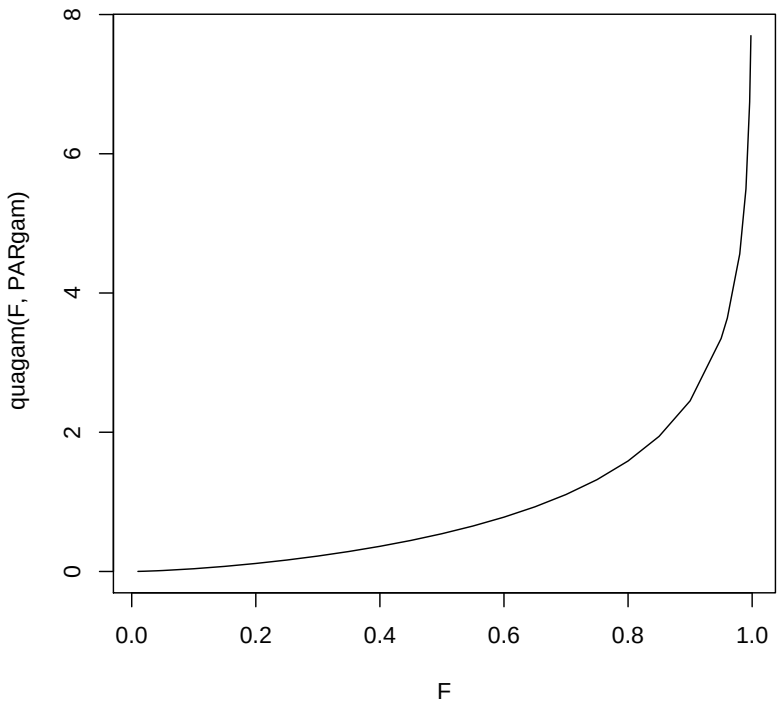

Figure 7.2. Quantile function of the Gamma distribution with $\alpha = 0.633$ and $\beta = 1.46$ from example 7–14

For the example, knowledgeable analysts might argue that an Exponential distribution should be considered because the Exponential, like the Gamma distribution, also has a lower bounds of zero. The *lmomco* package permits a quick comparison as shown in example 7–15.

7–15

```
LMRexp <- lmomexp(parexp(lmr)); T3exp <- round(LMRexp$TAU3,3)
cat(c("L-skew␣of␣Exponential␣=",T3exp,"\n"))
L-skew of Exponential = 0.333
```

The $\tau_3^{\text{exp}} = 0.333$ of the Exponential is much less than $\hat{\tau}_3 = 0.452$ of storm depth and much further from the $\tau_3^{\text{gam}} = 0.407$ of the fitted Gamma distribution. An immediate conclusion is that the Gamma distribution would be preferred for these sample L-moments because $\hat{\lambda}_1$ and $\hat{\lambda}_2$ are both fit when a Gamma is used, and τ_3^{gam} is closer to the $\hat{\tau}_3$ than τ_3^{exp}. This interpretation has broader ramifications related to the selection of distributions and is comprehensively explored in Chapter 10. ◀

The R environment has a built-in function named `qgamma()` for the quantiles of the Gamma distribution and the `quagam()` function uses this function. To demonstrate that

the functions are the same, the median of the example distribution is computed in example 7–16 in which the Gamma parameters in `PARgam` come from example 7–15.

7–16

```
# Native R code
qgamma(0.5, shape=PARgam$para[1], scale=PARgam$para[2])
[1] 0.5424176
# Using package lmomco
quagam(.5,PARgam)
[1] 0.5424176
```

When comparing the appearance of the calls to the QDF of the Gamma using the built-in R and *lmomco* styles, the author argues that the parameter list structure of *lmomco* provides a generally cleaner interface—as does the style used by the *lmom* package as well—but other factors certainly influence opinion. A feature of R is that it provides freedom of design (see Section 1.2). ◀

7.2.4 Cauchy Distribution

The Cauchy distribution is a very heavy-tailed distribution. The tails are so long in fact that the product moments and usual L-moments do not exist. However, if the smallest and largest values are trimmed, then the moments from the sample do exist. The trimmed L-moments (TL-moments) can be used with symmetrical trimming to provide a means of parameter estimation for the Cauchy.

DISTRIBUTION FUNCTIONS

The distribution functions of the Cauchy having parameters ξ (location) and α (scale, $\alpha > 0$) are

$$f(x) = \left(\pi\alpha\left[1 + \left(\frac{x-\xi}{\alpha}\right)^2\right]\right)^{-1} \tag{7.31}$$

$$F(x) = \frac{\arctan[(x-\xi)/\alpha]}{\pi} + 0.5 \tag{7.32}$$

$$x(F) = \xi + \alpha \times \tan[\pi(F - 0.5)] \tag{7.33}$$

The range of the distribution is $-\infty < x < \infty$.
The TL-moments with $t = 1$ symmetrical trimming are

$$\lambda_1^{(1)} = \xi \tag{7.34}$$
$$\lambda_2^{(1)} = 0.698\alpha \tag{7.35}$$
$$\tau_3^{(1)} = 0 \quad \text{(symmetrical)} \tag{7.36}$$
$$\tau_4^{(1)} = 0.343 \tag{7.37}$$

The parameters in terms of the L-moments are

$$\xi = \lambda_1^{(1)} \tag{7.38}$$
$$\alpha = \lambda_2^{(1)}/0.698 \tag{7.39}$$

Although the usual L-moments do not exist, the Cauchy distribution is the limiting point $\{\tau_3 \to 0, \tau_4 \to 1\}$ (Hosking, 2007b) on the L-moment ratio diagram of τ_3 and τ_4 (see Chapter 10).

USING R

The properties of the Cauchy distribution and some features of *lmomco* are now explored. In example 7–17, a Cauchy is specified using the `vec2par()` function. The commonly used (in this dissertation) `nonexceeds()` function returns a list of selected F values. The `par2qua()` function is used to convert the parameters into the quantiles of the distribution. For the example, the `quacau()` function could have been used instead because the `par2qua()` function simply dispatches to the `quacau()` function. The PDF of the distribution is created with the `pdfcau()` function and is shown in figure 7.3.

7–17

```
cau <- vec2par(c(100,200), type="cau")
F <- nonexceeds(); x <- par2qua(F,cau)
#pdf("cau1.pdf")
plot(x,pdfcau(x,cau), type="l", ylab="f(x)")
#dev.off()
```

◀

The L-moments of the Cauchy distribution do not exist because the extreme order statistics (minimum and maximum or $X_{1:n}$ and $X_{n:n}$) of the distribution are both infinite. However if these are trimmed, then the TL-moments can be computed. The largest and smallest values in other words must be discarded for moments to exist. However, an attempt is made to compute usual L-moments theoretically using the `theoLmoms()`

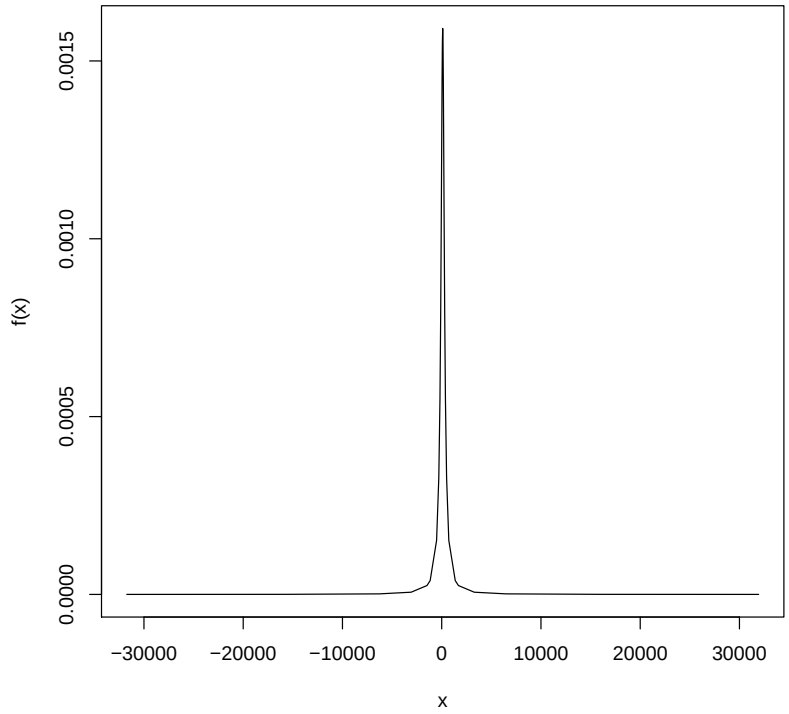

Figure 7.3. PDF of example Cauchy distribution from example 7–17

function in example 7–18. As the example shows, the `integrate()` function reports non-finite function values—the heavy tails of the Cauchy distribution.

7–18

```
cau <- vec2par(c(100,200), type="cau")
theoLmoms(cau) # first try regular L-moments and get error
Error in integrate(XofF, 0, 1) : non-finite function value
```

◀

The code in example 7–18 fails because of infinite extrema of the Cauchy distribution. Instead, the `theoTLmoms()` function is used in example 7–19 with symmetrical $t = 1$ trimming (`trim=1`) and three TL-moments on the return (`nmom=4`). The TL-moments are then shown by the `str()` function. Because the Cauchy is symmetrical, it is seen that $\tau_3^{(1)} = 0$. The theoretical integration shows that $\lambda_1^{(1)} = 100$ and $\lambda_2^{(1)} = 140$, which by eq. (7.35) should be $\lambda_2^{(1)} = 0.698 \times 100 = 139.6$.

7–19

```
the.lmr <- theoTLmoms(cau, trim=1, nmom=4)
str(the.lmr)
 $ lambdas : num [1:4]  1.00e+02  1.40e+02 -9.47e-15  4.78e+01
```

```
 $ ratios  : num [1:4]         NA  1.40e+00 -6.79e-17  3.43e-01
 $ trim    : num 1
 $ leftrim : NULL
 $ rightrim: NULL
 $ source  : chr "theoTLmoms"
```

◀

The ability for independent L-moment (or TL-moment) computation given a parameterized distribution is a feature of *lmomco*—the package has functions such as `theoLmoms()` and `theoTLmoms()` primarily for the purpose of permitting users to cross check the `lmomXXX()` (L-moments of distribution) functions. A check on the output of the function `theoTLmoms()` can be made by the `lmomcau()` function in example 7–20 and similarly of the output to that in example 7–19 is obvious.

7–20

```
lmomcau(cau)
$lambdas
[1] 100.0000 139.6000   0.0000  47.8828
$ratios
[1] 0.000 1.396 0.000 0.343
$trim
[1] 1
$source
[1] "lmomcau"
```

◀

7.2.5 Gumbel Distribution

Since the 1930s (Gilchrist, 2000, pp. 165–166), the two-parameter Gumbel distribution (**Extreme Value Type I**) has been an extensively studied and used distribution in the analysis of extremes such as floods, wave heights, rainfall, and lifetimes. The Gumbel has been formulated for positive skewness. A negatively skewed version is acquired by reflection and is known as the Reverse Gumbel distribution, which is described a separate section.

The Gumbel distribution often provides reasonable fits to many types of natural sciences data. For example, Thompson and others (2007) use the Gumbel to model a "dimensionless distribution with fixed scale parameter so only the mean earthquake magnitude

must be estimated for a region."[4] Clarke and Terrazas (1990) consider L-moments and the Gumbel for flood-flow regionalization of the Rio Uruguai. The three-parameter Generalized Extreme Value generally is now preferred over the Gumbel because the Gumbel distribution is a special case of the Generalized Extreme Value. Specifically, the Gumbel is not fit to the skewness of the data. Because the L-moments are such useful statistics for computation of distribution skewness, the preference for the Generalized Extreme Value is justified.

DISTRIBUTION FUNCTIONS

The distribution functions of the Gumbel having parameters ξ (location) and α (scale, $\alpha > 0$) are

$$f(x) = \alpha^{-1} \exp(Y) \exp[-\exp(Y)] \tag{7.40}$$

$$F(x) = \exp[-\exp(Y)] \tag{7.41}$$

$$x(F) = \xi - \alpha \log[-\log(F)] \tag{7.42}$$

where

$$Y = (x - \xi)/\alpha \tag{7.43}$$

The range of the distribution is $-\infty < x < \infty$.
The L-moments are

$$\lambda_1 = \xi + \alpha\rho \qquad \rho \text{ is Euler's constant, } 0.5772\ldots \tag{7.44}$$

$$\lambda_2 = \alpha \log(2) \tag{7.45}$$

$$\tau_3 = \log(9/8)/\log(2) = 0.1699 \tag{7.46}$$

$$\tau_4 = [16\log(2) - 10\log(3)]/\log(2) = 0.1504 \tag{7.47}$$

The parameters of the distribution are

$$\alpha = \lambda_2/\log(2) \tag{7.48}$$

$$\xi = \lambda_1 - \alpha\rho \tag{7.49}$$

[4] Thompson and others (2007) use the *lmomco* package for their L-moment computations. This paper provides the first known citation of *lmomco*.

Hershfield (1961) provides a venerable, but still authoritative, reference for the depth-duration frequency of rainfall in the United States. (Depth-duration frequency of rainfall also is considered in Section 11.1.) The Gumbel distribution was used by Hershfield in the regional study along with presumably considerable smoothing of contour lines of equal depth. The data listed in example 7–21 represent the 24-hour storm depths in inches having the respective annual recurrence intervals estimated by the author (Asquith) for the southern tip of Lake Michigan near the Illinois and Indiana border. An equivalent Gumbel distribution to these data is estimated—emphasis is needed that the data do not represent a random sample. Therefore, special processing is needed.

7–21

```
P <- c(2.8, 3.5, 4.0, 4.6, 5.2, 5.6) # precipitation data, inches
T <- c(2, 5, 10, 25, 50, 100) # recurrence interval, years
F <- T2prob(T) # re-express in nonexceedance probability

# custom quantile function of Gumbel, with no check on parameters
"myquagum" <- function(f, para) {
   return(para[1] - para[2] * log(-log(f)))
}

# objective function to minimize
"afunc" <- function(x, RHS=NULL, F=NULL) {
   return(sum((RHS - myquagum(F,x))^2))
}

# perform non-linear optimization
result <- optim(c(4,2), fn=afunc, RHS=P, F=F)
PAR <- vec2par(result$par, type="gum") # extraction of parameters

#pdf("tp40gum.pdf")
plot(F,quagum(F,PAR), type="l",
     xlab="NONEXCEEDANCE␣PROBABILITY",
     ylab="RAINFALL␣DEPTH,␣INCHES")
points(F,P)
#dev.off()
```

In the example, a custom QDF of the Gumbel distribution is created. This is done so that a currently (2011) hardwired parameter validation component of the `quagum()` function conducted by the `are.pargum.valid()` function is bypassed. The objective function `afunc()` returns the sum of square error for the $x(F)$ values in `P` for the desired F in `F`. The `optim()` function is used with initial starting parameter values of GUM(4, 2),

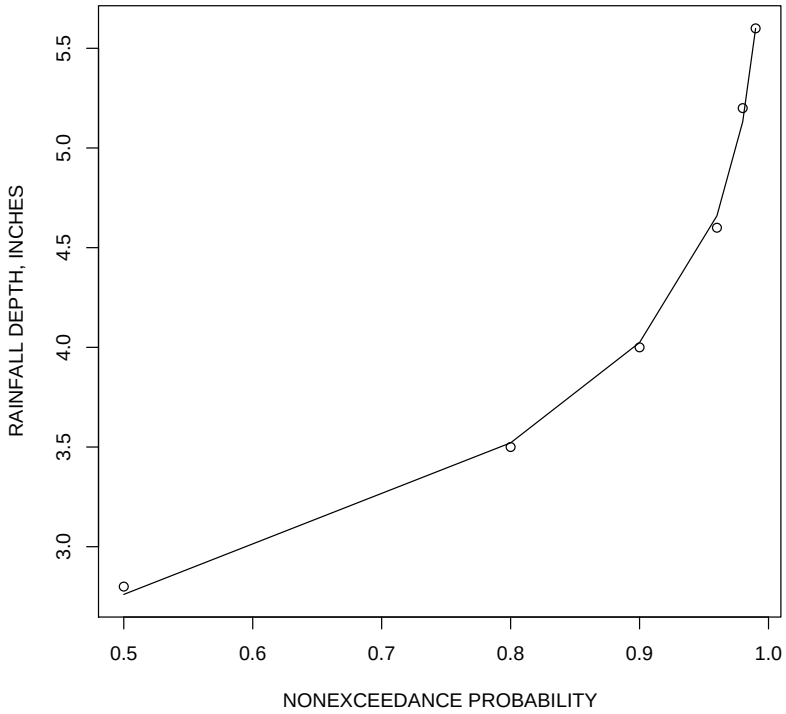

Figure 7.4. Gumbel distribution fit by non-linear optimization to data available from Hershfield (1961) from example 7–21

which were chosen by inspection of the values in `P`. The optimization converges by least squares, and the results (solid line) are shown in figure 7.4. The figure shows remarkably good agreement with the data points (open circles). Thus, example 7–21 shows how a distribution can be fit in a posterior manner to historical data or fit to selected information about the distribution. ◀

As another example with the Gumbel distribution, an alternative method for fitting distributions, which has not previously been described in this dissertation, is used—the **method of percentiles** (Gilchrist, 2000, p. 34). (Karian and Dudewicz (2000) provide an extensive discussion on the method in the context of the Generalized Lambda distribution.) The method estimates the distribution parameters such that specific quantiles are achieved. A different quantile is needed for each parameter in the distribution. For the Gumbel, two quantiles are needed. For demonstration, suppose that the sample $\hat{x}_{0.50} = 8{,}000$ (median) and $\hat{x}_{0.90} = 17{,}000$ (90th percentile) are available and a Gumbel fit to these values is needed. Using eq. (7.42), one can write

$$\hat{x}_{0.50} = \quad 8000 = \xi - \alpha \log[-\log(0.50)] \tag{7.50}$$

$$\hat{x}_{0.90} = 17000 = \xi - \alpha \log[-\log(0.90)] \tag{7.51}$$

and because of two equations and two unknowns, these become

$$\hat{x}_{0.90} - \hat{x}_{0.50} = 9000 = 2.250\alpha - 0.3665\alpha \tag{7.52}$$

and solving for α

$$\alpha = 4777 \tag{7.53}$$

and solving for ξ

$$\xi = 8000 + 4777 \log[-\log(0.50)] = 6249 \tag{7.54}$$

This solution by the method of percentiles is shown in figure 7.5, which was created by example 7–22. The example is unusual here in that F' (exceedance probability) is used instead of F on the horizontal axis. The $\hat{x}_{0.50}$ (on right) and $\hat{x}_{0.90}$ (on left) values are plotted as squares to show that GUM(6249, 4777) passes through the two points as the method of percentiles forced.

7–22

```
PARgum <- vec2par(c(6249,4777), type="gum")
F <- nonexceeds(); QUANTILE <- quagum(F,PARgum)
#pdf("gumMoP.pdf")
plot(1-F, QUANTILE, type="l", xlab="EXCEEDANCE␣PROBABILITY")
points(c(1-0.5,1-0.9), c(8000,17000), pch=15, cex=2)
#dev.off()
```

◀

7.2.6 Reverse Gumbel Distribution

The Reverse Gumbel distribution (Hosking, 1995) is a reflection (see page 36) of the Gumbel distribution. The Reverse Gumbel as implemented here supports right-tail censoring so could also be labeled as a Right-Censored Reverse Gumbel. The QDF of the Reverse Gumbel is

$$x(F)^{\text{revgum}} = -x(1 - F)^{\text{gum}} \tag{7.55}$$

where $x(F)^{\text{gum}}$ is the QDF of the Gumbel.

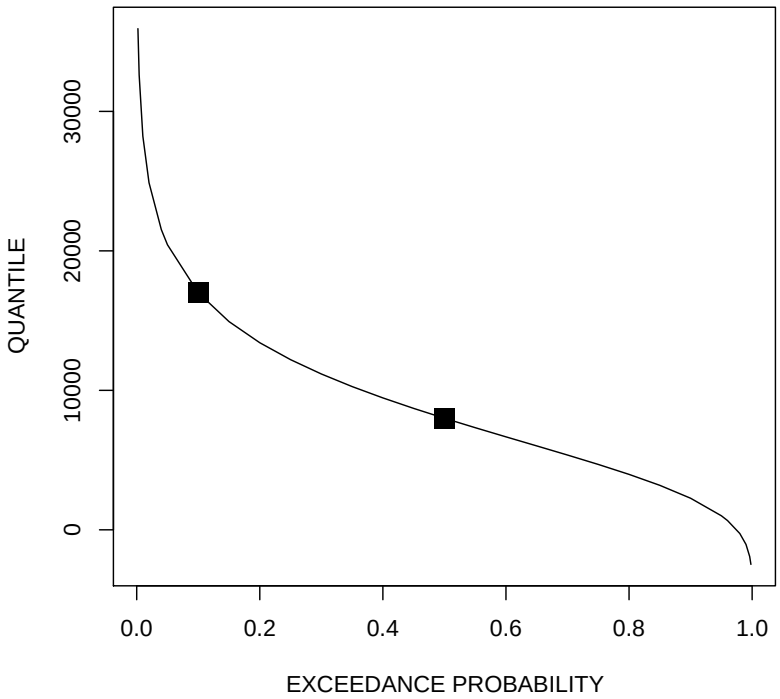

Figure 7.5. Gumbel distribution fit by method of percentiles from example 7–22

The distribution is useful in analysis of right-tail censored (type I and type II censoring, see Section 12.2) data. The distribution is the distribution of a log-transformed two-parameter Weibull distribution, which finds a place in distributional analysis of lifetime or reliability studies. To use the Reverse Gumbel distribution in the *lmomco* package, a right-tail censoring fraction ζ is needed because support for right-tail censoring is available. The censoring fraction could be estimated as the number of observed (noncensored) values m divided by the sample size n or

$$\zeta = m/n \tag{7.56}$$

The ζ parameter is not explicitly another parameter of the distribution in the sense that it only indirectly controls the geometry of the fitted distribution. If $\zeta = 1$, then a Reverse Gumbel distribution is fit without right censoring and the usual L-moments are used. When $\zeta < 1$, then the B-type L-moments, through the B-type probability-weighted moments, are used. Both of these "B-type" moments are described in Section 12.2.

DISTRIBUTION FUNCTIONS

The distribution functions of the Reverse Gumbel having parameters ξ (location), α (scale, $\alpha > 0$), and ζ (right-censoring fraction)

$$f(x) = -\alpha^{-1} \exp(Y)[\exp(\exp(Y))] \tag{7.57}$$

$$F(x) = 1 - \exp[-\exp(Y)] \tag{7.58}$$

$$x(F) = \xi + \alpha \log[-\log(1 - F)] \tag{7.59}$$

where

$$Y = (x - \xi)/\alpha \tag{7.60}$$

The range of the distribution is $-\infty < x < \infty$.
The B-type L-moments of the distribution are

$$\lambda_1^B = \xi - \alpha\rho - \alpha\zeta_1 \qquad \rho \text{ is Euler's constant, } 0.5772\ldots \tag{7.61}$$

$$\lambda_2^B = \alpha[\log(2) + \zeta_2 - \zeta_1] \tag{7.62}$$

where

$$\zeta_1 = \mathrm{Ei}[\ -\log(1 - \zeta)] \tag{7.63}$$

$$\zeta_2 = \mathrm{Ei}[-2\log(1 - \zeta)] \tag{7.64}$$

and

$$\mathrm{Ei}(x) = \int_x^\infty t^{-1} \exp(-t)\, \mathrm{d}t \tag{7.65}$$

is the **exponential integral** as defined by Hosking (1995, p. 558, A.9). Jeffrey (2004, p. 168) provides more details concerning the exponential integral.
The parameters of the distribution are

$$\alpha = \frac{\lambda_2^B}{\log(2) + \zeta_2 - \zeta_1} \tag{7.66}$$

$$\xi = \lambda_1^B + \alpha(\rho + \zeta_1) \tag{7.67}$$

USING R ——————————— ——————————— USING R

The *lmomco* package supports the Gumbel and Reverse Gumbel as separate distributions. Using the Reflection Rule (see page 36), the two distributions are now explored.

Example 7–23 begins the exploration, which will be based on simulated data from a Gumbel parent. A sample size of $n = 1{,}000$ is drawn from a Gumbel distribution, using `rlmomco()`, into variable `X`. The Gumbel has L-moments $\lambda_1 = 400$ and $\lambda_2 = 1200$, and `pargum()` computes the parameters. The Cunnane plotting positions are computed by `pp(X, a=0.40)`.

7–23

```
nsim <- 1000
lmr  <- vec2lmom(c(400,1200))
X    <- rlmomco(nsim, pargum(lmr))
PP   <- pp(X, a=0.40)
```

Continuing in example 7–24, for the random sample `X`, the first five L-moments are computed by `lmoms(X)`, and the L-moments also are computed for the negated sample by `lmoms(-X)`. Finally, four different *lmomco* parameter lists (see page 163 and ex. 7–1) are computed—two lists for the Gumbel and two lists for the Reverse Gumbel.

7–24

```
lmr    <- lmoms(X)  # L-moments of Gumbel
neglmr <- lmoms(-X) # L-moments of -X

PARgumC   <- pargum(lmr)         # Parameters of Gumbel
PARgumD   <- pargum(neglmr)      # Parameters of Gumbel

PARrevgumC <- parrevgum(lmr)     # Parameters of rev Gumbel
PARrevgumD <- parrevgum(neglmr) # Parameters of rev Gumbel
```

A comparison of five different Gumbel-like distributions plotted to the L-moments of example 7–24 is shown in figure 7.6, which was produced by example 7–25. Either by the constraints of grey-scale-only printing or the general complexity of the discussion of the example difficult. Therefore, step-by-step discussion is provided as code comments. It might be helpful for readers to run example 7–25 by plotting one curve at a time.

7–25

```
#pdf("revgum.pdf")
plot(PP, sort(X), type="n",
     xlab="NONEXCEEDANCE␣PROBABILITY",
     ylab="QUANTILE", ylim=c(-5000,15000))
lines(PP, quagum(PP,PARgumC), col=2, lwd=5) # red and thick
# Curve mimics the parent if nsim is large enough

lines(PP, quagum(PP,PARgumD))
# Thin black curve is Gumbel fit to negated values. The mean
# is reduced, L-scale is not. Curve plots under previous. Both
```

```
# distributions have the same L-skew.

lines(PP, -1*quagum((1-PP),PARgumD), col=2, lty=4, lwd=5)
# Curve is a manually reversed Gumbel fit to L-moments of
# negated values. Curve is thick, red, and dashed.

lines(PP, quarevgum(PP,PARrevgumC), col=4, lwd=2)
# Curve (solid blue) is a reversed Gumbel fit to L-moments of X
# and over plots previous curve. So manual reversal fit to -X
# is the same as using the reversed Gumbel.

lines(PP, -1*quarevgum((1-PP),PARrevgumD), col=5, lwd=2)
# Curve (light blue) is a manually reversed Gumbel fit to
# L-moments of negated values (overplots the thick red curve
# curve, first curve drawn). So reversing a reversed Gumbel fit
# to -X recovers Gumbel fit to X.

legend(0,10000,
       c("Gumbel␣fit␣to␣C␣PWMs",
         "Gumbel␣fit␣to␣D␣PWMs",
         "Hand-reversed␣Gumbel␣fit␣to␣D␣PWMs",
         "Reverse␣Gumbel␣fit␣to␣Type␣C␣PWMs",
         "Reverse␣Gumbel␣fit␣to␣Type␣D␣PWMs"),
       lwd=c(5,1,5,2,2), lty=c(1,1,4,1,1), col=c(2,1,2,4,5))
#dev.off()
```

To complete the discussion of the comparison of Gumbel-like distributions started in example 7–23, example 7–26 computes the theoretical L-moments for each of the four parameter lists. The results are not shown in example 7–26 but are listed in table 7.9.

7–26

```
theoLmoms(PARgumC);     theoLmoms(PARgumD)
theoLmoms(PARrevgumC); theoLmoms(PARrevgumD)
```

Table 7.9. Comparison of computed L-moments for four Gumbel distribution parameter lists from example 7–26

Function `theoLmoms()`	λ_1	λ_2	τ_3	τ_4	τ_5
`PARgumC`	502.5	1264	0.1699	0.1504	0.0559
`PARgumD`	−502.5	1264	.1699	.1504	.0559
`PARrevgumC`	502.5	1264	−.1699	.1504	−.0559
`PARrevgumD`	−502.5	1264	−.1699	.1504	−.0559

◀

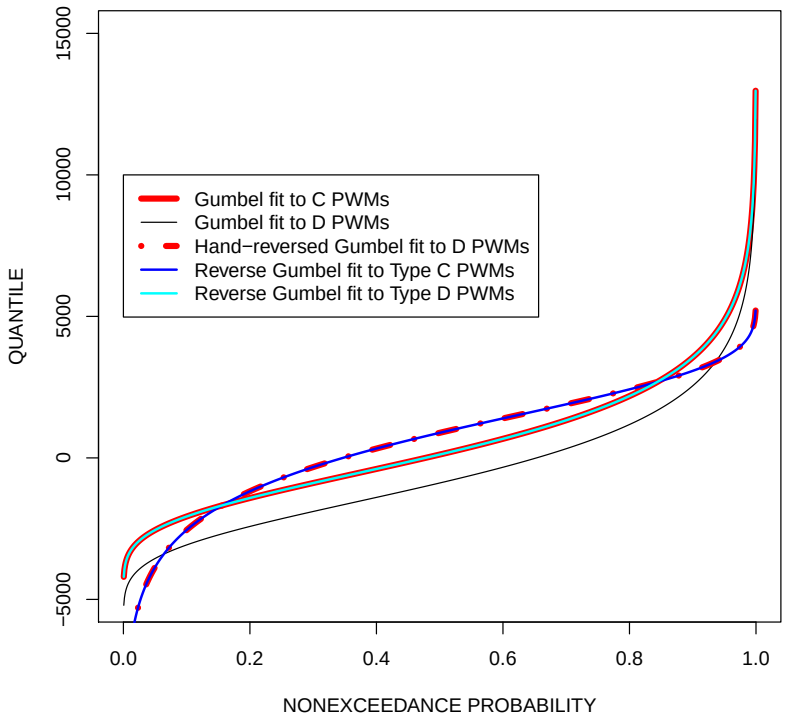

Figure 7.6. Comparison five Gumbel-like distributions as sequentially described in example 7–25

For another example of the Reverse Gumbel, Hosking (1995, p. 558) reports that the Reverse Gumbel is "the distribution of $\log X$ when X has a two-parameter Weibull distribution." The two-parameter Weibull is a three-parameter Weibull with a lower bounds of zero. This statement is evaluated using computational tools provided by R and the *lmomco* package in example 7–27.

7–27

```
nsam <- 40; nsim <- 10
for(i in 1:nsim) { # 1:nsim, is same as seq(1,nsim)
   x <- sort(rweibull(nsam, shape=1.5, scale=10))
   PP <- pp(x)
   lmr <- lmoms(x); lmrlg <- lmoms(log(x))
   PARrevgum <- parrevgum(lmr); PARrevgumlg <- parrevgum(lmrlg)
   plot(PP,x, ylim=c(0,40))
   lines(PP,quarevgum(PP,PARrevgum)) # thin line
   lines(PP,exp(quarevgum(PP,PARrevgumlg)), lwd=3) # thick line
   legend(0,30,c("Reverse␣Gumbel␣fit␣to␣X",
                 "Reverse␣Gumbel␣fit␣to␣ln(X)"),
                lwd=c(1,4))
   Sys.sleep(2)
}
```

The built-in Weibull distribution of R is a two-parameter version, and random variates are generated by `rweibull()`. The example proceeds by simulating $n = 40$ samples 10 times. The L-moments are computed for `x` and `log(x)` (natural logarithm) of the simulated and sorted Weibull sample. The Reverse Gumbel distribution is fit to the L-moments by the two `parrevgum()` calls. Although the plots are not shown here, a plot of the sample and the two fitted distributions is generated and the process repeated `nsim` times. The `Sys.sleep()` function causes the process to suspend for about 2 seconds before repeating so that the user can watch the results in a poor sort of animation. Alternatively, the user could bound the code with a `pdf()` function at at beginning and `dev.off()` and the end and then page through the resulting portable document format (PDF) file. ◀

7.2.7 Kumaraswamy Distribution

The Kumaraswamy distribution, which is named by Jones (2009), but was introduced *in the hydrologic literature* by Kumaraswamy (1980), is a relatively simple distribution which has doubly-bounded support on the interval $[0, 1]$. Jones (2009) provides the first extensive evaluation of the Kumaraswamy distribution and considers the L-moments of the distribution. The distribution mimics the Beta distribution but has explicit distributions functions and reliance on special beta functions is not necessary. However, Jones (2009, p. 70) states that the Beta distribution "[continues to] provide the premier family of continuous distributions on bounded support."

DISTRIBUTION FUNCTIONS

The distribution functions of the Kumaraswamy having parameters α (scale, $\alpha > 0$) and β (scale, $\beta > 0$) are

$$f(x) = \alpha\beta x^{\alpha-1}(1 - x^{\alpha})^{\beta-1} \tag{7.68}$$

$$F(x) = 1 - (1 - x^{\alpha})^{\beta} \tag{7.69}$$

$$x(F) = [1 - (1 - F)^{1/\beta}]^{1/\alpha} \tag{7.70}$$

The range of the distribution is $0 \leq x \leq 1$.
The L-moments with $\eta = 1 + 1/\alpha$ are

$$\lambda_1 = \beta \mathrm{B}(\eta, \beta) \tag{7.71}$$

$$\lambda_2 = \beta[\mathrm{B}(\eta, \beta) - 2\mathrm{B}(\eta, 2\beta)] \tag{7.72}$$

$$\tau_3 = \frac{\mathrm{B}(\eta, \beta) - 6\mathrm{B}(\eta, 2\beta) + 6\mathrm{B}(\eta, 3\beta)}{\mathrm{B}(\eta, \beta) - 2\mathrm{B}(\eta, 2\beta)} \tag{7.73}$$

$$\tau_4 = \frac{\mathrm{B}(\eta, \beta) - 12\mathrm{B}(\eta, 2\beta) + 30\mathrm{B}(\eta, 3\beta) - 40\mathrm{B}(\eta, 4\beta)}{\mathrm{B}(\eta, \beta) - 2\mathrm{B}(\eta, 2\beta)} \tag{7.74}$$

$$\tau_5 = \frac{\mathrm{B}(\eta, \beta) - 20\mathrm{B}(\eta, 2\beta) + 90\mathrm{B}(\eta, 3\beta) - 140\mathrm{B}(\eta, 4\beta) + 70\mathrm{B}(\eta, 5\beta)}{\mathrm{B}(\eta, \beta) - 2\mathrm{B}(\eta, 2\beta)} \tag{7.75}$$

where $\mathrm{B}(a, b)$ is the beta function that is shown in eq. (3.10). Readers are encouraged to compare this system of equations[5] for the L-moments to those for first five L-moments in terms of probability-weighted moments on page 121. The parameters can be solved numerically by minimizing the combined Pythagorean distance between the combined square errors $(\tau_3 - \hat{\tau}_3)^2$ and $(\tau_4 - \hat{\tau}_4)^2$. This technique is implemented for the `parkur()` function, which uses the `optim()` function of R for minimization.

The mode ($\alpha > 1, \beta > 1$) and antimode ($\alpha < 1, \beta < 1$) are

$$\mathrm{Mode/Antimode}^{\mathrm{kur}} = \left(\frac{\alpha - 1}{\alpha\beta - 1} \right)^{1/\alpha} \tag{7.76}$$

and finally, the Kumaraswamy distribution for $\alpha = \beta = 1$ becomes the Uniform distribution.

USING R

An example conversion of L-moments ($\lambda_1 = 0.7$, $\lambda_2 = 0.2$) to Kumaraswamy parameters and back again with the `lmorph()` function being used to shorten the output is shown in example 7–28.

7–28

```
lmorph(lmomkur(parkur(vec2lmom(c(0.7, 0.2)))))$lambdas[1:2]
[1] 0.7 0.2
```

The purpose of the example is to demonstrate how these four functions of *lmomco* can be chained together and more importantly how the numerical methods of the `parkur()` function can be tested because the values $\lambda_1 = 0.7$ and $\lambda_2 = 0.2$ are recovered. ◀

[5] The author derived the relation for τ_5 for this dissertation; whereas, Jones (2009) is the source for the others. However, the derivation is not too difficult given the established pattern (see eq. (6.37)).

For the Kumaraswamy, Jones (2009, figs. 2 and 4) provides contour plots of τ_3 and τ_4 based on natural logarithm values for α and β. Example 7–29 reproduces these two plots using the `contourplot()` of the *lattice* package. The two plots are respectively shown in figure 7.7.

7–29

```
A <- B <- exp(seq(-3,5, by=.05))
logA <- logB <- T3 <- T4 <- c(); i <- 0
for(a in A) {
  for(b in B) {
    i <- i + 1
    parkur  <- list(para=c(a,b), type="kur");
    lmr     <- lmomkur(parkur)
    logA[i] <- log(a)
    logB[i] <- log(b)
    T3[i]   <- lmr$TAU3
    T4[i]   <- lmr$TAU4
  }
}
library(lattice) # to acquire the contourplot function
#pdf("kurT3.pdf")
contourplot(T3~logA+logB, cuts=20, lwd=0.5, label.style="align",
  xlab="LOG␣OF␣ALPHA", ylab="LOG␣OF␣BETA",
  xlim=c(-3,5), ylim=c(-3,5),
  main="L-SKEW␣FOR␣KUMARASWAMY␣DISTRIBUTION")
#dev.off()
#pdf("kurT4.pdf")
contourplot(T4~logA+logB, cuts=10, lwd=0.5, label.style="align",
  xlab="LOG␣OF␣ALPHA", ylab="LOG␣OF␣BETA",
  xlim=c(-3,5), ylim=c(-3,5),
  main="L-KURTOSIS␣FOR␣KUMARASWAMY␣DISTRIBUTION")
#dev.off()
```

Inspection of figure 7.7 for $\alpha = \beta = 1$ (natural logarithms result in zero on each axis) shows $\tau_3 = 0$ and $\tau_4 = 0$, which is consistent with the Uniform distribution, which is symmetrical and has no peak. ◀

7.2.8 Rayleigh Distribution

The Rayleigh distribution has applications in modeling the lifetimes of rapidly aging subjects (Rizzo, 2008, p. 249) and as a "consequence of the central limit theorem, when the scattering medium contains a large number of randomly distributed [scattering angles]

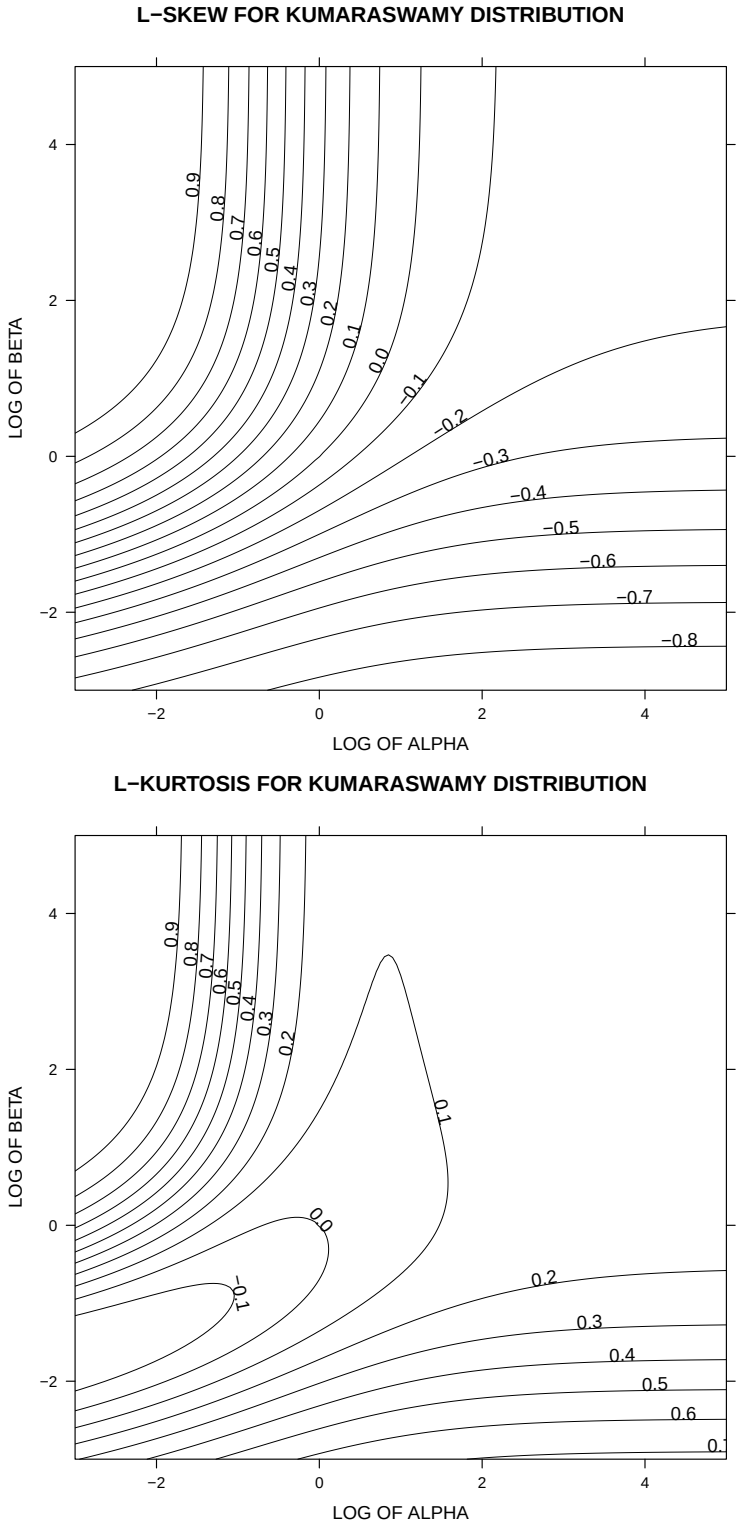

Figure 7.7. Relation between Kumaraswamy distribution parameters and L-skew and L-kurtosis from example 7−29

uniformly distributed between 0 and 2π" (Raju and Srinivasan, 2002, p. 872). Hosking (1986, p. 65) provides the L-moments of the Rayleigh. Like the other two-parameter distributions of this chapter, the Rayleigh is not fit to the skewness or higher measures of the shape of the data.

DISTRIBUTION FUNCTIONS

The distribution functions of the Rayleigh having parameters ξ (location) and α (scale, $\alpha > 0$) are

$$f(x) = \frac{(x-\xi)\exp(Y)}{\alpha^2} \tag{7.77}$$

$$F(x) = 1 - \exp(Y) \tag{7.78}$$

$$x(F) = \xi + \sqrt{-2\alpha^2 \log(1-F)} \tag{7.79}$$

where

$$Y = \frac{-(x-\xi)^2}{2\alpha^2} \tag{7.80}$$

The range of the distribution is $0 < x < \infty$.
The L-moments are

$$\lambda_1 = \xi + \alpha\sqrt{\pi/2} \tag{7.81}$$

$$\lambda_2 = \tfrac{1}{2}\alpha(\sqrt{2}-1)\sqrt{\pi} \tag{7.82}$$

and the L-moment ratios are

$$\tau_2 = 1 - \sqrt{1/2} \approx 0.2929 \text{ for } \xi = 0 \tag{7.83}$$

$$\tau_3 = \frac{1 - 3/\sqrt{2} + 2/\sqrt{3}}{1 - 1/\sqrt{2}} \approx 0.1140 \tag{7.84}$$

$$\tau_4 = \frac{1 - 6/\sqrt{2} + 10/\sqrt{3} - 5\sqrt{4}}{1 - 1/\sqrt{2}} \approx 0.1054 \tag{7.85}$$

The α parameter for a known ξ is

$$\alpha = \frac{\lambda_1 - \xi}{\sqrt{\pi/2}} \tag{7.86}$$

and the parameters for an unknown ξ are

$$\alpha = \frac{2\lambda_2}{\sqrt{\pi}(\sqrt{2}-1)} \tag{7.87}$$

$$\xi = \lambda_1 - \alpha\sqrt{\pi/2} \tag{7.88}$$

The mode of the distribution is

$$\text{Mode}^{\text{ray}} = \alpha \tag{7.89}$$

which, like discussed for the Gamma distribution, can be used for parameter estimation if the mode of the distribution is known or otherwise needs to be locked-in at a given position. This application is of interest in use of the Rayleigh distribution in streamflow hydrograph modeling in which the peak streamflow corresponds to the mode of the distribution.

USING R

Using L-moments from example 7–14, Rayleigh distributions as one- and two-parameter versions are fit in example 7–30. The PDFs of the two distributions are shown in figure 7.8. The figure shows that the general shape of the two distributions are similar, but that the location and shape vary when ξ is specified.

7–30

```
lmr <- vec2lmom(c(0.964,0.581,0.452), lscale=FALSE)
PARrayA <- parray(lmr); PARrayB <- parray(lmr, xi=0)
#pdf("rayleighA.pdf")
layout( matrix(1:2, byrow=TRUE) )
check.pdf(pdfray,PARrayA, plot=TRUE)
check.pdf(pdfray,PARrayB, plot=TRUE)
#dev.off()
```

◀

For a final example, suppose that streamflow hydrograph models having one unit of depth of runoff from a 36 square-mile watershed need to be created. The Gamma and Rayleigh distributions are chosen because each can attain shapes similar to observed streamflow hydrographs. Generally, streamflow hydrographs have a steep rising tail and drawn-out receding tail. The two distributions are readily fit to the time of peak or the mode of the distribution. Suppose also that this watershed generally peaks in about 5 hours from the inception of rainfall. The two distributions are fit and plotted next.

For a $\xi = 0$ Rayleigh distribution, the parameter $\alpha^{\text{ray}} = \text{Mode}$, so $\alpha^{\text{ray}} = 5$ for the problem at hand. The parameters of the Gamma now require estimation. Observing that

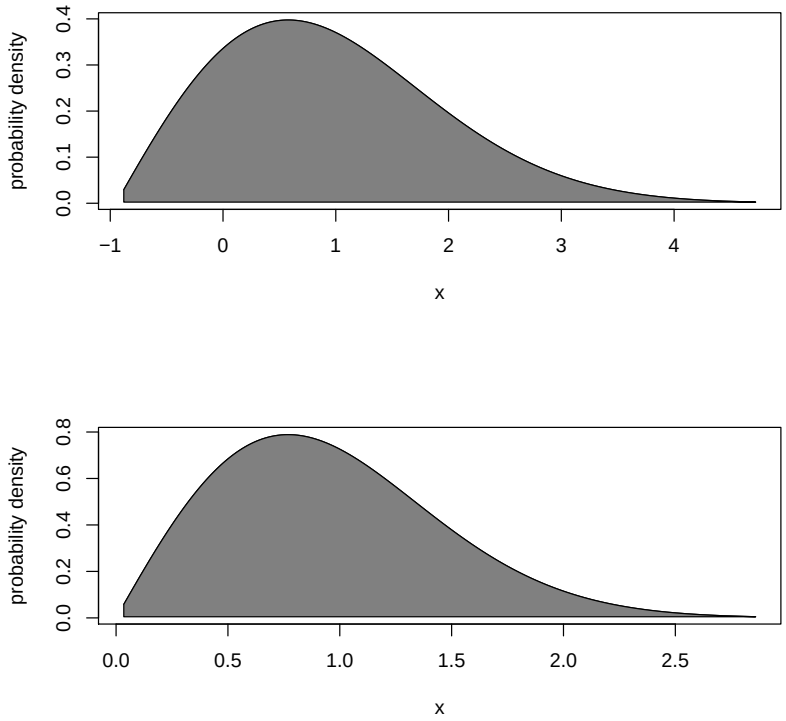

Figure 7.8. Comparison two Rayleigh distributions fit as one- or two-parameter versions to L-moments of $\lambda_1 = 0.964$ and $\lambda_2 = 0.581$ for unknown (top) and known ($\xi = 0$, bottom) lower bounds from example 7–30

the parameters of the Gamma in terms of the mode and mean λ_1 are

$$\alpha^{\text{gam}} = \lambda_1/(\lambda_1 - \text{Mode}) \tag{7.90}$$

$$\beta^{\text{gam}} = \lambda_1/\text{Mode} \tag{7.91}$$

all that is required therefore to fit the Gamma is the λ_1 of the Rayleigh fit that is computed by the `lmomray()` function using the parameters of the Rayleigh. These computations are made in example 7–31. The PDFs of the two distributions are shown in figure 7.9.

7–31

```
themode  <- 5 # the peak or mode is at 5 hours
PARray   <- vec2par(c(0,themode), type="ray") # fit the Rayleigh
LMRray   <- lmomray(PARray)
     mu  <- LMRray$L1 # extract the mean

# Now fit the Gamma distribution
GAMalpha <- mu/(mu - themode)
GAMbeta  <- mu/GAMalpha
```

```
PARgam   <- vec2par(c(GAMalpha,GAMbeta), type="gam")

F <- seq(0.0001,0.999, by=0.001) # nonexceedance probabilities
x.ray <- quaray(F,PARray)
x.gam <- quagam(F,PARgam)
x <- sort(c(x.ray, x.gam)) # combine x's into a single vector
y <- c(pdfray(x,PARray), pdfgam(x,PARgam)) # for plotting limits

#pdf("rayleighB.pdf")
plot( x, pdfray(x,PARray), type="l", ylim=c(min(y),max(y)),
      xlab="TIME, IN HOURS",
      ylab="UNITS OF WATERSHED DEPTH PER HOUR") # Rayleigh dist.
lines(x, pdfgam(x,PARgam), lty=2) # Gamma distribution (dashed)
#dev.off()
```

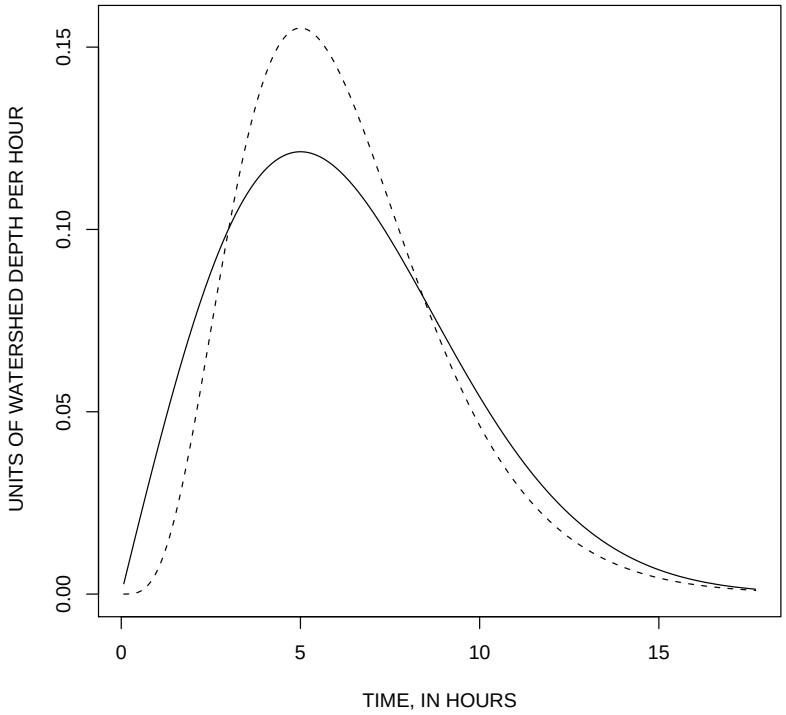

Figure 7.9. Comparison two Rayleigh distributions (solid line) and Gamma distribution (dashed line) fit to a time to peak (mode) of 5 hours from example 7–31

Although both streamflow hydrographs have unit volume (unit area under a PDF), the figure shows that Gamma has a larger peak or mode (about 50 percent more) than the Rayleigh. This difference comes by a contraction in general width of the Gamma hydrograph relative to the Rayleigh. The rising limbs of the hydrographs ($< \text{Mode}$) have dif-

ferent first derivative behavior. The Gamma has a rising limb inflection point and the Rayleigh does not. ◀

7.2.9 Rice Distribution

The Rice distribution (Sijbers and others, 1998; Raju and Srinivasan, 2002) has applications in modeling many types of backscatter, communications, and medical imaging applications. Like the other two-parameter distributions of this chapter, the Rice is not fit to the skewness or higher measures of shape of the data. The random variable R is RICE(ν,α) if $R = \sqrt{X^2 + Y^2}$ where the random variable X is NOR($\nu \cos\theta_X$, α^2) and Y is NOR($\nu \sin\theta_Y$, α^2) for random variable θ on the interval $[0, 2\pi]$. Readers are asked to note the application of two random variables of the θ. The author used numerical examples to confirm that two are needed as the literature is ambiguous. The parameter ν can be thought of as a signal and α as a noise, and the ratio between these concepts represents a form of signal-to-noise ratio or SNR. Sijbers and others (1998, p. 358) use a definition of $\mathrm{SNR} = \nu/\alpha$, whereas, Raju and Srinivasan (2002, p. 872) use a definition of $\mathrm{SNR} = \mu/\sigma$ where μ is the mean and σ is the standard deviation. For this dissertation and in the *lmomco* package, SNR is only considered as the ratio ν/α. The Rice distribution also is associated with a phenomena known as "Rician fading."

Rician fading[6] is

> A stochastic model for radio propagation anomaly caused by partial cancellation of a radio signal by itself—the signal arrives at the receiver by two different paths, and at least one of the paths is changing (lengthening or shortening). Rician fading occurs when one of the paths, typically a line of sight signal, is much stronger than the others. In Rician fading, the amplitude gain is characterized by a Rician [Rice] distribution.

DISTRIBUTION FUNCTIONS

The distribution functions of the Rice having parameters ν (location, $\nu \geq 0$) and α (scale, $\alpha > 0$) are

[6] This paragraph is verbatim from `http://en.wikipedia.org/wiki/Rician_fading`.

$$f(x) = \frac{x}{\alpha^2}\exp\left(-\frac{x^2+\nu^2}{2\alpha^2}\right) I_0\left(\frac{x\nu}{\alpha^2}\right) \tag{7.92}$$

$$F(x) = 1 - Q_1\left(\frac{\nu}{\alpha}, \frac{x}{\alpha}\right) \tag{7.93}$$

$x(F)$ has no explicit analytical form

If $\nu = 0$, the Rayleigh distribution with $\xi = 0$ and $\alpha = \alpha$ results.
The range of the distribution is $0 < x < \infty$.
A useful definition for the Rice distribution is the signal-to-noise ratio SNR

$$\mathrm{SNR} = \nu/\alpha = \zeta \tag{7.94}$$

For the PDF definition, where the function $I_0(z)$ is the modified **Bessel function** of the first kind for a real number z, which is defined in integral form as

$$I_v(z) = \frac{1}{\pi}\int_0^\pi \exp(z\cos\Theta)\cos(v\Theta)\,\mathrm{d}\Theta \tag{7.95}$$

and in series form as

$$I_v(z) = (z/2)^v \sum_{k=0}^{\infty} \frac{(z^2/4)^k}{\Gamma(k+1)\Gamma(v+k+1)} \tag{7.96}$$

Hence, $I_0(z)$ is

$$I_0(z) = \frac{1}{\pi}\int_0^\pi \exp(z\cos\Theta)\cos(\Theta)\,\mathrm{d}\Theta \tag{7.97}$$

or

$$I_0(z) = \sum_{k=0}^{\infty} \frac{(z^2/4)^k}{\Gamma(k+1)^2} \tag{7.98}$$

The Bessel function is implemented in R by the `besselI()` function.

For the CDF definition, Q_1 is the **Marcum Q function**.[7] The Marcum Q function is defined in integral form by

$$Q_M(a,b) = \frac{1}{a^{M-1}}\int_b^\infty x^M \exp[-(x^2+a^2)/2]\; I_{M-1}(ax)\,\mathrm{d}x \tag{7.99}$$

and in series form by

[7] Want an interesting tour through the mathematics of the signal processing and radar detection field? Google "Marcum Q function."

$$Q_M(a,b) = \exp[-(a^2+b^2)/2] \sum_{k=1-M}^{\infty} \left(\frac{a}{b}\right)^k I_k(ab) \tag{7.100}$$

where $I_k(z)$ is the Bessel function. For the Rice distribution, $M = 1$, which results in

$$Q_1(a,b) = \exp[-(a^2+b^2)/2] \sum_{k=0}^{\infty} \left(\frac{a}{b}\right)^k I_k(ab) \tag{7.101}$$

The product moments (mean and variance) of the Rice distribution require the **Laguerre polynomial** given by

$$L_{1/2}(z) = \exp(z/2) \times [(1-z)I_0(-z/2) - zI_1(-z/2)] \tag{7.102}$$

where $L_{1/2}(0) = 1$ and $\sqrt{\pi/2} \times L_{1/2}(-z^2/2) \to z$ as z becomes large. Using the Laguerre polynomial, the Rician mean is

$$\mu = \alpha\sqrt{\pi/2} \times L_{1/2}(-\tfrac{1}{2}(\nu/\alpha)^2) \tag{7.103}$$

and the Rician variance is

$$\sigma^2 = 2\alpha^2 + \nu^2 - \alpha^2(\pi/2)L_{1/2}^2(-\tfrac{1}{2}(\nu/\alpha)^2) \tag{7.104}$$

These two equations clearly are complex. For instance, two Bessel functions are involved but notice the square of the ν/α term or SNR that occurs in both the definitions of mean and variance. A key to working with the Rice distribution is understanding of the influence of ν/α on the moments.

The L-moments of the Rice distribution are difficult to express. However, the recognition that τ_2 can be interpreted as a signal-to-noise ratio provides a key—τ_2 should be proportional to SNR. The relations between τ_2 and functions based on the SNR can provide for parameter estimation by the method of L-moments. It was discovered by thought and numerical experimentation for this dissertation that

$$\nu/\alpha = \mathrm{SNR} = \zeta = \mathcal{F}(\tau_2) \tag{7.105}$$

where $\mathcal{F}(\tau_2)$ is an unknown function that is uniquely a function of τ_2. If the quantity ζ_{L05} is defined as

$$\zeta_{\mathrm{L05}} = \sqrt{\pi/2} \times L_{1/2}(-\tfrac{1}{2}\zeta^2) \tag{7.106}$$

it was discovered again by thought and numerical experimentation for this dissertation that

$$\zeta_{\text{L05}} = \mathcal{G}(\tau_2) \tag{7.107}$$

where $\mathcal{G}(\tau_2)$ is an unknown function, which also is uniquely a function of τ_2. The parameter α can be estimated using μ from eq. (7.103) by

$$\alpha = \mu \times \mathcal{G}(\tau_2) \tag{7.108}$$

and subsequently ν can be estimated by

$$\nu = \alpha \times \mathcal{F}(\tau_2) \tag{7.109}$$

Thus, two functional relations, empirical approximations, or simply lookup tables of $\mathcal{F}(\tau_2)$ and $\mathcal{G}(\tau_2)$ are needed. The direct application of linear interpolation through lookup tables is used by the *lmomco* package.

USING R ——— ——— USING R

The Marcum Q function $Q_1(a, b)$ function can be computed by functions defined in examples 7–32 and 7–33

7–32

```
"MarcumQ1" <- function(a, b, terms=10) {
   if(length(a) > 1) stop("a must be scalar")
   if(length(b) > 1) stop("b must be scalar")
   ab <- a*b;  a.over.b <- a/b
   A  <- exp(-(a^2 + b^2)/2)
   vals <- sapply(0:terms,
       function(v) { return(a.over.b^v * besselI(ab, n=v)) })
   return( A * sum(vals) )
}
```

7–33

```
"MarcumQ1sincos" <- function(a,b) {
   if(a < 0 | b < 0 | b <= a) stop("B > A >= 0 is not true")
   eta <- a/b; eta2 <- 2*eta; etasq <- eta^2
   fn <- function(t) {
      sint <- sin(t)
      tmp  <- 1 + eta2*sint + etasq
      K <- (1 + eta*sint) / tmp; L <- -(b^2/2)*tmp
      return(K*exp(L))
   }
```

```
   val <- integrate(fn, -pi, pi)$value
   return( 1/(2*pi) * val )
}

# Test the Marcum Q1 function
MarcumQ1(2,3)
[1] 0.2143619
MarcumQ1sincos(2,3)
[1] 0.2143621
```

where example 7–33 shows equivalence. ◀

The `cdfrice()` function in the *lmomco* package uses the `integrate()` function of R on `pdfrice()` instead of the Marcum Q function to mitigate for potential complexities in using the mathematics of examples 7–32 and 7–33. However for illustration, example 7–34 provides for the Rice CDF using the $Q_1(a, b)$ function of example 7–32.

7–34

```
"cdfrice.by.MarcumQ1" <- function(x, para=NULL, ...) {
    A <- para$para[1]; v <- para$para[2]
    a <- v/A
    f <- vector(mode="numeric")
    for(i in 1:length(x)) {
        if(x[i] < 0) {
           f[i] <- 0
        } else if(x[i] == Inf) {
           f[i] <- 1
        } else {
           b <- x[i]/A
           Q1 <- MarcumQ1(a,b, ...)
           f[i] <- 1 - Q1
        }
    }
    return(f)
}
```

Finally, example 7–35 creates in figure 7.10 a graphical check on equivalency between the integration of the `pdfrice()` and the CDF definition using the Marcum Q function.

7–35

```
PARrice <- vec2par(c(20,40), type="rice")
dF <- 0.1; x <- seq(0,100, by=dF)
testpdf <- pdfrice(x, para=PARrice)
sum(testpdf)*dF # following sum should be quite close to unity
[1] 0.9977918
```

```
#pdf("riceA.pdf")
layout(matrix(1:2, byrow=TRUE))
plot(x, testpdf, type="l", ylab="PROBABILITY␣DENSITY") # top plot
plot(x, cdfrice.by.MarcumQ1(x, para=PARrice),
     lwd=4, lty=3, type="l", ylab="CUMULATIVE␣PROBABILITY") #
        dots
lines(x,cdfrice(x, para=PARrice)) # line on bottom plot
#dev.off()
```

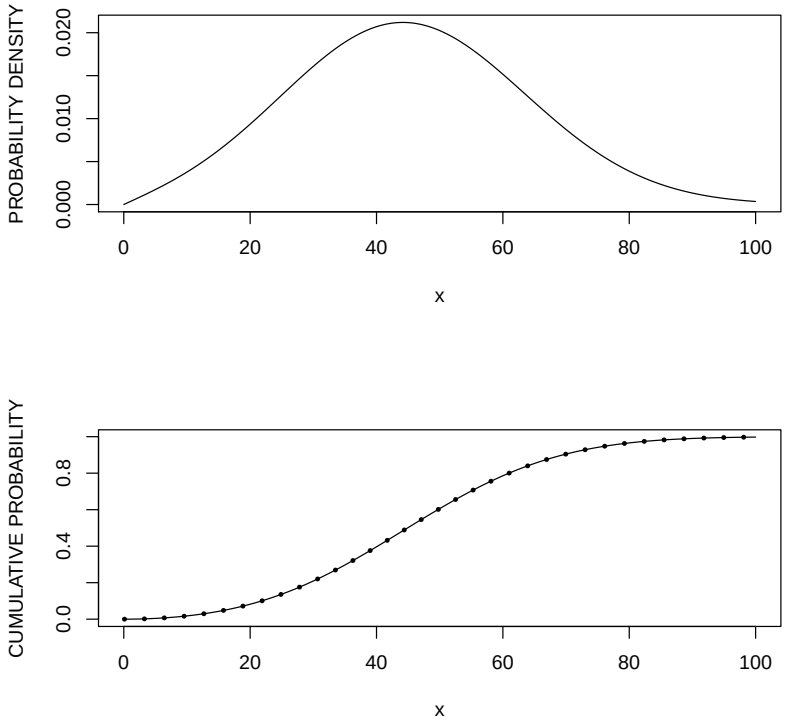

Figure 7.10. Example PDF and two computations of CDF of a RICE$(20, 40)$ distribution from example 7–35

◀

The Laguerre polynomial of eq. (7.102) is implemented by the `LaguerreHalf()` function of *lmomco*. This function is used in example 7–36. The example shows similar results for computed and simulated (999 simulations) of RICE$(20, 40)$ random variables.

7–36

```
PARrice <- vec2par(c(20,40), type="rice")
mu  <- 40*sqrt(pi/2)*LaguerreHalf(-(20/40)^2/2)
smu <- mean(quarice(runif(999), PARrice))
cat(c("#␣MEAN=",              round(mu,  digits=3),
```

```
      "␣and␣SIMULATED␣MEAN=", round(smu, digits=3),"\n"))
# MEAN= 53.218  and SIMULATED MEAN= 53.439
```

◀

The final example in 7–37 for the Rice distribution provides an extensive comparison of the shapes of the CDF and uses the several of the Rican functions of *lmomco* including `lmomrice()`, `cdfrice()`, and `quarice()` for a range of SNR from near that of the Rayleigh distribution ($\mathrm{SNR} \ll 1$) to the Normal distribution ($\mathrm{SNR} \gg 1$). The example creates figure 7.11. Detailed discussion of the figure is required.

7–37

```
nu  <- 17 # the signal
SNR <- c(0.07, 0.2, 0.3, 0.5, 1, 2, 5, 10, 22, 25, 40, 60)
WH  <- c(   4,   1,   1,   1, 1, 1, 1,  1,  1,  2,  2,  3)
ymax <- 0.9999; ymin <- 0.0001
ylim <- qnorm(c(ymin,ymax))
#pdf("ricecdfcomp.pdf")
plot(c(10,100), ylim, ylim=ylim, type="n", log="x",
     xlab="QUANTILE", ylab="STANDARD␣NORMAL␣VARIATE")
for(i in 1:length(SNR)) {
   snr  <- SNR[i]; wh <- WH[i]
   rice <- vec2par(c(nu,nu/snr), type="rice")
   ifelse(snr > 1, size <- 2, size <- 0.5)
    lmr <- lmomrice(rice, nmom=2) # 2 moments for speed
     mu <- lmr$lambdas[1]
      x <- quarice(c(ymin,nonexceeds(), ymax),rice)
   lines(x, qnorm(cdfrice(x,rice)), lty=wh, lwd=size)
   fmean <- cdfrice(mu,rice); xmed <- quarice(0.5,rice)
   points(c(mu,    mu), qnorm(c(fmean,fmean)), cex=2 )
   points(c(xmed,xmed), qnorm(c(0.5,    0.5)), pch=16)
}
legend(30,qnorm(0.05), bty="n", pt.cex=c(NA, NA, NA, NA, 2,  1),
       c("Rayleigh␣distribution",
         "Rice␣distribution", "Rice␣via␣normal␣with␣Laguerre",
         "Rice␣via␣normal␣without␣Laguerre", "Mean", "Median"),
       lty=c( 4, 1,  2,  3, 0,  0), pch=c(NA, NA, NA, NA, 1, 16))
#dev.off()
```

The Rice distributions in figure 7.11, albeit none all are fully drawn because of axis limits, are extremely varied. The minimum SNR is so low that Rice $\rightarrow$ Rayleigh and thus $\tau_2 \approx 0.2929$; as a result, the Rayleigh distribution is drawn on the far right (the thin dash-dot line). The solid lines represent true Rice distributions; whereas the two dashed lines that also the nearly vertical represent Rice $\rightarrow$ Normal; as a result, Normal distribution fits using the Rician mean and variance in eqs. (7.103) and (7.104) using the Laguerre

polynomial. The single dotted and near vertical lines represent a Normal distribution with the limiting mean ($\mu = \nu$) and variance ($\sigma^2 = \alpha^2$) of the Rice distribution for very high SNR. Finally, the distribution lines double in thickness when the $\mathrm{SNR} \geq 1$. The mean and median of the distributions are indicated by open and filled circular symbols, respectively.

Readers are encouraged to repeat example 7–37 with more or less expansive horizontal and vertical axis limits to see the real breadth of "Rician fits" for an extreme range of SNR. Readers also are encouraged to change the vertical axis transformation from `qnorm()` to just linear and repeat the example.

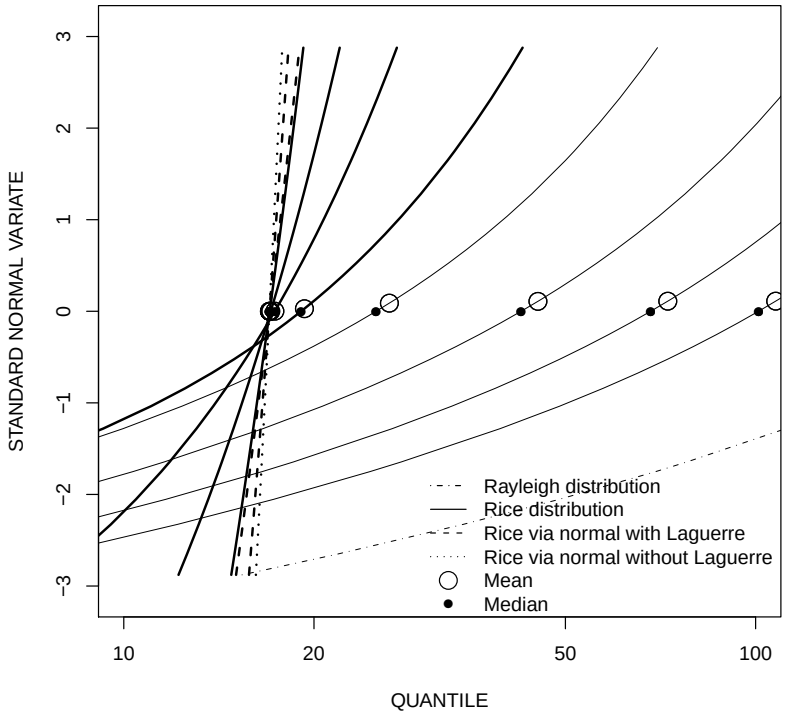

Figure 7.11. Comparison of CDF for signal $\nu = 17$ for a range of signal-to-noise (SNR) ratios for Rice distribution from example 7–37. The thick lines represent SNR greater than unity. Increasing SNR shifts the curves from right to left, and the curves become near vertical near $\nu = 17$.

The Rice distribution function of *lmomco* use the Rayleigh and Normal distributions as limiting conditions for hard-wired thresholds for SNR that have been determined by the author using numerical experiments to trap under and overflows. The documentation of *lmomco* provides these threshold values. ◀

The Rice distribution has an interesting $\{\tau_3, \tau_4\}$ relation to depict on an L-moment ratio diagram. L-moment ratio diagrams, which have not yet been introduced, are thoroughly described in Chapter 10, and some readers might need to consult that chapter first. The Rice distribution is not treated in that chapter but the trajectory the distribution on the diagram is shown in this section. The $\{\tau_3, \tau_4\}$ relation of the Rice distribution in particular is not a constant like most other two parameter distributions (a notable exception is the Gamma distribution, see page 300).

In example 7–38, an L-moment ratio diagram is plotted by the `plotlmrdia()` function, which relies on the `lmrdia()` function to provide lookup tables of $\{\tau_3, \tau_4\}$ by distribution. The diagram is shown in figure 7.12. For the example, five three-parameter distributions are drawn and their abbreviations are shown in the figure legend. Two distributions, the Normal and the Rayleigh, are represented by points; these are shown in the figure. Recalling that the Normal and Rayleigh are limiting conditions of the Rice, the Normal represents the condition of very large signal-to-noise ratio and hence plots on the left side of the diagram (solid square), whereas, the Rayleigh represents the condition of zero signal-to-noise ratio and hence plots on the right side (solid diamond). The $\{\tau_3, \tau_4\}$ relation of the Rice finally is drawn as the thick line and connects to the Normal and Rayleigh endpoints.

7–38

```
n <- 200; nsim <- 500 # no. samples and simulations
nu <- 5; alpha <- 3 # parameter values
RICEpar <- vec2par(c(nu,alpha), type="rice")
RICElmr <- lmomrice(RICEpar) # population L-moments
T3sim <- T4sim <- vector(mode="numeric", length=nsim)
#pdf("ricet3t4n200.pdf")
plotlmrdia(lmrdia(), xlim=c(0, 0.15), ylim=c(0, 0.2),
           autolegend=TRUE, xleg=0.13, yleg=0.08, nouni=TRUE,
           noexp=TRUE, nogum=TRUE, nolimits=TRUE)
for(i in 1:nsim) {
   X   <- rlmomco(n,RICEpar) # simulated values
   lmr <- lmoms(X)$ratios # sample L-moments
   points(lmr[3], lmr[4], cex=0.75, pch=16, col=8)
   T3sim[i] <- lmr[3]; T4sim[i] <- lmr[4]
}
T3 <- .lmomcohash$RiceTable$TAU3 # Tau3 of the Rice
T4 <- .lmomcohash$RiceTable$TAU4 # Tau4 of the Rice
lines(T3,T4, lwd=3) # thick line of the Rice
points(mean(T3sim), mean(T4sim), cex=3, lwd=2)
points(RICElmr$ratios[3], RICElmr$ratios[4], cex=2, pch=16)
#dev.off()
```

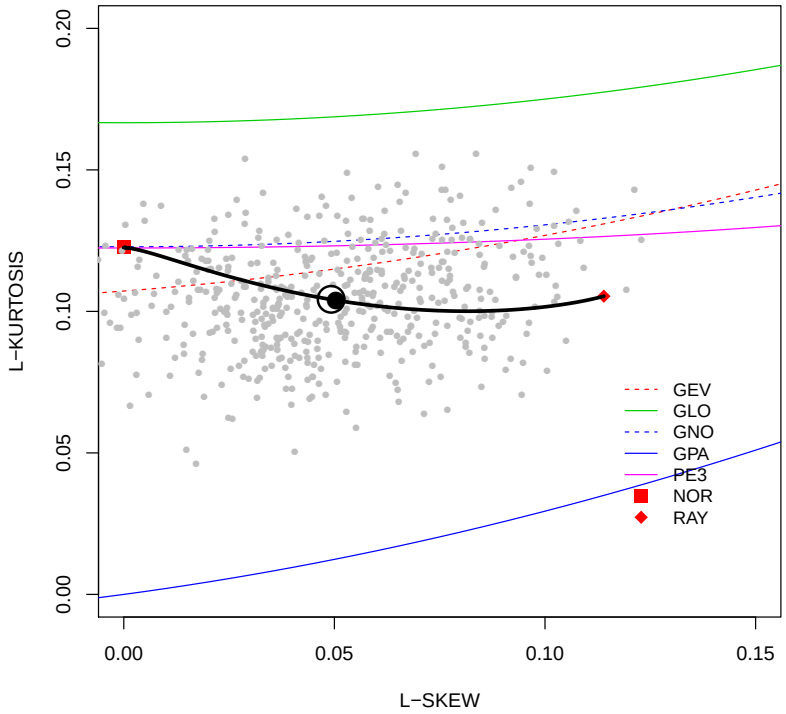

Figure 7.12. L-moment ratio diagram showing 500 simulations of $n = 200$ samples for a Rice having $\nu = 5$ and $\alpha = 3$ from example 7–38. The large open circle reprsents the pair-wise means of L-skew and L-kurtosis and large solid circle represents the population values.

The example continues by performing 500 simulations of $n = 200$ samples for a Rice having $\nu = 5$ and $\alpha = 3$. The $\hat{\tau}_3$ and $\hat{\tau}_4$ of the sample are computed and plotted as the small thin-lined open circles. The second-to-last line computes the mean values of the $\hat{\tau}_3$ and $\hat{\tau}_4$ and plots them as the large solid circle, and the example ends by plotting the population values of τ_3 and τ_4 that were computed by the `lmomrice()` function.

The L-moment ratio diagram in figure 7.12 shows that the numerically computed $\{\tau_3, \tau_4\}$ table in the `.lmomcohash$RiceTable` for signal-to-noise ratios spanning the Rayleigh to Normal intersect and stop at these two distributions. The diagram also shows that the Rice is less L-kurtotic for the range of τ_3 than all the distributions except the Generalized Pareto (the bottom curve of the diagram). Finally, the diagram shows that the sample L-moments estimate mean values for $\{\hat{\tau}_3, \hat{\tau}_4\}$ very close to those of the population. ◀

7.3 Summary

In this chapter, an overview of distributions supported by the *lmom* and *lmomco* packages is provided, but the emphasis of presentation obviously is on the *lmomco* package. The overview considered both the one- and two-parameter distributions that are the subject of this chapter, but also three-parameter (Chapter 8) and four- and more-parameter (Chapter 9) distributions were identified. The overview summarized the many functions of the two packages that provide convenient interaction with distributions or functions that otherwise generally support distributional analysis.

This chapter continued with presentation of the mathematics and examples for 9 two-parameter distributions. (This presentation structure also is used in Chapters 8 and 9.) The 38 examples in this chapter vary in size and scope, but collectively show how to estimate parameters using the method of L-moments, compute parameters from L-moments, and how to use the R functions supporting the PDF, CDF, and QDF of the distributions. Further, several of the examples demonstrate numerical exploration of sampling bias and other selected topics.

- The example for the Normal distribution investigates the sampling bias of the sample standard deviation and compares the bias to that from an L-moment-based estimate of the standard deviation. The biases reported in example 7–8 show that "on average" the estimation of σ using $\hat{\lambda}_2\sqrt{\pi}$ (L-scale$\times\sqrt{\pi}$) is less biased than $\hat{\sigma}$ when the parent is Normal.
- The examples for the Exponential distribution are comparatively simple as inspired by the simplicity of the distribution. The examples collectively demonstrate the function types of *lmomco* used for parameter estimation, the recovery of L-moments from distribution parameters, and both the CDF and QDF of the distribution.
- The examples for the Gamma distribution use sample L-moments from a previous study. The examples show how a two-parameter distribution attains a different L-skew τ_3 than the underlying data. These differences are not specific to the Gamma, but applicable to all two-parameter distributions and will become important in Chapter 10.
- The examples for the Cauchy distribution, because its extreme-order statistics are infinite, are used to demonstrate use of the TL-moments described in Section 6.4.

- The examples for the Gumbel distribution used non-linear optimization for posterior distribution fit and introduced the method of percentiles for parameter estimation.
- The examples for the Reverse Gumbel distribution explore the interrelations between the Gumbel and Reverse Gumbel distributions through the Reflection Rule (see page 36).
- The first example for the Kumaraswamy distribution shows the conversion of L-moments to parameters and back again. The second example is used to depict the $\{\tau_3, \tau_4\}$-parameter space for a wide range of the parameter values. The mapping reproduces figures previously published in the literature.
- The examples for the Rayleigh distribution demonstrate parameter estimation for both known and unknown location parameter and compare the fits of the distribution and the Gamma distribution to a common mode statistic.
- The examples for the Rice distribution are extensive and are used to show the reliability of the *lmomco* algorithms as these appear, after extensive searching, the first to implement L-moments for the distribution. The examples verify the numerical integration of the PDF to create the CDF by making comparisons to the definition using the Marcum Q function. Another example also verifies the quality of QDF, which is based on root solving the CDF by computation of the mean from simulation compared to the theoretical mean based on the Laguerre polynomial. Another example provides an extensive comparison of CDF shapes using a wide range of signal-to-noise ratios from less than to more than unity, and the example shows convergence to the Rayleigh distribution as the signal vanishes and shows convergence to the Normal distribution as the noise vanishes. Finally, an L-moment ratio diagram depicting the $\{\tau_3, \tau_4\}$-parameter space of the Rice is drawn (see Chapter 10, which does not encompass the Rice).

Chapter 8

L-moments of Three-Parameter Univariate Distributions

In this chapter, I present continued discussion of distribution support by L-moments in several R packages, but focus remains clearly on the *lmomco* package. The chapter provides a distribution-by-distribution discussion of mathematics, features, parameters, and L-moments of three-parameter distributions. In general, the mathematics of the distributions are more complex than seen in the previous chapter. Readers possessing considerable familiarity with statistics and R are likely to generally browse as needed through the distributions. Other readers are encouraged to at least review this chapter with the mindset that periodic return likely will be made. This chapter is central to distributional analysis with L-moment statistics using R.

8.1 Introduction

The distributions considered in this chapter have three-parameters and thus are fit to the mean, scale, and skewness (shape) of a sample distribution. As shown in Chapter 6 (see example 6–18) and in examples in this chapter, most notably those associated with the Pearson Type III distribution, L-moments can reliably estimate the skewness of a sample distribution through $\hat{\tau}_3$. Because of the general reliability of $\hat{\tau}_3$, the author suggests that three-parameter distributions should receive considerable attention, and often these might be preferred over lower-order distributions for magnitude and frequency analyses for skewed data sets unless mitigating factors or compelling reasons exist.

Some notes about the source of material, in particular, the mathematics of the three-parameter distributions, which are discussed in this chapter, are needed. Unless otherwise stated, the material is heavily based on collective review of Evans and others (2000),

Hosking (1996b), Hosking and Wallis (1997), and Stedinger and others (1993). Additional citations are provided as needed on a distribution-specific basis.

Finally, the chapter concludes with a summary of selected three-parameter distributions with existing L-moment derivations that are not yet (as of May 2011) implemented within the *lmomco* package. These additional distributions are associated with contemporary research into L-moments, but mostly are presented to show a front line in the continued development of the *lmomco* package.

8.2 Three-Parameter Distributions of the *lmomco* Package

8.2.1 Generalized Extreme Value Distribution

The Generalized Extreme Value distribution is a common distribution in applications involving extreme value analysis of natural phenomena. The two-parameter Gumbel distribution in Section 7.2.5 is a special case of the Generalized Extreme Value for shape parameter $\kappa = 0$. If $k > 0$, then the Generalized Extreme Value also is known as the **Fréchet** or **Extreme Value Type II distribution**.

DISTRIBUTION FUNCTIONS

The distribution functions of the Generalized Extreme Value having parameters ξ (location), α (scale, $\alpha > 0$), and κ (shape, $\kappa > -1$) are

$$f(x) = \alpha^{-1} \exp[-(1-\kappa)Y - \exp(-Y)] \tag{8.1}$$

$$F(x) = \exp[-\exp(-Y)] \tag{8.2}$$

where

$$Y = \begin{cases} -\kappa^{-1} \log\left[1 - \kappa(x-\xi)/\alpha\right] & \text{if } \kappa \neq 0 \\ (x-\xi)/\alpha & \text{if } \kappa = 0 \end{cases} \tag{8.3}$$

and

$$x(F) = \begin{cases} \xi + \alpha(1 - [-\log(F)]^{\kappa})/\kappa & \text{if } \kappa \neq 0 \\ \xi - \alpha \log[-\log(F)] & \text{if } \kappa = 0 \end{cases} \tag{8.4}$$

The ranges of the distribution are

$$-\infty < x \leq \xi + \alpha/\kappa \quad \text{if } \kappa > 0 \tag{8.5}$$

$$-\infty < x < \infty \quad \text{if } \kappa = 0 \tag{8.6}$$

$$\xi + \alpha/\kappa \leq x < \infty \quad \text{if } \kappa < 0 \tag{8.7}$$

The L-moments are

$$\lambda_1 = \xi + \alpha[1 - \Gamma(1+\kappa)]/\kappa \tag{8.8}$$

$$\lambda_2 = \alpha(1 - 2^{-\kappa})\Gamma(1+\kappa)/\kappa \tag{8.9}$$

$$\tau_3 = 2(1 - 3^{-\kappa})/(1 - 2^{-\kappa}) - 3 \tag{8.10}$$

$$\tau_4 = \frac{5(1 - 4^{-\kappa}) - 10(1 - 3^{-\kappa}) + 6(1 - 2^{-\kappa})}{(1 - 2^{-\kappa})} \tag{8.11}$$

where $\Gamma(a)$ is the complete gamma function that is shown in eq. (8.85). No explicit solution for the κ parameter in terms of the L-moments is possible and a hybrid of numerical methods are used by *lmomco*. The other two parameters are

$$\alpha = \frac{\lambda_2 \kappa}{(1 - 2^{-\kappa})\Gamma(1+\kappa)} \tag{8.12}$$

$$\xi = \lambda_1 - \alpha[1 - \Gamma(1+\kappa)]/\kappa \tag{8.13}$$

USING R ——————————— ——————————— USING R

The Generalized Extreme Value distribution is demonstrated in example 8–1 using parameters from L-moments reported by (Hosking and Wallis, 1997, table 2.5). These are listed in table 8.1. The parameters represent a site-specific characterization of the annual maximum windspeed for the indicated location.

Table 8.1. L-moments of wind speed data reported by Hosking and Wallis (1997, table 2.5)

Location	ξ (miles per hour)	α (miles per hour)	κ (--)
Brownsville, Texas	39.8	6.26	−0.037
Corpus Christi, Texas	47.5	4.87	−.471
Port Arthur, Texas	48.5	7.15	−.059

8–1

```
Br <- vec2par(c(39.8, 6.26, -0.037), type="gev")
Cr <- vec2par(c(47.5, 4.87, -0.471), type="gev")
Pr <- vec2par(c(48.5, 7.15, -0.059), type="gev")
F <- nonexceeds(); F <- F[F >= 0.75]
the.quantiles <- data.frame(F=F,
                      Brownsville=round(  quagev(F,Br)),
                      CorpusChristi=round(quagev(F,Cr)),
                      PortArthur=round(   quagev(F,Pr)))
print(the.quantiles)
       F Brownsville CorpusChristi PortArthur
1  0.750          48            56         58
2  0.800          49            58         60
3  0.850          52            61         62
4  0.900          54            67         66
5  0.950          59            79         72
6  0.960          61            84         74
7  0.980          66           102         80
8  0.990          71           127         86
9  0.996          78           176         95
10 0.998          84           230        102
```

The quantiles show that for the 99th-percentile ($F = 0.99$) or 100-year recurrence interval (`prob2T(0.99)` from *lmomco*) for Corpus Christi, Texas is estimated to be about 56 and 41 miles per hour more than for Brownsville and Port Arthur, Texas, respectively. It is unknown whether these differences are reliable and show that Corpus Christi has higher wind risk than the other two locales or whether the differences exist because of sampling and uncertainties of the basic form of the parent distribution. Regional analysis of these sample L-moments and those from other observation points on the Texas Gulf Coast would be recommended. ◀

For another demonstration of the Generalized Extreme Value distribution, a small application is created to read-in a selected file of annual peak streamflow data from the U.S. Geological Survey streamflow-gaging station 08167000 Guadalupe River near Comfort, Texas. The data resides in file `lmomco/inst/testdata/sta08167000.txt`. For illustration, the fit of the Generalized Extreme Value is inverted into equivalent T-year recurrence intervals to judge the historical context of the data. For the example, an explicit assumption is made that the Generalized Extreme Value is an appropriate distribution for the problem.

The application begins with example 8–2 by prompting the user for the file name using the `file.choose()` function. The contents of the selected file are read-in by the `read.table()` function. The data reside in `peak_va`, and these are extracted and sorted

into the variable `Q`. The example ends with the computation of the Weibull plotting positions.

8–2
```
file <- file.choose() # select the sta08167000.txt file
D    <- read.table(file, header=TRUE, sep="\t") # open the file
Q    <- sort(D$peak_va) # extract and sort peak data
Fs   <- pp(Q) # compute Weibull plotting positions
```

These data also can be accessed by `data(USGSsta08167000peaks)`, but for the example, a more "manual" style is shown by the file specification using the `file.choose()` function, which launches a conventional file browser of the host operating system. The text file `sta08167000.txt` is provided within the *lmomco* source distribution to support the example.

The application continues in example 8–3 in which the L-moments of the data are computed, the parameters of the Generalized Extreme Value distribution are computed as `GEVpar`, and finally, the nonexceedance probabilities F are stored in variable `GEV_Fs` by the CDF of the Generalized Extreme Value distribution or `cdfgev()`.

8–3
```
lmr <- lmoms(Q) # compute L-moments
GEVpar <- pargev(lmr) # GEV parameter estimation
GEV_Fs <- cdfgev(Q,GEVpar) # inversion of the GEV
```

The application is completed in example 8–4. This example handles the portable document format (PDF) generation of the output in which two pages will be contained. The `gsub()` function is used to strip out the trailing `.txt` of the file name and replacing it with `.pdf`. The two calls to `plot()` generate the first and second pages of the file `pdffile()` (`sta08167000.pdf` for the example). The `abline()` function is used to draw a one-to-one sloped line on the T-year recurrence interval plot. Readers should note the use of the `prob2T()` function to convert nonexceedance probabilities into recurrence intervals. Finally, the two plots are shown in figures 8.1 and 8.2.

8–4
```
pdffile <- gsub(".txt",".pdf",file) # make pdf file name
pdf(pdffile) # open pdf device
  plot(prob2T(GEV_Fs),prob2T(Fs),
       xlab = "GEV␣RECURRENCE␣INTERVAL,␣IN␣YEARS",
       ylab = "ANNUAL␣RECURRENCE␣INTERVAL␣BY␣PLOTTING␣POS.")
  abline(0,1) # one-to-one sloped line
  plot(qnorm(Fs),quagev(Fs,GEVpar), type="l",
       xlab = "STANDARD␣NORMAL␣DEVIATE",
```

```
       ylab = "STREAMFLOW, IN CUBIC FEET PER SECOND")
  points(qnorm(Fs),Q)
dev.off() # close up the pdf device, now the pdf is viewable.
```

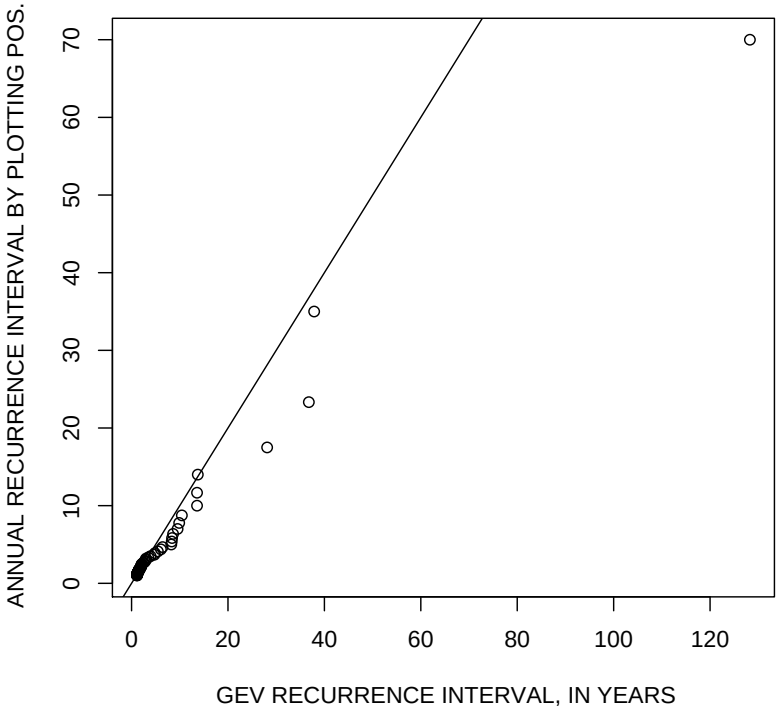

Figure 8.1. Comparison of T-year recurrence interval of individual annual peak streamflow data points estimated by CDF of Generalized Extreme Value distribution and those from Weibull plotting positions for U.S. Geological Survey streamflow-gaging station 08167000 Guadalupe River at Comfort, Texas from example 8–4 [first `plot()` call]. The line is one-to-one sloped.

◀

8.2.2 Generalized Logistic Distribution

The Generalized Logistic distribution likely is a less commonly used distribution than the Generalized Extreme Value. The distribution has three-parameters and thus is fit to the mean, scale, and shape of a data set. As will be seen the Generalized Logistic is more kurtotic than the other three-parameter distributions described herein. As reported by Hosking and Wallis (1997, p. 197), the Generalized Logistic is a reparametrization (generalization) of the **log-logistic distribution**. Alkasasbeh and Raqab (2008) provide an exten-

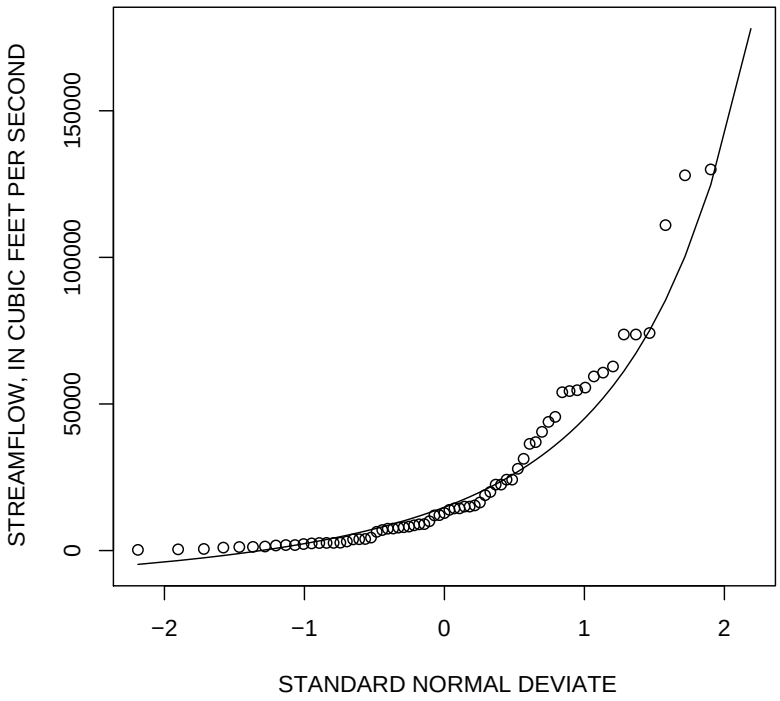

Figure 8.2. Comparison of empirical distribution of annual peak streamflow data (open circles) and fitted Generalized Extreme Value distribution (solid line) for U.S. Geological Survey streamflow-gaging station 08167000 Guadalupe River at Comfort, Texas from example 8–4 [second `plot()` call]

sive study of five parameter estimation methods for the distribution; the methods include maximum likelihood, method of moments, method of percentiles, least and weighted-least squares, and method of L-moments.

DISTRIBUTION FUNCTIONS

The distribution functions of the Generalized Logistic having parameters ξ (location), α (scale, $\alpha > 0$), and κ (shape, $-1 < k < 1$) are

$$f(x) = \frac{\alpha^{-1} \exp[-(1-\kappa)Y]}{[1+\exp(-Y)]^2} \tag{8.14}$$

$$F(x) = 1/[1+\exp(-Y)] \tag{8.15}$$

where

$$Y = \begin{cases} -\kappa^{-1} \log[1 - \kappa(x - \xi)/\alpha] & \text{if } \kappa \neq 0 \\ (x - \xi)/\alpha & \text{if } \kappa = 0 \end{cases} \tag{8.16}$$

and

$$x(F) = \begin{cases} \xi + \alpha(1 - [(1 - F)/F]^{\kappa})/\kappa & \text{if } \kappa \neq 0 \\ \xi - \alpha \log[(1 - F)/F] & \text{if } \kappa = 0 \end{cases} \tag{8.17}$$

The ranges of the distribution are

$$-\infty < x \leq \xi + \alpha/\kappa \quad \text{if } \kappa > 0 \tag{8.18}$$

$$-\infty < x < \infty \quad \text{if } \kappa = 0 \tag{8.19}$$

$$\xi + \alpha/\kappa \leq x < \infty \quad \text{if } \kappa < 0 \tag{8.20}$$

The L-moments are

$$\lambda_1 = \xi + \alpha[1/\kappa - \pi/\sin(\kappa\pi)] \tag{8.21}$$

$$\lambda_2 = \alpha\kappa\pi/\sin(\kappa\pi) \tag{8.22}$$

$$\tau_3 = -\kappa \tag{8.23}$$

$$\tau_4 = (1 + 5\kappa^2)/6 \tag{8.24}$$

and the relation between τ_3 and τ_4 is

$$\tau_4 = \frac{1 + 5(\tau_3)^2}{6} \tag{8.25}$$

The parameters are

$$\kappa = -\tau_3 \tag{8.26}$$

$$\alpha = \frac{\lambda_2 \sin(\kappa\pi)}{\kappa\pi} \tag{8.27}$$

$$\xi = -\lambda_1 - \alpha \left(\frac{1}{\kappa} - \frac{\pi}{\sin(\kappa\pi)} \right) \tag{8.28}$$

USING R ——————————————— ——————————————— USING R

Asquith (1998) in a large study of the L-moments and parameters of Generalized Logistic and Generalized Extreme Value distributions for annual maximum rainfall in Texas

concludes that the Generalized Logistic distribution is appropriate for rainfall durations less than 24-hours and the Generalized Extreme Value distribution was appropriate for larger durations. Parameters of the Generalized Logistic distribution of 1-hour annual maximum rainfall for Travis County, Texas are listed in table 8.2.

Table 8.2. Parameters and corresponding L-moments of Generalized Logistic distribution for 1-hour annual maximum rainfall for Travis County, Texas derived from Asquith (1998)

ξ (inches)	α (inches)	κ (--)	λ_1 (inches)	λ_2 (inches)	τ_3 (--)	τ_4 (--)
1.7	0.35	−0.20	1.82	0.374	0.200	0.200

The CDF and QDF of the fitted Generalized Logistic distribution are produced with example 8–5 and are shown in figure 8.3. The variable `PARglo` stores the *lmomco* parameter list (see page 163 and ex. 7–1) for the distribution. The `quaglo()` and `cdfglo()` provide the QDF and CDF of the distribution, respectively.

8–5

```
PARglo <- vec2par(c(1.7,0.35,-0.20), type="glo")
F <- nonexceeds()  # list of nonexceedance probs.
Q <- quaglo(F,PARglo) # will need Q in cdf generation
#pdf("glo1.pdf")
layout(matrix(1:2, nrow=1))
plot(Q,cdfglo(Q,PARglo), ylab="F", type="l")
plot(F,Q, type="l")
#dev.off()
```

◀

The corresponding L-moments of the Generalized Logistic parameters are listed in table 8.2 and are computed in example 8–6. The example uses the `lmomglo()` and `par2lmom()` functions to the same effect. In each case, the `lmorph()` function is used to convert the returned L-moments to a more succinct data structure—the *lmomco* L-moment list (see page 127 and exs. 6–7 – 6–9). The L-moments are listed in table 8.2. It is a coincidence for this particular example that τ_3 and τ_4 are effectively equal.

8–6

```
PARglo <- vec2par(c(1.7,0.35,-0.20), type="glo")
the.Lmoms.shown    <- lmorph(lmomglo( PARglo))
the.Lmoms.notshown <- lmorph(par2lmom(PARglo))
str(the.Lmoms.shown)
List of 6
```

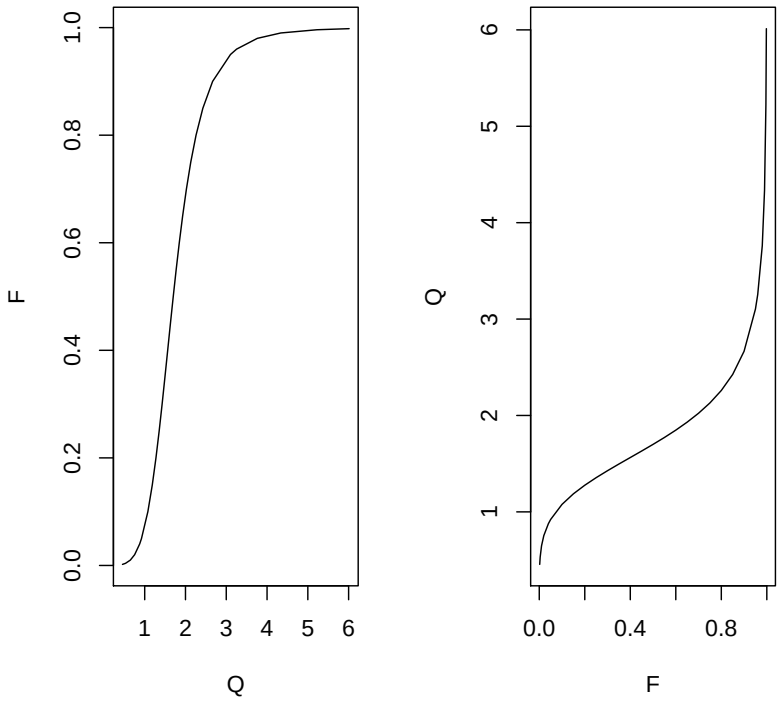

Figure 8.3. CDF and QDF of Generalized Logistic distribution fit to L-moments in table 8.2 from example 8–5

```
$ lambdas : num [1:4] 1.8207 0.3741 0.0748 0.0748
$ ratios  : num [1:4]    NA 0.205 0.200 0.200
$ trim    : num 0
$ leftrim : NULL
$ rightrim: NULL
$ source  : chr "lmorph"
```

◀

The Generalized Logistic distribution is more kurtotic—that is, has a larger τ_4^{glo} for a given τ_3—than the τ_4^{glo} of the Generalized Extreme Value distribution. The relations between τ_4 for these two distributions and many others are discussed further in Chapter 10.

Example 8–7, for a symmetrical distribution ($\tau_3 = 0$), demonstrates the effect of larger τ_4^{glo} on the far tails of each distribution. A symmetrical distribution provides a means to explicitly consider the interpretations of kurtosis—L-kurtosis τ_4 in the context here. The L-moments are set by the `vec2lmom()` function, and distributions will be fit to these L-moments. The two fitted distributions set into `PARgev` and `PARglo` using the

`pargev()` and the `parglo()` functions. The `lmomgev()` and `lmomglo()` functions compute the respective distribution L-moments. As the example shows, τ_4^{glo} (`T4.glo = 0.167`) is larger than τ_4^{gev} (`T4.gev = 0.107`).

8–7

```
lmr <- vec2lmom(c(2000,500,0))
PARgev <- pargev(lmr);          PARglo <- parglo(lmr)
LMRgev <- lmomgev(PARgev);      LMRglo <- lmomglo(PARglo)
T4.gev <- round(LMRgev$TAU4,3); T4.glo <- round(LMRglo$TAU4,3)
cat(c(  "T4.gev=",T4.gev,
      "␣␣T4.glo=",T4.glo,"\n"))
T4.gev= 0.107   T4.glo= 0.167
```

◀

Continuing from example 8–7, the code in example 8–8 produces the comparison shown in figure 8.4. The thin line `lwd=1` in the figure is the Generalized Extreme Value and the thick line `lwd=3` is the Generalized Logistic. Thus, although the distributions mathematically differ, the distributions are fit to the same L-moments. For the example, the two distributions generally have similar (near identical) quantiles in the central part of the range of F values.

8–8

```
F <- nonexceeds() # vector of selected nonexceedance
   probabilities
#pdf("glo2.pdf")
plot(F,quagev(F,PARgev), type="l", lwd=1,
     xlab="Nonexceedance␣Probability", ylab="Quantile")
lines(F,quaglo(F,PARglo), lwd=3)
legend(0.2, 4000, c("GEV","GLO"), lwd=c(1,3), box.lty=0, bty="n")
#dev.off()
```

◀

Finally, for the distribution parameters considered in example 8–7 and the F values, comparison of the PDFs of the two fitted distributions is made using example 8–9. The example uses the `quaglo()` and `quagev()` functions to compute distribution-specific ranges `x.glo` and `x.gev`. These ranges in turn are used with the PDF functions `pdfglo()` and `pdfgev()`. The PDFs are shown in figure 8.5.

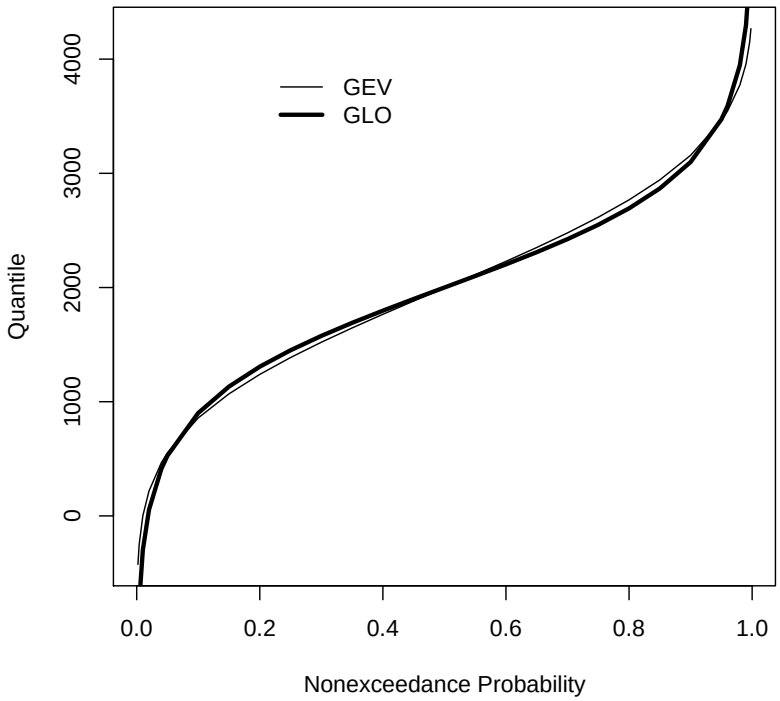

Figure 8.4. Comparison of QDF for Generalized Extreme Value and Generalized Logistic distributions fit to L-moments of $\lambda_1 = 2000$, $\lambda_2 = 500$, and $\tau_3 = 0$ from example 8–8

8–9

```
x.glo <- quaglo(F,PARglo) # arguments F, PARglo, and PARgev
x.gev <- quagev(F,PARgev) # derived from previous example
#pdf("glo3.pdf")
plot(x.glo,pdfglo(x.glo,PARglo), type="l",
     xlab="x", ylab="f(x)", lwd=3)
lines(x.gev,pdfgev(x.gev,PARgev), lwd=1)
legend(-500, 0.0005, c("GEV","GLO"),
       lwd=c(1,3), box.lty=0, bty="n")
#dev.off()
```

◀

8.2.3 Generalized Normal Distribution

The Generalized Normal is a Normal distribution whose generalization accommodates non-zero skewness and retains the Normal as a special case. The Generalized Normal is lauded as a replacement for the log-Normal distribution by avoiding the introduction of

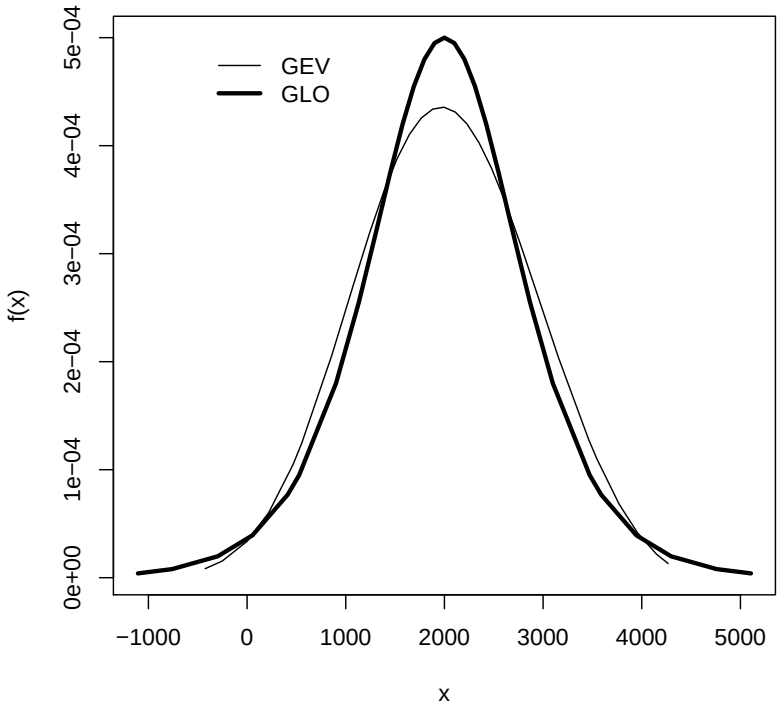

Figure 8.5. Comparison of PDF for Generalized Extreme Value and Generalized Logistic distributions fit to L-moments of $\lambda_1 = 2000$, $\lambda_2 = 500$, and $\tau_3 = 0$ from example 8–9

logarithmic transformation of the data prior to computation of sample statistics. Logarithmic transformation can be problematic for circumstances involving negative or zero values. Further, logarithmic transformation can accentuate the influence of small values (low outliers) on sample statistics while the influence of large values is decreased (see Section 4.4). Comparisons between the Generalized Normal and log-Normal distributions are made in this section.

DISTRIBUTION FUNCTIONS

The distribution functions of the Generalized Normal having parameters ξ (location), α (scale, $\alpha > 0$), and κ (shape) are

$$f(x) = \frac{\exp(\kappa Y - Y^2/2)}{\alpha\sqrt{2\pi}} \tag{8.29}$$

$$F(x) = \Phi(Y) \tag{8.30}$$

$x(F)$ has no explicit analytical form

where $\Phi(Y)$ is the CDF of the standard Normal distribution and Y is

$$Y = \begin{cases} -\kappa^{-1}\log[1-\kappa(x-\xi)/\alpha] & \text{if } \kappa \neq 0 \\ (x-\xi)/\alpha & \text{if } \kappa = 0 \end{cases} \tag{8.31}$$

The ranges of the distribution are

$$-\infty < x \leq \xi + \alpha/\kappa \quad \text{if } \kappa > 0 \tag{8.32}$$

$$-\infty < x < \infty \quad \text{if } \kappa = 0 \tag{8.33}$$

$$\xi + \alpha/\kappa \leq x < \infty \quad \text{if } \kappa < 0 \tag{8.34}$$

The first two L-moments are

$$\lambda_1 = \xi + \frac{\alpha}{\kappa}[1-\exp(\kappa^2/2)] \tag{8.35}$$

$$\lambda_2 = \frac{\alpha}{\kappa}[\exp(\kappa^2/2)][1-2\Phi(-\kappa/\sqrt{2})] \tag{8.36}$$

There are no simple expressions for τ_3, τ_4, and τ_5. There are no simple expressions for the parameters in terms of the L-moments. Numerical methods are required.

Emphasis is needed that logarithmic transformation of the data prior to fitting of the Generalized Normal distribution is not required. Whereas, logarithmic transformation is needed for the log-Normal distribution having parameters ξ, $\mu_{\log}$, and $\sigma_{\log}$. A closely related distribution to the Generalized Normal is the **3-parameter log-Normal distribution** (log-Normal3). In particular, the log-Normal3 distribution for $x > 0$ has the same distribution functions with the substitution of Y in eq. (8.31) for the following

$$Y = \frac{\log(x-\zeta) - \mu_{\log}}{\sigma_{\log}} \tag{8.37}$$

where ζ is the lower bounds (real space) for which $\zeta < \lambda_1 - \lambda_2$, $\mu_{\log}$ is the mean in log-space, and $\sigma_{\log}$ is the standard deviation in log-space for which $\sigma_{\log} > 0$.

The parameter equalities between the Generalized Normal and log-Normal3, by letting $\eta = \exp(\mu_{\log})$, are

$$\xi = \zeta + \eta \tag{8.38}$$

$$\alpha = \eta\sigma_{\log} \tag{8.39}$$

$$\kappa = -\sigma_{\log} \tag{8.40}$$

from which the L-moments can be computed by algorithms for the Generalized Normal. The parameters of the log-Normal3 in terms of the parameters of the Generalized Normal, by letting $\eta = \lambda_1 - \zeta$, are

$$\sigma_{\log} = \sqrt{2} \times \Phi^{(-1)}(0.5[1 + \lambda_2/\eta]) \tag{8.41}$$

$$\mu_{\log} = \log(\eta) - 0.5\sigma_{\log}^2 \tag{8.42}$$

for a known ζ and, by letting $\eta = \alpha/\sigma_{\log}$, are

$$\sigma_{\log} = -\kappa \tag{8.43}$$

$$\mu_{\log} = \log(\eta) \tag{8.44}$$

$$\zeta = \xi - \eta \tag{8.45}$$

for an unknown ζ. Readers should note that natural logarithms are represented by the $\log()$ function in the prior typeset mathematics, and this mimics the syntax of natural logarithms `log()` in R. For an example of a study using the log-Normal3 and within in the context of L-moments, Benson (1993) concludes that the log-Normal3 and Generalized Extreme Value distributions are appropriate for modeling hydraulic conductivity of compacted soil liners (a common landfill liner and cover).

USING R

The Generalized Normal accommodates skewness. The generalized nature of the distribution is demonstrated by plotting a PDF for each of three ensembles of L-moments. Example 8–10 sets the L-moments in `lmr1`, `lmr2`, and `lmr3` by the `vec2lmom()` function. The parameters for each are set in `PAR1`, `PAR2`, and `PAR3` by the `lmom2par()` and `pargno()` functions. The two functions are purposefully used to show two alternative methods in *lmomco* for parameter estimation. The `lmom2par()` dispatches to `pargno()` based on the `type` argument. The `layout()` function is used to set up three stacked plots. The high-level function `check.pdf()` is used to succinctly plot the PDF of each

distribution. (The function was created as a tool to verify that PDF functions properly integrate to unity—see the documentation.) The three PDFs are shown in figure 8.6.

8–10

```
lmr1 <- vec2lmom(c(0, 1, -0.4)) # set three suites
lmr2 <- vec2lmom(c(0, 1,  0.0)) # of L-moments
lmr3 <- vec2lmom(c(0, 1,  0.2)) # for this example
PAR1 <- lmom2par(lmr1, type="gno") # not parallel style, but
PAR2 <- pargno(lmr2) # different dialect to perform
PAR3 <- pargno(lmr3) # parameter estimation
#pdf("gnopdf.pdf")
layout(matrix(1:3, ncol=1))
check.pdf(pdfgno,PAR1, plot=TRUE)
check.pdf(pdfgno,PAR2, plot=TRUE)
check.pdf(pdfgno,PAR3, plot=TRUE)
#dev.off()
```

◀

Returning to the distribution of annual peak streamflow for U.S. Geological Survey streamflow-gaging station 05405000 Baraboo River near Baraboo, Wisconsin considered in example 2–32 on page 57 and associated discussion, example 8–11 loads the data and prepares the annual peak streamflow data for plotting by use of functions `sort()` and `pp()`. The sample L-moments are computed by `lmoms()` and are listed in the first three columns in table 8.3. The product moment values of the logarithms are shown in the last two columns and are repeated from the output of example 2–32. The Generalized Normal parameters are computed by `pargno()` into `GNOpar` and are GNO(13811, 19049, −1.0710).

8–11

```
#pdf("gnolognor.pdf")
data(USGSsta05405000peaks) # from lmomco package
attach(USGSsta05405000peaks)
Q   <- sort(peak_va) # sort the annual peak streamflow values
PP  <- pp(Q) # compute Weibull plotting positions
lmr    <- lmoms(Q);        GNOpar <- pargno(lmr)
lmr.lg <- lmoms(log10(Q)); NORpar <- parnor(lmr.lg)
plot(qnorm(PP),Q, xlab="STANDARD␣NORMAL␣DEVIATE",
     ylab="STREAMFLOW,␣IN␣FT^3/S")
lines(qnorm(PP), quagno(PP,GNOpar)) # plot gno as solid line
lines(qnorm(PP), 10^quanor(PP,NORpar),
      lty=2) # plot lognormal distribution as dashed line
#dev.off()
```

These parameters can be reverted to L-moment vectors by a combination of the `lmomgno()` and `lmorph()` functions as shown in example 8–12.

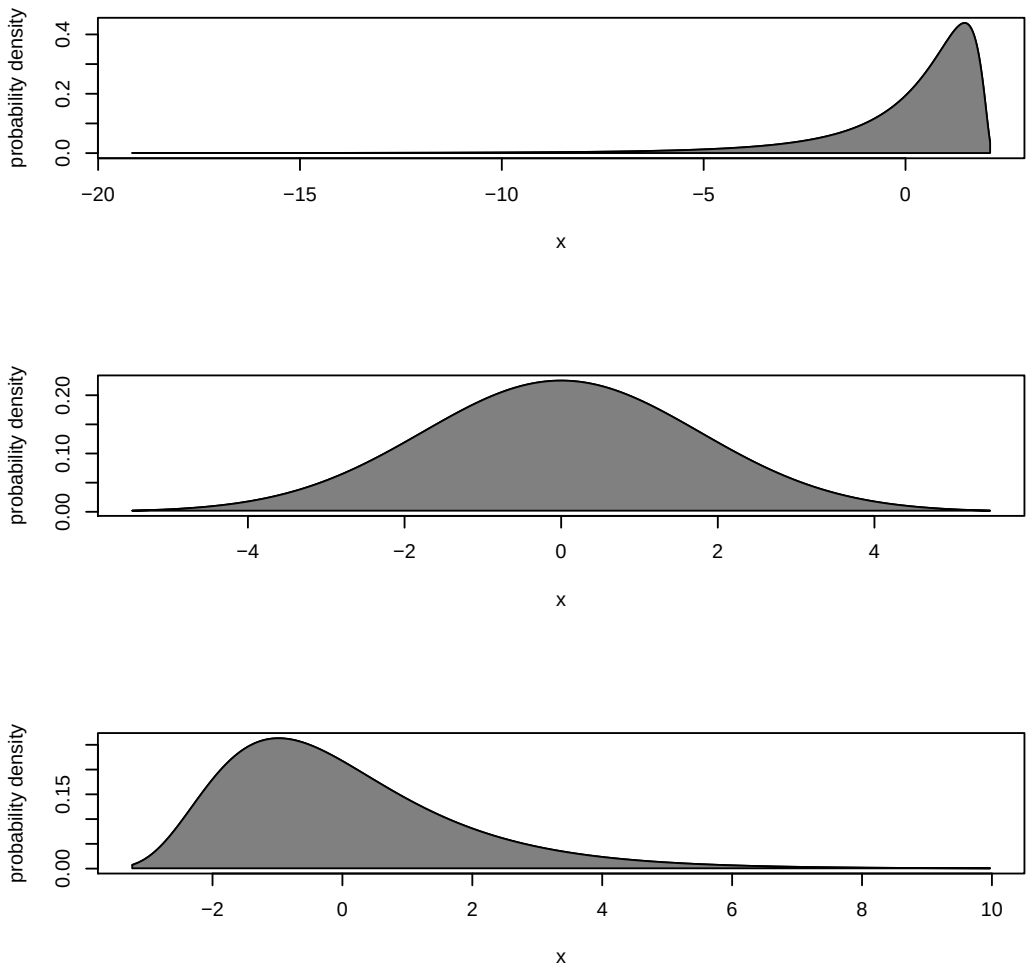

Figure 8.6. Probability density functions for three selected Generalized Normal distributions.

8–12

```
str(lmorph(lmomgno(GNOpar)))
List of 6
 $ lambdas : num [1:5] 3134.6  893.9  159.7  132.1   52.9
 $ ratios  : num [1:5]     NA 0.2852 0.1786 0.1477 0.0592
 $ trim    : num 0
 $ leftrim : NULL
 $ rightrim: NULL
 $ source  : chr "lmorph"
```

Continuing with the primary demonstration, example 8–11 also computes the sample L-moments of $\log_{10}$-transformed data and fits a Normal distribution using `parnor()`. The empirical distribution and the fitted Generalized Normal and log-Normal distributions are then plotted, and the results are shown in figure 8.7.

The plot in figure 8.7 shows that the Generalized Normal provides a preferable fit—the distribution is more reliably fit by the method of L-moments and has avoided the use of logarithms. Using L-moments, the analyst can work in the natural units of the data. By better representing the first three sample L-moments, the Generalized Normal is preferable to the log-Normal for the current data. Readers might compare figure 2.11 on page 59 to figure 8.7, and note that the log-Normal distribution is represented by a dashed line in each of the two figures. ◀

Table 8.3. L-moments of annual peak streamflow data for 05405000 Baraboo River near Baraboo, Wisconsin and parameters for fitted Generalized Normal distribution

λ_1 (ft^3/s)	λ_2 (ft^3/s)	τ_3 (--)	ξ (ft^3/s)	α (ft^3/s)	κ (--)	$\mu(\log_{10})$ (ft^3/s)	$\sigma(\log_{10})$ (ft^3/s)
3135	894	0.1786	2849	1497	−0.3683	3.438	0.2356

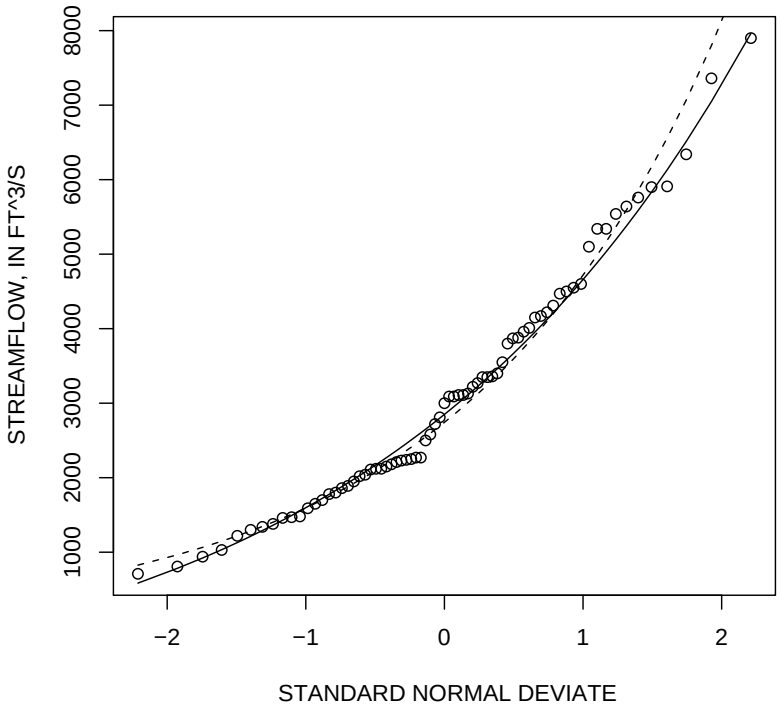

Figure 8.7. Empirical distribution of annual peak streamflow data for U.S. Geological Survey streamflow-gaging station 05405000 Baraboo River near Baraboo, Wisconsin and Generalized Normal (solid line) and log-Normal (dashed line) distributions fit by method of L-moments from example 8–11

An exploration of the sampling properties and general applicability of the Generalized Normal in the context that the parent distribution is log-Normal is made using example 8–13. The parent log-Normal distribution is defined using the product moments listed in table 8.3. A sample size $n = 70$ is chosen. The quantile of interest has $F = 0.99$, which means $X_{0.99} = 9{,}680$ (`true.Quantile`).

8–13

```
mu <- 3.438; sig <- 0.2356
n <- 70; F <- 0.99; nsam <- 10000
NORpar <- vec2par(c(mu,sig), type="nor")
true.Quantile <- 10^(quanor(F,NORpar))
eps.bylognor <- vector(mode = "numeric")
eps.bygno <- eps.bylognor
for(i in seq(1:nsam)) {
  logQ <- rlmomco(n,NORpar)

  smu <- mean(logQ); ssig <- sd(logQ)
  sNORpar <- vec2par(c(smu,ssig), type="nor")

  lmr <- lmoms(10^logQ)
  sGNOpar <- pargno(lmr)

  eps.bylognor[i] <- 10^(quanor(F,sNORpar)) - true.Quantile
  eps.bygno[i]    <-     quagno(F,sGNOpar)  - true.Quantile
}
```

Next, through the `for()` loop, the `nsam` differences between the two estimated $\hat{X}_{0.99}$ and $X_{0.99}$ are computed. The summary statistics of the differences then are computed in example 8–14.

8–14

```
summary(eps.bylognor) # errors by log-Normal
    Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
-3816.00  -819.40   -61.61    31.98   789.80  5588.00

summary(eps.bygno) # errors by Gen. Normal
    Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
-4509.00 -1097.00  -123.00    54.26   991.60 10270.00

# Relative efficiency
RE <- sum((eps.bylognor - mean(eps.bylognor))^2) /
      sum((eps.bygno    - mean(eps.bygno))^2)

print( round(RE, digits=3) )
[1] 0.549
```

The summary statistics in example 8–14 show that the product moments, through the assumption that the parent is log-Normal, provide slightly less biased estimates of $X_{0.99}$ than the L-moments through the Generalized Normal distribution. The statistics also show that the spread or variability of the $\hat{X}_{0.99}$ estimates from the Generalized Normal is larger. The relative efficiency is $\mathrm{RE}(\mathrm{lognor}, \mathrm{gno}) \approx 0.549$. This value shows that the product moments can outperform the L-moments, but do so here because the simulated parent is log-Normal (G of the logarithms is zero), simulated from log-space, and the estimated distribution also is log-Normal. ◀

The Generalized Normal and log-Normal3 distributions are closely related. In example 8–15, a Generalized Normal is fit to some L-moments, and the QDF is plotted in figure 8.8.

8–15

```
lmr <- vec2lmom(c(1000,300,0.2))
F <- seq(0,1, by=0.05) # 5-percent intervals
#pdf("ln3.pdf")
plot(F,qlmomco(F,pargno(lmr)), ylim=c(-600,2500),
     type="l", lwd=2, lty=2) # dashed line
lines(F,qlmomco(F,parln3(lmr, zeta=NULL)), col=2) # red line
lines(F,qlmomco(F,parln3(lmr, zeta=-600)))
lines(F,qlmomco(F,parln3(lmr, zeta=0   )))
lines(F,qlmomco(F,parln3(lmr, zeta=200 )))
lines(F,qlmomco(F,parln3(lmr, zeta=400 )))
#dev.off()
```

The figure shows open circles for the Generalized Normal values by 5-percent increments. Additionally, the lines depicted various solutions for the log-Normal3 distribution, in which the unknown ζ-parameter solution is plotted in red. In the figure, the red line (log-Normal3) plots along the dashed line (Generalized Normal)—the two distributions are the same. ◀

8.2.4 Generalized Pareto Distribution

The Generalized Pareto distribution (Hosking and Wallis, 1987) likely is a less commonly used distribution than the Generalized Extreme Value in distributional analysis of earth-systems data such as floods, droughts, and rainfall. The Generalized Pareto distribution is more common in financial studies as a historical model of income distribution. The

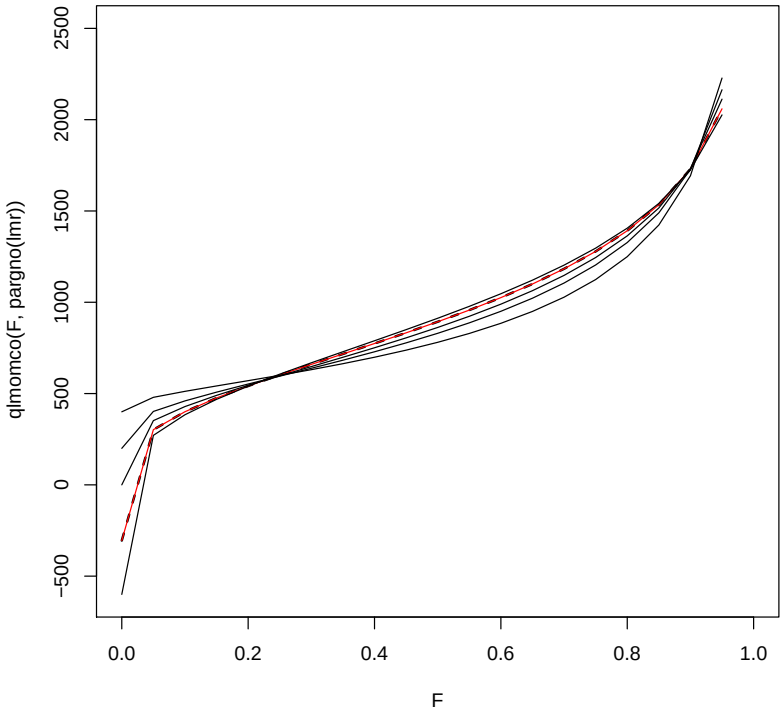

Figure 8.8. Quantile function by 5-percent intervals for a Generalized Normal (dashed line) distribution and several log-Normal3 fits using selected lower limits and fit (red line) treating lower limit as unknown from example 8–15

Generalized Pareto generally is less kurtotic (τ_4) than the other three-parameter distributions described here and much less so for negatively skewed or left-tail heavy data (see Chapter 10).

The Generalized Pareto distribution is especially useful as a distribution for pedagogical purposes: (1) it is a three-parameter distribution that supports both known and unknown lower limits, (2) the distribution functions are readily computed and theoretical integrations by eqs. (3.4) and (6.1) are straightforward, and (3) the L-moments in terms of the parameters and parameters in terms of L-moments are readily computed. These factors make the Generalized Pareto attractive for educational settings including examination purposes.

DISTRIBUTION FUNCTIONS

The distribution functions of the Generalized Pareto having parameters ξ (location), α (scale, $\alpha > 0$), and κ (shape, $\kappa > -1$) are

$$f(x) = \alpha^{-1} \exp[-(1-\kappa)Y] \tag{8.46}$$

$$F(x) = 1 - \exp(-Y) \tag{8.47}$$

where

$$Y = \begin{cases} -\kappa^{-1} \log[1 - \kappa(x-\xi)/\alpha] & \text{if } \kappa \neq 0 \\ (x-\xi)/\alpha & \text{if } \kappa = 0 \end{cases} \tag{8.48}$$

and

$$x(F) = \begin{cases} \xi + \alpha[1-(1-F)^{\kappa}]/\kappa & \text{if } \kappa \neq 0 \\ \xi - \alpha \log(1-F) & \text{if } \kappa = 0 \end{cases} \tag{8.49}$$

The ranges of the distribution are

$$\xi < x \leq \xi + \alpha/\kappa \quad \text{if } \kappa > 0 \tag{8.50}$$

$$\xi \leq x < \infty \qquad \text{if } \kappa \leq 0 \tag{8.51}$$

The L-moments are

$$\lambda_1 = \xi + \alpha/(1+\kappa) \tag{8.52}$$

$$\lambda_2 = \alpha/[(1+\kappa)(2+\kappa)] \tag{8.53}$$

$$\tau_3 = (1-\kappa)/(3+\kappa) \tag{8.54}$$

$$\tau_4 = (1-\kappa)(2-\kappa)/[(3+\kappa)(4+\kappa)] \tag{8.55}$$

The parameters for a known ξ are

$$\kappa = [(\lambda_1 - \xi)/\lambda_2] - 2 \tag{8.56}$$

$$\alpha = (1+\kappa)(\lambda_1 - \xi) \tag{8.57}$$

and the relation between τ_3 and τ_4 is

$$\tau_4 = \frac{\tau_3(1+5\tau_3)}{5+\tau_3} \tag{8.58}$$

and the parameters for an unknown ξ are

$$\kappa = (1 - 3\tau_3)/(1 + \tau_3) \tag{8.59}$$

$$\alpha = (1 + \kappa)(2 + \kappa)\lambda_2 \tag{8.60}$$

$$\xi = \lambda_1 - (2 + \kappa)\lambda_2 \tag{8.61}$$

The Generalized Pareto distribution is a true three-parameter distribution and is fit to the L-moments λ_1, λ_2, and τ_3 when the lower limit ξ is unknown, but when ξ is known, the Generalized Pareto becomes a two-parameter distribution, and the distribution is not fit to the skewness of the data. If $\kappa = 0$, the Exponential distribution results, and if $\kappa = 1$, the Uniform distribution results. The Generalized Pareto can be formulated as quite similar to the Generalized Lambda (see Section 9.2.2) when $\kappa^{\text{gld}} = 0$.

USING R ——————— ——————— USING R

Suppose a Generalized Pareto is specified as GPA(-6000, 400, -0.5), the first four L-moments can be manually computed. Using analytical expressions (eqs. (8.52) – (8.55)) for the L-moments in terms of the parameters that follow

$$\lambda_1 = -6000 + 400/(1 - 0.5) = -5200 \tag{8.62}$$

$$\lambda_2 = 400/[(1 - 0.5)(2 - 0.5)] = 533 \tag{8.63}$$

$$\tau_3 = (1 + 0.5)/(3 - 0.5) = 0.600 \tag{8.64}$$

$$\tau_4 = (1 + 0.5)(2 + 0.5)/[(3 - 0.5)(4 - 0.5)] = 0.429 \tag{8.65}$$

a double check of the arithmetic with the `par2lmom()` and `lmomgpa()` functions is now made in example 8–16. In the example, the `lmorph()` function is used for format conversion to show the two *lmomco* L-moment lists (see page 127 and exs. 6–7 – 6–9).

8–16

```
GPApar <- vec2par(c(-6000, 400, -0.5), type="gpa")
lmomGPA <- lmomgpa(GPApar)

str(lmomGPA)
List of 10
 $ L1    : num -5200
 $ L2    : num 533
 $ TAU3  : num 0.6
 $ TAU4  : num 0.429
 $ TAU5  : num 0.333
 $ LCV   : num -0.103
 $ L3    : num 320
```

```
 $ L4    : num 229
 $ L5    : num 178
 $ source: chr "lmomgpa"

str(lmorph(lmomGPA))
List of 6
 $ lambdas : num [1:5] -5200   533   320   229   178
 $ ratios  : num [1:5]     NA -0.103  0.600  0.429  0.333
 $ trim    : num 0
 $ leftrim : NULL
 $ rightrim: NULL
 $ source  : chr "lmorph"
```

◀

8.2.5 Right-Censored Generalized Pareto Distribution

The Right-Censored Generalized Pareto distribution is a right-censored version of the Generalized Pareto distribution having parameters ξ (location), α (scale, $\alpha > 0$), and κ (shape, $\kappa > -1$).

DISTRIBUTION FUNCTIONS

The distribution functions of the Right-Censored Generalized Pareto are the same as those for the Generalized Pareto so reference to Section 8.2.4 is made. The relations by Hosking (1995) between the parameters and the B-type L-moments (through the B-type probability-weighted moments of Section 12.2) of the data under right-tail censoring are

$$\lambda_1^B = \xi + \alpha m_1 \tag{8.66}$$

$$\lambda_2^B = \alpha(m_1 - \quad m_2) \tag{8.67}$$

$$\lambda_3^B = \alpha(m_1 - \ 3m_2 + \ 2m_3) \tag{8.68}$$

$$\lambda_4^B = \alpha(m_1 - \ 6m_2 + 10m_3 - \ 5m_4) \tag{8.69}$$

$$\lambda_5^B = \alpha(m_1 - 10m_2 + 30m_3 - 35m_4 + 14m_5) \tag{8.70}$$

where $m_r = [1-(1-\zeta)^{r+\kappa}]/(r+\kappa)$ and ζ is the right-tail censor fraction or the probability $\Pr[]$ that x is less than the quantile at ζ nonexceedance probability: ($\Pr[x < X(\zeta)]$).

The Right-Censored Generalized Pareto distribution is demonstrated in example 12–1 by an application beginning on page 342, and readers are directed there for details. ◀

8.2.6 Trimmed Generalized Pareto Distribution

Elamir and Seheult (2003) describe a $t_1 = t_2 = 1$ symmetrically trimmed version of the Generalized Pareto distribution. In the *lmomco* package, this distribution is the Trimmed Generalized Pareto distribution. Hosking (2007b) evaluates asymmetrically trimmed versions ($t_2 = 0, 1, 2$) of the Generalized Pareto—these are not considered here. The parameters are estimated by the TL-moments (see Section 6.4), but the distribution functions rely on those for the Generalized Pareto in Section 8.2.4.

DISTRIBUTION FUNCTIONS

The distribution functions of a $t = 1$ symmetrically-trimmed Trimmed Generalized Pareto having parameters ξ (location), α (scale, $\alpha > 0$), and κ (shape, $\kappa > 1$) are defined as for the Generalized Pareto on page 236.

The TL-moments of the Generalized Pareto (Trimmed Generalized Pareto) with symmetrical trimming of smallest and largest values ($\lambda_r^{(1)}$ or $\tau_r^{(1)}$) are

$$\lambda_1^{(1)} = \xi + \frac{\alpha(\kappa + 5)}{(\kappa + 3)(\kappa + 2)} \tag{8.71}$$

$$\lambda_2^{(1)} = \frac{6\alpha}{(\kappa + 4)(\kappa + 3)(\kappa + 2)} \tag{8.72}$$

$$\tau_3^{(1)} = \frac{10(1 - \kappa)}{9\ (\kappa + 5)} \tag{8.73}$$

$$\tau_4^{(1)} = \frac{5(\kappa - 1)(\kappa - 2)}{4(\kappa + 6)(\kappa + 5)} \tag{8.74}$$

The parameters are

$$\kappa = \frac{10 - 45\tau_3^{(1)}}{9\tau_3^{(1)} + 10} \tag{8.75}$$

$$\alpha = \lambda_2^{(1)}(\kappa + 2)(\kappa + 3)(\kappa + 4)/6 \tag{8.76}$$

$$\xi = \lambda_1^{(1)} - \frac{\alpha(\kappa + 5)}{(\kappa + 2)(\kappa + 3)} \tag{8.77}$$

USING R ——————————————— ——————————————— USING R

An example of the Trimmed Generalized Pareto distribution in the context of computing theoretical $t = 1$ TL-moments using the `theoTLmoms()` function is provided in example 6–15 on page 142. The results of that example are compared to analytical results computed by the `lmomTLgpa()` function, which implements eqs. (8.71)–(8.74) for the same TLGPA(10, 5, 0.5) in example 8–17. Comparison between the two examples shows that $\lambda_1^{(1)} = 13.14$, $\lambda_2^{(1)} = 0.762$, $\tau_3^{(1)} = 0.101$, and $\tau_4^{(1)} = 0.0262$ for a TLGPA(10, 5, 0.5).

8–17

```
PARgpa <- vec2par(c(10,5,0.5), type="gpa")
lmr <- lmomTLgpa(PARgpa); print(lmr)
$lambdas
[1] 13.14285714  0.76190476  0.07696008  0.01998002
$ratios
[1] 0.00000000 0.05797101 0.10101010 0.02622378
$trim
[1] 1
$source
[1] "lmomTLgpa"
```

◀

The robustness of the TL-moments in the presence of some contrived contamination by outliers to a sample is now explored. In example 8–18, a GPA(1000, 1000, −0.5) is specified and a sample of size $n = 30$ is chosen for evaluation. The evaluation will use 1,000 simulations. Each simulated sample is contaminated by shifting the decimal of the largest value to the left one place. The output of a simulation run with focus on the 99th percentile $F = 0.99$ is shown and indicates that both techniques underestimate considerably in the right tail of the distribution. However, the bias using the TL-moments is about half compared to the L-moments [$(-4972)/(-8603) = 0.58$].

```
PARgpa <- vec2par(c(1000,1000,-0.5), type="gpa")
e1 <- e2 <- vector(mode = "numeric")
n  <- 30;  nsim <- 1000
F  <- 0.99
QF <- quagpa(F,PARgpa)
for(i in seq(1,nsim)) {
   Q <- sort(rlmomco(n,PARgpa)) # generate random GPA values
   # contaminate the sample, shift largest by order of magnitude
   Q[n] <- Q[n]/10   # shift decimal to left of largest value
    lmr <- TLmoms(Q) # technically same as lmr <- lmoms(Q)
   TLmr <- TLmoms(Q, trim=1) # TL-moments for trim = 1

   # Parameter estimation via method of L-moments
   PARgpa1 <- pargpa(lmr)
   PARgpa2 <- parTLgpa(TLmr)

   # Now estimate the 99th percentile
   QF1 <- quagpa(F,PARgpa1)
   QF2 <- quagpa(F,PARgpa2)

   e1[i] <- QF1 - QF
   e2[i] <- QF2 - QF # storing the results
}
b1 <- round(mean(e1)); b2 <- round(mean(e2))
cat(c(   "Bias␣using␣␣L-moments=",b1,
      "␣␣␣Bias␣using␣TL-moments=",b2,"\n"))
Bias using  L-moments= -8603    Bias using TL-moments= -4972
```

The robustness of the TL-moments is shown. Unfortunately in practice, knowledge of the type and degree of contamination by outliers that a sample might be exposed to mostly is unknown. In circumstances in which decimal shifts are likely, including situations involving transcription or even optical character recognition errors, the TL-moments might offer protection against such errors in the data values or protection against either low or high outliers (or both). ◀

8.2.7 Pearson Type III Distribution

The Pearson Type III distribution is a three-parameter distribution that is a widely used probability distribution in the hydrologic sciences. It is a particularly interesting distribution for the purposes of this dissertation because the product moments are explicit

parameters. This fact greatly simplifies comparisons between parameter estimates from product moments and L-moments.

DISTRIBUTION FUNCTIONS

The distribution functions of the Pearson Type III having parameters μ (mean, location), σ (standard deviation, scale), and γ (skew, shape), but expressed with alternative parameters ξ (location), β (scale, $\beta > 0$), and α (shape, $\alpha > 0$) are

$$f(x) = \begin{cases} \beta^{-\alpha}(x-\xi)^{\alpha-1}\exp(-Y_1)/\Gamma(\alpha) & \text{if } \gamma > 0 \\ \beta^{-\alpha}(\xi-x)^{\alpha-1}\exp(-Y_2)/\Gamma(\alpha) & \text{if } \gamma < 0 \\ \varphi((x-\mu)/\sigma) & \text{if } \gamma = 0 \end{cases} \tag{8.78}$$

$$F(x) = \begin{cases} G(\alpha, Y_1)/\Gamma(\alpha) & \text{if } \gamma > 0 \\ 1 - G(\alpha, Y_2)/\Gamma(\alpha) & \text{if } \gamma < 0 \\ \Phi((x-\mu)/\sigma) & \text{if } \gamma = 0 \end{cases} \tag{8.79}$$

$x(F)$ has no explicit analytical form

where

$$Y_1 = (x-\xi)/\beta \quad \text{and} \quad Y_2 = (\xi - x)/\beta \tag{8.80}$$

and where $G(a, b)$ is the incomplete gamma function, $\Gamma(a)$ is the complete gamma function, $\varphi(a)$ is the PDF of the Normal distribution, $\Phi(a)$ is the CDF of the Normal distribution. The relations between the product moments and the three alternative parameters for $\gamma \neq 0$ are

$$\alpha = 4/\gamma^2 \tag{8.81}$$

$$\beta = \sigma|\gamma|/2 \tag{8.82}$$

$$\xi = \mu - 2\sigma/\gamma \tag{8.83}$$

The **incomplete gamma function** $G(a, b)$ is

$$G(a, b) = \int_0^b t^{(a-1)}\exp(-t)\,\mathrm{d}t \tag{8.84}$$

and the **complete gamma function** $\Gamma(a)$ is

$$\Gamma(a) = \int_0^\infty t^{(a-1)} \exp(-t)\, \mathrm{d}t \tag{8.85}$$

The particular parameterization of the Pearson Type III shown is useful. For hydrologic data, more common situations of positive skewness (right-tail heavy), less common negative skewness (left-tail heavy), and zero skewness (Normal distribution) are accommodated. The ranges of the distribution are

$$\xi \leq x < \infty \quad \text{if } \gamma > 0 \tag{8.86}$$

$$-\infty < x < \infty \quad \text{if } \gamma = 0 \text{ (Normal distribution)} \tag{8.87}$$

$$-\infty < x \leq \xi \quad \text{if } \gamma < 0 \tag{8.88}$$

The L-moments are

$$\lambda_1 = \xi + \alpha\beta \tag{8.89}$$

$$\lambda_2 = \pi^{-1/2}\beta\, \Gamma(\alpha + 1/2)/\Gamma(\alpha), \text{ and} \tag{8.90}$$

$$\tau_3 = 6\, I_{1/3}(\alpha, 2\alpha) - 3 \tag{8.91}$$

where $I_x(p, q)$ denotes the **incomplete Beta function ratio**, **regularized incomplete Beta function**, **regularized Beta function** for short

$$I_x(p, q) = \frac{\Gamma(p+q)}{\Gamma(p)\,\Gamma(q)} \int_0^x t^{p-1}(1-t)^{q-1}\mathrm{d}t \tag{8.92}$$

which also is the same as the CDF of the Beta distribution $B(x, p, q)$.

The parameters have quasi-analytical solutions (Hosking and Wallis, 1997, p. 202). The following approximations have a relative accuracy better than 5×10^{-5} for all values of α. If $0 < |\tau_3| < 1/3$, let $z = 3\pi\tau_3^2$ and use minimax approximations by Hosking (1996b) for α

$$\alpha \approx \frac{1 + 0.2906z}{z + 0.1882z^2 + 0.0442z^3} \tag{8.93}$$

if $1/3 \leq |\tau_3| < 1$, let $z = 1 - |\tau_3|$ and use

$$\alpha \approx \frac{0.36067z - 0.59567z^2 + 0.25361z^3}{1 - 2.78862z + 2.56096z^2 - 0.77045z^3} \tag{8.94}$$

The parameters in terms of α and the L-moments are

$$\gamma = \text{sign}(\tau_3)\frac{2}{\sqrt{\alpha}} \tag{8.95}$$

$$\sigma = \frac{\lambda_2 \Gamma(\alpha)\sqrt{\alpha\pi}}{\Gamma(\alpha + 0.5)} \tag{8.96}$$

$$\mu = \lambda_1 \tag{8.97}$$

Finally, the **log-Pearson Type III distribution** is a Pearson Type III fit to the logarithms of a random variable.

USING R

Daily mean streamflow for U.S. Geological Survey streamflow-gaging station 06766000 Platte River at Brady, Nebraska is available in the `USGSsta06766000dvs` data for the period from 03/01/1939 to 09/30/1991. The flow-duration curve is a plot of the sorted daily mean streamflow values plotting against nonexceedance probability computed by plotting positions. Example 8–19 loads in these data and plots the time series of streamflow as measured daily.

8–19

```
data(USGSsta06766000dvs) # from lmomco package
flow <- USGSsta06766000dvs$X01_00060_00003
#pdf("fdc1pe3.pdf")
plot(flow, type="l", xlab="DAY", ylab="FLOW, IN FT^3/S")
#dev.off()
```

Subsequently, example 8–20 fits the Pearson Type III distribution by the method of L-moments. The fitted distribution is then plotted on the empirical distribution. Unlike other examples herein, the empirical distribution is represented by a line instead of points.

Specific judgements of Pearson Type III fit are not made for these daily mean streamflows with the exception that there are considerable differences in the far-right (drought) tail. The data trail off towards zero (no-flow), which is otherwise not representable on the logarithmic scale. The Pearson Type III distribution would provide for, that is, estimate, an order of magnitude or more streamflow than the data show or suggest for $F \ll 0.1$ for drought. Hence, the Pearson Type III greatly overestimates the availability of a natural resource under drought conditions in this particular example.

```
lmr <- lmoms(flow) # compute L-moments
# now compute the Pearson III parameters
PE3.par <- lmom2par(lmr, type="pe3")
PP <- pp(flow) # compute plotting positions
#pdf("fdc2pe3.pdf")
sflow <- sort(flow)
plot(PP, log10(sflow), type="l", lty=2,
     xlab="NONEXCEEDANCE␣PROBABILITY",
     ylab="LOG10(FLOW),␣IN␣FT^3/S")
lines(PP,log10(par2qua(PP,PE3.par)))
legend(0,4, c("DATA", "Pearson␣Type␣III␣distribution"),
            lty=c(2,1))
#dev.off()
```

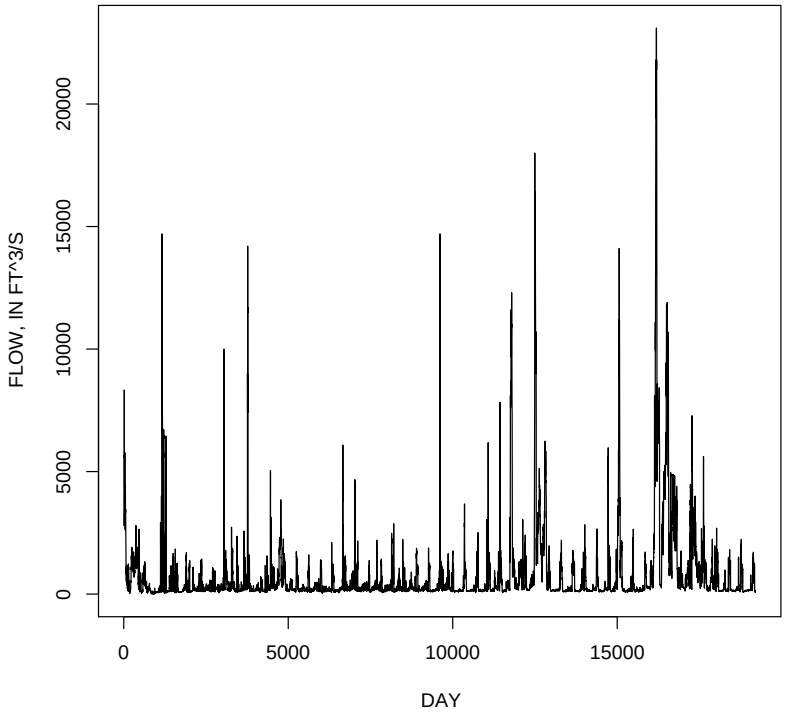

Figure 8.9. Time series by day of daily mean streamflow for U.S. Geological Survey streamflow-gaging station 06766000 Platte River at Brady, Nebraska from example 8–19

◀

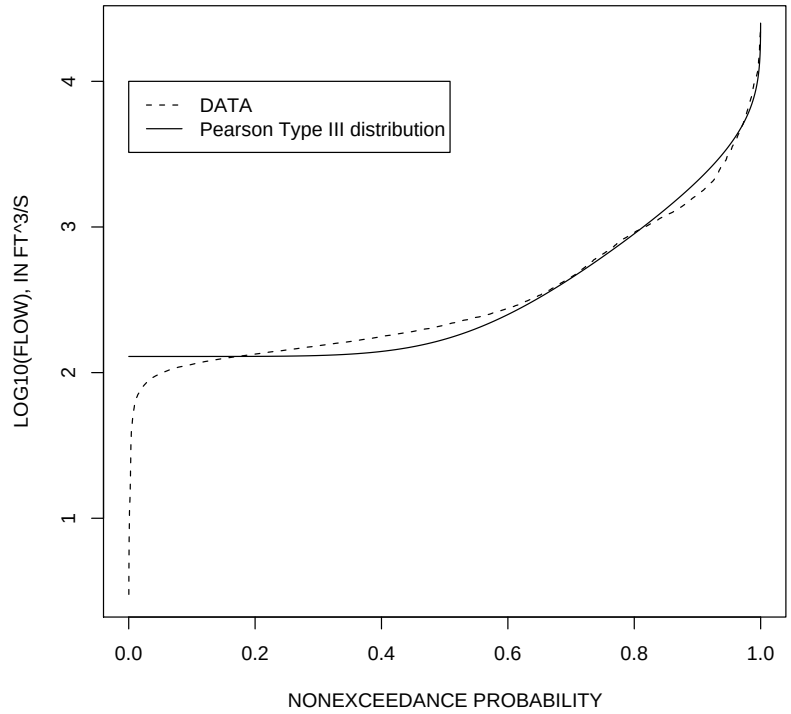

Figure 8.10. Flow-duration curve of daily mean streamflow for U.S. Geological Survey streamflow-gaging station 06766000 Platte River at Brady, Nebraska from example 8–20

8.2.8 Weibull Distribution

The Weibull distribution as implemented by *lmomco* is a three-parameter version, whereas, the built-in R version has two parameters. The three-parameter version offers additional flexibility. The Weibull is comprehensively summarized in Rinne (2008).

DISTRIBUTION FUNCTIONS

The distribution functions of the Weibull having parameters ζ (location), β (scale), and δ (shape) are

$$f(x) = \delta Y^{\delta-1} \exp(-Y^{\delta})/\beta \tag{8.98}$$

$$F(x) = 1 - \exp(-Y^{\delta}) \tag{8.99}$$

$$x(F) = \beta[-\log(1-F)]^{1/\delta} - \zeta \tag{8.100}$$

where

$$Y = (x - \zeta)/\beta \tag{8.101}$$

The range of the distribution is

$$\zeta \leq x < \infty \tag{8.102}$$

The Weibull distribution is a **Reverse Generalized Extreme Value distribution**. As result, the Generalized Extreme Value algorithms are used for implementation of the Weibull in *lmomco*. The relations between the Generalized Extreme Value parameters (ξ, α, κ) from Hosking and Wallis (1997) are

$$\kappa = 1/\delta \tag{8.103}$$

$$\alpha = \beta/\delta \tag{8.104}$$

$$\xi = \zeta - \beta \tag{8.105}$$

The Weibull distribution is popular in the analysis of lifetimes in which case x is time t. If $\delta < 1$, then as t increases the failure rate decreases—this condition is known as "infant mortality." If $\delta > 1$, then as t increases the failure rate increases—this condition is known as "wear out." If $\delta = 1$, the Weibull $\rightarrow$ Exponential distribution, and the failure rate is constant: $h(x) = 1/\beta$ (see example 2–4).

USING R

In the R environment, the CDF of the Weibull distribution is `pweibull()`. Given an *lmomco* parameter list (see page 163 and ex. 7–1) for the Weibull distribution as `para`, the equivalent R syntax is `pweibull(a, b, c)` for `a=x+para$para[1]`, `b=para$para[3]`, and `c=para$para[2]`. For the current implementation for the *lmomco* package, the Reverse Generalized Extreme Value distribution is used `1-cdfgev(-x,gevpara)` where the `gevpara` holds the converted Weibull parameters.

The Weibull and Generalized Extreme Value distribution are related. A comparison is made in example 8–21 of the fits between the distributions to the number of Internal Revenue Service refunds by state in the data `IRSrefunds.by.state` for fiscal year 2006 (`http://www.irs.gov/taxstats/article/0,,id=168593,00.html` accessed in December 2007). The data are loaded, `attach()`ed, and `sort()`ed. The plotting positions are computed by `pp()`, and the sample L-moments are computed by `lmoms()`. The parameters for the Weibull and Generalized Extreme Value distributions are respectively computed by the `parwei()` and `pargev()` functions. The `layout()` function sets up

two plots. The high-level `check.pdf()` function plots the two PDFs. The `mtext()` function renders the respective plot titles. The plots are shown in figure 8.11.

8–21

```
data(IRSrefunds.by.state) # from lmomco package
attach(IRSrefunds.by.state)
REFUNDS <- sort(REFUNDS); PP <- pp(REFUNDS)
lmr <- lmoms(REFUNDS)
PARwei <- parwei(lmr); PARgev <- pargev(lmr)
#pdf("weigevpdfA.pdf")
layout(matrix(1:2, ncol=1)) # setup to plots
check.pdf(pdfwei,PARwei, plot=TRUE) # function returns unity
mtext("Weibull␣distribution") # provide the plot with a title
check.pdf(pdfgev,PARgev, plot=TRUE) # function returns unity
mtext("Generalized␣Extreme␣Value␣distribution") # another title
#dev.off()
```

Although each is fit to the same L-moments, the two PDFs shown in figure 8.11 appear quite different. A logical line of inquiry is: How different do the fitted distributions look compared to the empirical distribution? This question is answered in the next example. ◀

Example 8–22 and resulting plot in figure 8.12 make the comparison of the empirical distribution to the fitted CDFs. Subsequently, a vector of nonexceedance probabilities is created and set into the variable `F`. The intersection of the quantiles for the two distributions is created by sorting the values returned by the `quawei()` and `quagev()` functions.

8–22

```
F <- seq(0.05,0.99, by=0.001)
x <- sort(c(quawei(F,PARwei), quagev(F,PARgev)))
#pdf("weigevcdfA.pdf")
plot(log10(x), qnorm(cdfwei(x,PARwei)), type="l", lwd=3,
     xlab="LOG10␣OF␣NUMBER␣OF␣REFUNDS␣BY␣STATE",
     ylab="STANDARD␣NORMAL␣DEVIATE")
lines(log10(x), qnorm(cdfgev(x,PARgev)))
points(log10(REFUNDS), qnorm(PP), cex=2)
#dev.off()
```

Next, the plot is drawn in example 8–22. The `cdfwei()` and `cdfgev()` functions compute the F values for the empirical distribution. The vertical axis is a probability axis by casting the F values into standard normal deviates using the `qnorm()` function. The base-10 logarithms of the quantiles `x` are used to reduce visual curvature of the plot, note however, that the distributions are not fit to the logarithms of the data. The Weibull (thick line) and the Generalized Extreme Value (thin line) distributions are plotted. The

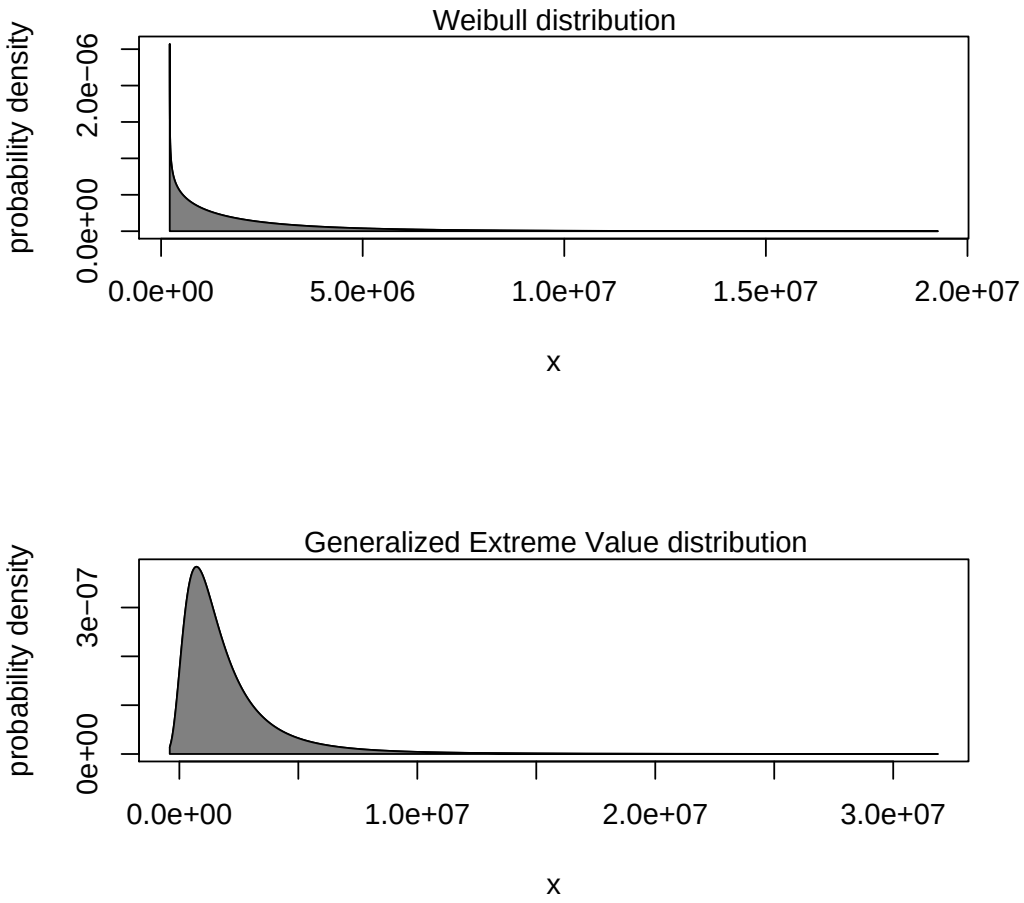

Figure 8.11. Comparison of probability density functions for Weibull and Generalized Extreme Value distributions fit to same L-moments of number of Internal Revenue Service refunds by state from example 8–21

empirical distribution finally is plotted by the `points()` function. The data points are drawn unnecessarily large for demonstration of the `cex` argument, which scales the points larger or smaller depending on the `cex` argument value.

Several observations of figure 8.12 can be made. Because each is fit to λ_1, λ_2, and τ_3, both distributions generally mimic the data between -1 and 1 standard deviations. Substantial differences exist primarily in the tails. Neither distribution exhibits quite enough straightness in the right tail as suggested by the data and the respective plotting positions. The figure shows that the largest four values have been underestimated. For the left tail, the Weibull distribution has the preferable fit compared to that of the Generalized

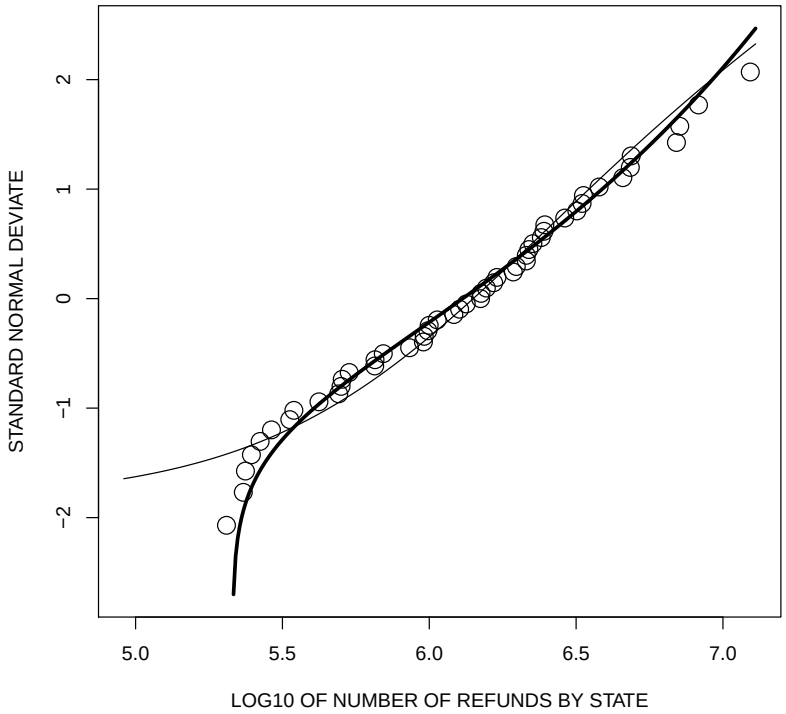

Figure 8.12. Comparison of cumulative probability functions for Weibull (thick line) and Generalized Extreme Value (thin line) distributions fit to same L-moments and empirical distribution of number of Internal Revenue Service refunds by state from example 8–22

Extreme Value. Whether the tail differences are important or have ramifications for later interpretations is a discipline-specific problem. ◀

8.3 Three-Parameter Distributions not yet supported by the *lmomco* Package

This section provides a summary of selected three-parameter distributions with existing L-moment support that are currently (May 2011) not implemented in the *lmomco* package. These distributions are discussed and included in this dissertation because they represent the current (May 2011) front line of *lmomco* development regarding three-parameter distributions.

8.3.1 Polynomial Density-Quantile3 Distribution

The Polynomial Density-Quantile3 distribution was developed by Hosking (2007a) as part of an investigation into derived distributions from the maximization of the entropy (or information content) of a distribution subject to constraints. In particular the Polynomial Density-Quantile3 is the distribution that has maximum entropy resulting from the maximization conditional on having specified values for the L-moments of λ_1, λ_2, and λ_3 are specified.

DISTRIBUTION FUNCTIONS

The distribution functions of the Polynomial Density-Quantile3 having parameters ξ (location), α (scale, $\alpha > 0$), and κ (shape, $-1 < \kappa < -1$) are

$$\begin{aligned} &f(x) \text{ has no explicit analytical form} \\ &F(x) \text{ has no explicit analytical form} \\ &x(F) = \xi + \alpha\left[\log\left(\frac{F}{1-F}\right) + \kappa\log\left(\frac{[1-\kappa(2F-1)^2]}{4F(1-F)}\right)\right] \end{aligned} \tag{8.106}$$

The range of the distribution is

$$-\infty < x < \infty \tag{8.107}$$

The L-moments are

$$\lambda_1 = \xi + \alpha[(1+\kappa)\log(1+\kappa) - (1-\kappa)\log(1-\kappa) - \kappa\log(4)] \tag{8.108}$$

$$\lambda_2 = \frac{\alpha(1-\kappa^2)}{(1-\kappa\tau_3)} \tag{8.109}$$

$$\tau_3 = \frac{1}{\kappa} - \frac{1}{\operatorname{arctanh}(\kappa)} \tag{8.110}$$

$$\tau_4 = (5\tau_3/\kappa - 1)/4 \tag{8.111}$$

The parameter κ requires numerical solution of eq. (8.110), and the other parameters are

$$\xi = \lambda_1 - \alpha[(1+\kappa)\log(1+\kappa) - (1-\kappa)\log(1-\kappa) - \kappa\log(4)] \tag{8.112}$$

$$\alpha = \frac{\lambda_2(1-\kappa\tau_3)}{(1-\kappa^2)} \tag{8.113}$$

The Polynomial Density-Quantile3 distribution has larger τ_4^{pdq3} than τ_4^{glo} of the Generalized Logistic distribution. For example, for a sample distribution having $\hat{\tau}_3 = 0.1795$ has, by eq. (8.25), a $\tau_4^{\text{glo}} = 0.1935$, and by eq. (8.111), a $\tau_4^{\text{pdq3}} = 0.1988$ for a Polynomial Density-Quantile3, by eq. (8.110), having $\kappa = 0.5$.

8.3.2 Polynomial Density-Quantile4 Distribution

The Polynomial Density-Quantile4 distribution was developed by Hosking (2007a) and is a symmetrical distribution that has maximum entropy conditional on having specified values for the L-moments of λ_1, λ_2, and λ_4 are specified.

DISTRIBUTION FUNCTIONS

The distribution functions of the Polynomial Density-Quantile4 having parameters ξ (location), α (scale, $\alpha > 0$), and κ (shape, $0 < \kappa < 1$) are

$$\begin{aligned} &f(x) \text{ has no explicit analytical form} \\ &F(x) \text{ has no explicit analytical form} \\ &x(F) = \xi + \alpha\left[\log\left(\frac{F}{1-F}\right) - 2\kappa \operatorname{arctanh}(\kappa[2F-1])\right] \end{aligned} \tag{8.114}$$

and for κ (shape, $-\infty < \kappa < 0$) are

$$\begin{aligned} &f(x) \text{ has no explicit analytical form} \\ &F(x) \text{ has no explicit analytical form} \\ &x(F) = \xi + \alpha\left[\log\left(\frac{F}{1-F}\right) + 2\kappa \arctan(\kappa[2F-1])\right] \end{aligned} \tag{8.115}$$

The range of the distribution is

$$-\infty < x < \infty \tag{8.116}$$

The L-moments are

$$\lambda_1 = \xi \tag{8.117}$$

$$\lambda_2 = \begin{cases} \alpha(1-\kappa^2)\text{arctanh}(\kappa)/\kappa & \text{if } \kappa > 0 \\ \alpha(1+\kappa^2)\ \ \arctan(\kappa)/\kappa & \text{if } \kappa < 0 \end{cases} \tag{8.118}$$

$$\tau_3 = 0 \tag{8.119}$$

$$\tau_4 = \begin{cases} -\frac{1}{4} + \frac{4}{5\kappa}\left(\frac{1}{\kappa} - \frac{1}{\text{arctanh}(\kappa)}\right) & \text{if } \frac{1}{6} \leq \tau_4 < 1 \\ -\frac{1}{4} - \frac{4}{5\kappa}\left(\frac{1}{\kappa} - \frac{1}{\arctan(\kappa)}\right) & \text{if } -\frac{1}{4} < \tau_4 < \frac{1}{6} \end{cases} \tag{8.120}$$

The parameter κ requires numerical solution of eq. (8.120), and the other parameters are

$$\xi = \lambda_1 \tag{8.121}$$

$$\alpha = \begin{cases} \lambda_2\kappa/[(1-\kappa^2)\text{arctanh}(\kappa)] & \text{if } \kappa > 0 \\ \lambda_2\kappa/[(1+\kappa^2)\ \ \arctan(\kappa)] & \text{if } \kappa < 0 \end{cases} \tag{8.122}$$

The Polynomial Density-Quantile4 distribution is symmetrical about ξ and is quite similar to the Normal but exhibits heavier tails. Using the standard Normal for reference, Hosking (2007a, p. 2883) reports that "PDF and QDF functions [of the two distributions] are very similar except in the extreme tails." In particular, the distributions differ by less than 0.03 in the quantiles for $0.011 < F < 0.989$. The tails of the Polynomial Density-Quantile4 are exponentially decreasing and the distribution could be useful in distributional analysis with data exhibiting similar tail characteristics.

8.3.3 Student t (3-parameter) Distribution

The Student t (3-parameter) distribution is used by Hosking (1999) for the modeling of IBM stock prices. The L-moments of the ST3($\xi = 0, \alpha = 1, \nu = 2$) are considered by Jones (2002, p. 48).

DISTRIBUTION FUNCTIONS

The distribution functions of the Student t (3-parameter) having parameters ξ (location), α (scale, $\alpha > 0$), and ν (degrees of freedom, shape, $\nu > 1$) are

$$f(x) = \frac{\Gamma(\frac{1}{2} + \frac{1}{2}\nu)}{\alpha\nu^{1/2}\,\Gamma(\frac{1}{2})\Gamma(\frac{1}{2}\nu)}(1 + t^2/\nu)^{-(\nu+1)/2} \tag{8.123}$$

$$F(x) \text{ has no explicit analytical form}$$

$$x(F) \text{ has no explicit analytical form}$$

where

$$t = \frac{(x - \xi)}{\alpha} \tag{8.124}$$

The range of the distribution is

$$-\infty < x < \infty \tag{8.125}$$

The L-moments are

$$\lambda_1 = \xi \tag{8.126}$$

$$\lambda_2 = 2^{6-4\nu}\pi\alpha\nu^{1/2}\,\Gamma(2\nu - 2)/[\Gamma(\tfrac{1}{2}\nu)]^4 \tag{8.127}$$

$$\tau_3 = 0 \tag{8.128}$$

$$\tau_4 = \frac{15}{2}\frac{\Gamma(\nu)}{\Gamma(\frac{1}{2})\Gamma(\nu - \frac{1}{2})}\int_0^1 \frac{(1-x)^{\nu-3/2}[I_x(\frac{1}{2}, \frac{1}{2}\nu)]^2}{\sqrt{x}}\,\mathrm{d}x - \frac{3}{2} \tag{8.129}$$

where $I_x(\frac{1}{2}, \frac{1}{2}\nu)$ is the CDF of the Beta distribution. Hosking (1999) does not provide details as to the definition of $I_x(\frac{1}{2}, \frac{1}{2}\nu)$.[1] In Hosking and Wallis (1997, p. 201), $I_x(p, q)$ is the incomplete Beta function ratio, which is eq. (8.92) of this dissertation; numerical experiments, not presented here, seem to confirm that eq. (8.129) using the CDF of the Beta distribution is correct.

The parameters require numerical methods. Hosking (1999) reports that a one-to-one relation between τ_4 and ν exists and a table could be computed and ν found by linear

[1] This is quite unusual for Jonathan and such ambiguity is surprising. This author (Asquith) is a fan of Jonathan's work and eagerly awaits the discovery of each new reference by him and commends Jonathan for a long history of well written and documented articles that are especially approachable for non-mathematicians.

interpolation.[2] The parameters ξ and α readily follow by

$$\xi = \lambda_1 \tag{8.130}$$

$$\alpha = \frac{\lambda_2}{2^{6-4\nu}\pi\nu^{1/2}\,\Gamma(2\nu-2)/[\Gamma(\frac{1}{2}\nu)]^4} \tag{8.131}$$

8.4 Summary

Three-parameter distributions often are preferable to two-parameter distributions in the application of distributional analysis where the skewness of the data is expected to be different from zero (asymmetrically distributed data about the mean or different from that of other distributions). The 8 three-parameter distributions formally considered in this chapter are fit to the first three L-moments of the data. Both the *lmomco* and *lmom* packages provide support for many three-parameter distributions. The 22 examples demonstrate a variety of applications and generally have expanded complexity relative to the examples in Chapter 7. Further, additional comparisons between product moments and L-moments to those in that chapter also are made.

- The examples for the Generalized Extreme Value distribution consider the distribution of annual wind speed data reported by Hosking and Wallis (1997) in which the parameters of the Generalized Extreme Value were already provided for three cities in Texas. A table of selected quantiles of the three Generalized Extreme Value is provided. The examples also create a small application using the Generalized Extreme Value to generate a quantile-quantile plot (expressed in annual recurrence interval) of some annual peak streamflow data in Texas contained within the *lmomco* package.
- The examples for the Generalized Logistic distribution consider the distribution of 1-hour annual maxima of rainfall for a county in Texas based on Generalized Logistic parameters provided by Asquith (1998). CDF and QDF plots are created and the L-moments of the given parameters computed.
- The examples for the Generalized Normal distribution create three representations of the PDF for three ensembles of L-moments in order to demonstrate the effect of τ_3 on

[2] This is a similar method of parameter estimation as the author (Asquith) has implemented for the Rice distribution in the *lmomco* package.

the fitted distribution. The examples continue with a return to annual peak streamflow data considered in Chapter 2. The Generalized Normal and log-Normal are fit to the L-moments (real space) and product moments (logarithms) of the annual peak streamflow data and a QDF plot along with the sample data is created. The Generalized Normal provides a preferable fit. The examples continue with an exploration of the sampling properties of the Generalized Normal for a log-Normal parent. The statistical simulations show that the product moments might perform better than L-moments when a parent is truly log-Normal. Finally, an example is provided comparing the Generalized Normal to the log-Normal3 and various lower limits of the log-Normal3 are considered.

- The examples for the Generalized Pareto distribution show manual computations of the L-moments from a given set of parameters. The computations are shown because the Generalized Pareto has some readily used analytical solutions for the parameters but also are more complex than the elementary solutions for the Exponential distribution.
- The examples for the Trimmed Generalized Pareto distribution compute some TL-moments by analytical and numerical methods and equivalency is shown. The robustness of the TL-moments in the presence of contrived contamination is explored with the focus on the $F = 0.99$ quantile. The bias of the TL-moments for fitting and estimation of the quantile is shown to be considerably less than that from use of L-moments.
- No examples for the Right-Censored Generalized Pareto distribution are provided in this chapter.
- The examples for the Pearson Type III distribution involve the exploration of the flow-duration curve for some daily mean streamflow data in Nebraska. A comparison between the fitted distribution and the data is made along with several plots. Finally, example computations comparing the Pearson Type III to the Normal distribution are made.
- The examples for the Weibull distribution consider some income tax data for the United States. PDFs of the Weibull and Generalized Extreme Value are created by fitting to the L-moments and plotted. Finally, CDF plots are created by computation of appropriate distribution ranges using selected nonexceedance probabilities and

QDF functions. The examples show that the Weibull is preferable to the Generalized Extreme Value for these data.

Finally, the chapter concludes with a summary of selected three-parameter distributions with existing L-moment derivations that are not yet (as of May 2011) implemented within the *lmomco* package.

Chapter 9

L-moments of Four- and More Parameter Univariate Distributions

In this chapter, I present continued discussion of distribution support by L-moments in several R packages, but focus remains clearly on the *lmomco* package. The chapter provides a distribution-by-distribution discussion of mathematics, features, parameters, and L-moments of four- and more parameter distributions. These distributions are not as well known as those in the two previous chapter but are remarkablely useful in L-moment applications. In general, the mathematics of the distributions are even more complex than seen in the previous chapter. Readers possessing considerable familiarity with statistics and R are likely to generally browse as needed through the distributions. Other readers are encouraged to at least review this chapter with the mindset that periodic return likely will be made. Because of the ubiquitous two- and three-parameter distribution in practice, this chapter might be of secondary importance to readers pursuing mastery of distributional analysis with L-moment statistics using R.

9.1 Introduction

Distributions having four- and more parameters are described in this chapter. These distributions are less well-known than many of the other lower-order (lower-parameter) distributions described in the previous two chapters. However, it will be seen that four- and more parameter distributions are very attractive for mimicking the geometry of heavy-tailed distributions.

The four- and more parameter distributions are fit to the mean, scale, shape (skewness), and kurtosis (seen simply as a higher order measure of shape) of a data set. For sufficiently large sample sizes (vagueness on how large is "large" is intentional), sample L-moments can reliably estimate distribution shape through $\hat{\tau}_4$ and even distribution shape by the

fifth L-moment through $\hat{\tau}_5$. For some types of distributional analyses, four- and more parameter distributions are flexible and might provide useful fits that are not attainable by lower-order distributions.

The flexibility is particularly useful in the study three-parameter distributions because four- and more parameter distributions, being simultaneously fit to τ_3, τ_4, and higher, can mimic the shapes of many three-parameter distributions. The flexibility does come at the price of having to estimate additional moments at fourth or fifth order. It is important to note that, as a general rule, parameter estimation for four- and more parameter distributions is considerably more complex than lower parameter distributions—numerical methods for minimization or root-solving generally are required.

Final notes about the source of material and in particular the mathematics of the four- and more parameter distributions is needed. Unless otherwise stated, the material is heavily based on Asquith (2007), Hosking (1996b), Hosking (1994), Hosking and Wallis (1997), Karian and Dudewicz (2000), and Stedinger and others (1993). These and additional references are provided on a distribution-specific basis.

9.2 Four- and More Parameter Distributions of the *lmomco* Package

9.2.1 Kappa Distribution

A particularly useful distribution for both applied and research investigations is the four-parameter Kappa distribution throughly documented by Hosking (1994) and used in several hydrometeorologic investigations (Hosking and Wallis, 1993, 1997; Parida, 1999; Dupuis and Winchester, 2001; Asquith and others, 2006). The Kappa distribution of today is a generalization of a three-parameter version introduced at the end of a paper on precipitation amount modeling by Mielke (1973) who states that "further investigation" is needed.[1] Because the Kappa distribution has four parameters, it can acquire a wider range of shapes than two- or three-parameter distributions such as the Normal (two parameter) or Generalized Extreme Value (three parameter) distributions. The Kappa parameter space as measured by the pairing $\{\tau_3, \tau_4\}$ is large enough[2] to make it especially useful for

[1] It seems that Hosking (1994) was the first to take up the mantle of four-parameter Kappa investigation with vigor.

[2] Although not as large as the Generalized Lambda and Wakeby distributions described later in this chapter.

many types of distributional analyses. Stress is needed that sample sizes should be sufficiently large for reliable estimation of $\hat{\tau}_4$. Finally, the Kappa is attractive because parameter estimation for the Kappa is much more straightforward than for the Generalized Lambda distribution.

The Kappa distribution is of particular interest to L-moment practitioners because with $h = -1$, the distribution is the Generalized Logistic distribution; with $h = 0$, it is the Generalized Extreme Value; and with $h = 1$, it is the Generalized Pareto. Because of the Kappa's parentage over these collectively popular distributions and the large range of $\{\tau_3, \tau_4\}$-parameter space attained, the distribution is very useful in simulation studies to assess performance of the Generalized Logistic, Generalized Extreme Value (and Gumbel), Generalized Normal, and Generalized Pareto distributions.

DISTRIBUTION FUNCTIONS

The distribution functions of the Kappa having parameters ξ (location), α (scale), κ (shape1), h (shape2) subject to the constraint that $h \geq 0$ and $\kappa > -1$ or if $h < 0$ and $-1 < \kappa < -1/h$ are

$$f(x) = \alpha^{-1}[1 - \kappa(x - \xi)/\alpha]^{1/k-1} \times F^{1-h} \tag{9.1}$$

$$F(x) = [1 - h(1 - \kappa(x - \xi)/\alpha)^{1/k}]^{1/h} \tag{9.2}$$

$$x(F) = \xi + \frac{\alpha}{\kappa}\left[1 - \left(\frac{1 - F^h}{h}\right)^{\kappa}\right] \tag{9.3}$$

The ranges of the distribution $x_L \leq x \leq x_U$ are

$$x_L = \begin{cases} \xi + \alpha(1 - h^{-\kappa})/\kappa & \text{if } h > 0 \\ \xi + \alpha/\kappa & \text{if } h \leq 0 \text{ and } \kappa < 0 \\ -\infty & \text{if } h \leq 0 \text{ and } \kappa \geq 0 \end{cases} \tag{9.4}$$

$$x_U = \begin{cases} \xi + \alpha/\kappa & \text{if } \kappa > 0 \\ \infty & \text{if } \kappa \leq 0 \end{cases} \tag{9.5}$$

The L-moments are

$$\lambda_1 = \xi + \alpha(1 - g_1)/\kappa \tag{9.6}$$

$$\lambda_2 = \alpha(g_1 - g_2)/\kappa \tag{9.7}$$

$$\tau_3 = (-g_1 + 3g_2 - \ 2g_3)/(g_1 - g_2) \tag{9.8}$$

$$\tau_4 = (-g_1 + 6g_2 - 10g_3 + 5g_4)/(g_1 - g_2) \tag{9.9}$$

where g_r is

$$g_r = \begin{cases} r\,\Gamma(1+\kappa)\,\Gamma(r/h)/[h^{1+\kappa}\,\Gamma(1+\kappa+r/h)] & \text{if } h > 0 \\ \\ r\,\Gamma(1+\kappa)\,\Gamma(-\kappa-r/h)/[(-h)^{1+\kappa}\,\Gamma(1-r/h)] & \text{if } h < 0 \end{cases} \tag{9.10}$$

where $\Gamma(a)$ is the complete gamma function that is shown in eq. (8.85).

There are no simple expressions for the parameters in terms of the L-moments. Numerical methods must be used. Algorithmically, the condition of $\kappa = 0$ or $h = 0$ for the distribution functions is accommodated by the following limiting property of logarithms and exponents

$$\exp(a) = \lim_{b \to 0}(1 + ab)^{1/b} \tag{9.11}$$

The availability, flexibility, and suitability of the Kappa in distributional analysis has generated additional interest in parameter estimation methods. Although relatively straight forward equations are involved for the method of L-moments, iterative solutions are still required to simultaneously solve for κ and h and then ξ and α and convergence "sometimes fails" (Park and Park, 2002, p. 65) or is not available by Hosking's algorithms (Hosking, 1996b) because τ_4 lies above the Generalized Logistic line (see Chapter 10). Park and Park (2002) explore the maximum likelihood method with a penalty method for parameter estimation. The authors present two examples of parameter estimation by methods of L-moments and maximum likelihood. Park and Park, (p. 68) state that "maximum likelihood [parameter] estimates can always be calculated" and that "more extensive study is needed for sophisticated comparison [between] the two estimation methods [of L-moments and maximum likelihood]." The Kappa parameter estimation is further considered by Singh and Deng (2003) in which an entropy-based parameter estimation method is compared to the method of L-moments. The results of their study are ambiguous in that these two estimation methods perform well or are "comparable" (Singh and

Deng, 2003, p. 90), but use of L-moments is far simpler than entropy. Singh and Deng conclude that "the combinations of the two methods can further improve parameter estimation [for the Kappa]."

USING R ——— ——— USING R

The Kappa distribution is demonstrated on the annual peak streamflow data for U.S. Geological Survey streamflow-gaging station 08190000 Nueces River near Laguna, Texas. The data are available in the data set `USGSsta08190000peaks`. Example 9–1 produces, using algorithmic similarity to other examples with similar themes in this dissertation, the empirical distribution by Weibull plotting positions and, by the method of L-moments, fits a Kappa distribution using the `parkap()` function. The quantiles of the Kappa are computed by the `quakap()` function. The distributions are shown in figure 9.1. A standard Normal tranformation (`qnorm()`) is used for the horizontal axis, and a $\log_{10}()$ transformation `log10()` is used for the vertical axis. These transformations increase the linearity in the figure.

9–1

```
data(USGSsta08190000peaks) # from lmomco package
attach(USGSsta08190000peaks)
Q <- sort(peak_va) # sort data for plotting
detach(USGSsta08190000peaks) # detach names from the workspace
PP <- pp(Q) # compute Weibull plotting positions
lmr <- lmoms(Q)
PARkap <- parkap(lmr) # L-moments and parameters

#pdf("nueces1.pdf")
plot(qnorm(PP),log10(Q),
     xlab="STANDARD␣NORMAL␣DEVIATE",
     ylab="LOG10(STREAMFLOW,␣IN␣FT^3/S)")
lines(qnorm(PP),log10(quakap(PP,PARkap)), lwd=3)
legend(-2,5.5, c("Kappa␣by␣L-moments"),
       lwd=c(3), lty=c(1), box.lty=0, bty="n")
#dev.off()
```

Some interpretations of the Kappa fit in figure 9.1 can be made. The empirical distribution has some interesting sinuous or seemingly distinguishable (steepness, curvature) parts. The two "hinge" points appear near -0.55 and 0.10 standard deviations. For the data from this particular location, values with about $F < 0.29$ (`pnorm(-0.55)`) likely represent drought-like conditions in which the annual peak streamflow does not represent storm runoff. For values with the approximate range $0.29 < F < 0.54$ (`pnorm(0.10)`),

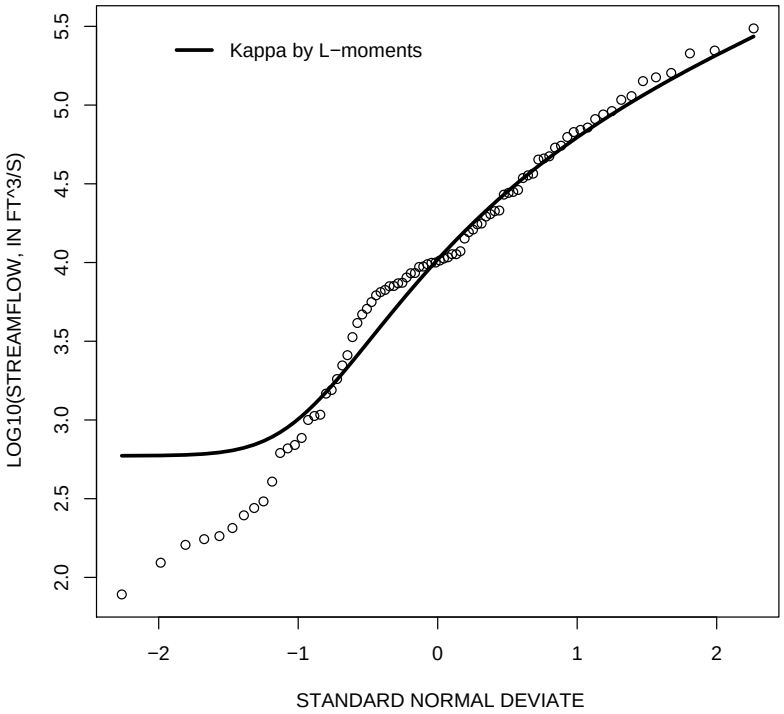

Figure 9.1. Empirical distribution of annual peak streamflow for U.S. Geological Survey streamflow-gaging station 08190000 Nueces River near Laguna, Texas and Kappa distribution fit by the method of L-moments from example 9–1

the relatively flatter portion of the data could represent a flow regime during years for which flows are relatively stable and do not represent periods (years) for which the flows are not caused by "full reaction" of the approximately 737 square-mile watershed to large rainfall events. From a perspective of distributional analysis of flood flows, the right-tail portion of the distribution is of primary interest. The figure shows that the Kappa has an acceptable fit to the general curvature of the empirical distribution.

The Kappa has an apparently acceptable fit in the right-tail of the empirical distribution in figure 9.1. The Kappa distribution is compared to the log-Normal distribution, fit by the method of moments, and the Generalized Normal distribution, fit by the method of L-moments in example 9–2. The results are shown in figure 9.2. The horizontal and vertical limits have been changed by the `xlim` and `ylim` arguments to the `plot()` function. The variable `F` contains F values conveniently produced by the `nonexceeds()` function. Finally for this example, the `qlmomco()` is used to create parallel syntax (see lines labeled `# Kappa` and `# GNO`) for computation of Kappa and Generalized Normal quantiles.

9–2

```
PARgno <- pargno(lmr)
mu.lg <- mean(log10(Q)); sig.lg <- sd(log10(Q))
F <- nonexceeds()
#pdf("nueces2.pdf")
plot(qnorm(PP),log10(Q), xlim=c(-1,3), ylim=c(2.5,6),
     xlab="STANDARD␣NORMAL␣DEVIATE",
     ylab="LOG10(STREAMFLOW,␣IN␣FT^3/S)")
lines(qnorm(F),log10(qlmomco(F,PARkap)), lwd=3) # Kappa
lines(qnorm(F),log10(qlmomco(F,PARgno)), lwd=1) # GNO
lines(qnorm(F),
      qnorm(F, mean=mu.lg, sd=sig.lg),
      lty=2) # plot lognormal distribution as dashed line
legend(1,4.5, c("Kappa␣by␣L-moments",
                "GNO␣by␣L-moments",
                "Log-normal␣by␣\n␣product␣moments"),
       lwd=c(3,1,1), lty=c(1,1,2), box.lty=0, bty="n")
#dev.off()
```

◀

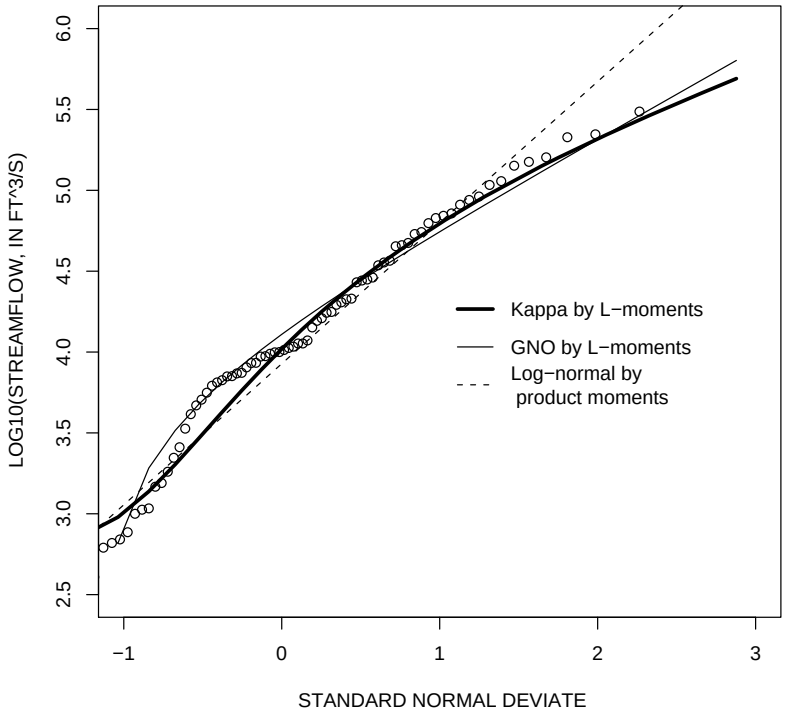

Figure 9.2. Empirical distribution of annual peak streamflow for U.S. Geological Survey streamflow-gaging station 08190000 Nueces River near Laguna, Texas and three selected distributions from example 9–2

Dupuis and Winchester (2001) provide a study of the Kappa in the context of the methods of L-moments and maximum likelihood with a focus on "infeasible" parameters and circumstances in which τ_4 is above the τ_4^{glo} of the Generalized Logistic and hence Kappa parameters are "uncomputable." Their definition of infeasible is the same as that on page 104 of this dissertation and treated by Chen and Balakrishnan (1995) for different distributions. Dupuis and Winchester conduct a more than $1{,}000$ drawing simulation study for $n = 50$ samples for sequences of κ and h parameters using $\xi = 0$ and $\alpha = 1$ without a loss of generality. A select part of their study is reproduced by the code in example 9–3.

9–3

```
n <- 50; nsim <- 1000 # sample size and no. of simulation
Ks <- c(-0.4, -0.2, 0, 0.2, 0.4) # kappas
Hs <- c(-1.2, -0.8, -0.4, 0, 0.4, 0.8, 1.2) # h's

for(k in Ks) {   # for each kappa
  for(h in Hs) { # for each h
    failed <- 0  # reset
    infeas <- 0  # reset

    for(i in 1:nsim) { # for each simulation
      KAPsim <- NA

      while(1) { # IMPORTANT, until computable param are found
        Xsim   <- rlmomco(n, vec2par(c(0,1,k,h), type="kap"))
        KAPsim <- parkap(lmoms(Xsim))
        ifelse(KAPsim$ifail > 0, failed <- failed + 1, break)
      }

      kapsup <- KAPsim$support # support of fitted Kappa dist.

      if(kapsup[1] > min(Xsim) | kapsup[2] < max(Xsim)) {
        infeas <- infeas + 1 # found infeasible
      }
    }
    DWinf <- round( 100 * infeas/nsim, digits=1)
    DWcom <- round( 100 * failed/(nsim+failed), digits=1)
    # Make nice rows of output
    # kappa   h    infeasible.percent[uncomputable.percent]
    cat(c(k,"␣␣",h,"␣␣",DWinf,"[",DWcom,"]","\n"), sep="")
  }
}
# One example line from the cat() is shown below
# -0.2  0.4  33.6[13.3]
```

The example reproduces the κ and h sequences used by Dupuis and Winchester (2001) and then mimics the quadruple loop that the authors must of used by following their written description of their algorithm. The example computes the percent of time that the parameters are infeasible or uncomputable. Feasible means that the support of the fitted distribution is inside the range of the observed (or simualted here) data. Readers are asked to note the addition of the number of failed attempts in the denominator used to compute the `DWcom` variable, which Dupuis and Winchester also are careful to point out. The last line of the example shows the results for $\kappa = -0.2$ and $h = 0.4$, and for these parameters, infeasible parameters were found 33.6 percent of the time and 13.3 percent of the time the parameters could not even be computed (τ_4 is above the τ_4^{glo} of the Generalized Logistic distribution, see Chapter 10). These two percentages compare favorably with those in Dupuis and Winchester (2001, p. 108), who report "36.6[12.4]" compared to "33.6[13.3]" in example 9–3. The remainder of the output (not shown) by example 9–3 also compares favorably with Dupuis and Winchester. This $\{\tau_3, \tau_4\}$-parameter space restriction is removed when using the Generalized Lambda distribution described in the next section. But the parameter space expansion comes at the expensive of more complex parameter estimation nuances. ◀

9.2.2 Generalized Lambda Distribution

The Generalized Lambda distribution (Karian and Dudewicz, 2000; Asquith, 2007; Karvanen and Nuutinen, 2008) is a flexible distribution with deceptively simple PDF and QDF functions compared to the difficulty of parameter estimation in terms of its moments (product moment or L-moment). A reason that the Generalized Lambda is of interest is that it has a larger $\{\tau_3, \tau_4\}$-parameter space than the Kappa. However, the Generalized Lambda is problematic to work with in part because multiple parameter solutions are possible, and demonstration and accommodation of this possibility is made in this dissertation. In circumstances in which the analyst is to produce an equation for the fitted distribution, the Generalized Lambda is more attractive because of the simpler, that is, easier to "deploy," QDF form than that for the Kappa. Mercy and Kumaran (2010) provide extensive derivations of probability-weighted moments for the Generalized Lambda under censoring conditions.

DISTRIBUTION FUNCTIONS

The distribution functions of the Generalized Lambda having parameters ξ (location), α (scale), κ (shape1), h (shape2) are

$$f(x) = \frac{1}{\alpha[\kappa(F^{\kappa-1}) - h(1-F)^{h-1}]} \tag{9.12}$$

$F(x)$ has no explicit analytical form

$$x(F) = \xi + \alpha[F^{\kappa} - (1-F)^{h}] \tag{9.13}$$

The ranges of the distribution are listed below where (or) note exclusion of ∞ and the brackets [or] note inclusion of the indicated limit:

κ	h	Range
$\kappa > 0$	$h > 0$	$[\xi - \alpha, \xi + \alpha]$
$\kappa > 0$	$h = 0$	$[\xi, \xi + \alpha]$
$\kappa = 0$	$h > 0$	$[\xi - \alpha, \xi]$
$\kappa < 0$	$h < 0$	$(-\infty, \infty)$
$\kappa < 0$	$h = 0$	$(-\infty, \xi + \alpha]$
$\kappa = 0$	$h < 0$	$[\xi - \alpha, \infty)$

The first three L-moments are

$$\lambda_1 = \xi + \alpha\left(\frac{1}{\kappa+1} - \frac{1}{h+1}\right) \tag{9.14}$$

$$\lambda_2 = \alpha\left(\frac{\kappa}{(\kappa+2)(\kappa+1)} + \frac{h}{(h+2)(h+1)}\right) \tag{9.15}$$

$$\lambda_3 = \alpha\left(\frac{\kappa(\kappa-1)}{(\kappa+3)(\kappa+2)(\kappa+1)} - \frac{h(h-1)}{(h+3)(h+2)(h+1)}\right) \tag{9.16}$$

The fourth L-moment of the Generalized Lambda is

$$K_{\lambda_4} = \frac{\kappa(\kappa-2)(\kappa-1)}{(\kappa+4)(\kappa+3)(\kappa+2)(\kappa+1)}$$

$$H_{\lambda_4} = \frac{h(h-2)(h-1)}{(h+4)(h+3)(h+2)(h+1)}$$

$$\lambda_4 = \alpha(K_{\lambda_4} + H_{\lambda_4}) \tag{9.17}$$

The fifth L-moment of the Generalized Lambda is

$$\begin{aligned} K_{\lambda_5} &= \frac{\kappa(\kappa-3)(\kappa-2)(\kappa-1)}{(\kappa+5)(\kappa+4)(\kappa+3)(\kappa+2)(\kappa+1)} \\ H_{\lambda_5} &= \frac{h(h-3)(h-2)(h-1)}{(h+5)(h+4)(h+3)(h+2)(h+1)} \\ \lambda_5 &= \alpha(K_{\lambda_5} - H_{\lambda_5}) \end{aligned} \tag{9.18}$$

Let L_λ be defined as follows:

$$L_\lambda = \kappa(h+2)(h+1) + h(\kappa+2)(\kappa+1) \tag{9.19}$$

The L-moment ratio τ_3 is

$$\begin{aligned} K_{\tau_3} &= \kappa(\kappa-1)(h+3)(h+2)(h+1) \\ H_{\tau_3} &= h(h-1)(\kappa+3)(\kappa+2)(\kappa+1) \\ \tau_3 &= \frac{K_{\tau_3} - H_{\tau_3}}{(\kappa+3)(h+3)L_\lambda} \end{aligned} \tag{9.20}$$

The L-moment ratio τ_4 is

$$\begin{aligned} K_{\tau_4} &= \kappa(\kappa-3)(\kappa-2)(\kappa-1)(h+5)(h+4)(h+3)(h+2)(h+1) \\ H_{\tau_4} &= h(h-3)(h-2)(h-1)(\kappa+5)(\kappa+4)(\kappa+3)(\kappa+2)(\kappa+1) \\ \tau_4 &= \frac{K_{\tau_4} + H_{\tau_4}}{(\kappa+4)(h+4)(\kappa+3)(h+3)L_\lambda} \end{aligned} \tag{9.21}$$

The L-moment ratio τ_5 is

$$\begin{aligned} K_{\tau_5} &= \kappa(\kappa-2)(\kappa-1)(h+4)(h+3)(h+2)(h+1) \\ H_{\tau_5} &= h(h-2)(h-1)(\kappa+4)(\kappa+3)(\kappa+2)(\kappa+1) \\ \tau_5 &= \frac{K_{\tau_5} - H_{\tau_5}}{(\kappa+4)(h+4)(\kappa+3)(h+3)L_\lambda} \end{aligned} \tag{9.22}$$

Karvanen and Nuutinen (2008) provide a general equation for L-moment computation for $r \geq 2$

$$\lambda_r = \alpha \sum_{j=0}^{r-1} (-1)^{r-j-1} \binom{r-1}{j} \binom{r+j-1}{j} \left(\frac{1}{j+1+\kappa} + \frac{(-1)^r}{j+1+h} \right) \tag{9.23}$$

Finally, the L-moments are potentially defined for

$$\kappa > -1 \text{ and } h > -1 \tag{9.24}$$

There are no simple expressions for the parameters in terms of the L-moments. Numerical methods must be employed and multiple solutions in different regions of $\{\kappa, h\}$-space are common. Besides demonstration in the remainder of this section, the multiple solution nature of the Generalized Lambda is considered extensively near the end of Section 11.2. The distribution with $\kappa^{\text{gld}} = 0$ is a form of the Generalized Pareto distribution.

USING R

The Generalized Lambda distribution is demonstrated using the annual peak streamflow data for U.S. Geological Survey streamflow-gaging station 08190000 Nueces River near Laguna, Texas. The data are available in the data set `USGSsta08190000peaks`. Example 9–4, using algorithmic similarity to other examples, produces the empirical distribution by Weibull plotting positions and by the method of L-moments fits a Generalized Lambda distribution using the `pargld()` function. The quantiles of the Generalized Lambda are computed by the `quagld()` function. For this example, two solutions of the Generalized Lambda appear available for the tolerance on the minimization set by `eps=1e-2`. The two Generalized Lambda fits are shown in figure 9.3 along with the fit for the Kappa as a reference.

9–4

```
data(USGSsta08190000peaks) # from lmomco package
attach(USGSsta08190000peaks)
Q   <- sort(peak_va) # sort the annual peak streamflow values
PP  <- pp(Q) # compute Weibull plotting positions
lmr <- lmoms(Q); PARkap <- parkap(lmr) # and fit a Kappa

# Now for the GLD distribution
PARgld1 <- pargld(lmr, eps=1e-2); # print(PARgld1)
# output has been suppressed, values lifted from the output
other <- unlist(PARgld2$rest[4,1:4]); # print(other)
PARgld2 <- vec2par(c(5541.6, 245064,
                     7.551838, 308.4899734), type="gld")
```

The textual output of the `pargld()` function has been suppressed in the example, but two viable solutions exist.[3] The parameters for one solution are shown in the `vec2par()`

[3] The author has chosen a large tolerance to cause two solutions to be found for this example.

function. Because numerical methods are used, the precise numerical values for the parameters will be different in subsequent trials of the `pargld()` function.

As shown in example 9–4, the Generalized Lambda has two solutions for the provided L-moments. The minimum least-squares solution on τ_3 and τ_4 that is computed by `pargld()` is manually set into the `PARgld2` variable, and the error is $\epsilon \approx 1\mathrm{E}{-9}$. The first and primary solution in `PARgld1` has a much larger $\epsilon \approx 2\mathrm{E}{-3}$. Refining interpretation and using the difference $\Delta\tau_5 = \tau_5^{\mathrm{gld}} - \tau_5$, it is seen that $\Delta\tau_5 \approx 0.07$ for solution in `PARgld1` and $\Delta\tau_5 \approx -0.11$ for solution in `PARgld2`. Asquith (2007) and documentation of *lmomco* provides further details of this algorithm. The preferable solution in `PARgld1` is about GLD(-58839, -54582, 59.23810, -0.414052). Example 9–5 plots both Generalized Lambda solutions and that for the Kappa. The author suggests that the preferred solution in `PARgld1` is visually more consistent with the empirical distribution for right-tail, annual peak streamflow estimation, and this is the tail of interest here.

9–5

```
#pdf("nueces3.pdf")
plot(qnorm(PP),log10(Q),
     xlab="STANDARD␣NORMAL␣DEVIATE",
     ylab="LOG10(STREAMFLOW,␣IN␣FT^3/S)")
lines(qnorm(PP), log10(quagld(PP,PARgld1)), lwd=3)
lines(qnorm(PP), log10(quagld(PP,PARgld2)), lwd=1)
lines(qnorm(PP), log10(quakap(PP, PARkap)), lwd=1, lty=2)
legend(-2,5.5, c("GLD1(preferred)","GLD2","Kappa␣by␣L-moments"),
       lwd=c(3,1,1), lty=c(1,1,2), box.lty=0, bty="n")
#dev.off()
```

Further solution justification beyond the author's visual assessment of the previous paragraph is needed. The convergence error on τ_3 and τ_4 for `PARgld2` is $\epsilon \approx 1\mathrm{E}{-9}$, which is about 7 orders of magnitude better than that for `PARgld1`. The $\Delta\tau_5$ performance however is substantially better in `PARgld1` than `PARgld2`. These statements are made to point out that `PARgld2` is numerically superior in terms of fit to the L-moments but lacks a visibly appropriate fit. A QDF mixture might be an alternative model for analysis of these data.[4] The author suggests that the choice of a general value for `eps` for the `pargld()` (and `parTLgld()` by association) function is an open problem for additional research.

To conclude, the author recognizes the greater complexity and interpretation required for parameter estimation and subsequent selection of a preferred fit for the distribution

[4] Such a mixture could be constructed using the Intermediate Rule on page 35 in which two distributions used and each fit to the upper and lower halves of the data. The weight factor for the Intermediate Rule could be chosen to satisfy the overall mean.

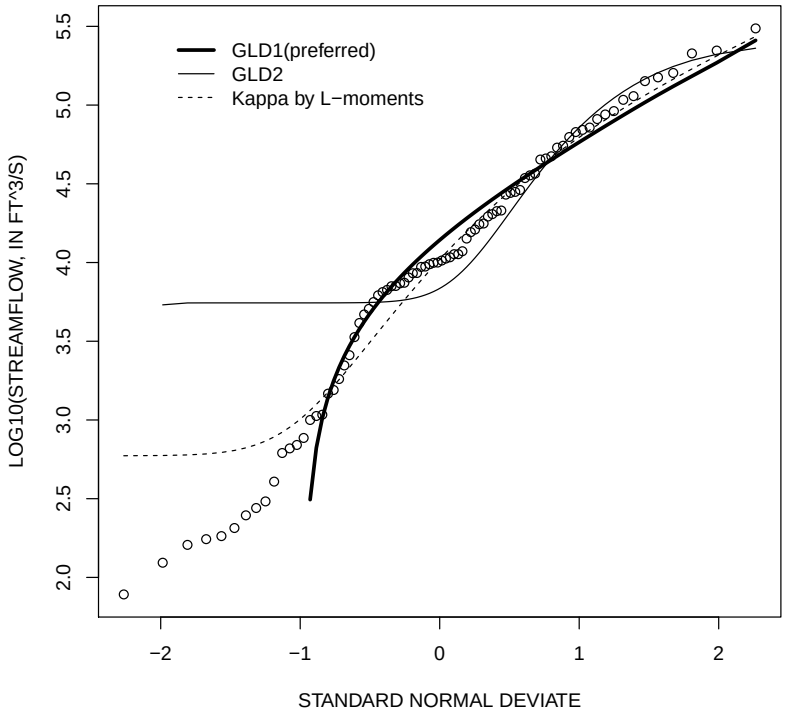

Figure 9.3. Empirical distribution of annual peak streamflow for U.S. Geological Survey streamflow-gaging station 08190000 Nueces River near Laguna, Texas and two Generalized Lambda distributions and Kappa fit to sample L-moments from example 9–4

relative to the lack of analyst intervention needed for the Kappa. The author suggests that the Generalized Lambda could be useful for circumstances in which τ_4 is greater than that of the Generalized Logistic distribution and therefore the Kappa cannot be fit. Thus, the Generalized Lambda thus can have a complementary role to the Kappa in circumstances in which "hyper" L-kurtosis occurs. ◀

Su (2010) provides the *GLDEX* package for R to fit the Generalized Lambda distribution and small part of that package is parameter estimation by L-moments (see page 10 of this dissertation for more discussion). The L-moment functions provided by *GLDEX* are `Lmoments()`,`Lcoefs()`,`Lmomcov()`,`Lmoments_calc()`,`Lmomcov_calc()`,and `t1lmoments()`. These functions apparently are authored by Karvanen (2009) because credit is given; the *GLDEX* does not require the `library()` loading of the *Lmoments* package by Karvanen (2009).

Example 9–6 is derived from examples in the *GLDEX* package. The example simulates $n = 500$ standard Normal values and then computes two parameter suites for the Gener-

alized Lambda. There are two parameterizations of the distribution by the package: `RPRS` and `RMFMKL` and interest here is the former. The example ends by creating the `gldex` variable, which holds the `RPRS` parameterization because this is most similar (nearly identical) to that shown in this dissertation (see eq. (9.13)). The `fun.RPRS.lm()` function and inversion of its second returned parameter are of primary interest in the example.

9–6

```
library(GLDEX)
set.seed(1) # author would like others to mimic
fake.dat <- rnorm(500,0,1) # simulate 500 standard normals
fun.data.fit.lm(fake.dat) # GLDEX has TWO GLD parameterizations
#               RPRS     RMFMKL # manually commented out
#[1,] -4.645127e-04 0.01371437 # manually commented out
#[2,] -4.678373e-07 1.53954741 # manually commented out
#[3,] -2.593875e-07 0.10017806 # manually commented out
#[4,] -2.701999e-07 0.08378585 # manually commented out
# fun.RMFMKL.lm(fake.dat) # manually commented out

gldvec    <- fun.RPRS.lm(fake.dat) # "RPRS" is approx. lmomco
gldvec[2] <- 1/gldvec[2] # inversion of the 2nd parameter
gldex <- vec2par(gldvec, type="gld") # parameters for lmomco
```

The exploration of *GLDEX* capabilities continues in example 9–7. This example uses the *lmomco* package to compute the Generalized Lambda parameters. The output shows the `pargld()` function making various attempts that originate from within several distinct parameter regions, which are throughly described by Karian and Dudewicz (2000). The chosen solution, by small error in $\{\tau_3, \tau_4\}$ space and smallest $\Delta\tau_5$, is GLD(-0.0108, 7.0158, 0.0899, 0.0955). Close inspection shows that GLD(0.0782, 2.2597, 4.8869, 4.1421) might also be appropriate because this fit also has the smallest error in $\{\tau_3, \tau_4\}$ space and provides a solution that also is not much worse in terms of $\Delta\tau_5$. These two solutions are set into the `lmomco1` and `lmomco2` variables of example 9–8. Readers are asked to consult example 11–20 on page 332 and associated discussion for more details concerning multiple Generalized Lambda solutions.

9–7

```
PARgld <- pargld(lmoms(fake.dat))

print(PARgld) # some editing for space has been done
$type
[1] "gld"
$para
         xi       alpha       kappa           h
-0.01076771  7.01578127  0.08985130  0.09553745
```

```
$delTau5
[1] 0.009355544
$error
[1] 9.619003e-08
$source
[1] "pargld"
$rest
         xi    alpha    kappa        h   delTau5       error
1 -0.011357 7.078270 0.088893 0.094619 0.009394 1.520184e-10
2  0.078243 2.259686 4.886910 4.142094 0.014976 6.740724e-11
3  0.078250 2.259649 4.886804 4.141916 0.014976 6.700292e-13
```

Example 9–8, after setting the two solutions for the Generalized Lambda from the *lmomco* package, creates a QDF plot shown in figure 9.4. The figure shows the simulated data values and the preferred solution by the `pargld()` function in `lmomco1` as the solid thick line. The solid thin line is the alternative solution in `lmomco2` appears slightly less favorable. Finally, the solution from the `fun.RPRS.lm()` function is drawn as the dashed line. The example demonstrates consistency between the *lmomco* and *GLDEX* packages.

9–8

```
lmomco1 <- vec2par(c(-0.01076771, 7.01578127,
                      0.08985130, 0.09553745), type="gld")
lmomco2 <- vec2par(c(0.078250, 2.259649,
                     4.886804, 4.141916), type="gld")
F <- seq(.01,.99, by=.01)
#pdf("gldex_norm.pdf")
plot(pp(fake.dat),sort(fake.dat), pch=16, cex=0.75, col=8,
     xlab="NONEXCEEDANCE␣PROBABILITY",
     ylab="QUANTILE")
lines(F, qlmomco(F, lmomco1), lwd=3)
lines(F, qlmomco(F, lmomco2), lwd=1)
lines(F, qlmomco(F, gldex),   lty=2)
#dev.off()
```

◀.

Example 9–9 shows for $\kappa^{\text{gld}} = 0$ that the Generalized Lambda and Generalized Pareto are equivalent.

9–9

```
qlmomco(c(0.25,0.75), vec2par(c(100,30,      0.4), type="gpa"))
[1] 108.1524 131.9238
qlmomco(c(0.25,0.75), vec2par(c(100,30/0.4,0,0.4), type="gld"))
[1] 108.1524 131.9238
```

◀

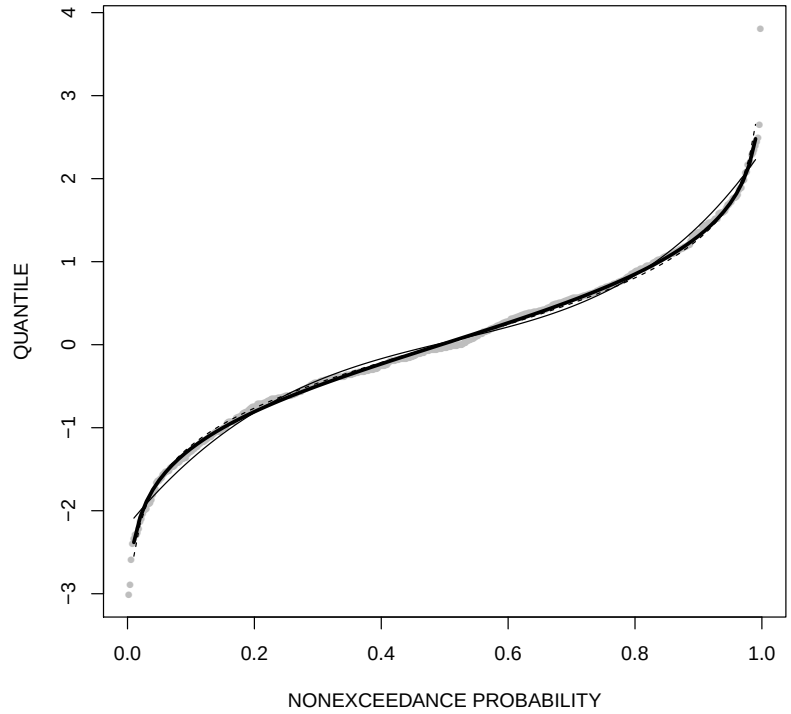

Figure 9.4. Simulated standard Normal distribution for $n = 500$ and three fitted Generalized Lambda distributions using algorithms of the *GLDEX* and *lmomco* packages from example 9–8. The two solid lines are from *lmomco* and the dashed line is from *GLDEX*.

9.2.3 Trimmed Generalized Lambda Distribution

The Trimmed Generalized Lambda distribution (Asquith, 2007) is a trimmed version of the Generalized Lambda presented in the previous section. The Trimmed Generalized Lambda is defined by Asquith (2007) in terms of the symmetrically $t = 1$ trimmed TL-moments. However, extension of the TL-moments to other and asymmetrical trimming is made for this dissertation (see eq. (9.37)). Partial motivation for a Trimmed Generalized Lambda is to provide a Generalized Lambda capable of reliably representing the Cauchy and potentially useful for experimentation with TL-moments in general. A Generalized Lambda fit to the TL-moments has been fit by the method of TL-moments.

DISTRIBUTION FUNCTIONS

The distribution functions of the Trimmed Generalized Lambda are the same as those for the Generalized Lambda so reference to Section 9.2.2 is made. The extended listings of

the TL-moments are shown for $t = 1$ symmetrical trimming. The first two TL-moments ($\lambda_1^{(1)}$ and $\lambda_2^{(1)}$) of the Generalized Lambda are

$$\lambda_1^{(1)} = \xi + 6\alpha\left(\frac{1}{(\kappa+3)(\kappa+2)} - \frac{1}{(h+3)(h+2)}\right) \tag{9.25}$$

$$\lambda_2^{(1)} = 6\alpha\left(\frac{\kappa}{(\kappa+4)(\kappa+3)(\kappa+2)} + \frac{h}{(h+4)(h+3)(h+2)}\right) \tag{9.26}$$

The third TL-moment ($\lambda_3^{(1)}$) of the Trimmed Generalized Lambda is

$$\begin{aligned} K_{\lambda_3^{(1)}} &= \frac{\kappa(\kappa-1)}{(\kappa+5)(\kappa+4)(\kappa+3)(\kappa+2)} \\ H_{\lambda_3^{(1)}} &= \frac{h(h-1)}{(h+5)(h+4)(h+3)(h+2)} \\ \lambda_3^{(1)} &= \frac{20}{3}\alpha(K_{\lambda_3^{(1)}} - H_{\lambda_3^{(1)}}) \end{aligned} \tag{9.27}$$

The fourth TL-moment ($\lambda_4^{(1)}$) of the Trimmed Generalized Lambda is

$$\begin{aligned} K_{\lambda_4^{(1)}} &= \frac{\kappa(\kappa-2)(\kappa-1)}{(\kappa+6)(\kappa+5)(\kappa+4)(\kappa+3)(\kappa+2)} \\ H_{\lambda_4^{(1)}} &= \frac{h(h-2)(h-1)}{(h+6)(h+5)(h+4)(h+3)(h+2)} \\ \lambda_4^{(1)} &= \frac{30}{4}\alpha(K_{\lambda_4^{(1)}} + H_{\lambda_4^{(1)}}) \end{aligned} \tag{9.28}$$

The fifth TL-moment ($\lambda_5^{(1)}$) of the Trimmed Generalized Lambda is

$$\begin{aligned} K_{\lambda_5^{(1)}} &= \frac{\kappa(\kappa-3)(\kappa-2)(\kappa-1)}{(\kappa+7)(\kappa+7)(\kappa+6)(\kappa+5)(\kappa+4)(\kappa+3)(\kappa+2)} \\ H_{\lambda_5^{(1)}} &= \frac{h(h-3)(h-2)(h-1)}{(h+7)(h+6)(h+5)(h+4)(h+3)(h+2)} \\ \lambda_5^{(1)} &= \frac{42}{5}\alpha(K_{\lambda_5^{(1)}} - H_{\lambda_5^{(1)}}) \end{aligned} \tag{9.29}$$

The sixth TL-moment ($\lambda_6^{(1)}$) of the Trimmed Generalized Lambda is

$$\begin{aligned}
K_{\lambda_6^{(1)}} &= \frac{\kappa(\kappa-4)(\kappa-3)(\kappa-2)(\kappa-1)}{(\kappa+8)(\kappa+7)(\kappa+6)(\kappa+5)(\kappa+4)(\kappa+3)(\kappa+2)} \\
H_{\lambda_6^{(1)}} &= \frac{h(h-4)(h-3)(h-2)(h-1)}{(h+8)(h+7)(h+6)(h+5)(h+4)(h+3)(h+2)} \\
\lambda_6^{(1)} &= \frac{56}{6}\alpha(K_{\lambda_6^{(1)}} + H_{\lambda_6^{(1)}})
\end{aligned} \tag{9.30}$$

Let $L_{\lambda^{(1)}}$ be defined as follows:

$$L_{\lambda^{(1)}} = \kappa(h+4)(h+3)(h+2) + h(\kappa+4)(\kappa+3)(\kappa+2) \tag{9.31}$$

The TL-moment ratio $\tau_3^{(1)}$ of the Trimmed Generalized Lambda is

$$\begin{aligned}
K_{\tau_3^{(1)}} &= \kappa(\kappa-1)(h+5)(h+4)(h+3)(h+2) \\
H_{\tau_3^{(1)}} &= h(h-1)(\kappa+5)(\kappa+4)(\kappa+3)(\kappa+2) \\
\tau_3^{(1)} &= \frac{10}{9}\left(\frac{K_{\tau_3^{(1)}} - H_{\tau_3^{(1)}}}{(\kappa+5)(h+5)L_{\lambda^{(1)}}}\right)
\end{aligned} \tag{9.32}$$

The TL-moment ratio $\tau_4^{(1)}$ of the Trimmed Generalized Lambda is

$$\begin{aligned}
K_{\tau_4^{(1)}} &= \kappa(\kappa-2)(\kappa-1)(h+6)(h+5)(h+4)(h+3)(h+2) \\
H_{\tau_4^{(1)}} &= h(h-2)(h-1)(\kappa+6)(\kappa+5)(\kappa+4)(\kappa+3)(\kappa+2) \\
\tau_4^{(1)} &= \frac{5}{4}\left(\frac{K_{\tau_4^{(1)}} + H_{\tau_4^{(1)}}}{(\kappa+6)(h+6)(\kappa+5)(h+5)L_{\lambda^{(1)}}}\right)
\end{aligned} \tag{9.33}$$

The TL-moment ratio $\tau_5^{(1)}$ of the Trimmed Generalized Lambda is

$$\begin{aligned}
K^1_{\tau_5^{(1)}} &= \kappa(\kappa-3)(\kappa-2)(\kappa-1) \\
K^2_{\tau_5^{(1)}} &= (h+7)(h+6)(h+5)(h+4)(h+3)(h+2) \\
H^1_{\tau_5^{(1)}} &= h(h-3)(h-2)(h-1) \\
H^2_{\tau_5^{(1)}} &= (\kappa+7)(\kappa+6)(\kappa+5)(\kappa+4)(\kappa+3)(\kappa+2) \\
\tau_5^{(1)} &= \frac{7}{5}\left(\frac{K^1_{\tau_5^{(1)}}K^2_{\tau_5^{(1)}} - H^1_{\tau_5^{(1)}}H^2_{\tau_5^{(1)}}}{(\kappa+7)(h+7)(\kappa+6)(h+6)(\kappa+5)(h+5)L_{\lambda^{(1)}}}\right)
\end{aligned} \tag{9.34}$$

The TL-moment ratio $\tau_6^{(1)}$ of the Trimmed Generalized Lambda is

$$\begin{aligned}
K^1_{\tau_6^{(1)}} &= \kappa(\kappa-4)(\kappa-3)(\kappa-2)(\kappa-1) \\
K^2_{\tau_6^{(1)}} &= (h+8)(h+7)(h+6)(h+5)(h+4)(h+3)(h+2) \\
H^1_{\tau_6^{(1)}} &= h(h-4)(h-3)(h-2)(h-1) \\
H^2_{\tau_6^{(1)}} &= (\kappa+8)(\kappa+7)(\kappa+6)(\kappa+5)(\kappa+4)(\kappa+3)(\kappa+2) \\
K_{\tau_6^{(1)}} &= (\kappa+8)(\kappa+7)(\kappa+6)(\kappa+5) \\
H_{\tau_6^{(1)}} &= (h+8)(h+7)(h+6)(h+5) \\
\tau_6^{(1)} &= \frac{14}{9}\left(\frac{K^1_{\tau_6^{(1)}} K^2_{\tau_6^{(1)}} + H^1_{\tau_6^{(1)}} H^2_{\tau_6^{(1)}}}{K_{\tau_6^{(1)}} H_{\tau_6^{(1)}} L_{\lambda^{(1)}}}\right)
\end{aligned} \tag{9.35}$$

Finally, these $t = 1$ TL-moments are potentially defined for

$$\kappa > -2 \quad \text{and} \quad h > -2 \tag{9.36}$$

As with the Generalized Lambda distribution, there are no simple expressions for the parameters in terms of the TL-moments. Numerical methods must be employed and multiple solutions in different regions of $\{\kappa, h\}$-space are common.

Finally, for this dissertation, the author has derived the following general expression (a special situation for $r = 1$ exists) for the L-moments and TL-moments of the Generalized Lambda. It is fitting to include it within this section.

$$\begin{aligned}
\lambda_r^{(t_1,t_2)} = \alpha(r^{-1})(r+t_1+t_2)\sum_{j=0}^{r-1}(-1)^r \binom{r-1}{j}\binom{r+t_1+t_2-1}{r+t_1-j-1}\times \\
\left(\frac{\Gamma(\kappa+r+t_1-j)\Gamma(t_2+j+1)}{\Gamma(\kappa+r+t_1+t_2+1)} - \frac{\Gamma(r+t_1-j)\Gamma(h+t_2+j+1)}{\Gamma(h+r+t_1+t_2+1)}\right)
\end{aligned} \tag{9.37}$$

where for the special condition of $r = 1$, the mean is

$$\text{mean} = \xi + \lambda_1^{(t_1,t_2)} \tag{9.38}$$

Inspection of the $\Gamma(\ )$ arguments, which must be > 0, in eq. (9.37) shows that

$$\kappa > -(1+t_1) \quad \text{and} \quad h > -(1+t_2) \tag{9.39}$$

The Trimmed Generalized Lambda distribution is demonstrated by comparison to the Cauchy in example 9–10. In the example, the location and scale parameters of the Cauchy are set respectively in `myloc` and `myscal`. A sample size of $n = 300$ is set for simulation by the `rcauchy()` function. The sample TL-moments from the random sample in `fake.dat` are computed the by `TLmoms()` function, and the *lmomco* parameter lists (see page 163 and ex. 7–1) for the two distributions are respectively set in `PARcau` and `PARgld` by the `parcau()` and `parTLgld()` functions.

9–10

```
myloc    <- 3000; myscal <- 40000; n <- 300
set.seed(10) # see comments about random seed in text
fake.dat <- rcauchy(n, location=myloc, scale=myscal)
TLlmr    <- TLmoms(fake.dat, trim=1)
PARcau   <- parcau(TLlmr)
PARgld   <- parTLgld(TLlmr, eps=1e-3, verbose=TRUE)
```

The demonstration continues in example 9–11 in which two vectors of quantiles for each distribution are set in `x.cau` and `x.gld` by the `quacau()` and `quagld()` functions. The PDFs of the distributions are subsequently computed by the `pdfcau()` and `pdfgld()` functions and are shown in figure 9.5.

9–11

```
F <- seq(0.1,0.9, by=0.01)
x.cau <- quacau(F,PARcau)
x.gld <- quagld(F,PARgld)
#pdf("TLgldcau.pdf")
plot( x.cau, pdfcau(x.cau,PARcau), type="l")
lines(x.gld, pdfgld(x.gld,PARgld), lty=2)
legend(0,1e-6,c("TL-moment␣fitted␣Cauchy",
                "TL-moment␣fitted␣GLD"),
       lty=c(1,2), box.lty=0, bty="n", xjust=0.5, yjust=0)
#dev.off()
```

The figure shows general mimicry of the Cauchy by the Generalized Lambda. However, the scale of the extremely heavy-tailed Cauchy is large enough to cause solution difficulties with the Generalized Lambda. Readers are asked to repeat examples 9–10 and example 9–11 with `set.seed(1)` and a substantial departure from the Cauchy will be seen. Experimentation with the sample size n is also advised. ◀

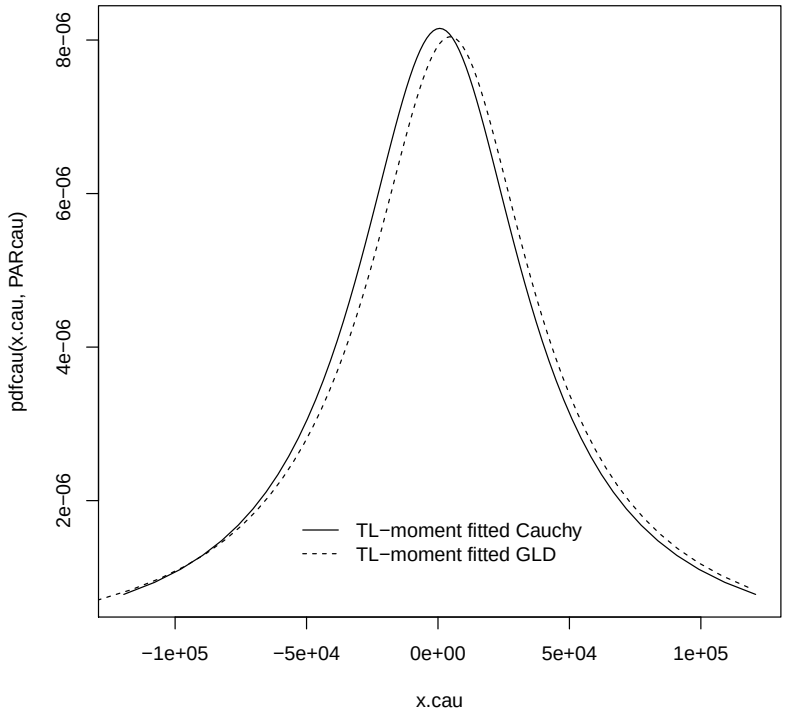

Figure 9.5. Comparison of PDF of Cauchy and Generalized Lambda distributions fit to 300 random samples of CAU($3000, 40000$) by method of TL-moments from example 9–11

9.2.4 Wakeby Distribution

The Wakeby distribution (Houghton, 1978; Landwehr and others, 1979a) is a wildly flexible distribution because it has five parameters: ξ (location), α (scale1), β (shape1), γ (scale2), and δ (shape2). The distribution is attractive because it is fit to four or five L-moments depending on whether ξ is either known or unknown. The Wakeby distribution has the following properties as identified by Hosking and Wallis (1997, p. 205), which make the Wakeby useful as a tool for advanced distributional analysis:

1. The Wakeby distribution, similarly to the Kappa, can mimic the shapes or even subsume many of the skewed distributions described herein, including the Generalized Extreme Value, Generalized Normal, and Pearson Type III;
2. The Wakeby distribution is particularly useful for simulation to study the properties of simpler distributions and for the study of distribution-form sensitivity during the study of a problem;

3. The Wakeby can acquire a very heavy upper tail, much like the Generalized Lambda (greater τ_4 values than can be acquired by the Kappa) and thus can generate large outliers; and
4. The ξ parameter represents the lower bounds of the Wakeby. For some data sets, imposition of a lower bound can be useful, but the Wakeby is readily solved if the lower bounds is unknown.

The Hosking and Wallis (1997) algorithm for Wakeby parameter estimation, which is used by the *lmomco* package, fits a Generalized Pareto distribution for some combinations of L-moments that would otherwise provide parameters that are inconsistent with constraints listed below. Although the Wakeby is flexible, according to the author's experience with hydrologic data sets, the Wakeby often cannot be fit in practice to a nontrivial number of data sets (well at least data sets of Texas hydrology). However, when pooled or regional mean or weighted mean L-moments are used (see Section 11.1.2, ex. 11–7), such L-moments generally can be used to estimate Wakeby parameters.

DISTRIBUTION FUNCTIONS

The distribution functions of the Wakeby having parameters ξ (location), α (scale1), γ (scale2), β (shape1), δ (shape2) are

$$f(x) = \alpha(1-F)^{\beta-1} + \gamma(1-F)^{-\delta-1} \tag{9.40}$$

$F(x)$ has no explicit analytical form

$$x(F) = \xi + \frac{\alpha}{\beta}[1-(1-F)^{\beta}] - \frac{\gamma}{\delta}[1-(1-F)^{-\delta}] \tag{9.41}$$

The constraints of the Wakeby parameters are:

$$\begin{gathered} \beta + \delta > 0 \quad \text{or} \quad \beta = \gamma = \delta = 0 \\ \text{if} \quad \alpha = 0 \quad \text{then} \quad \beta = 0 \\ \text{if} \quad \gamma = 0 \quad \text{then} \quad \delta = 0 \\ \gamma \geq 0 \quad \text{and} \quad \alpha + \gamma \geq 0 \end{gathered}$$

The range of the distribution is $\xi \leq x \leq x_U$ where the upper limit is

$$x_U = \begin{cases} \infty & \text{if } \delta \geq 0 \text{ and } \gamma > 0 \\ \xi + \alpha/\beta - \gamma/\delta & \text{if } \delta < 0 \text{ or } \gamma = 0 \end{cases} \tag{9.42}$$

The L-moments for $r \leq 3$ are defined for $\delta < 1$

$$\lambda_1 = \xi + \frac{\alpha}{(1+\beta)} + \frac{\gamma}{(1-\delta)} \tag{9.43}$$

$$\lambda_2 = \frac{\alpha}{(1+\beta)(2+\beta)} + \frac{\gamma}{(1-\delta)(2-\delta)} \tag{9.44}$$

$$\lambda_3 = \frac{\alpha(1-\beta)}{(1+\beta)(2+\beta)(3+\beta)} + \frac{\gamma(1+\delta)}{(1-\delta)(2-\delta)(3-\delta)} \tag{9.45}$$

and the L-moments for $r > 3$ are

$$\lambda_4 = \frac{\alpha(1-\beta)(2-\beta)}{(1+\beta)(2+\beta)(3+\beta)(4+\beta)} + \frac{\gamma(1+\delta)(2+\delta)}{(1-\delta)(2-\delta)(3-\delta)(4-\delta)} \tag{9.46}$$

$$\lambda_5 = \frac{\alpha(1-\beta)(2-\beta)(3-\beta)}{(1+\beta)(2+\beta)(3+\beta)(4+\beta)(5+\beta)} + \frac{\gamma(1+\delta)(2+\delta)(3+\delta)}{(1-\delta)(2-\delta)(3-\delta)(4-\delta)(5-\delta)} \tag{9.47}$$

The following algorithm can be used for computation of the parameters in terms of the L-moments. If ξ (lower bounds) is unknown, let

$$N_1 = 3\lambda_2 - 25\lambda_3 + 32\lambda_4 \tag{9.48}$$

$$N_2 = -3\lambda_2 + 5\lambda_3 + 8\lambda_4 \tag{9.49}$$

$$N_3 = 3\lambda_2 + 5\lambda_3 + 2\lambda_4 \tag{9.50}$$

and

$$C_1 = 7\lambda_2 - 85\lambda_3 + 203\lambda_4 - 125\lambda_5 \tag{9.51}$$

$$C_2 = -7\lambda_2 + 25\lambda_3 + 7\lambda_4 - 25\lambda_5 \tag{9.52}$$

$$C_3 = 7\lambda_2 + 5\lambda_3 - 7\lambda_4 - 5\lambda_5 \tag{9.53}$$

The parameters β and $-\delta$ are the larger and smaller roots, respectively, of the quadratic equation

$$(N_2C_3 - N_3C_2)z^2 + (N_1C_3 - N_3C_1)z + (N_1C_2 - N_2C_1) = 0 \tag{9.54}$$

and

$$\alpha = \frac{(1+\beta)(2+\beta)(3+\beta) \times [(1+\delta)\lambda_2 - (3-\delta)\lambda_3]}{4(\beta+\delta)} \tag{9.55}$$

$$\gamma = \frac{-(1-\delta)(2-\delta)(3-\delta) \times [(1-\beta)\lambda_2 - (3+\beta)\lambda_3]}{4(\beta+\delta)} \tag{9.56}$$

$$\xi = \lambda_1 - \frac{\alpha}{(1+\beta)} - \frac{\gamma}{(1-\delta)} \tag{9.57}$$

If ξ is known, assume without loss of generality that $\xi = 0$, let

$$N_1 = 4\lambda_1 - 11\lambda_2 + 9\lambda_3 \tag{9.58}$$

$$N_2 = -\lambda_2 + 3\lambda_3 \tag{9.59}$$

$$N_3 = \lambda_2 + \lambda_3 \tag{9.60}$$

and

$$C_1 = 10\lambda_1 - 29\lambda_2 + 35\lambda_3 - 16\lambda_4 \tag{9.61}$$

$$C_2 = -\lambda_2 + 5\lambda_3 - 4\lambda_4 \tag{9.62}$$

$$C_3 = \lambda_2 - \lambda_4 \tag{9.63}$$

Then as before, β and $-\delta$ are the larger and smaller roots of eq. (9.54) and

$$\alpha = \frac{(1+\beta)(2+\beta) \times [\lambda_1 - (2-\delta)\lambda_2]}{(\beta+\delta)} \tag{9.64}$$

$$\gamma = \frac{-(1-\delta)(2-\delta) \times [\lambda_1 - (2+\beta)\lambda_2]}{(\beta+\delta)} \tag{9.65}$$

USING R ——————————— ——————————— USING R

For a demonstration of the Wakeby distribution, a Wakeby is defined with a $\lambda_1 = 0$, $\lambda_2 = 1/\sqrt{\pi}$, and $\tau_4 = 0.1226$, which are L-moments consistent with those of the standard Normal distribution. Such a Wakeby is defined in example 9–12. The example also sets up limits for τ_3 and τ_5. These two L-moments describe successive asymmetry of the distribution.

9–12

```
F  <- seq(0.001,0.999, by=0.001) # useful nonexceedance values
L1 <- 0; L2 <- 1/sqrt(pi); T4 <- 0.1226 # consistent with
      # standard Normal distribution

t3edge  <- 0.7 # upper limit
t3range <- t3edge - -t3edge # mirrored limits
T3 <- seq(-t3edge,t3edge, by=0.05)

t5edge  <- 0.1 # upper limit
t5range <- t5edge - -t5edge # mirrored limits
T5 <- seq(-t5edge,t5edge, by=0.05)

nowak <- 0 # number of Wakeby's fit
nogpa <- 0 # number of Generalized Pareto's fit instead
```

The demonstration continues in example 9–13 by calling an effectively empty plot with proper limits. Nested `for()` loops are then sweep through τ_3 and τ_4 between the respective limits of ± 0.7 and ± 0.1, and plot the Wakeby for these L-moments in figure 9.6. The example skips invalid combinations of the L-moments, and the warnings normally produced by `are.lmom.valid()` are suppressed using the `options()` function.

9–13

```
ops <- options(warn = -1) # save options, and turn warnings off
#pdf("wakskewness_sweep.pdf", version="1.4")
plot(c(),c(), xlim=c(-3,3), ylim=c(-5,5),
             xlab="STANDARD␣NORMAL␣DEVIATE",
             ylab="QUANTILE") # empty plot with good limits
for(t3 in T3) {
  for(t5 in T5) {
    lmr <- vec2lmom(c(L1,L2,t3,T4,t5)) # set the L-moments
    if(! are.lmom.valid(lmr)) next # skip if they are invalid
    wak <- parwak(lmr) # compute Wakeby parameters
    cat(c("tau3␣=", t3, "␣and␣tau5␣=", t5, "\n"))

    if(wak$ifail == 2) { # GPA fit instead, red lines
       lines(qnorm(F),quawak(F,wak),
             col=rgb(1,0,0,0.3), lwd=0.5, lty=2)
       nogpa <- nogpa + 1 # count of Generalized Pareto fits
       cat("␣␣␣Generalized␣Pareto␣fit\n")
       next
    }

    r <- 0   # change colors according to tau3 and tau5 vals
    g <-  (t3edge - t3)/t3range # less green, tau3 increasing
    b <- (t5 - -t5edge)/t5range #  more blue, tau5 increasing
```

```
    lines(qnorm(F),quawak(F,wak), lwd=1,
          col=rgb(r,g,b, 0.8))
    nowak <- nowak + 1 # count of Wakeby fits
    cat("␣␣␣Wakeby␣fit\n")
    Sys.sleep(1) # to pause before moving on to next tau5
  }
  Sys.sleep(2) # to pause before moving on to next tau3
}
legend(-3,4,c("Generalized␣Pareto␣distribution",
              "Wakeby␣distribution"),
           lwd=c(0.5,1), lty=c(2,1))
#dev.off()
options(ops) # restore the options, warnings turned back on

# Now show how many of each distribution was returned by the
# Wakeby parameter estimation algorithm of parwak()
cat(c("No.␣Wakeby=",nowak,"␣and␣no.␣GPA=",nogpa,"\n"))
No. Wakeby= 196  and no. GPA= 222
```

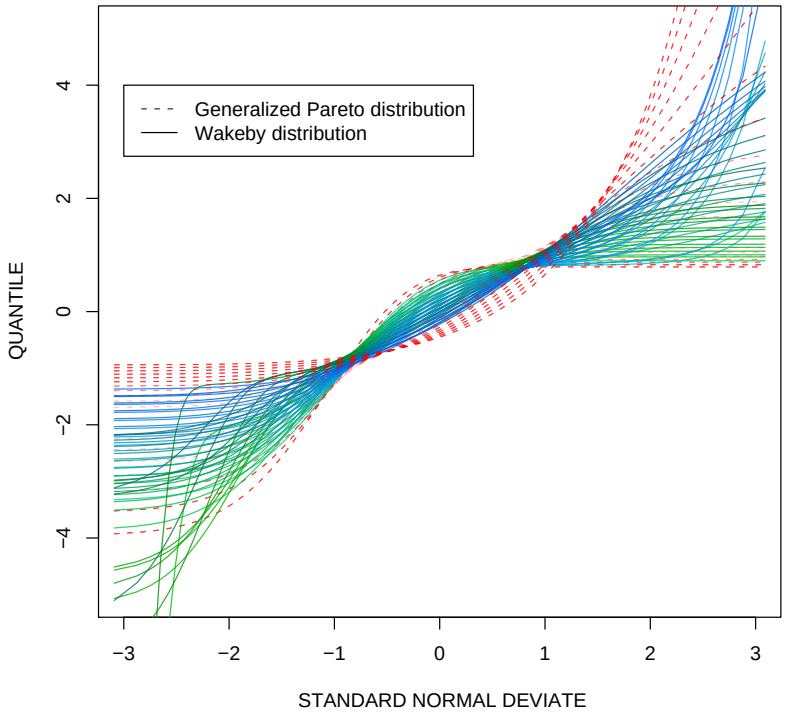

Figure 9.6. Comparison of Wakeby distributions (and Generalized Pareto, if applicable, dashed lines) for λ_1, λ_2, and τ_4 consistent with standard Normal distribution and τ_3 and τ_5 swept through ± 0.7 and ± 0.1, respectively from example 9–13. Decreasing green is equivalent to increasing τ_3, and increasing blue is equivalent to τ_5 increasing.

The figure shows the substantial range in distribution geometry that can be represented by the Wakeby as computed by the algorithm in `parwak()`. The color of each Wakeby distribution changes according to the values for τ_3 and τ_5. Unfortunately, the constraints of grey-scale printing limit the effectiveness of the figure. Decreasing green is equivalent to increasing τ_3, and increasing blue is equivalent to τ_5 increasing. Finally, the counts of Wakeby and Generalized Pareto fits are shown in the last line of example 9–13. ◀

The Wakeby distribution clearly has some complex shapes, and the PDF of the distribution aids in visualization. In example 9–14, consistency with select L-moments of the standard Normal distribution ($\lambda_1 = 0$, $\lambda_2 = \pi^{-1/2}$, and $\tau_4 = 0.1226$) is set up as was done in example 9–12. Next, a function `myWAK()` is created to plot PDFs for values of τ_3 and τ_5 that are passed as arguments in `t3` and `t5`, respectively. The function uses `cdfwak()`, `check.pdf()`, and `quawak()` functions.

9–14

```
F  <- seq(0.001,0.999, by=0.001) # useful nonexceedance values
L1 <- 0; L2 <- 1/sqrt(pi); T4 <- 0.1226 # consistent with
      # standard Normal distribution

"myWAK" <- function(t3,t5) {
  lmr <- vec2lmom(c(L1,L2,t3,T4,t5)) # set the L-moments
  if(! are.lmom.valid(lmr)) { # test the validity
     warning("L-moments␣are␣not␣valid")
     return(1)
  }
  PARwak <- parwak(lmr) # compute Wakeby parameters
  mydis <- NULL # which distribution chosen by the algorithm
  if(PARwak$ifail == 0) {
    mydis <- "Wakeby␣fit"
  }
  else if(PARwak$ifail == 1) {
    mydis <- "Wakeby␣fit␣with␣xi␣=␣0"
  }
  else if(PARwak$ifail == 2) {
    mydis <- "Generalized␣Pareto␣fit␣instead"
  }
  else {  return(1)  }
  mystr <- paste("Tau3=",t3,"␣␣Tau5=",t5,sep="␣")
  # Perform the plotting operations
  layout(matrix(1:2,nrow=2)) # two plots, top and bottom
  x <- quawak(F,PARwak)
  plot(x, cdfwak(x,PARwak), type="l", lwd=2,
       col=PARwak$ifail+1, lty=PARwak$ifail+1,
       xlab="x", ylab="F")
  mtext(mydis)
```

```
  check.pdf(pdfwak, PARwak, plot=TRUE)
  lines(x,pdfwak(x,PARwak), lty=2, col=1, lwd=5)
  mtext(mystr)
  return(PARwak) # return the parameters in case needed
}
```

The `myWAK()` function is called six separate times in example 9–15 and the results are shown in figures 9.7–9.12. The parameters of the Wakeby are returned with each call to `myWAK()`, but these are not shown in the example. The thick dashed line on the PDF (bottom plot) is the PDF of the Wakeby superimposed on the results of the `check.pdf()` function. For these six figures, the Generalized Pareto is shown only in figure 9.11 because the Wakeby could not be fit to $\tau_3 = 0.1$ and $\tau_5 = 0.5$.

9–15

```
myWAK(   0,     0);   myWAK( 0.1,     0)
myWAK(-0.1,     0);   myWAK( 0.1,  0.1)
myWAK( 0.1,   0.5);   myWAK(-0.2, -0.1)
```

The figures follow on the next three pages. ◀

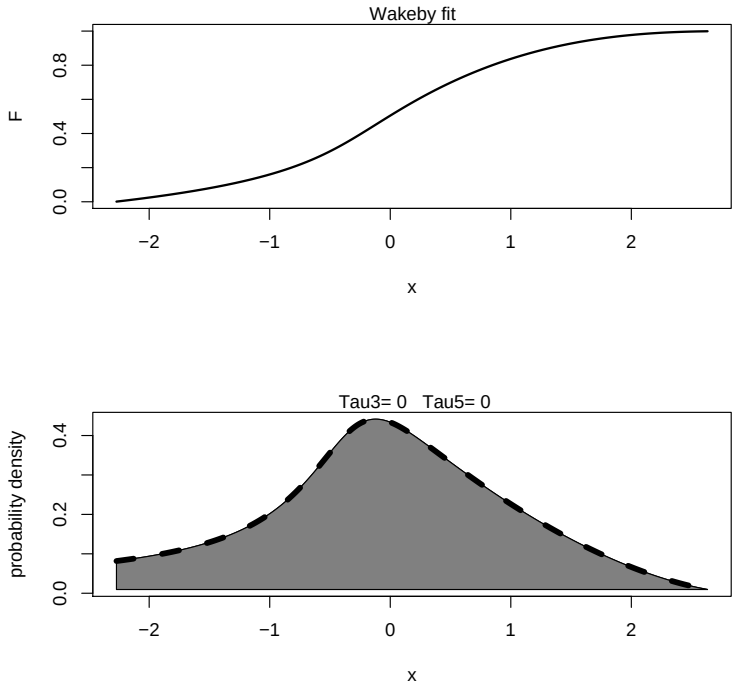

Figure 9.7. Comparison of QDF and PDF of Wakeby distribution (or Generalized Pareto, if applicable) for λ_1, λ_2, and τ_4 consistent with standard Normal distribution and $\tau_3 = 0$ and $\tau_5 = 0$ from example 9–15

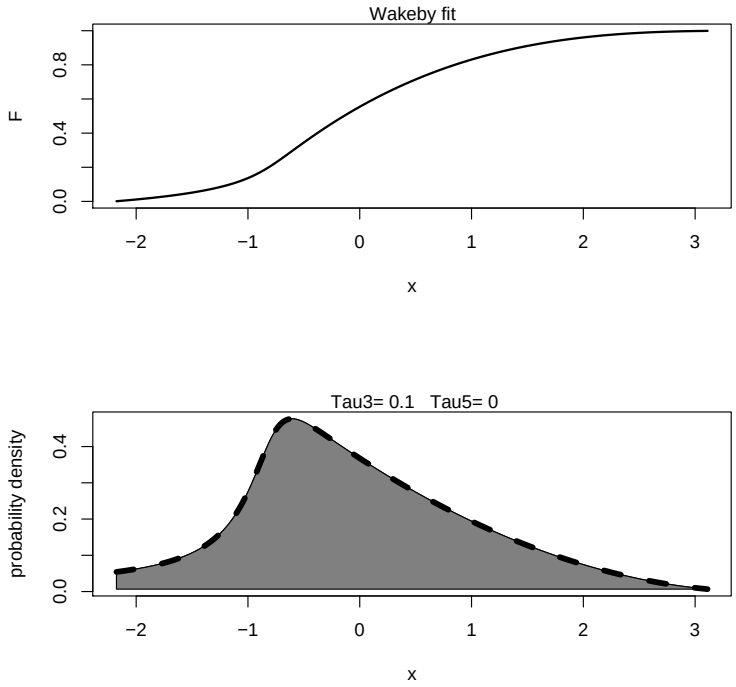

Figure 9.8. Comparison of QDF and PDF of Wakeby distribution (or Generalized Pareto, if applicable) for λ_1, λ_2, and τ_4 consistent with standard Normal distribution and $\tau_3 = 0.1$ and $\tau_5 = 0$ from example 9–15

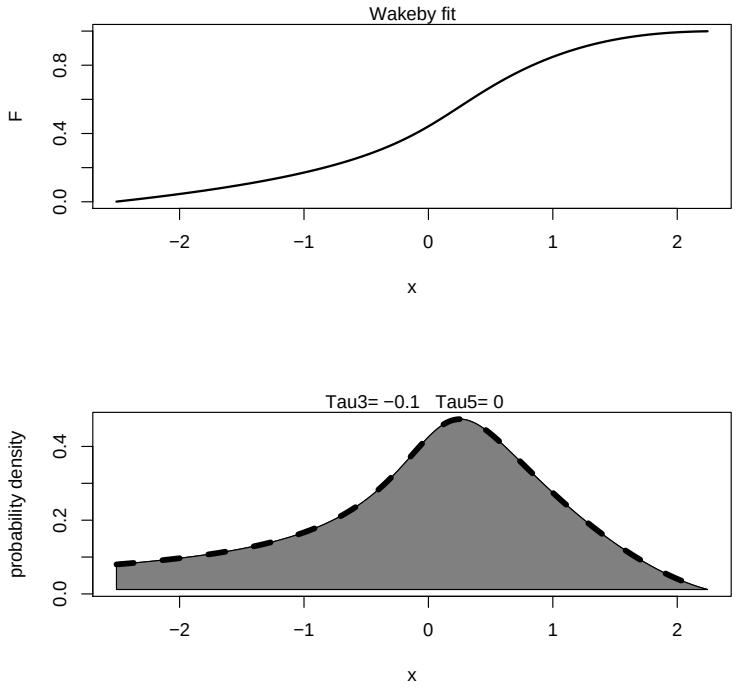

Figure 9.9. Comparison of QDF and PDF of Wakeby distribution (or Generalized Pareto, if applicable) for λ_1, λ_2, and τ_4 consistent with standard Normal distribution and $\tau_3 = -0.1$ and $\tau_5 = 0$ from example 9–15

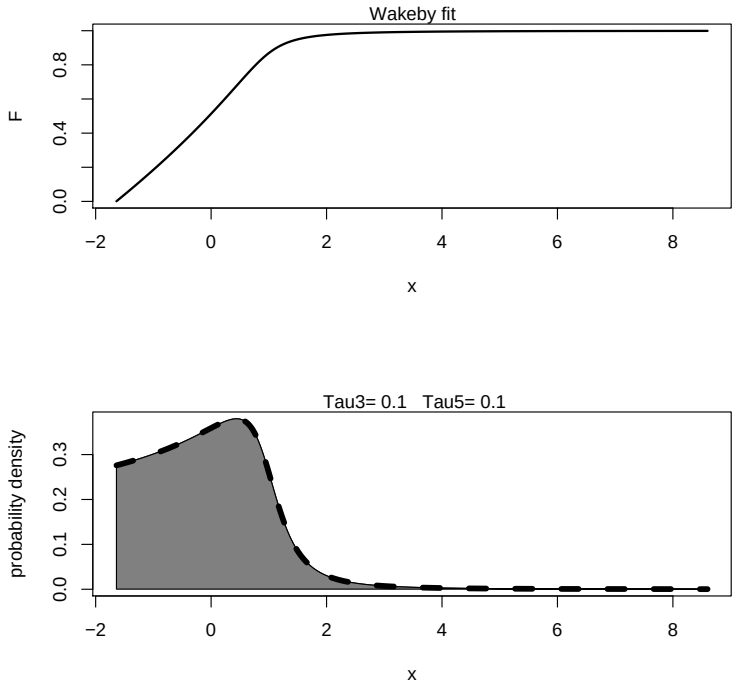

Figure 9.10. Comparison of QDF and PDF of Wakeby distribution (or Generalized Pareto, if applicable) for λ_1, λ_2, and τ_4 consistent with standard Normal distribution and $\tau_3 = 0.1$ and $\tau_5 = 0.1$ from example 9–15

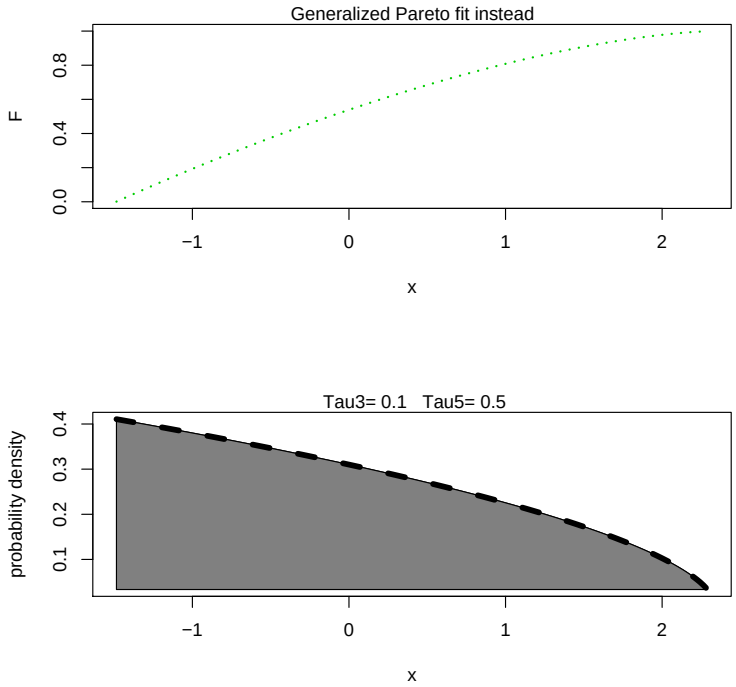

Figure 9.11. Comparison of QDF and PDF of Wakeby distribution (or Generalized Pareto, if applicable) for λ_1, λ_2, and τ_4 consistent with standard Normal distribution and $\tau_3 = 0.1$ and $\tau_5 = 0.5$ from example 9–15

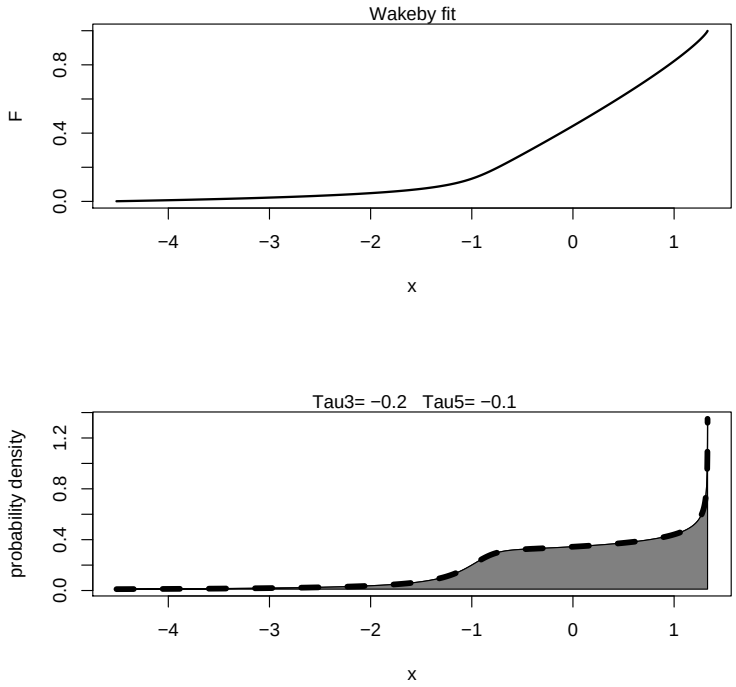

Figure 9.12. Comparison of QDF and PDF of Wakeby distribution (or Generalized Pareto, if applicable) for λ_1, λ_2, and τ_4 consistent with standard Normal distribution and $\tau_3 = -0.2$ and $\tau_5 = -0.1$ from example 9–15

9.3 Special Demonstration of Distributions

The distributions supported by the *lmomco* package are extensively documented and demonstrated in Chapters 7–9. A special demonstration of most of these distributions is now appropriate.

The shapes are compared to the PDF for twelve distributions fit to the same L-moments. Example 9–16 shows some suggested steps how the comparison can be effectively done with the function nomenclature of the *lmomco* package. The example begins by specifying some L-moments using the now familiar construct with the `vec2lmom()` function. A full listing of distribution abbreviations for the *lmomco* package is returned by the `dist.list()` function. A `for()` loop is used to iterate through the distribution list and determine the global end points of the quantiles so that for later plotting purposes the horizontal axis for each PDF will have the same extent. Seven distributions are left off—the Cauchy because TL-moments are needed, the Generalized Lambda to avoid treatment for multiple solutions, Kumaraswamy to avoid a nonconvergence error and log-Normal3, Rayleigh, Reverse Gumbel, and Rice because four more distributions would simply compress the graphical output in figure 9.13 too much for effective presentation.

9–16

```
L1 <- 900; L2 <- 500; T3 <- 0.1; T4 <- -0.1; T5 <- 0.04
lmr <- vec2lmom(c(L1,L2,T3,T4,T5))
dist <- dist.list() # return list of dist. abbreviations
dist <- dist[dist != "cau" & dist != "gld"    &
             dist != "kur" & dist != "ln3"    &
             dist != "ray" & dist != "revgum" &
             dist != "rice"]
F <- nonexceeds()   # convenient values
my.min <- Inf; my.max <- -Inf # for global end points
for(d in dist) { # used to find global end points
  my.para <- lmom2par(lmr, type=d)
  x <- qlmomco(F,my.para)
  my.min <- min(x,my.min); my.max <- max(x,my.max)
}
```

The demonstration continues in example 9–17 with similar structure to example 9–16 but with the addition of plotting operations. The results are shown in figure 9.13. For the example the `qlmomco()` and `dlmomco()` functions respectively provide the QDF and PDF of the respectively distribution in the variable `d`. A previously not identified function, `prettydist()`, returns the corresponding full name for the distribution so that the individual plots can be labeled.

9–17

```
#pdf("alldist.pdf")
n <- length(dist) # how long is that list?
layout(matrix(1:n, ncol=n%/%4)) # at time of writing---three cols
for(d in dist) {
  my.para <- lmom2par(lmr, type=d)
  x <- qlmomco(F,my.para)
  plot(x, dlmomco(x,my.para), type="l", ylab = "DENSITY",
       xlim=c(my.min,my.max))
  mtext(prettydist(d)) # place distribution name above plot
}
#dev.off()
```

Inspection of the figure suggests that the PDF of the Generalized Pareto and Wakeby distributions are similar. Actually, they are identical. The Wakeby algorithm could not find a solution and reverted to that for the Generalized Pareto. Therefore, the plot is technically mislabeled in the figure. The Generalized Pareto looks so different from the other five three-parameter distributions because at $\tau_3 = 0.1$, the Generalized Pareto has much less τ_4 than the other distributions. Readers are guided to figure 10.6 in the context of τ_3 and τ_4 comparison. Finally, readers might find it informative to experiment with examples 9–16 and 9–17 by changing the τ_3, τ_4, and τ_5 values in `T3`, `T4`, and `T5`, respectively. ◀

Now, considering τ_4 in more detail, the PDF for the Kappa distribution in figure 9.13 has considerable distortion on the left and right tails because of extremely large probability density on the edges (in the tails). Why does the PDF of the Kappa look so different from the other examples?

Before answering the question, an alternative plot of the PDF is useful and created by example 9–18. The results are shown in figure 9.14. The figure clearly provides for better resolution and shows that numerical singularities do not exist as the lower resolution of figure 9.13 suggests.

9–18

```
#pdf("kappdf.pdf")
check.pdf(pdfkap,parkap(lmr), plot=TRUE,
          plotlowerF=0.1, plotupperF=0.9)
#dev.off()
```

Similar plotting of narrower tails, such as provided by the options `plotlowerF` and `plotupperF` to `check.pdf()`, occasionally is needed for this and other distributions for effective graphical depiction of the PDF structure for specific combinations of parameters. The `check.pdf()` function thus provides a convenient interface for PDF plotting purposes.

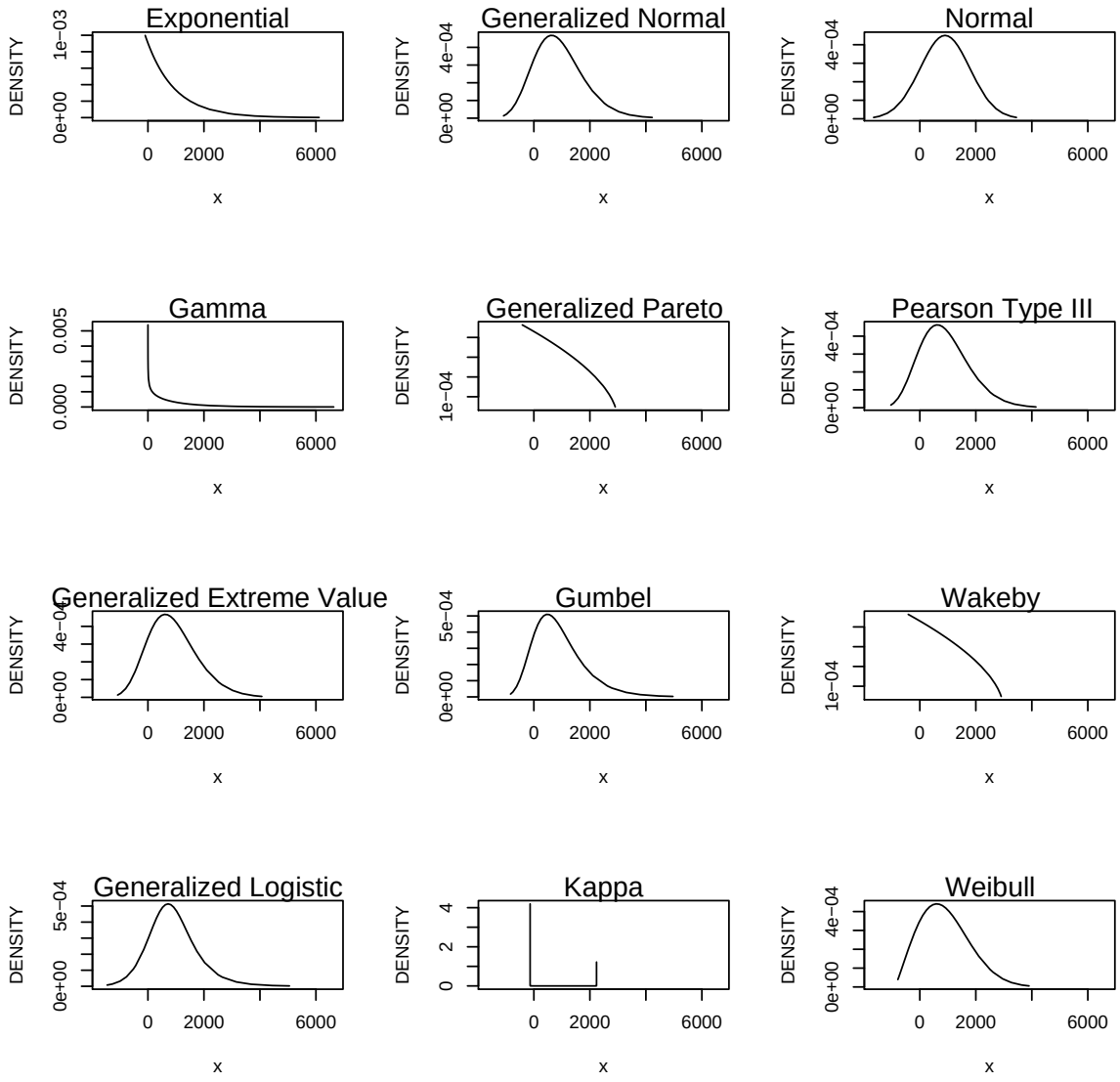

Figure 9.13. Comparison of PDF for twelve distributions fit to L-moments from example 9–17

Returning to the answer posed by the earlier question, for the example L-moments here, the negative τ_4 used in the example produces anti-peaking. (There is no central peak or mode.) This is the effect of $\tau_4 < 0$. The other 11 (well 10, considering that the Generalized Pareto is shown twice) in figure 9.13 all have less than four parameters. As a result, none of the distributions is actually fit to the specified τ_4. Each has its own τ_4 of course, but none can acquire the condition of $\tau_4 < 0$. Therefore, the anti-peaking distribution geometry of

the L-moments in variable `lmr` in example 9–16 is only mimicked—that is, seen—by the Kappa distribution. ◀

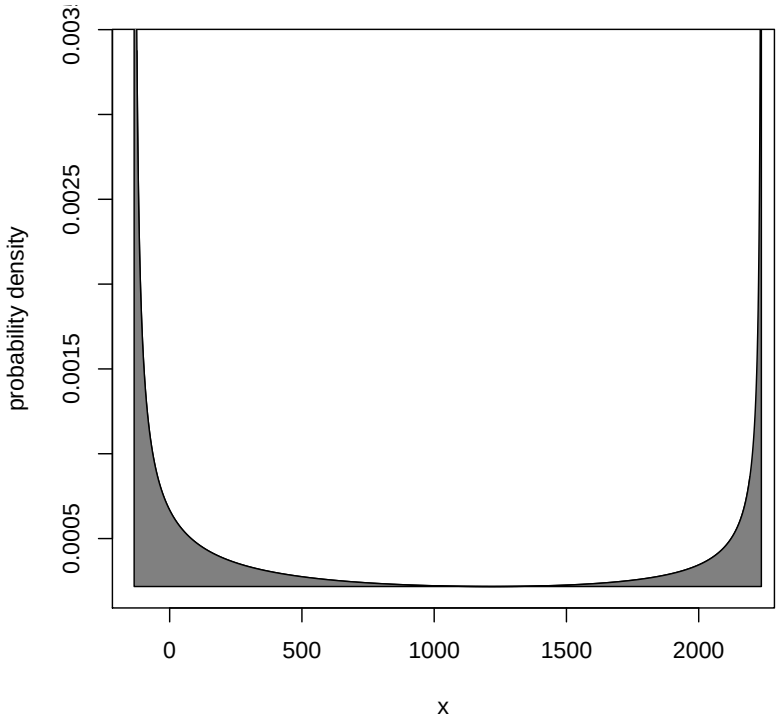

Figure 9.14. Alternative depiction of PDF of Kappa distribution shown in figure 9.13 from example 9–18

Finally, to end this section and again using the Kappa distribution, the influence of positive and negative τ_4 and the relation to the peakedness of the distribution is demonstrated. In example 9–19, two Kappa distributions with $\mu = 0$, $\sigma = 1$, $\lambda_2 = \sigma\sqrt{\pi}$, $\tau_3 = 0$, and two τ_4 values are plotted in figure 9.15.

9–19

```
#pdf("kappeak.pdf")
layout(matrix(1:2, ncol=1))
top <- vec2lmom(c(0,1/sqrt(pi),0, 0.01)) # positive L-kurtosis
bot <- vec2lmom(c(0,1/sqrt(pi),0,-0.01)) # negative L-kurtosis
check.pdf(pdfkap,parkap(top), plot=TRUE,
          plotlowerF=0.1, plotupperF=0.9)
check.pdf(pdfkap,parkap(bot), plot=TRUE,
          plotlowerF=0.1, plotupperF=0.9)
#dev.off()
```

The differences in the PDFs are striking, yet figure 9.15 has a pleasant sort of symmetry for two distributions having $\tau_3 = 0$ and equivalent magnitude τ_4 that differ in sign. ◀

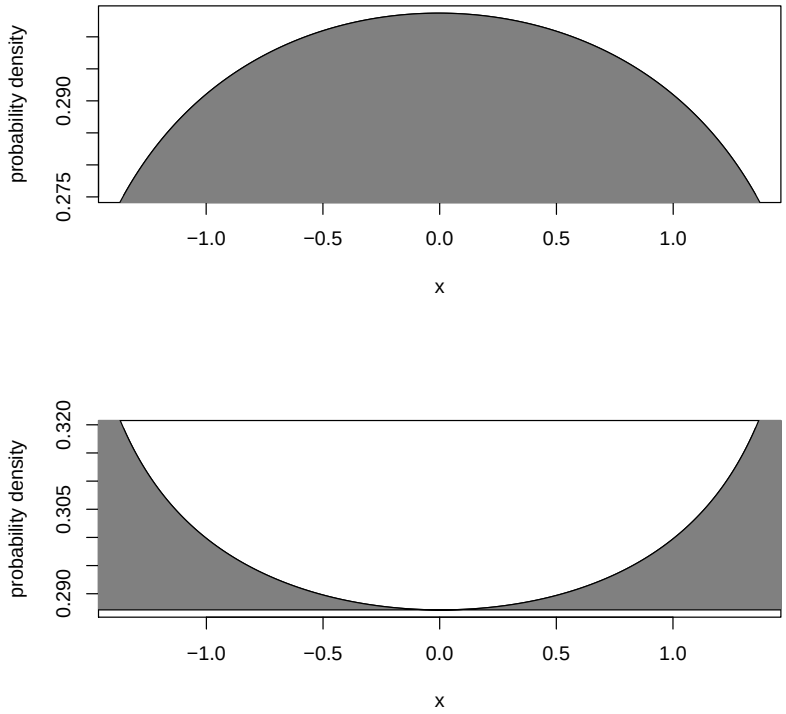

Figure 9.15. Comparison of symmetrical PDFs of two Kappa distributions having positive (top) and negative (bottom) τ_4 values of equivalent magnitude from example example 9–19

9.4 Summary

Four- and more parameter distributions are substantially more flexible than are three-parameter distributions. The four- and more parameter distributions considered in this chapter are fit to at least the first four L-moments of the data. Both the *lmomco* and *lmom* packages provide support for the Kappa and Wakeby distributions. The *lmomco* package in particular offers the addition of the Generalized Lambda distribution; the Generalized Lambda can be problematic in practice because multiple solutions can exist. The additional flexibility of four- and more parameter distributions requires reliable estimation of at least four L-moments and thus larger samples sizes are required than are needed for lower-order distributions. The tail shapes of four- and more parameter distributions

might provide for more accurate quantile estimation in the far tails provided that sample sizes are sufficiently large to support reliable $\hat{\tau}_4$ and $\hat{\tau}_5$ estimation. Conversely, these distributions might provide for substantially biased quantile estimates in circumstances in which sample sizes are insufficient to make reliable estimation of distribution parameters. The 19 examples repeat themes established previously in this dissertation, but some of the Generalized Lambda and Wakeby examples demonstrate the considerable nuances of these distributions in practice. Finally, Section 9.3 shows that the high-level functionality of the *lmomco* package facilitates experimentation with L-moments and distributional form, and that section effectively closes the content of Chapters 7 and 8.

- The examples for the Kappa distribution consider the distribution of annual peak streamflow data for a location in Texas that are contained within the *lmomco* package. The streamflow exhibits remarkable variability and skewness. The Kappa distribution is fit to the L-moments and plotted. The Kappa acceptably fits the right tail (the flood-flow tail and tail of interest). Subsequent examples fit the Generalized Normal distribution to the L-moments along with the log-Normal. Although not attaining the quality of fit of the Kappa mostly because of having only three parameters, the Generalized Normal also mimics the geometry of the data. The log-Normal (two parameter) does not have an acceptable fit—the data possess more curvature than this distribution can attain.
- The examples for the Generalized Lambda distribution repeat consideration of the distribution of annual peak streamflow data used for the Kappa examples. The Generalized Lambda is fit to the L-moments of the data and is plotted along side the Kappa. The default Generalized Lambda solution returned by a function of *lmomco* exhibits excessive flattening in the right tail (again the flood-flow tail and tail of interest). Therefore, a secondary solution is found that more closely matches the Kappa. The difficulty in using the Generalized Lambda in practice thus is shown.
- The examples for the Trimmed Generalized Lambda distribution demonstrate an approximation to the Cauchy distribution.
- The examples for the Wakeby distribution are extensive because of the complexity of the five-parameter version of the Wakeby. Various QDF variations that are departures from the λ_1, λ_2, and τ_4 of the standard Normal distribution are created by changing the τ_3 and τ_5 of the distribution and all plotted on the same figure. The examples also provide various PDF variations that are departures from an otherwise standard

Normal distribution. The PDF variations are created by changing the τ_3 and τ_5 values, and these PDFs are separately plotted. For one of the PDFs, a Wakeby could not be fit and the backup fit of the Generalized Pareto is shown instead. So the examples, do show the fitting of a Generalized Pareto in circumstances in which the Wakeby is not applicable.

Chapter 10
L-moment Ratio Diagrams

In this chapter, I present discussion of L-moment ratio diagrams. These diagrams are extremely useful in application of L-moment statistics because they permit differentiating between many distributions. The diagrams permit visual description of distribution-specific interrelations between selected L-moment ratios. The diagrams are commonly used for ad-hoc selection (a sort of goodness-of-fit) of a distribution from a ensemble of candidate distributions. The diagrams are quite common in the L-moment literature and are an important part of exploratory analysis. Compared to other chapters, this chapter is the most similar to a conventional paper on the topic and should be especially accessible to many readers. This chapter is central to distributional analysis with L-moment statistics using R.

10.1 Introduction

Probability distributions are distinguished by their formal mathematical definition, moments, and respective parameter values. As a result, distributions have specific and generally unique intra-relations (within distribution) between moments and parameters. As seen in this chapter, the intra-moment relations of L-moments are a convenient and powerful tool for discriminating between distributional form. The intra-moment relations also provide a means for ad hoc, yet reliable, judgement of goodness-of-fit, and hence, the relations guide the process of distribution evaluation and selection from a suite of candidate distributions.

As a means to guide the selection of a distribution, a convenient graphical construct, which is termed a **moment ratio diagram**, provides a visualization of intra-moment relations. The moment ratio diagram often is used to depict the relation between relative vari-

ability and skewness (CV versus $\hat{G}$ or τ_2 versus τ_3) or more often the relation between skewness and kurtosis ($\hat{K}$ versus $\hat{G}$ or τ_3 versus τ_4). For the purpose of this dissertation, the focus is on **L-moment ratio diagrams** as detailed in Hosking (1990), Vogel and Fennessey (1993), Peel and others (2001), and many others, in general, and L-moment ratio diagrams of τ_3 and τ_4, in particular. L-moment ratio diagrams of τ_3 and τ_4 are especially useful in evaluation of distributional form in a framework that is largely independent of the location and scale characteristics of the distribution. Such L-moment ratio diagrams are frequently depicted in contributions to the L-moment literature. The importance of these diagrams and their variants to L-moment practitioners is hard to overemphasize. For example, Hosking (2007b) suggests that the L-moment ratio diagram of τ_4 and τ_6 is useful for distinguishing between symmetric distributions and a financial application is seen in Hosking and others (2000) and Kerstens and others (2010).

L-moment ratio diagrams address, but not completely solve, the nontrivial problem of distribution evaluation and selection of distributional form for arbitrary data. The diagrams also provide convenient tools for studying the applicability of selected distributions. Because distributions have unique points, lines, or regions on an L-moment ratio diagram, the diagram can be used to evaluate the portion of the $\{\tau_3, \tau_4\}$-parameter space occupied by the distribution that is most similar to that of the data. Data often are represented by samples from different locations, individuals, measurement campaigns, or studies, and hence form a localized cloud on the diagram.

This chapter describes L-moment ratio diagrams for complete distributions or whole samples. The diagrams for censored distributions and samples are described in much literature that includes Hosking (1995) and Zafirakou-Koulouris and others (1998). Such diagrams are not considered here. Hypothesis testing involving goodness-of-fit also is not considered here. However, Liou and others (2008) do provide one of relatively few studies related to quantification of distribution-fit judgement using L-moment ratio diagrams; Liou and others (2008) provide a few additional citations.

10.2 Assessment of Distribution Form using L-moment Ratio Diagrams

L-moment ratio diagrams are straightforward to use for assessment of distribution form and are in widespread use for this purpose by L-moment practitioners. In their most common form, the diagrams depict the relation between τ_3 (horizontal axis) and τ_4 (vertical

axis). Although perhaps obtuse at first introduction, the diagram has numerous components that will be familiar by the end of this chapter.

The diagrams are effective tools for visualizing the relations between the pair $\{\tau_3, \tau_4\}$ of a distribution and the locations of the $\{\hat{\tau}_3, \hat{\tau}_4\}$ from samples. The relations or spatial differences on the diagram help guide the analyst in the selection of distributions. An example of a well-typeset L-moment ratio diagram, which is derived from Asquith and others (2006), is shown in figure 10.1. The diagram shows the intra-moment relation between τ_3 and τ_4 for selected distributions as well as hundreds of samples from two phenomena types. Several interpretations of the contents of the figure are possible as shown by the following discussion.

Two-parameter distributions, when possessing location and scale parameters, have constant values for both τ_3 and τ_4 regardless of the values for λ_1 and λ_2, and such two-parameter distributions plot as special points on the diagram. For example, the Normal distribution, which is not plotted in the figure because of the selected horizontal axis scale, has $\tau_3^{\text{nor}} = 0$ and $\tau_4^{\text{nor}} \approx 0.123$ (see Section 7.2.1). Whereas the Exponential distribution, which is shown in figure 10.1, has $\{\tau_3^{\text{exp}}, \tau_4^{\text{exp}}\} = \{0.333, 0.167\}$. The two-parameter Gamma distribution is different because the distribution lacks a location parameter and does not have constant values for τ_3 and τ_4. Instead, the Gamma exists along the line of the Pearson Type III distribution. The Gamma is further discussed later in this chapter.

In contrast to two-parameter distributions, three-parameter distributions, when possessing location, scale, and shape parameters, trace a unique trajectory or curvilinear path through the $\{\tau_3, \tau_4\}$-parameter space of the diagram. The trajectories of four selected three-parameter distributions are shown in figure 10.1: the Generalized Extreme Value, Generalized Logistic, Generalized Pareto, and Pearson Type III distributions.

In contrast to three-parameter distributions, distributions with more than one shape parameter—generally distributions with four or more parameters—cover or occupy a region or regions of the diagram. The Kappa distribution is such an example, and, for a given τ_3, the Kappa occupies the region below the τ_4 of the Generalized Logistic and above the τ_4 of the theoretical limits of the L-moments. The bottom graph in figure 10.1 shows the range of the Kappa by the extent of the line with arrows on both ends. The Generalized Lambda (Karvanen and Nuutinen, 2008) and Wakeby (Hosking and Wallis, 1997) distributions have more complex parameter spaces and are not easily illustrated.

The L-moment ratio diagram in figure 10.1 also shows individual $\hat{\tau}_3$ and $\hat{\tau}_4$ values (the two circle types) computed from real-world data. The L-moment ratio diagram was

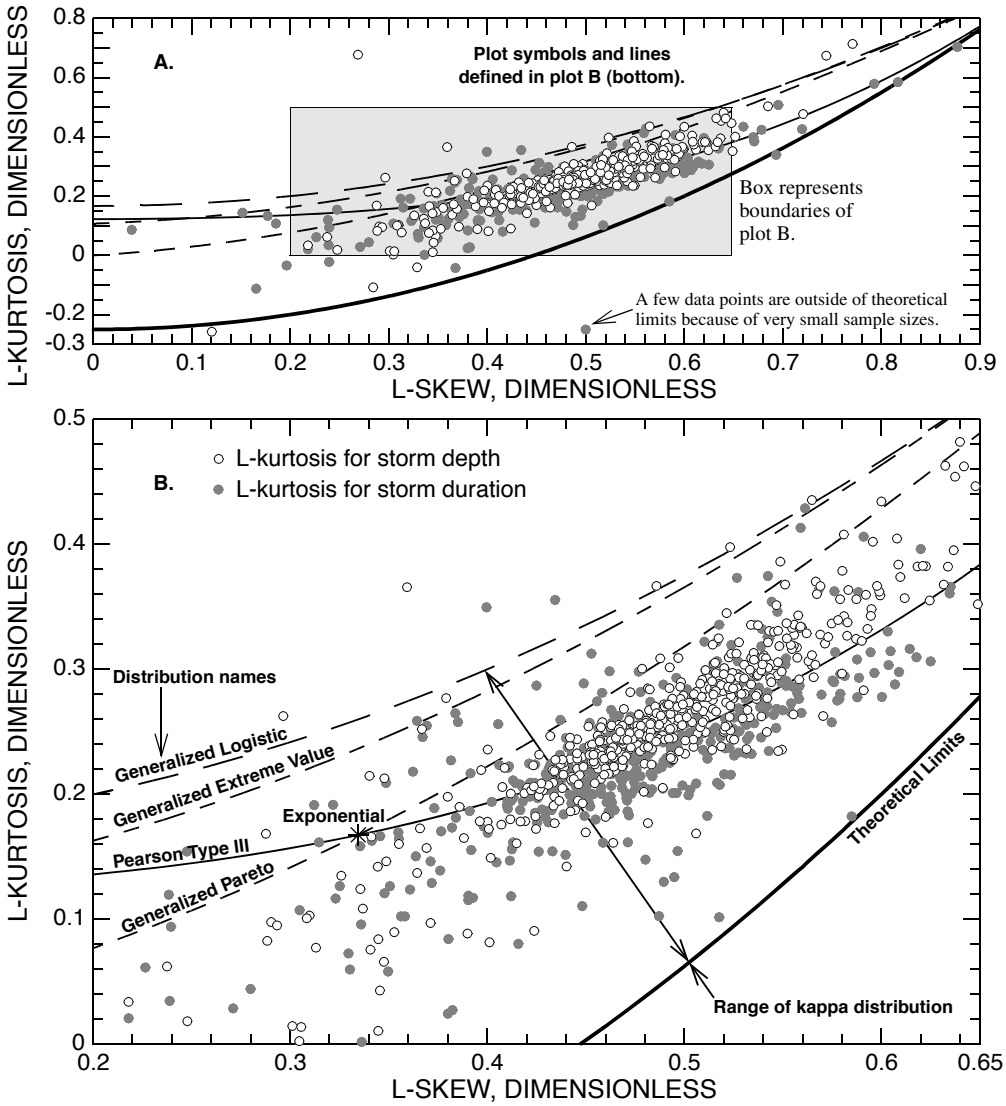

Figure 10.1. High-quality L-moment ratio diagram showing L-skew and L-kurtosis of selected distribuions and sample values for storm depth and storm duration from Asquith and others (2006)

developed as part of a large-scale study of the statistics of storms throughout eastern New Mexico, Oklahoma, and Texas (Asquith and others, 2006). For each of 774 locations with hourly raingages, literally thousands of storms per location were extracted, and the sample L-moments of storm depth and the sample L-moments of storm volume for each location were computed. The $\hat{\tau}_3$ and $\hat{\tau}_4$ for storm depth (total depth of rainfall) are plotted in the figure as open circles, and the $\hat{\tau}_3$ and $\hat{\tau}_4$ for storm duration are plotted as grey circles.

Several interpretive observations of the data shown in the diagram (fig. 10.1) can be made:

1. The top graph shows that both rainfall phenomena have positive skewness and mild kurtosis;
2. The data have a strong tendency to plot in a relatively restricted portion of the L-moment ratio diagram. It is important to note that the locations of the raingages are distributed throughout a large geographic region (on the order of $370{,}000$ square miles); and
3. The central tendency of $\hat{\tau}_3$ and $\hat{\tau}_4$ is important. The $\hat{\tau}_3$ and $\hat{\tau}_4$ for either storm phenomena (storm depth and storm duration) cluster around $\{\hat{\tau}_3, \hat{\tau}_4\} = \{0.50, 0.27\}$ for storm depth and $\{\hat{\tau}_3, \hat{\tau}_4\} = \{0.48, 0.23\}$ for storm duration. These two pairs of $\hat{\tau}_3$ and $\hat{\tau}_4$ are derived from Asquith and others (2006, tables 5 and 6). The overlap of the data clouds and general numerical similarity of the $\hat{\tau}_3$ and $\hat{\tau}_4$ of the two phenomena suggests that the general asymmetry and shape of the two unknown parent distributions of distinct phenomena are similar.

Continuing with interpretations of figure 10.1, the Pearson Type III distribution curve passes close to the $\{\tau_3, \tau_4\}$ pairings for both storm depth and storm duration. Hence, a conclusion is, should a three-parameter distribution be selected, that the Pearson Type III would provide a favorable choice for generalized modeling of these rainfall phenomena. For additional interpretation, the $\hat{\tau}_3$ and $\hat{\tau}_4$ values are almost universally below the lines for the Generalized Extreme Value and Generalized Logistic distributions, and the sample sizes are large enough to judge that τ_3 and τ_4 are reliably estimated. Both the Generalized Extreme Value and Generalized Logistic distributions would be poor choices from the perspective of the regional (geographic) form of the parent distribution. The Generalized Pareto, although a better choice than either the Generalized Extreme Value or Generalized Logistic, generally would have τ_4^{gpa} larger than the sample data. An alternative choice over the Pearson Type III would be the four-parameter Kappa distribution because the Kappa could match the record-length weighted mean values of $\hat{\tau}_3$ and $\hat{\tau}_4$. Asquith and others (2006) concluded that the Kappa distribution is preferable to model the unknown parent distribution of rainfall depth and duration in their study area.

To further illustrate interpretations of L-moment ratio diagrams, numerical experiments are performed using functions from the *lmomco* package. The demonstration begins in example 10–1 by specifying the first four L-moments using the `vec2lmom()` function. For later reference, primary interest concerns the values for $\{\tau_3, \tau_4\} = \{0.4, 0.4\}$. Next, the parameters are computed using the `parXXX()` functions (see table 7.4 for a listing) for three selected distributions: Gamma (`gam`), Generalized Logistic (`glo`), and Pearson Type III (`pe3`) by the `parglo()`, `parpe3()`, and `pargam()` functions, respectively.

10–1

```
t3  <- 0.4 # set L-skew
t4  <- 0.4 # set L-kurtosis
lmr <- vec2lmom(c(1000,500,t3,t4)) # create full list of L-
   moments
PARgam <- pargam(lmr)
PARglo <- parglo(lmr)
PARpe3 <- parpe3(lmr)
```

Continuing in example 10–2, two vectors are created to store sample values $\hat{\tau}_3$ and $\hat{\tau}_4$. These vectors are used for subsequent plotting operations in later examples.

10–2

```
t3gam <- vector(mode = "numeric")
t3glo <- t4glo <- t3pe3 <- t4pe3 <- t4gam <- t3gam
```

Example 10–3 establishes an arbitrary sample size of $n = 20$ and performs 50 simulations of three independent $n = 20$ drawings from the three distributions by the `rlmomco()` function. The `nsim=50` simulation size is too small for rigorous numerical study but is sufficiently large for effective demonstration of key concepts of L-moment ratio diagrams. The `rlmomco()` function dispatches n random F values to the appropriate `quaXXX()` functions (see table 7.3 for a listing). Following each sample drawing for each distribution, the L-moments are computed by the `lmoms()` function, and the $\hat{\tau}_3$ and $\hat{\tau}_4$ are stored in respective vectors.

10–3

```
n <- 20; nsim <- 50
for(i in seq(1,nsim)) {
  Q <- rlmomco(n,PARgam); tmp <- lmoms(Q)
  t3gam[i] <- tmp$ratios[3]; t4gam[i] <- tmp$ratios[4]

  Q <- rlmomco(n,PARglo); tmp <- lmoms(Q)
  t3glo[i] <- tmp$ratios[3]; t4glo[i] <- tmp$ratios[4]
```

```
  Q <- rlmomco(n,PARpe3); tmp <- lmoms(Q)
  t3pe3[i] <- tmp$ratios[3]; t4pe3[i] <- tmp$ratios[4]
}
```

The results of the simulation are plotted finally in figure 10.2 using example 10–4. In the example, the plot is initiated, and the $\{\hat{\tau}_3, \hat{\tau}_4\}$ values for Gamma, Generalized Logistic, and Pearson Type III distributions are plotted as open squares (`pch=0`), open triangles (`pch=2`), and filled circles (`pch=16`), respectively. For each of the three distributions, the mean values of the 50 values of $\hat{\tau}_3$ and $\hat{\tau}_4$ are plotted as the large symbol shapes. The intersection of the horizontal and vertical lines in the interior of the plot, which are drawn by the `segments()` function, cross at the location of the τ_3 and τ_4 of the population.

10–4

```
#pdf("lmr1.pdf")
plot(t3pe3,t4pe3, ylim=c(0,0.7), xlim=c(0,0.8), type="n",
                  xlab="L-SKEW", ylab="L-KURTOSIS")
points(t3gam,t4gam, pch=0)
points(t3glo,t4glo, pch=2)
points(t3pe3,t4pe3, pch=16, col=rgb(0.6,0.6,0.6))
points(mean(t3gam),mean(t4gam),pch=22,bg=rgb(1,1,1),   cex=3)
points(mean(t3glo),mean(t4glo),pch=24,bg=rgb(1,1,1),   cex=3)
points(mean(t3pe3),mean(t4pe3),pch=21,bg=rgb(.6,.6,.6),cex=3)
segments(t3,-1, t3,1); segments(-1,t4, 1,t4)
#dev.off()
```

Considerable interpretation of figure 10.2 can be made. Although substantial overlap is present, the points in the figure show differences in general plotting region, which are dependent on distribution type. For example, the data points for the Pearson Type III distribution (filled circles) generally plot as a cluster or otherwise define a region with τ_4 values less than $\hat{\tau}_4$ from the other two distributions. ◀

To further illustrate the interpretation of L-moment ratio diagrams, the sample size is increased by an order of magnitude to $n = 200$, and examples 10–3 and 10–4 rerun. The results are shown in figure 10.3. The larger sample size reduces sample variability, and therefore, the data points in figure 10.3 define more visually distinct or separable regions that have more consistency with the trajectory of each parent distribution. These regions are identifiable as separate—this is a major feature of the diagrams. The regions show that the three distributions (and others) are readily distinguished on L-moment ratio diagrams of τ_3 and τ_4. The diagrams hence can guide analysts towards a distribution that might be

most appropriate or at a minimum suggest less appropriate distributions to model the phenomena under study.

Because each is a three-parameter distribution, the Generalized Logistic and Pearson Type III distributions are each fit to the same $\tau_3 = 0.4$ (the vertical line), but each is not fit to the given $\tau_4 = 0.4$ (the horizontal line). As a result, the mean $\hat{\tau}_3$ of these two distributions (large triangle and filled circle, respectively) should plot reasonably close to the $\tau_3 = 0.4$ line. Because the τ_4 of the Generalized Logistic distribution is intrinsically larger than that of the Pearson Type III, the large triangle is plotted above the large filled circle—that is, the large triangle (Generalized Logistic) is more L-kurtotic than the large filled circle (Pearson Type III).

The two-parameter Gamma distribution conversely cannot be fit to $\tau_3 = 0.4$ because of even higher moment order and obviously not $\tau_4 = 0.4$. The Gamma distribution only is fit to $\lambda_1 = 1{,}000$ and $\lambda_2 = 500$ as provided by example 10–1. In this situation, the fitted Gamma has a τ_3^{gam} that is substantially less than the population τ_3. The Gamma and Pearson Type III distributions are closely related distributions. It is not a coincidence that the large square plots in the region where it does because the Gamma distribution exists along the Pearson Type III trajectory of $\{\tau_3^{\text{pe3}}, \tau_4^{\text{pe3}}\}$. Both the Gamma and Pearson Type III distributions are further discussed in the next section. ◀

10.3 Construction of L-moment Ratio Diagrams

This section addresses the construction of L-moment ratio diagrams using features of the *lmomco* package. L-moment ratio diagrams of τ_3 and τ_4 are readily constructed using a combination of distribution-specific tabulated values, analytical expressions, or numerical approximations.

Hosking and Wallis (1997, p. 208) report polynomial approximations for the characteristic $\{\tau_3, \tau_4\}$ relations for construction of L-moment ratio diagrams. The polynomial approximations are of the form

$$\tau_4 = \sum_{j=0}^{8} A_j \, {\tau_3}^j \tag{10.1}$$

where the coefficients A_j for selected three parameter distributions are listed in table 10.1. Hosking and Wallis (1997, p. 208) also report that the approximations produce τ_4 values

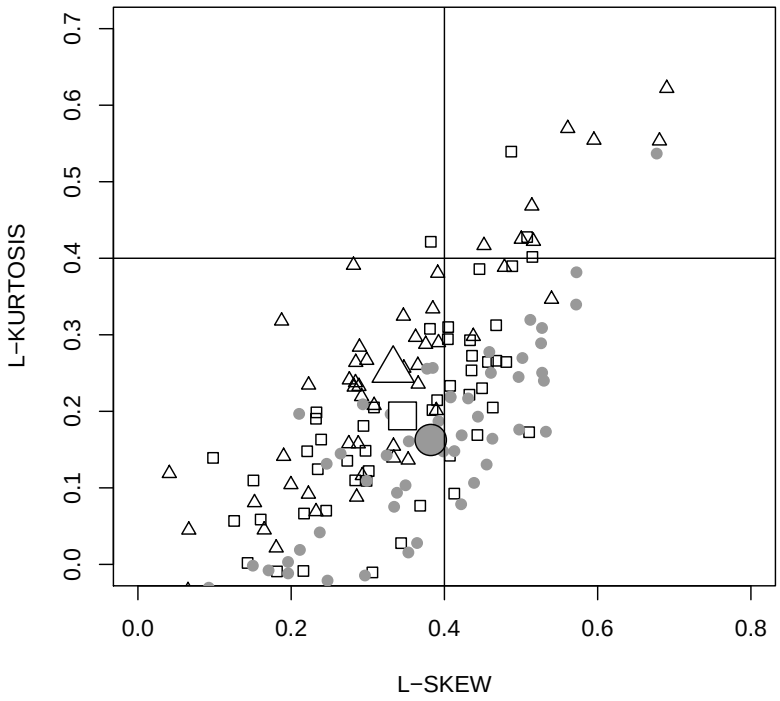

Figure 10.2. L-moment ratio diagram showing 50 sample simulations of L-skew and L-kurtosis for $n = 20$ samples drawn from three distributions from example 10–4

within 0.0005 over the range $-0.9 \leq \tau_3 \leq 0.9$, except for the Generalized Extreme Value distribution, for which the 0.0005 accuracy is available only when $-0.6 \leq \tau_3 \leq 0.9$.

USING R

The `lmrdia()` function provides characteristic $\{\tau_3, \tau_4\}$ relations as either constants or matrices for common distributions used in L-moment-based distributional analysis. The distributions supported by `lmrdia()`, identified by the *lmomco* abbreviation style and considered in this dissertation, include `exp`, `gev`, `glo`, `gpa`, `gum`, `gno` (log-normal), `nor`, `pe3`, and `ray`. For example, to access the characteristic $\{\tau_3, \tau_4\}$ pairings for the Gumbel distribution, example 10–5 can be used.

10–5

```
lmrdia <- lmrdia() # extract ordinates for LMR diagrams
lmrdia$gum[1,]     # display values for the Gumbel distribution
[1] 0.169925 0.150375
```

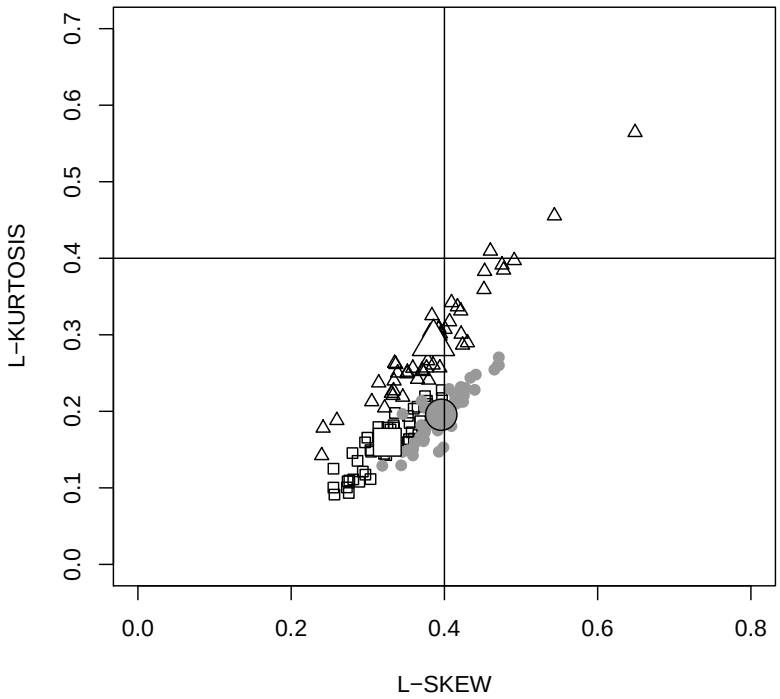

Figure 10.3. L-moment ratio diagram showing 50 sample simulations of L-skew and L-kurtosis for $n = 200$ samples drawn from three distributions based on examples 10–3 and 10–4

Table 10.1. Coefficients for polynomial approximations of L-kurtosis as a function of L-skew for selected distributions

[GEV, Generalized Extreme Value distribution; GLO, Generalized Logistic distribution; GNO, Generalized Normal distribution; PE3, Pearson Type III distribution; --, implies a coefficient of zero]

	GEV	GLO	GNO	GPA	PE3
A_0	0.10701	0.16667	0.12282	0.	0.12240
A_1	.11090	--	--	.20196	--
A_2	.84838	.83333	.77518	.95924	.30115
A_3	$-.06669$	--	--	$-.20096$	--
A_4	.00567	--	.12279	.04061	.95812
A_5	$-.04208$	--	--	--	--
A_6	.03763	--	$-.13638$	--	$-.57488$
A_7	--	--	--	--	--
A_8	--	--	.11368	--	.19383

For the Gumbel distribution, these values are $\{\tau_3^{\text{gum}}, \tau_4^{\text{gum}}\} = \{0.17, 0.15\}$ as shown. The `lmrdia()` function also returns the theoretical limits of τ_3 and τ_4 in the `limits` field, which are accessed using `lmrdia$limits`.

In order to draw the trajectories for several of the distributions, the `lmrdia()` function uses (1) the polynomial approximations in table 10.1 for the Generalized Normal and Pearson Type III distributions; (2) analytical expressions for the Generalized Logistic distribution by eq. (8.25) and the Generalized Pareto distribution by eq. (8.58); and (3) the iteration of parameter κ through eqs. (8.10) and (8.11) for the Generalized Extreme Value distribution. ◀

Considering again figure 10.3 and the examples from the previous section, if the following three lines in example 10–6 are added to the end of example 10–4 and executed after examples 10–1 – 10–3, then the trajectories of τ_3 and τ_4 for the Pearson Type III and the Generalized Logistic distributions become superimposed on the plot. The results are shown in figure 10.4. The Generalized Logistic is shown by the thin line and the Pearson Type III by the thick line.

10–6

```
lmrdia <- lmrdia() # data structure
lines(lmrdia$glo[,1],lmrdia$glo[,2]) # thin line
lines(lmrdia$pe3[,1],lmrdia$pe3[,2], lwd=3) # thick line
```

The Gamma distribution is closely related to the Pearson Type III; the Gamma can acquire the same τ_3 and τ_4 combinations as the Pearson Type III. However, a fitted Gamma distribution will not plot at the same τ_3 and τ_4 values as a fitted Pearson Type III for a given data set (L-moment combination)—the two distributions are different. The Gamma has two parameters and the Pearson Type III has three. The large square (the mean of the $\{\tau_3^{\text{gam}}, \tau_4^{\text{gam}}\}$ loci) in figure 10.4 thus is plotted effectively on the Pearson Type III curve and will plot on the curve for sufficiently large sample sizes. ◀

The general construction of an L-moment ratio diagram is made in example 10–7. The resulting diagram is shown in figure 10.5. The `plotlmrdia()` function provides a high-level interface for plotting L-moment ratio diagrams. The diagram shown in figure 10.5 is a "full perspective" diagram because the entire range of τ_3 and τ_4 is depicted. The range of τ_3 is $-1 < \tau_3 < 1$ (eq. (6.28)), and the "Theoretical limits" line demarks the base of the $\frac{1}{4}(5{\tau_3}^2 - 1) \leq \tau_4 < 1$ relation (eq. (6.29)). The figure is generated with selected colors for some of the lines.

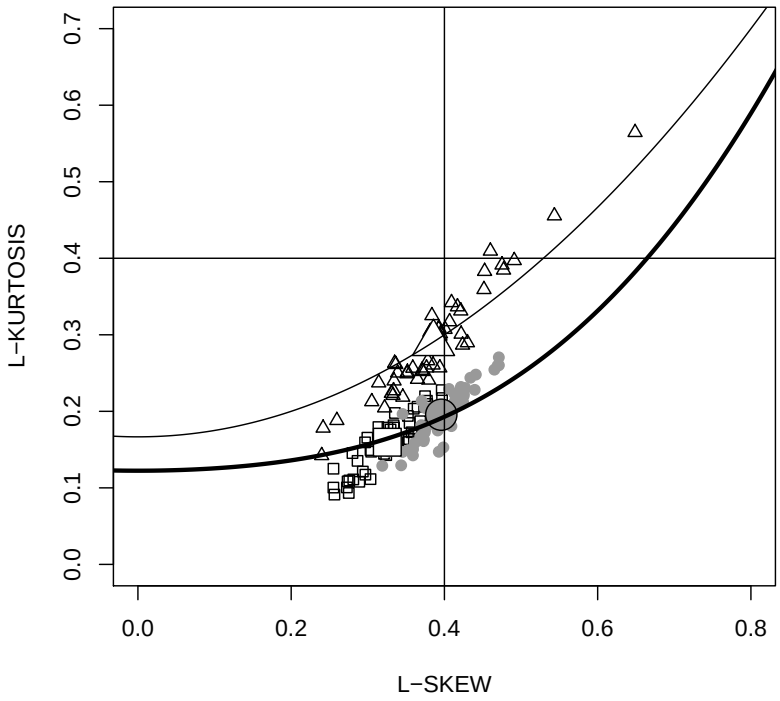

Figure 10.4. L-moment ratio diagram showing 50 sample simulations of L-skew and L-kurtosis values for $n = 200$ samples drawn from three distributions with superimposed theoretical lines for the Generalized Logistic distribution (thin line) and Pearson Type III distribution (thick line) from examples 10–4 and 10–6

10–7

```
lmrdia <- lmrdia() # function takes no arguments
#pdf("lmr4.pdf")
plotlmrdia(lmrdia, autolegend=TRUE, xleg=0, yleg=1)
# the plotlmrdia() function takes many arguments
#dev.off()
```

The `lmrdia()` function returns an R list containing matrices of the τ_3 and τ_4 values for selected distributions. The `plotlmrdia()` function accepts (expects) the list returned by `lmrdia()`. The `plotlmrdia()` function has a variety of named arguments to configure the diagram. Example 10–7 show the autogeneration of a distribution legend with the origin of the legend at $\{\tau_3, \tau_4\} = \{0, 1\}$.

The distributions depicted in figure 10.5 plot as either points or lines. Some large-parameter distributions such as the Generalized Lambda (four parameter), Wakeby (five parameter), and quantile mixtures (see Karvanen, 2009) occupy difficult to depict regions

of the diagram. The region of $\{\tau_3^{\text{kap}}, \tau_4^{\text{kap}}\}$ for the Kappa distribution was shown by annotation of arrows and text in figure 10.1. ◀

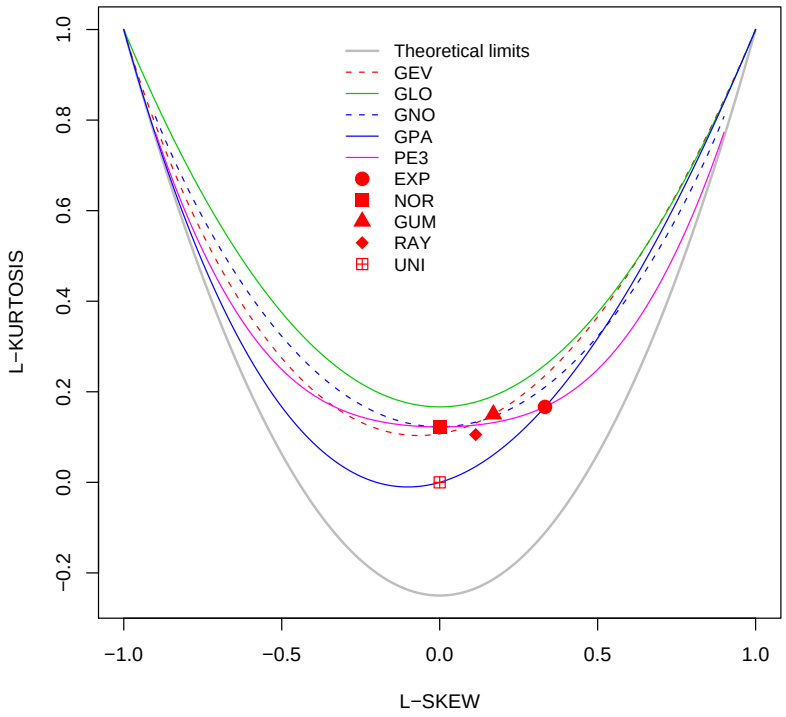

Figure 10.5. Default L-moment ratio diagram provided by package *lmomco* from example 10–7

Natural phenomena often are generated from random variables that are strictly positive. As a result, it is common for sample distributions to be positively skewed. A particularly useful L-moment ratio diagram for studying such distributions is shown in figure 10.6, which was produced by example 10–8. The diagram encompasses generally positive, but not strictly positive τ_3, to accommodate vagaries of sampling. An L-moment ratio diagram with the limits as shown in the figure will often provide an appropriate base figure for many situations of distributional analysis of natural phenomena.

10–8

```
#pdf("lmr5.pdf")
plotlmrdia(lmrdia(), autolegend=TRUE, nopoints=TRUE,
           xleg=0.1, yleg=0.41,
           xlim=c(-0.1,0.5), ylim=c(-0.1,0.4))
#dev.off()
```

◀

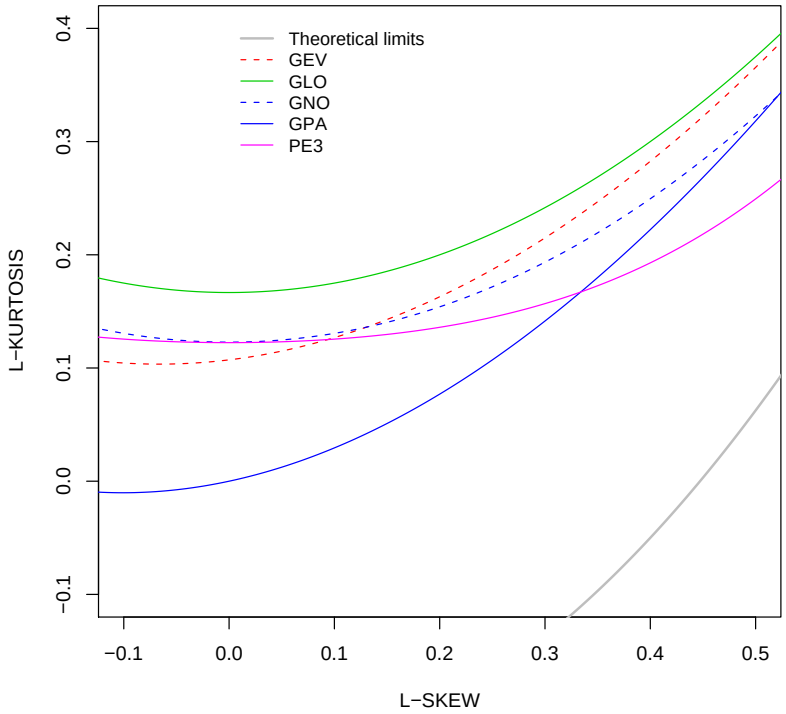

Figure 10.6. More typical L-moment ratio diagram for generally positively skewed phenomena provided by package *lmomco* from example 10–8

In example 10–9, some arbitrary population L-moment values are selected ($\lambda_1 = 10000$, $\lambda_2 = 7500$, $\tau_3 = 0.3$, and $\tau_4 = 0.2$). A sample size of $n = 30$ and the number of simulations to demonstrate `nsim` are set. The `vec2lmom()` and `parkap()` functions are used to set the L-moments and compute the Kappa parameters. Two temporary vectors `t3` and `t4` also are created. These vectors are filled within the `for()` loop with values of $\hat{\tau}_3$ and $\hat{\tau}_4$ from simulated Kappa quantiles computed by the `rlmomco()` function that are then passed to the `lmoms()` function. After the `t3` and `t4` vectors are populated, each vector is plotted on an L-moment ratio diagram in figure 10.7. The lines of code containing the two `points()` functions show how the plotting operations were made for the figure.

10–9

```
T3 <- 0.3; T4 <- 0.2; n <- 30; nsim <- 50

lmr <- vec2lmom(c(10000,7500,T3,T4))
kap <- parkap(lmr)
t3  <- t4 <- vector(mode = "numeric")
for(i in seq(1,nsim)) {
  sim.lmr <- lmoms(rlmomco(n,kap))
```

```
  t3[i] <- sim.lmr$ratios[3]
  t4[i] <- sim.lmr$ratios[4]
}

#pdf("lmr6.pdf")
plotlmrdia(lmrdia(), autolegend=TRUE, nopoints=TRUE,
           xleg=0.1, yleg=0.41,
           xlim=c(-0.1,0.5), ylim=c(-0.1,0.4))
points(t3,t4) # small open circles
points(mean(t3),mean(t4), pch=16, cex=3) # filled circle
segments(T3,-1, T3,1)
segments(-1,T4, 1,T4)
#dev.off()
```

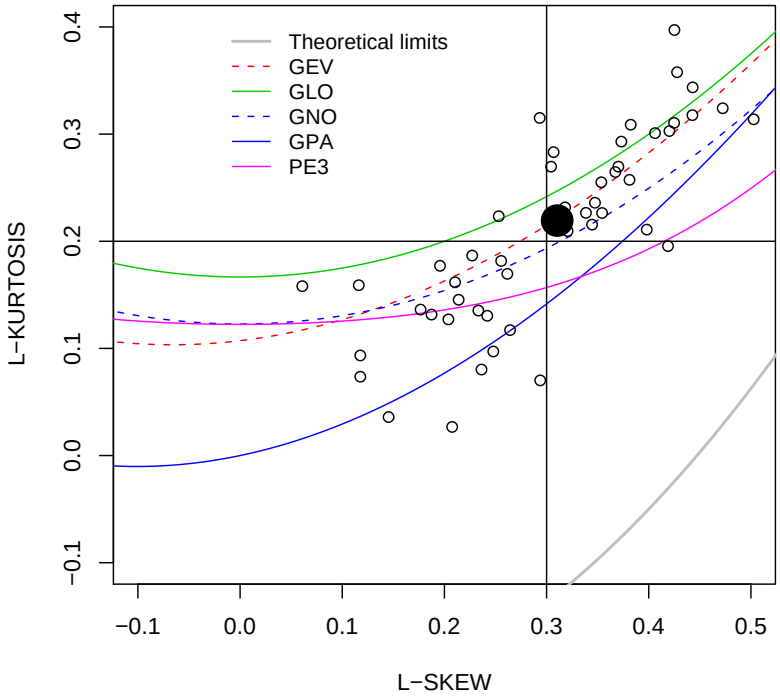

Figure 10.7. L-moment ratio diagram shown distribution of 50 sample simulations of of L-skew and L-kurtosis for $n = 30$ samples drawn from a KAP$(10000, 7500, 0.3, 0.2)$ distribution from example 10–9

The diagram also shows the intersection of the population $\{\tau_3, \tau_4\}$ values by the horizontal and vertical crossing lines, and the large filled circle is plotted at the `mean()` values of the `t3` and `t4` vectors. The diagram shows, like the simulations that produced

figures 10.2 and 10.3, that $\{\hat{\tau}_3, \hat{\tau}_4\}$ from a given parent distribution will exhibit scatter on an L-moment ratio diagram.

A potentially paradoxical situation in the case for the Kappa is that the Kappa cannot be fit to some of the $\{\hat{\tau}_3, \hat{\tau}_4\}$ because these pairs plot above the Generalized Logistic line. The simulations show that the Kappa can generate data having the sample pair $\{\hat{\tau}_3, \hat{\tau}_4\}$ plot above the Generalized Logistic. For the example, there are nine simulations for which this is true. The paradox is actually no different from the three-parameter distribution simulations that were made in the context of figures 10.2 and 10.3. Those simulations also produced $\hat{\tau}_4$ values for a given $\hat{\tau}_3$, which are unattainable when the distribution is fit by the method of L-moments.

To further clarify, if one were to take the simulated sample L-moments (the sample L-moments for an individual execution of the `for()` loop in example 10–9) and turn around and attempt to fit the Kappa distribution, one would be unable to do so because of inherent limitations of shape for the Kappa distribution. As mitigation for the paradox, so-called regional L-moments or pooled L-moments of many quasi-independent data sets, which are assumed to be generated by a common parent distribution, can be used. An example of such practice is provided in Section 11.1 and by citation in Section 12.7.2. ◀

10.4 Summary

In this chapter, L-moment ratio diagrams are introduced and specifically diagrams of τ_3 and τ_4 are described. Such diagrams are useful for distinguishing between distributional form because specific intra-moment—that is conceptually unique—relations exist for each distribution. Detailed description of the general interpretation of the diagrams is provided. The 9 examples in the chapter demonstrate how sampling variability affects graphical interpretation and demonstrate the construction of L-moment ratio diagrams using the *lmomco* package.

There are other forms of L-moment ratio diagrams in use including diagrams of τ_2 versus τ_3 (Vogel and others, 2008) and τ_4 versus τ_6 (Hosking and others, 2000; Hosking, 2007b). The former are useful for evaluation of two-parameter distributions, whereas the latter are useful for evaluation of distribution form for generally symmetrical distributions. Neither of these diagrams are otherwise discussed in this dissertation.

Chapter 11
Short Studies of Statistical Regionalization

In this chapter, I present two short studies concerning original distributional analysis of hydrologic data using L-moment statistics. This chapter is intended to provide a "look and feel" of distributional analysis using L-moments and the *lmomco* package for decidedly non-Normal data and provide guidance into regionalization of hydrometeorological data. This chapter is dependent on many concepts and functions described and demonstrated in previous chapters, and general familiarity with L-moments, distributions, and L-moment ratio diagrams is assumed. Several nuances of distribution fit and solution choice are described. This chapter could be especially useful to some less experienced readers expecting to conduct their own distributional analysis. Therefore, this chapter in a way "blue prints" a simple form distributional analysis with L-moment statistics using R.

11.1 Analysis of Annual Maxima for 7-Day Rainfall for North-Central Texas Panhandle

A small region of the north-central Texas Panhandle is chosen for analysis of the magnitude and frequency of 7-day annual maxima rainfall.[1] For the study area of the north-central Texas Panhandle, an estimate of the 100-year recurrence interval is sought for 7-day annual maximum depth of rainfall. A comparison of this estimate to that from a previous study is made. A fascinating discussion of identification of similar rainfall climates using L-moments is available in Guttman (1993) and application in Guttman and others (1993).

[1] Specifically, the largest total rainfall for 7 consecutive days per year.

11.1.1 Background and Data

Asquith (1998) and Asquith and Roussel (2004) provide a comprehensive L-moment-based analysis of regional characteristics of depth-duration frequency of rainfall in Texas. For the analysis shown in this section, the data used are 7-day annual maxima for seven daily rainfall stations operated by or in cooperation with the National Weather Service. Each station has at least 10 years of record through 1994. As will be seen, these data are available in the *lmomco* package. The text in the remainder of this background and data section is derived from Asquith (1998).

Distributional analysis of rainfall data is important because rainfall depths for various durations and frequencies, referred to as depth-duration frequency (DDF), have many uses. A common use (Asquith, 1998) of DDF is for the design of structures that control and route localized runoff, such as parking lots, storm drains, and culverts. Another use of DDF is to drive river-flow models that incorporate rainfall characteristics. Accurate DDF estimates are important for cost-effective structural designs at stream crossings and for developing reliable flood-prediction models.

Accurate DDF analysis using data from any one rainfall-monitoring station is difficult (see Wallis, 1988, p. 305) because the data for one station represent a poor spatial and (or) temporal sampling of rainfall distributions. For example, storms occur over areas that might or might not contain a station; and generally, comparatively short records (small samples) are available at a single station. Additionally, the distribution of rainfall associated with any one station tends to be highly non-Normal. More accurate DDF estimates can be developed by "pooling" or "regionalizing" data from many nearby stations (see Stedinger and others, 1993, chap. 18, p. 33). Schaefer (1990) provided a highly influential paper on rainfall regionalization for the Asquith (1998) study.

11.1.2 Distributional Analysis

The underlying assumption of the rainfall regionalization for the north-central Texas Panhandle is the assumption that the L-moments of the unknown parent distribution of annual maxima can be reliably estimated by weighted means of station-specific L-moments from the observation network within the study area. The assumption also implies that a single distribution is appropriate for modeling the frequency of annual max-

ima in the study area. Hosking and Wallis (1993; 1997) describe an extensive L-moment-based regionalization method based partly on this assumption, and other studies using this assumption are readily found in the literature of this discipline.

The analysis begins by loading seven individual data sets that are provided in the *lmomco* package. Each data set is identified in example 11–1 and represents a time series of annual 7-day rainfall maxima. The communities are Amarillo, Canyon, Claude, Hereford, Tulia, and Vega. The `tulia6Eprecip` location is about 6 miles east of Tulia. Collectively, these communities represent an area of approximately $1{,}400$ square miles.

11–1

```
data(amarilloprecip)   # from lmomco package
data(canyonprecip)     #     .. ditto ..
data(claudeprecip)     #     .. ditto ..
data(herefordprecip)   #     .. ditto ..
data(tuliaprecip)      #     .. ditto ..
data(tulia6Eprecip)    #     .. ditto ..
data(vegaprecip)       #     .. ditto ..
```

The loading of the data in example 11–2 is followed by placing the `DEPTH`s into variables with abbreviated names of the community. A `sort()` operation also is made because only sorted data are needed for the analysis; no evaluation of climatic cycles or trends is made and an assumption of stationarity is implicitly made.

11–2

```
AMAR <- sort(amarilloprecip$DEPTH)
CANY <- sort(canyonprecip$DEPTH)
CLAU <- sort(claudeprecip$DEPTH)
HERF <- sort(herefordprecip$DEPTH)
TULA <- sort(tuliaprecip$DEPTH)
TUL6 <- sort(tulia6Eprecip$DEPTH)
VEGA <- sort(vegaprecip$DEPTH)
```

The distributional analysis initiates with a graphical review using box plots of the distribution of the rainfall data for each community. The box plots are shown in figure 11.1, which was created by example 11–3. In the example, the lengths of the record are computed and set into the `w` variable. These lengths are used as weights for a weighted-mean computation in a subsequent example.

11–3

```
x <- list(AMAR=AMAR, CANY=CANY, CLAU=CLAU,
          HERF=HERF, TULA=TULA, TUL6=TUL6,
          VEGA=VEGA) # combine all into short variable name
w <- sapply(x,length) # w will be used in a later example
print(w) # show the lengths of the individual records
AMAR CANY CLAU HERF TULA TUL6 VEGA
  47   72   91   67   48   50   61
#pdf("texas_panhandle_boxplot.pdf")
boxplot(x, ylab="7-DAY␣ANNUAL␣MAX␣RAINFALL,␣IN␣INCHES", range=0)
#dev.off()
```

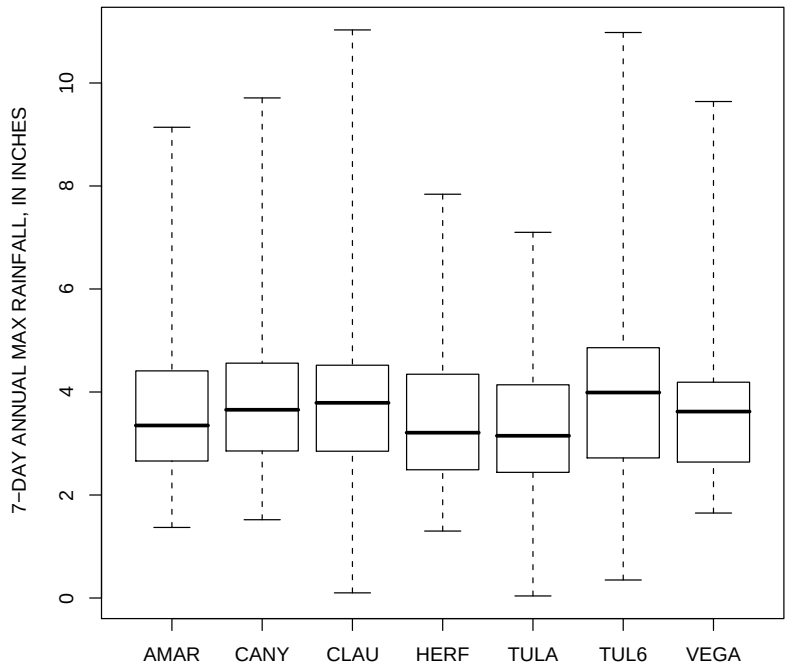

Figure 11.1. Box plots of the distributions of 7-day annual maxima rainfall for seven communities in the north-central Texas Panhandle from example 11–3

The box plots show that the typical location or central tendency of the seven distributions is about 3.75 inches. The individual interquartile range or IQRs of the seven distributions also are similar. Although exhibiting apparent differences in the distal-tail regions, the data clearly have positive τ_3. For the analysis here, it is assumed that the observed differences in distribution geometry represent vagaries of sampling from a common parent distribution.

Additional variations on box plots exist to assess distributional geometry. The R packages *beanplot* by Kampstra (2008a) and *vioplot* by Adler (2005) provide functions of the respective names that produce **bean plots** and **violin plots**. These two plot types can be used to depict the probability density of the data in a style that is unattainable by the conventional box plot. Kampstra (2008b) provides additional description of bean plots. Example 11–4 demonstrates the application of these plots and the results are shown in figure 11.2.

11–4

```
library("beanplot"); library("vioplot")
rng  <- sapply(x, range) # x from previous example
ylim <- c(min(rng[1,]), max(rng[2,]))
#pdf("texas_panhandle_beanvio.pdf");
par(mfrow=c(2,1), mai=c(0.5,1,0.5,0.5) )
beanplot(x, ll=0.04, main="BEAN␣PLOT:␣beanplot()", log="",
 ylim=ylim, ylab="7-DAY␣ANNUAL␣MAX␣RAINFALL,\n␣IN␣INCHES",
 overallline="median")
cities <- names(x); data <- x # get names and make a copy
names(data)[1] <- "x" # modify the copy
do.call("vioplot",c(data, list(ylim=ylim, names=cities,
                    col="white")))
title(main="VIOLIN␣PLOT:␣vioplot()",
      ylab="7-DAY␣ANNUAL␣MAX␣RAINFALL,\n␣IN␣INCHES")
#dev.off()
```

The bean plots show the density as the curved hull around the individual data points ("beans" or "kernels"). The `beanplot()` function also depicts that overall median of the seven data groups as the dotted horizontal line. The violin plots also show the density, but truncate the density at the minimum and maximum values. Inside each violin is a more-or-less conventional box plot. Both `beanplot()` and `vioplot()` functions each have numerous configuration options.

Continuing with the analysis, the Weibull plotting-positions and sample L-moments of the rainfall data are computed in example 11–5, and set into 14 concise variable names. The plotting positions are used in subsequent plotting operations, and the L-moments are used to fit probability distributions using the method of L-moments.

11–5

```
AMAR.pp <- pp(AMAR);  CANY.pp <- pp(CANY)
CLAU.pp <- pp(CLAU);  HERF.pp <- pp(HERF)
TULA.pp <- pp(TULA);  TUL6.pp <- pp(TUL6)
VEGA.pp <- pp(VEGA)
```

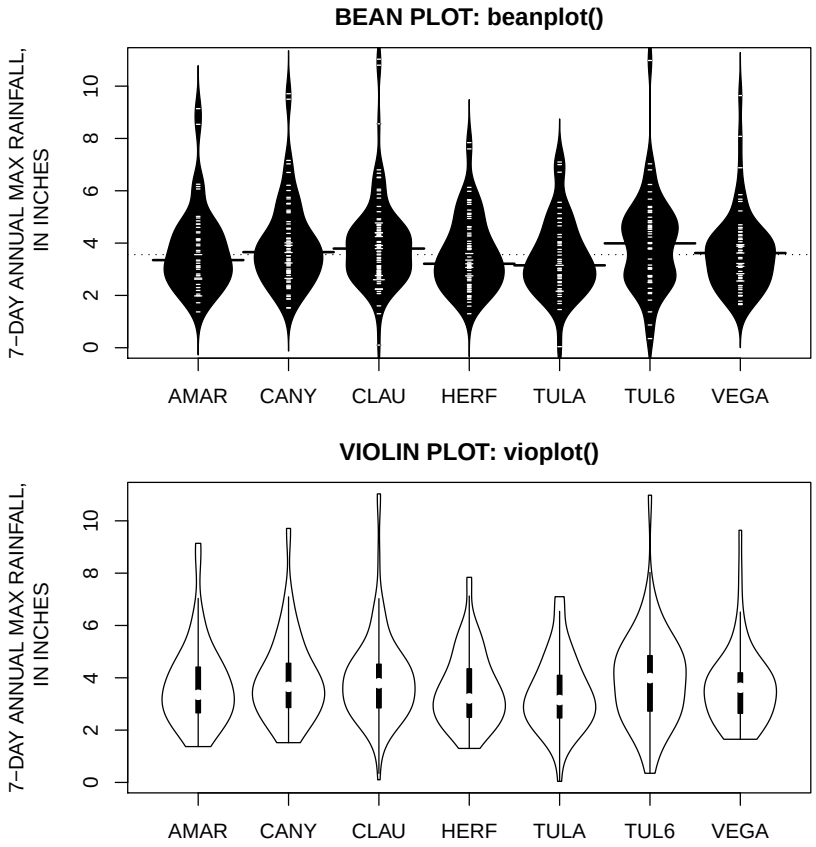

Figure 11.2. Bean and violin plots of the distributions of 7-day annual maxima rainfall for seven communities in the north-central Texas Panhandle from example 11–4

```
AMAR.lmr <- lmoms(AMAR);  CANY.lmr <- lmoms(CANY)
CLAU.lmr <- lmoms(CLAU);  HERF.lmr <- lmoms(HERF)
TULA.lmr <- lmoms(TULA);  TUL6.lmr <- lmoms(TUL6)
VEGA.lmr <- lmoms(VEGA)
```

As part of the analysis, weighted-mean values of the sample L-moments are needed. To simplify later code, it is useful to have the sample L-moments collected into individual variables. This is done in example 11–6 by long-hand placement into the variable `L1` for $\hat{\lambda}_1$, and using a convenience function named `afunc()` for $\hat{\tau}_2$, $\hat{\tau}_3$, and $\hat{\tau}_4$, in variables `T2`, `T3`, and `T4`, respectively. The example shows the multiplication of $1.018\hat{\lambda}_1$ to account for a recording bias attributable to a 7-day interval as developed by Weiss (1964) and used by Asquith (1998, table 1).

11–6

```
"afunc" <- function(r) {
  return(c(AMAR.lmr$ratios[r], CANY.lmr$ratios[r],
           CLAU.lmr$ratios[r], HERF.lmr$ratios[r],
           TULA.lmr$ratios[r], TUL6.lmr$ratios[r],
           VEGA.lmr$ratios[r]))
}
L1 <- c(AMAR.lmr$lambdas[1], CANY.lmr$lambdas[1],
        CLAU.lmr$lambdas[1], HERF.lmr$lambdas[1],
        TULA.lmr$lambdas[1], TUL6.lmr$lambdas[1],
        VEGA.lmr$lambdas[1])*1.018; # bias correction factor
           Weiss (1964)

T2 <- afunc(2); T3 <- afunc(3); T4 <- afunc(4)
```

Regional values for the sample L-moments ("regional L-moments") are computed in example 11–7 using the `weighted.mean()` function using the weights in `w` from example 11–3. The example continues with the selection of the Kappa distribution for modeling. The Kappa distribution often is a highly suitable distribution to model hydrometeorological data sets provided that $\hat{\tau}_4$ values are less than those of the Generalized Logistic distribution (see Chapter 10).

11–7

```
reg.L1  <- weighted.mean(L1, w);  reg.T2  <- weighted.mean(T2, w)
reg.T3  <- weighted.mean(T3, w);  reg.T4  <- weighted.mean(T4, w)
reg.lmr <- vec2lmom(c(reg.L1, reg.L1*reg.T2, reg.T3, reg.T4))
reg.kap <- parkap(reg.lmr) # parameters of the Kappa distribution

str(reg.lmr) # output the regional L-moments
List of 9
 $ L1  : num 3.83
 $ L2  : num 0.85
 $ TAU3: num 0.186
 $ TAU4: num 0.188
 $ TAU5: NULL
 $ LCV : num 0.222
 $ L3  : num 0.158
 $ L4  : num 0.16
 $ L5  : NULL

print(reg.kap) # output the regional Kappa distribution
$type
[1] "kap"
$para
        xi      alpha      kappa          h
 3.4142310  0.9135191 -0.1389698 -0.5673624
$source
```

```
[1] "parkap"
$ifail
[1] 0
$ifailtext
[1] "Successful␣parameter␣estimation."
```

Example 11–7 shows that the Kappa is successfully fit to the regional L-moments and forms a regional Kappa distribution, and the fitted distribution is

$$P_{\text{7-day}}(F) = 3.414 + \frac{0.9135}{-0.1390}\left[1 - \left(\frac{1 - F^{-0.5674}}{-0.5674}\right)\right] \tag{11.1}$$

where $P_{\text{7-day}}$ is the 7-day annual maximum rainfall in inches for F (nonexceedance probability).

The L-moment ratio diagram is created in example 11–8 and shown in figure 11.3. The diagram shows that the regional value (filled circle) for τ_4 (and generally the $\hat{\tau}_4$ for each community) is larger than that for all the three-parameter distributions with the exception of the Generalized Logistic. In fact, the regional value for τ_4 is almost as large as that for the Generalized Logistic; in this circumstance the Kappa distribution can just barely be fit.

11–8

```
#pdf("texas_panhandle_lmrdia.pdf")
lmrdia <- lmrdia()
plotlmrdia(lmrdia,autolegend=TRUE, nopoints=TRUE,
           nolimits=TRUE, xlim=c(0,0.3), ylim=c(-0.1,0.4),
           xleg=0.05, yleg=0.3)
points(T3,T4)
points(reg.T3,reg.T4, pch=16, cex=2)
#dev.off()
```

The distributional analysis is effectively completed in example 11–9. In the example, the quantiles of the Kappa distribution for the selected F values are computed. These are shown in figure 11.4 connected as a curved line, which is superimposed on the individual data points of the empirical distribution for each of the seven communities. The figure shows the similarity of the seven empirical distributions. Because the communities are close to each other, are at similar elevations, and thus have similar climate, there is anticipation that a more accurate estimate in the far right tail of the unknown parent distribution is acquired by pooling (averaging together) the L-moments together (Hosking and Wallis, 1993, 1997).

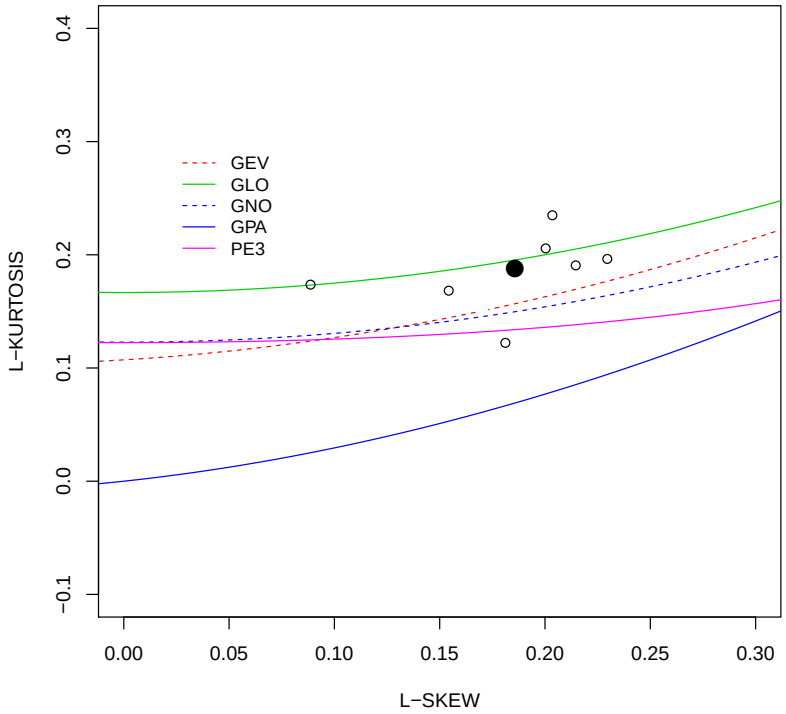

Figure 11.3. L-moment ratio diagram showing $\hat{\tau}_3$ and $\hat{\tau}_4$ of 7-day annual maximum rainfall for seven communities in Texas Panhandle (open circles) and weighted mean value (filled circle) from example 11–8

11–9

```
F <- seq(0.001,0.999, by=0.001)
#pdf("texas_panhandle.pdf", version="1.4")
plot(F,quakap(F,reg.kap), type="n",
       xlim=c(0,1), ylim=c(0,12),
       xlab="NONEXCEEDANCE␣PROBABILITY",
       ylab="7-DAY␣RAINFALL␣DEPTH,␣IN␣INCHES")
points(AMAR.pp, AMAR, pch=16, col=rgb(0, 0, 0, 0.15))
points(CANY.pp, CANY, pch=16, col=rgb(0, 0, 0, 0.20))
points(CLAU.pp, CLAU, pch=16, col=rgb(0, 0, 0, 0.25))
points(HERF.pp, HERF, pch=16, col=rgb(0, 0, 0, 0.35))
points(TULA.pp, TULA, pch=16, col=rgb(0, 0, 0, 0.40))
points(TUL6.pp, TUL6, pch=16, col=rgb(0, 0, 0, 0.45))
points(VEGA.pp, VEGA, pch=16, col=rgb(0, 0, 0, 0.50))
T.YEAR <- 100
PT <- quakap(T2prob(T.YEAR),reg.kap)
lines(c(T2prob(T.YEAR), T2prob(T.YEAR)), c(0,20), lty=2)
lines(c(0,1),c(PT,PT), lty=2)
lines(F,quakap(F,reg.kap), lwd=3)
#dev.off()
```

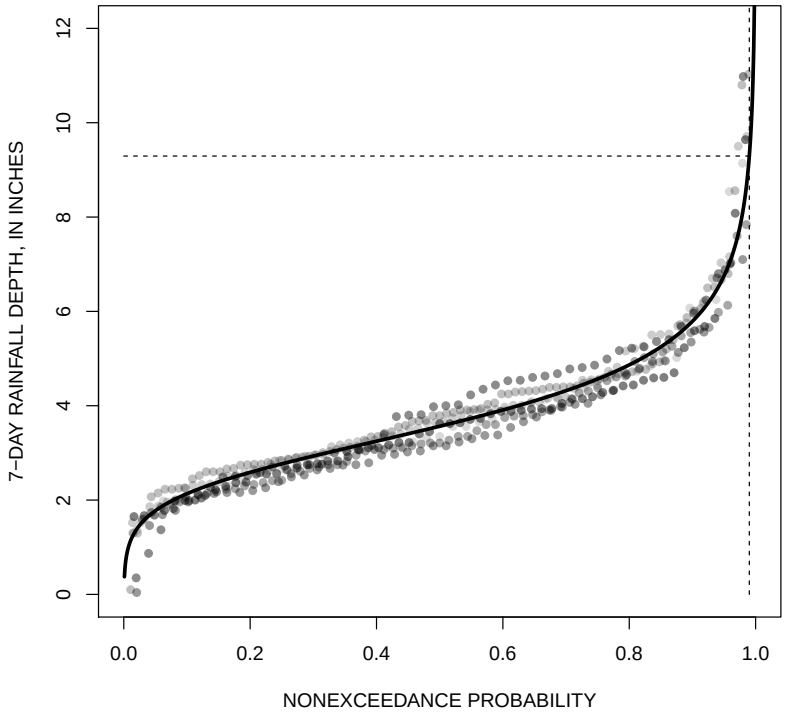

Figure 11.4. Empirical distribution of 7-day annual maxima rainfall for seven communities in the Texas Panhandle—Dashed lines show intersection of 100-year event from regional Kappa distribution (thick line) from example 11–9.

The purpose of the distributional analysis here is to estimate the 100-year, 7-day annual maximum rainfall depth. In the example, the `T2prob()` function is used to compute the F of the 100-year event ($F = 0.99$). The dashed lines in figure 11.4 indicate the solution. The analysis is completed by outputting the 100-year event in variable `PT` in example 11–10. The output shows that the estimated 100-year, 7-day rainfall depth is about 9.29 inches.

11–10

```
print(PT)
[1] 9.293368
```

Asquith and Roussel (2004, p. 82) report that the 100-year, 7-day annual maximum rainfall for the approximate geographic center of the seven communities is about 9.3 inches. Their analysis was based on regionalized parameters of a Generalized Extreme Value distribution fit by the method of L-moments to each station (the 7 here and another 858 across Texas) by Asquith (1998).The two rainfall depths of 9.29 and 9.3 inches compare favorably. The computational reliability of some of the *lmomco* algorithms (circa 2008) com-

pared to algorithms implemented by the author in the period 1996–98 using the FORTRAN algorithms of Hosking (1996b) is demonstrated. ◀

11.2 Peak-Streamflow Distributions for Selected River Basins in the United States

In Section 11.1, a style of distributional analysis is conducted in which the sample L-moments of rainfall data collected from different localities having similar climate are pooled together. Such a practice is done with the intent of making more secure inferences of distributional form and extreme-tail quantile estimation.

In this section, however, distributional analysis is conducted using L-moments to document distributional differences of the annual peak streamflow for five climatically and physiographically different rivers of the United States that have similar drainage areas. Repeating the descriptions on page 56, annual peak streamflows are the largest instantaneous volumetric rate of flow in a stream for a given year, and such data provide the backbone for statistical analyses that govern the management of flood plains and contribute to the design of water-related infrastructure such as bridges.

11.2.1 Background and Data

The five rivers along with corresponding U.S. Geological Survey streamflow-gaging station number and drainage area are listed in table 11.1. These five rivers were selected because they are similarly sized and represent fundamentally different hydrologic processes because the climate and physiographic settings of the five unique river basins are very diverse.

11.2.2 Distributional Analysis

The analysis of the annual peak streamflow data is initiated by loading in five individual data sets that are provided within the *lmomco* package. Each data set is seen in example 11–11, and each data set represents a time series of annual peak streamflow for the

Table 11.1. Summary of selected U.S. Geological Survey streamflow-gaging stations for distributional analysis using L-moments

Station number	Station name	Drainage area (square miles)
01515000	Susquehanna River near Waverly, New York	4,773
02366500	Choctawhatchee River near Bruce, Florida	4,384
08151500	Llano River at Llano, Texas	4,203
09442000	Gila River near Clifton, Arizona	4,010
14321000	Umpqua River near Elkton, Oregon	3,683

respective streamflow-gaging stations. The example also shows that the `Streamflow` data are placed into five concise variable names.

11–11

```
data(USGSsta01515000peaks)  # load data from lmomco package
data(USGSsta02366500peaks)  #     .. ditto ..
data(USGSsta08151500peaks)  #     .. ditto ..
data(USGSsta09442000peaks)  #     .. ditto ..
data(USGSsta14321000peaks)  #     .. ditto ..

susque.Q <- USGSsta01515000peaks$Streamflow # concise names
chocta.Q <- USGSsta02366500peaks$Streamflow #  .. ditto ..
llano.Q  <- USGSsta08151500peaks$Streamflow #  .. ditto ..
gila.Q   <- USGSsta09442000peaks$Streamflow #  .. ditto ..
umpqua.Q <- USGSsta14321000peaks$Streamflow #  .. ditto ..
```

As in Section 11.1.2, the distributional analysis begins with a graphical review using box plots of the distribution of the data for each community. The box plots are shown in figure 11.5, which was created by example 11–12.

11–12

```
#pdf("usrivers_boxplot.pdf")
allQ <- list(Susque=susque.Q, Chocta=chocta.Q,
             Llano=llano.Q,   Gila=gila.Q,     Umpqua=umpqua.Q)
boxplot(allQ, ylab="PEAK␣STREAMFLOW,␣IN␣CFS", range=0)
#dev.off()
```

The Weibull plotting positions and sample L-moments of the streamflow data are computed in example 11–13 and set into 10 concise variable names. The plotting positions are used in subsequent plotting operations, and the L-moments are used to fit probability distributions using the method of L-moments.

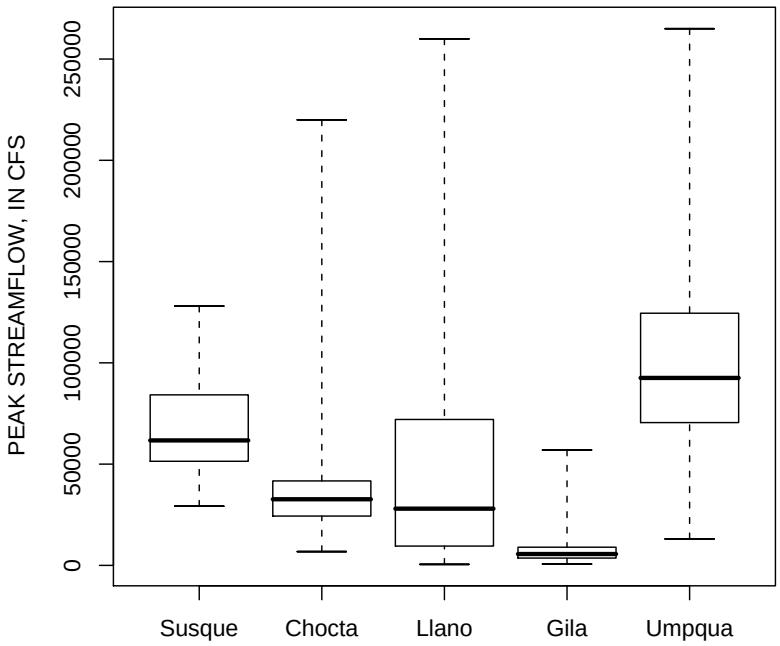

Figure 11.5. Box plots of empirical distribution of annual peak streamflow for five selected river basins in the United States from example 11–12

11–13

```
susque.pp <- pp(susque.Q);  chocta.pp <- pp(chocta.Q)
llano.pp  <- pp(llano.Q);   gila.pp   <- pp(gila.Q)
umpqua.pp <- pp(umpqua.Q)

susque.lmr <- lmoms(susque.Q);  chocta.lmr <- lmoms(chocta.Q)
llano.lmr  <- lmoms(llano.Q);   gila.lmr   <- lmoms(gila.Q)
umpqua.lmr <- lmoms(umpqua.Q)
```

Following the custom of the author's preference for initial forays into distributional analysis of hydrologic data, a Kappa distribution is fit to the L-moments in example 11–14. Inspection of the Kappa parameters using the `print()` or `str()` functions (results not shown) in the five variables shows that the distribution cannot be fit to the Choctawhatchee River data because the data are too L-kurtotic. So for site-to-site comparison of a common fitted distribution, the Kappa is not optimal in this circumstance.

11–14

```
susque.kap <- parkap(susque.lmr)
chocta.kap <- parkap(chocta.lmr); llano.kap <- parkap(llano.lmr)
gila.kap   <- parkap(gila.lmr);  umpqua.kap <- parkap(umpqua.lmr)
```

An L-moment ratio diagram with the sample L-moments of the five rivers is created and plotted in example 11–15 and shown in figure 11.6. The open-circle symbol for each river is scaled somewhat according to the magnitude of τ_5, and the first letter of each river name is superimposed on the open circles.

11–15

```
Stau <- c(susque.lmr$ratios[3], susque.lmr$ratios[4])
Ctau <- c(chocta.lmr$ratios[3], chocta.lmr$ratios[4])
Ltau <- c( llano.lmr$ratios[3],  llano.lmr$ratios[4])
Gtau <- c(  gila.lmr$ratios[3],   gila.lmr$ratios[4])
Utau <- c(umpqua.lmr$ratios[3], umpqua.lmr$ratios[4])

#pdf("usrivers_lmrdia.pdf", version="1.4")
lmrdiastuff <- lmrdia()
plotlmrdia(lmrdiastuff, autolegend=TRUE, xleg=0, yleg=0.6,
           xlim=c(-0.2,0.6), ylim=c(-0.2,0.6))

# plot first letter of the river over the circle
points(Stau[1], Stau[2], cex=1.5, pch="S")
points(Ctau[1], Ctau[2], cex=1.5, pch="C")
points(Ltau[1], Ltau[2], cex=1.5, pch="L")
points(Gtau[1], Gtau[2], cex=1.5, pch="G")
points(Utau[1], Utau[2], cex=1.5, pch="U")

mycol <- rgb(0, 0, 0, 0.75)
# plot circle with diameter scaled somewhat with Tau5
points(Stau[1], Stau[2], cex=1+2*susque.lmr$ratios[5], col=mycol)
points(Ctau[1], Ctau[2], cex=1+2*chocta.lmr$ratios[5], col=mycol)
points(Ltau[1], Ltau[2], cex=1+2* llano.lmr$ratios[5], col=mycol)
points(Gtau[1], Gtau[2], cex=1+2*  gila.lmr$ratios[5], col=mycol)
points(Utau[1], Utau[2], cex=1+2*umpqua.lmr$ratios[5], col=mycol)

legend(0.2,0.6,
       c("S␣␣␣Susquehanna␣River,␣New␣York",
         "C␣␣␣Choctawhatchee␣River,␣Florida",
         "L␣␣␣Llano␣River,␣Texas",
         "G␣␣␣Gila␣River,␣Arizona",
         "U␣␣␣Umpqua␣River,␣Oregon"), box.lty=0, bty="n")
#dev.off()
```

It is obvious that the Choctawhatchee River plots in figure 11.6 above the trajectory of the Generalized Logistic distribution on the diagram, and hence, as already mentioned, a Kappa distribution cannot be fit to the sample L-moments of the Choctawhatchee River data. The analyst could choose to fall back to the Generalized Logistic distribution at the expense of the distribution not being L-kurtotic enough, but other distributions such as the Wakeby or Generalized Lambda could also be consulted.

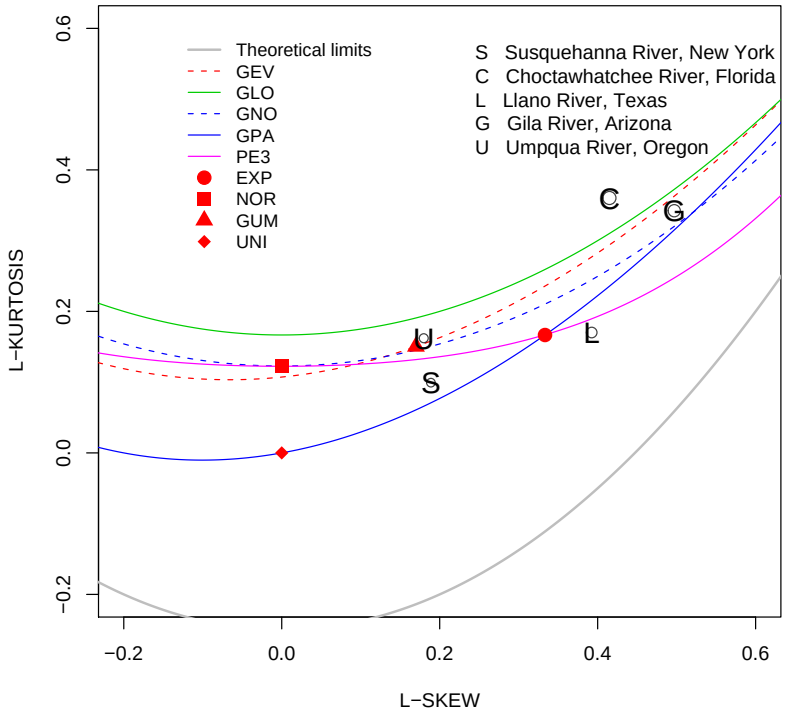

Figure 11.6. L-moment ratio diagram showing $\hat{\tau}_3$ and $\hat{\tau}_4$ of annual peak streamflow data for five selected river basins in the United States from example 11–15. The size of the open circles is scaled somewhat in proportion with $\hat{\tau}_5$.

In example 11–16, the Wakeby distribution is fit. Inspection of the Wakeby parameters in the five variables shows that the distribution could be fit. The Susquehanna and Gila Rivers require that $\xi = 0$ in order for the Wakeby to be fit but the remaining three do not.

11–16

```
susque.wak <- parwak(susque.lmr)
chocta.wak <- parwak(chocta.lmr)
llano.wak  <- parwak(llano.lmr)
gila.wak   <- parwak(gila.lmr)
umpqua.wak <- parwak(umpqua.lmr)
```

Before continuing with rather complex examples, a few useful variables are set in example 11–17, which are related to F values (`F` and `qF`) for pending horizontal axis and vertical axis limits (`mymin` and `mymax`). The `qnorm()` function is used to transform F into standard normal deviates.

11–17

```
F  <- nonexceeds()
qF <- qnorm(F)
```

```
mymin <- 0
mymax <- 300000
```

Plots of the QDF of the fitted Wakeby distribution are laid out and created in example 11–18. The five plots are shown in figure 11.7. The QDFs are superimposed on the actual data values.

11–18

```
#pdf("usrivers_qdf.pdf")
layout(matrix(1:6, ncol=2))
qpp <- qnorm(susque.pp); Q <- sort(susque.Q)
plot(qpp,Q, ylim=c(mymin,mymax), xlim=c(-2,3),
           xlab="STANDARD␣NORMAL␣DEVIATE",
           ylab="STREAMFLOW,␣IN␣CFS")
lines(qF,par2qua(F,susque.wak), col=2)
mtext("Susquehanna␣River,␣New␣York")
lines(c(qnorm(0.99),qnorm(0.99)), c(1,1000000), lty=2)

qpp <- qnorm(chocta.pp); Q <- sort(chocta.Q)
plot(qpp,Q, ylim=c(mymin,mymax), xlim=c(-2,3),
           xlab="STANDARD␣NORMAL␣DEVIATE",
           ylab="STREAMFLOW,␣IN␣CFS")
lines(qF,par2qua(F,chocta.wak),col=2)
mtext("Choctawhatchee␣River,␣Florida")
lines(c(qnorm(0.99),qnorm(0.99)), c(1,1000000), lty=2)

qpp <- qnorm(llano.pp); Q <- sort(llano.Q)
plot(qpp,Q, ylim=c(mymin,mymax), xlim=c(-2,3),
           xlab="STANDARD␣NORMAL␣DEVIATE",
           ylab="STREAMFLOW,␣IN␣CFS")
lines(qF,par2qua(F,llano.wak), col=2)
mtext("Llano␣River,␣Texas")
lines(c(qnorm(0.99),qnorm(0.99)), c(1,1000000), lty=2)

qpp <- qnorm(gila.pp); Q <- sort(gila.Q)
plot(qpp,Q, ylim=c(mymin,mymax), xlim=c(-2,3),
           xlab="STANDARD␣NORMAL␣DEVIATE",
           ylab="STREAMFLOW,␣IN␣CFS")
lines(qF,par2qua(F,gila.wak), col=2)
mtext("Gila␣River,␣Arizona")
lines(c(qnorm(0.99),qnorm(0.99)), c(1,1000000), lty=2)

qpp <- qnorm(umpqua.pp); Q <- sort(umpqua.Q)
plot(qpp,Q, ylim=c(mymin,mymax), xlim=c(-2,3),
           xlab="STANDARD␣NORMAL␣DEVIATE",
           ylab="STREAMFLOW,␣IN␣CFS")
lines(qF,par2qua(F,umpqua.wak), col=2)
mtext("Umpqua␣River,␣Oregon")
```

```
lines(c(qnorm(0.99),qnorm(0.99)), c(1,1000000), lty=2)
#dev.off()
```

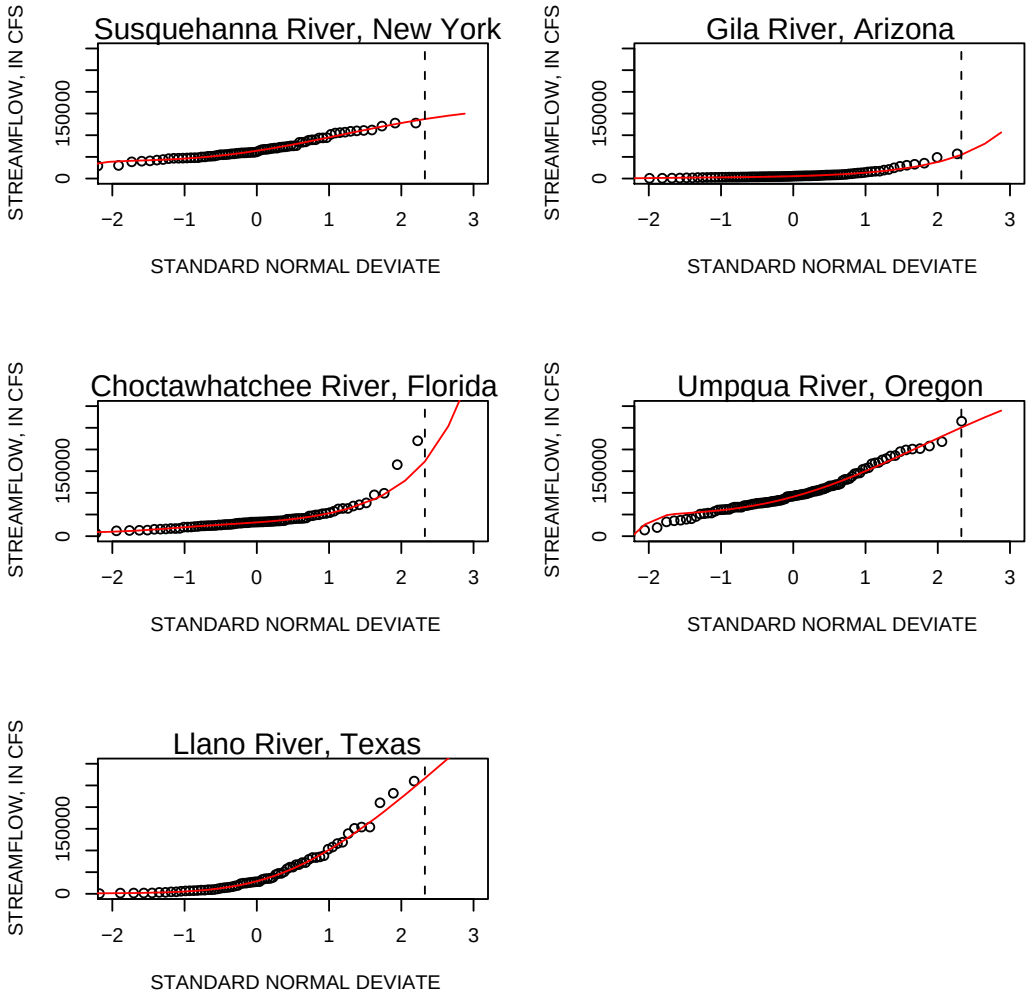

Figure 11.7. Plots of QDF of fitted Wakeby distribution of annual peak streamflow and empirical distribution for five selected river basins in the United States from example 11–18

Next, plots of the PDF of the data are created using the function `check.pdf()` in example 11–19 and the results are shown in figure 11.8. The PDFs each have their own unique geometry (shape); however, as also seen in figure 11.5, the fitted Wakeby distributions are clearly right-tail heavy (positive skewness). The fit for the Umpqua River shows the beginnings of asymmetric behavior towards $-\infty$ in figure 11.7 and more clearly in figure 11.8. Truncation of this fit to a zero lower bound could be made following the material in Section 12.6 but is not made here.

11–19

```
#pdf("usrivers_pdf.pdf")
layout(matrix(1:6, ncol=2))

check.pdf(pdfwak,susque.wak, plot=TRUE)
mtext("Susquehanna␣River,␣New␣York")

check.pdf(pdfwak,chocta.wak, plot=TRUE)
mtext("Choctawhatchee␣River,␣Florida")

check.pdf(pdfwak,llano.wak, plot=TRUE)
mtext("Llano␣River,␣Texas")

check.pdf(pdfwak,gila.wak, plot=TRUE)
mtext("Gila␣River,␣Arizona")

check.pdf(pdfwak,umpqua.wak, plot=TRUE)
mtext("Umpqua␣River,␣Oregon")
#dev.off()
```

◀

Interest in the distribution of annual peak streamflow is primarily in the right tail because the design of that water-related (drainage) infrastructure is dependent on high-magnitude events. Focused attention, therefore, is made on the fit in the right tail. Inspection of figure 11.7 suggests that the Wakeby distribution can mimic the empirical distribution of the data. However, there are concerns about underestimation of peak-streamflow magnitude for large standard normal deviates for the Choctawhatchee and Llano Rivers because the fitted distribution plots to the right of the two (Chocatawhatchee) and three (Llano) largest values. The lack of apparent tail fit for the Choctawhatchee is especially troublesome and more investigation is warranted.

The Chocatawhatchee River distribution is further considered using the Generalized Lambda distribution (see Section 9.2.2) in example 11–20. In the example, the parameter solution, which has an acceptable least-square error (ϵ, see eq. (11.3)) for τ_3 and τ_4 and $\Delta\tau_5$ (see eq. (11.2)), is returned from 14 optimization attempts by the `pargld()` function. Each attempt is started differing combinations of Generalized Lambda parameter space (see Karian and Dudewicz, 2000). The values for `chocta.gld1` are shown by the `print()` function. The output also shows the contents of `chocta.gld4$rest`, which shows other solutions. These are collectively treated in a later example.

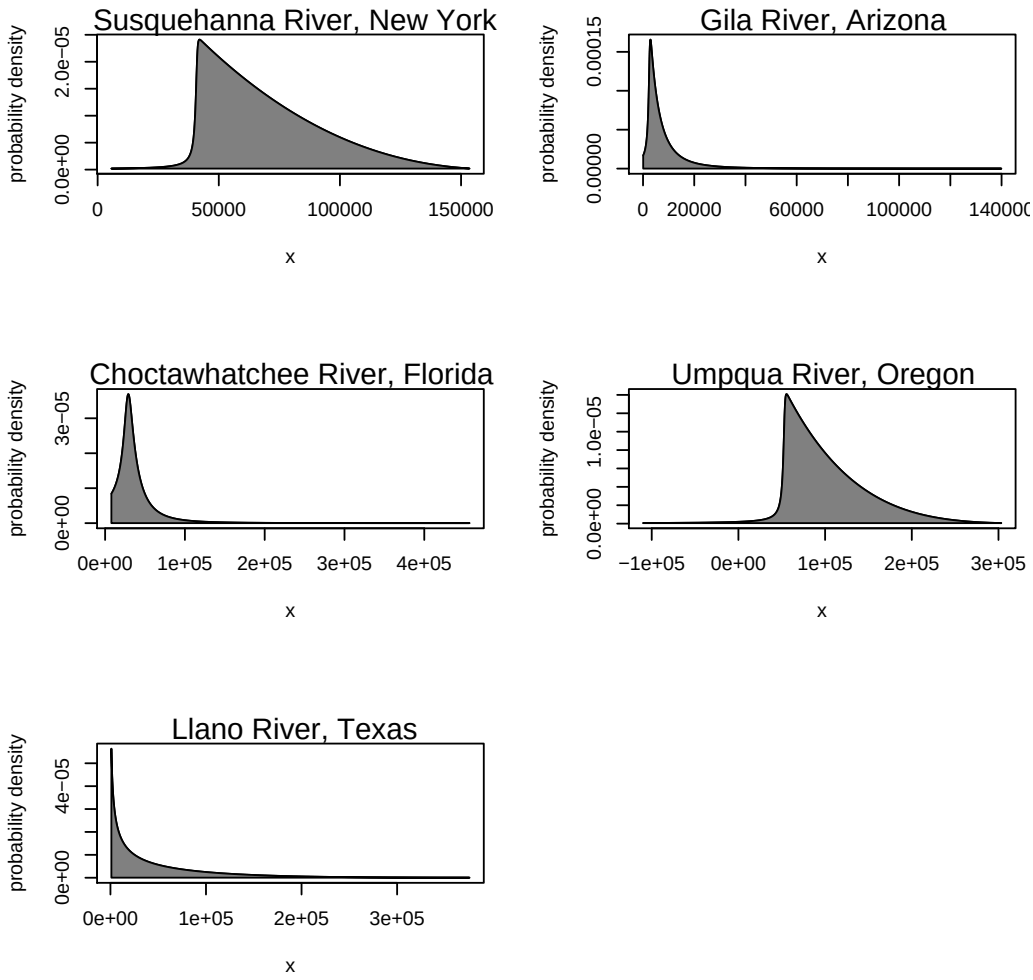

Figure 11.8. Plots of PDF of fitted Wakeby distribution of annual peak streamflow for five selected river basins in the United States from example 11–19

11–20

```
chocta.gld4 <- pargld(chocta.lmr, eps=1e-2)
# print(chocta.gld4) # edited for brevity
$type
[1] "gld"
$para
          xi        alpha        kappa            h
1.666864e+05 1.505572e+05 3.356197e+01 1.462652e-01
$delTau5
[1] -0.002264453
$error
[1] 0.004366765
$source
[1] "pargld"
```

```
$rest
      xi   alpha      kappa          h     delTau5        error
1  26140  -22019 -0.144155 -0.439496 -0.01103901 4.872965e-09
2  26131  -22049 -0.143872 -0.439247 -0.01109174 5.873891e-09
3 -87708 -101009  5.848814 -0.289431  0.09507022 3.123230e-10
4 -87713 -101014  5.848653 -0.289428  0.09507910 6.235505e-10
5  26841  120943  7.315955 70.672263 -0.24461201 2.798407e-06
6  26855  120648  7.298149 70.072798 -0.24552523 3.524567e-09
```

The values under the `delTau5` and `error` headings represent the following quantities

$$^{(i)}\Delta\tau_5 = {}^{(i)}\tau_5^{\text{gld}} - \hat{\tau}_5 \tag{11.2}$$

$$^{(i)}\epsilon = \left({}^{(i)}\tau_3^{\text{gld}} - \hat{\tau}_3\right)^2 + \left({}^{(i)}\tau_4^{\text{gld}} - \hat{\tau}_4\right)^2 \tag{11.3}$$

where (i) is the ith attempt, τ_r^{gld} represent the rth L-moment of the fitted Generalized Lambda and τ_r represent sample values. The $\Delta\tau_5$ represents the difference between τ_5 of the fitted distribution and that of the data. This difference is not explicitly minimized by the algorithm in the `pargld()` function, but this difference can be used to judge the merit of a given solution for the Generalized Lambda. On the other hand, the ϵ values do represent minimizations performed by the `optim()` function that is repetitively called by the `pargld()` function.

Example 11–20 shows that four different "solutions" exist that could be acceptable. A solution is represented by attempts (1 and 14). In example 11–21, the three additional Generalized Lambda solutions are set into descriptive variable names.

11–21

```
chocta.gld1 <-
vec2par(c(26140, -22019, -0.144155, -0.439496),
        type="gld")

chocta.gld2 <-
vec2par(c(-87708, -101009, 5.848814, -0.289431),
        type="gld")

chocta.gld3 <-
vec2par(c(26841, 120943, 7.315955, 70.672263),
        type="gld")
```

To clarify, the contents of variables `chocta.gld1`, `chocta.gld2`, `chocta.gld3`, and `chocta.gld4`, these Generalized Lambda solutions and their diagnostic errors are shown in the following equation ensemble

$$\mathrm{GLD}_1(F) = 26140 - 22019[F^{-0.144155} - (1-F)^{-0.439496}] \tag{11.4}$$
$$\Delta\tau_5 \approx -0.011,\ \epsilon < 10^{-8}$$

$$\mathrm{GLD}_2(F) = -87708 - 101009[F^{5.848814} - (1-F)^{-0.289431}] \tag{11.5}$$
$$\Delta\tau_5 \approx 0.095,\ \epsilon < 10^{-9}$$

$$\mathrm{GLD}_3(F) = 26841 + 120943[F^{7.315955} - (1-F)^{70.672263}] \tag{11.6}$$
$$\Delta\tau_5 \approx -0.245,\ \epsilon < 10^{-5}$$

$$\mathrm{GLD}_4(F) = 166686 + 150557[F^{33.56197} - (1-F)^{0.146265}] \tag{11.7}$$
$$\Delta\tau_5 \approx -0.002,\ \epsilon < 10^{-2}$$

where the nomenclature $\mathrm{GLD}_1(F)$ is equivalent to `chocta.gld1` and extends to the other three solutions.

Which of the solutions is most preferable? The choice of solution is a problem shows that a major feature and yet misfeature of the Generalized Lambda. The $\mathrm{GLD}_2(F)$ solution is optimal as measured by the ϵ (the smallest of the 14 attempts), but the $\Delta\tau_5$ is not as small as two of the other solutions. The $\mathrm{GLD}_3(F)$ solution also has a small ϵ, but the $\Delta\tau_5$ is extremely large—one would expect this solution to appear quite different from the data. It appears on further consideration that the $\mathrm{GLD}_1(F)$ and $\mathrm{GLD}_4(F)$ solutions might be preferable. Readers are asked to recall that $\mathrm{GLD}_4(F)$ was returned by the `pargld()` function.

In example 11–22, empirical distribution along with the four Generalized Lambda solutions and the Wakeby distribution are plotted and are shown in figure 11.9.

11–22

```
#pdf("usrivers_gld.pdf")
xs <- qnorm(chocta.pp)

layout(1) # Previous layouts are matrices, so this might be
          # needed to reset things following earlier examples

plot(qnorm(chocta.pp),  log10(sort(chocta.Q)),
     xlab="STANDARD␣NORMAL␣DEVIATE",
     ylab="LOG10(STREAMFLOW,␣IN␣CFS)")
lines(xs, log10(par2qua(chocta.pp,chocta.wak)), lwd=0.7, lty=2)
lines(xs, log10(par2qua(chocta.pp,chocta.gld1)),  lwd=1, lty=1)
lines(xs, log10(par2qua(chocta.pp,chocta.gld2)),  lwd=2, lty=3)
lines(xs, log10(par2qua(chocta.pp,chocta.gld3)),  lwd=3, lty=4)
lines(xs, log10(par2qua(chocta.pp,chocta.gld4)),  lwd=3, lty=1)
```

```
legend(-2,5, lty=c(2,1,3,4,1), lwd=c(0.7,1,2,3,3),
       c("Wakeby", "GLD:␣chocta.gld1", "GLD:␣chocta.gld2",
         "GLD:␣chocta.gld3", "GLD:␣chocta.gld4"))
#dev.off()
```

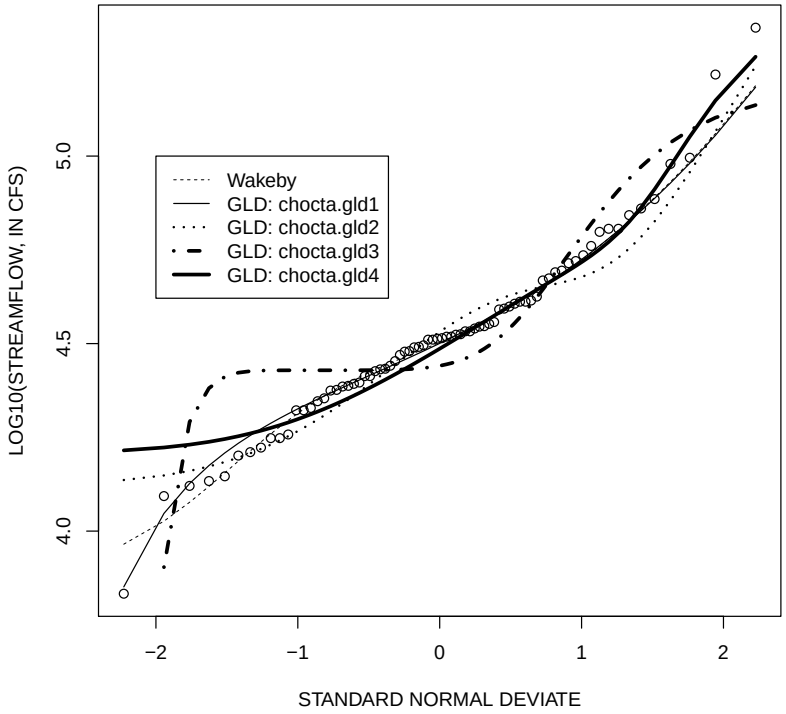

Figure 11.9. Empirical distribution of annual peak streamflow data for U.S. Geological Survey streamflow-gagin station 02366500 Choctawhatchee River near Bruce, Florida and Wakeby and four Generalized Lambda distributions fit by method of L-moments from example 11–22

Inspection of the figure suggests that the solution `chocta.gld3` clearly has a poor fit, which is consistent with the large $\Delta\tau_5 \approx 0. - 0.245$. Also solution `chocta.gld2` has a questionable fit; although the τ_3 and τ_4 match that of the data, the $\Delta\tau_5 \approx 0.095$ is not "small" compared to `chocta.gld1` and `chocta.gld4`.

The two solutions, `chocta.gld1` and `chocta.gld4`, both have potentially acceptable fits depending on how the analyst interprets the diagnostics. If the analyst is interested in distributional analysis for the right tail of the distribution as in the case of assessment of flood risk to drainage infrastructure (say a bridge), then the fit of `chocta.gld4` might be preferable. However, this solution diverges from the data in the left tail. If the objective of

distributional analysis is to generally mimic the full spectrum of the flood potential, then solution `chocta.gld1` might be preferable.

In conclusion, the choice between different distributions or multiple solutions for a distribution is not a trivial one and serious reflection on the topic is needed in many real-world circumstances. Ambiguity never-the-less often remains or is intrinsically expected in distributional analysis of hydrometeorological data or other non-Normal data with small sample sizes. Whereas goodness-of-fit tests, which are outside the scope here, or L-moment ratio diagrams provide guidance, it can be difficult to separate the purity of statistical computations from the context to model building process of the distributional analysis for a given problem. ◀

11.3 Summary

In this chapter, 22 examples for two short studies of distributional characteristics involving annual maximum rainfall, and annual peak streamflow data are presented. In the former study, the rainfall data from the observation network are assumed to be drawn from a common parent distribution and the sample L-moments could be pooled together by weighted-mean values to estimate a regional Kappa distribution. This distribution is used to estimate a 100-year rainfall event that is similar to one derived from a previous study. In the later study, the streamflow data from the observation network represent distinct distributions that reflect the unique climatic and physiographic characteristics of five widely dispersed river basins. The Wakeby distribution is used, but concerns over fit for one river basin in particular led to a detailed presentation of the use of the Generalized Lambda distribution.

Chapter 12
Advanced Topics

In this chapter, I present many generally distinct topics on advanced topics of distributional analysis using L-moments and probability-weighted moments. Inclusion of this material in earlier chapters would, in my opinion, unnecessarily detract from the central theme up until this point in the narrative. Familiarity L-moments and ancillary statistics and their support in the *lmomco* package is now assumed. Primary contributions of this chapter are detailed treatment of both left- and right-tail censoring, conditional probability adjustment, and multivariate L-moments. Secondary contributions are an exploration of quantile uncertainty by simulation and "journal article" like comparison of product moments and L-moments, which are each applied real- and logarithmic space, for the Pearson Type III distribution. This chapter demonstrates a significant expansion of capabilities for distributional analysis with L-moment statistics using R.

12.1 Introduction

Several generally more advanced L-moment-related topics are discussed in this chapter. These topics do not pigeon-hole well into other portions of this dissertation, but these topics never-the-less contribute to distributional analysis with L-moment statistics. The use of L-moments and probability-weighted moments for **right-tail censoring** is described in Section 12.2. Following in parallel, the use of L-moments and probability-weighted moments for **left-tail censoring** is described in Section 12.3. The censoring discussion subsequently expands in Sections 12.4 and 12.5 to include censoring by indicator variable and detailed discussion of a method known as flipping to support **left-tail censoring** from right-tail censoring operators.

Following the censoring material, conditional probability adjustment for zero values is shown through **blipped-distribution modeling**, which is followed by an exploration of **quantile uncertainty** in the context of **sampling error** and **model-selection error** (error attributable to distribution selection). An extensive comparison between product moments and L-moments for the Pearson Type III distribution follows. Finally, L-comoments, which are multivariate extensions of L-moments, are described in the last section of this chapter and thus appropriately conclude this dissertation.

12.2 L-moments from Probability-Weighted Moments for Right-Tail Censored Distributions

Limited discussion of distributional analysis with L-moments for censored distributions is provided in this and the next three sections. More thorough treatment of L-moments for both (or either) right-tail and left-tail censoring is found in Wang (1990a; 1990b; 1996a), Hosking (1995), Kroll and Stedinger (1996), Zafirakou-Koulouris and others (1998), and Wang and others (2010). The probability-weighted moments under conditions of censoring are commonly referred to as **partial probability-weighted moments** by those authors. Finally, the L-moment ratio diagrams for censored distributions and samples are described in Hosking (1995) and Zafirakou-Koulouris and others (1998) but are not considered in this dissertation. The author recognizes the substantial nuances associated with analysis of censored data and accordingly recommends the book by Helsel (2005) and the ancillary R package *NADA* by Lee (2009).

This section concerns right-tail censoring that is restricted to circumstances involving a constant censoring threshold T. The threshold might be known or unknown, but invariate during the course of the sampling of the random variable. A different style of right-tail censoring is discussed in Section 12.4. Much of the material in this section is drawn from Hosking (1995).

A common data type in studies of lifetimes, survival, and reliability are right-tail censored. Specifically, the sample of size n is not fully measured on the high-magnitude portion of the distribution. Two types of right-tail censoring (and left-tail censoring by analogy) are recognized. For **right-tail type I censoring**, a **right-tail censoring threshold** T is known and m values are less than this value and $n - m$ values are greater than or equal to T. For **right-tail type II censoring**, only the m smallest values are observed and

the $n-m$ values are censored above the threshold $X_{m:n}$, which is the largest noncensored order statistic.

The **right-tail censoring fraction** (ζ) is a convenient parameter to accommodate data censoring in probability-weighted moment computations. The censoring fraction satisfies the relation $\zeta = F(T)$ for the CDF of random variable X with a QDF of $x(F)$. Differences between type I and type II censoring exist by definition and in sampling properties. These differences become less important as sample size becomes large. Values for ζ can be estimated by $\zeta = m/n$; this is not necessarily an optimal choice, but for convenience, it is all that is considered by Hosking (1995) as well as in this dissertation and the *lmomco* package.

Zafirakou-Koulouris and others (1998, p. 1246) provide additional discussion of type I and type II censoring: "Since the censoring threshold T is fixed in type I censoring, m is a random variable with a binomial distribution. Otherwise, type II censoring results, and T becomes the random variable, with m fixed."

12.2.1 Theoretical Probability-Weighted Moments for Right-Tail Censored Distributions

The theoretical probability-weighted moments of a right-tail censored distribution having a QDF of $x(F)$ are defined as two types (Hosking, 1995). The theoretical "A"- and "B"-type probability-weighted moments for a right-tail censored random variable $X_{1:n} < X_{2:n} < \cdots < X_{m:n} < T = X_{m+1:n} = X_{m+2:n} = \cdots = X_{n:n}$, where the censoring threshold remains denoted as T, are now defined.

The definition requires a conceptualization of two sample types. Consider first the uncensored values of random sample of size m has a QDF expressed as

$$y^A(F) = x(\zeta F) \tag{12.1}$$

Whereas second, the complete random sample of size n has a QDF expressed as

$$y^B(F) = \begin{cases} x(F) & \text{for } 0 < F < \zeta \\ x(\zeta) = T & \text{for } \zeta \leq F < 1 \end{cases} \tag{12.2}$$

Using the two definitions for QDF, Hosking (1995) shows that the probability-weighted moments for moment order r for $r \geq 0$, for the respective QDF are

$$\begin{aligned}
\beta_r^A &= \int_0^1 F^r \, y^A(F) \, \mathrm{d}F \\
&= \frac{1}{\zeta^{r+1}} \int_0^\zeta F^r \, x(F) \, \mathrm{d}F \\
&= \frac{1}{[F(T)]^{r+1}} \int_{-\infty}^T [F(x)]^r \, x \, \mathrm{d}F(x)
\end{aligned} \tag{12.3}$$

for the "A"-type probability-weighted moments and

$$\begin{aligned}
\beta_r^B &= \int_0^1 F^r \, y^B(F) \, \mathrm{d}F \\
&= x(\zeta)\frac{1-\zeta^{r+1}}{r+1} + \int_0^\zeta F^r \, x(F) \, \mathrm{d}F \\
&= T\frac{1-\zeta^{r+1}}{r+1} + \int_0^\zeta F^r \, x(F) \, \mathrm{d}F \\
&= T\frac{1-[F(T)]^{r+1}}{r+1} + \int_{-\infty}^T [F(x)]^r \, x \, \mathrm{d}F(x)
\end{aligned} \tag{12.4}$$

for the "B"-type probability-weighted moments. Finally, the relation between A- and B-type probability-weighted moments is

$$\beta_{r-1}^B = \frac{1}{r}[r \, \zeta^r \, \beta_{r-1}^A + (1-\zeta^r)T] \tag{12.5}$$

where $x(\zeta)$ is the value of the QDF at nonexceedance probability $F = \zeta$. In other words, ζ is the right-tail censoring fraction or the probability $\Pr[]$ that x is less than the quantile at ζ nonexceedance probability: ($\Pr[x < X(\zeta)]$). The choice of A- and B-type in derivations of probability-weighted moments or L-moments for censored distributions can be made by mathematical convenience according discussion by Hosking (1995).

12.2.2 Sample Probability-Weighted Moments for Right-Tail Censored Data

The sample A- and B-type probability-weighted moments (Hosking, 1995) are computed for a right-tail censored sample $x_{1:n} < x_{2:n} < \cdots < x_{m:n} < T = x_{m+1:n} = x_{m+2:n} = \cdots = x_{n:n}$, where the censoring threshold is denoted as T. The data possess m values that are observed (noncensored, $< T$) out of a total of n samples. The ratio of m to n is defined as $\zeta = m/n$, which plays an important role in parameter estimation. The ζ is interpreted as the probability that x is less than the QDF at $F = \zeta$: $\Pr[x < x(\zeta)]$. The sample A-type probability-weighted moments are defined by

$$\hat{\beta}_r^A = \frac{1}{m}\binom{m-1}{r}^{-1} \sum_{j=1}^{m} \binom{j-1}{r} x_{j:n} \tag{12.6}$$

which, to reiterate the definition, are the already familiar probability-weighted moments of the uncensored sample of Chapter 5 for m observed values.

The sample B-type probability-weighted moments conversely are computed from the "complete" sample, in which the $n - m$ censored values are replaced by the T right-tail censoring threshold. The B-type probability-weighted moments are defined by

$$\hat{\beta}_r^B = \frac{1}{n}\binom{n-1}{r}^{-1} \left[\sum_{j=1}^{m} \binom{j-1}{r} x_{j:n} + \sum_{j=m+1}^{n} \binom{j-1}{r} T \right] \tag{12.7}$$

When there are more than a few censored values, the $\hat{\beta}_r^A$ and $\hat{\beta}_r^B$ are readily estimated by computing the β_r^A and using the expression

$$\hat{\beta}_r^B = Z\beta_r^A + \frac{1-Z}{r+1}T \tag{12.8}$$

where

$$Z = \frac{m}{n}\binom{m-1}{r} \Big/ \binom{n-1}{r} \tag{12.9}$$

to make the conversion, as it were, to the $\hat{\beta}_r^B$.

The A- and B-type probability-weighted moments are easily converted to A- and B-type L-moments by the usual linear methods (see eq. (6.32)), such as supported by the `pwm2lmom()` function.

As identified by Hosking (1995), Hamada (1995, table 9.3) provides a table of lifetime-to-breakage measured in cycles for drill bits used for producing small holes in printed circuit boards. The data are originally credited to an F. Montmarquet. The data were collected under various control and noise factors to perform reliability assessment to maximize bit reliability with minimization of hole diameter. Smaller holes permit higher density of placed circuitry, and thus small holes are economically attractive.

The lifetime-to-breakage testing was completed at $3{,}000$ cycles—the right-tail censoring threshold or $T = 3{,}000$. For purposes of demonstration of right-tail censoring using A- and B-type probability-weighted moments, these data have been merged into a single sample in data `DrillbitLifetime` of the *lmomco* package.

Beginning in example 12–1, the drill-bit lifetime data are set into `X` and the Weibull plotting positions computed by the `pp()` function. The right-tail censored probability-weighted moments are computed for the sample using the `pwmRC()` function. Subsequently, the parameters for the Generalized Pareto and Right-Censored Generalized Pareto distributions are computed by the `pargpa()` (usual L-moments and probability-weighted moments) and `pargpaRC()` functions, respectively.

12–1

```
data(DrillBitLifetime) # from lmomco package
X <- DrillBitLifetime$LIFETIME
PP <- pp(X); RCpwm <- pwmRC(X,3000)
paragpa   <- pargpa(pwm2lmom(pwm(X))) # usual PWMs (no censoring)
paragpaRC <- pargpaRC(pwm2lmom(RCpwm$Bbetas), RCpwm$zeta)
```

The demonstration continues in example 12–2, and the results are shown in figure 12.1. The fits of the two Generalized Pareto distributions differ considerably. With special attention to the approximate interval F: $[0.7, 0.9]$, it is obvious that the Right-Censored Generalized Pareto provides a preferable fit over the Generalized Pareto. The Generalized Pareto fit is swung too far to the right because this fit improperly "feels" the many values equal to, that is censored, to the value of $3{,}000$.

12–2

```
#pdf("lifetime.pdf")
plot(1-PP,qlmomco(1-PP,paragpaRC), type="l", lwd=3,
     xlab="EXCEEDANCE␣PROBABILITY", ylim=c(0,4000),
     ylab="LIFE␣TIME,␣CYCLES")    # thick line
lines(1-PP,qlmomco(1-PP,paragpa)) #  thin line
points(1-PP,sort(X, decreasing=TRUE))
```

```
legend(0,4000, lwd=c(3,1), lty=c(1,1), box.lty=0, bty="n",
       c("right-censored␣Gen.␣Pareto␣distribution",
         "Gen.␣Pareto␣distribution"))
#dev.off()
```

◀

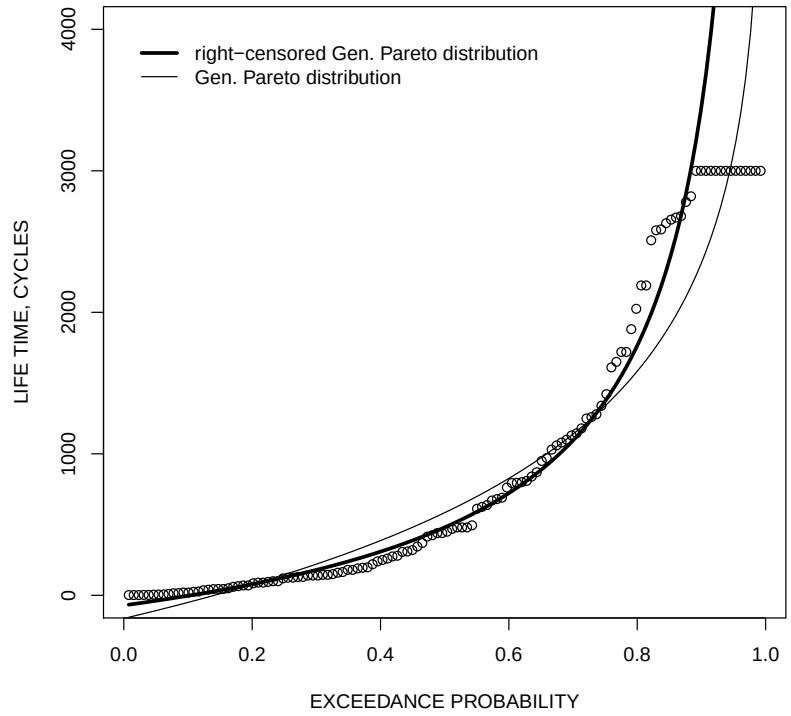

Figure 12.1. Comparison of Right-Censored Generalized Pareto distribution fit to right-tail censored probability-weighted moments (thick line) and Generalized Pareto fit to whole sample probability-weighted moments (thin line) from example 12–2. The thick line represents the preferable fit to the data.

12.3 L-moments from Probability-Weighted Moments for Left-Tail Censored Distributions

This section concerns left-tail censoring that is restricted to circumstances involving a constant censoring threshold T. The threshold might be known or unknown, but invariate during the course of the sampling of the random variable. A different style of left-tail

censoring is discussed in Section 12.5. Much of the material in this section is drawn from Zafirakou-Koulouris and others (1998).

It is common in studies of hydrologic or environmental data (particularly water quality or chemistry) to be left-tail censored. Specifically, the sample of size n is not fully measured on the low-magnitude portion of the distribution. This is known as the **detection limit** problem. Two types of left-tail censoring are recognized. For **left-tail type I censoring**, a **left-tail censoring threshold** T is known and $n-m$ values are greater than this value and m values are smaller than or equal to T. For **left-tail type II censoring**, only the $n-m$ largest values are observed and the m values are censored below the threshold $X_{m:n}$, which is the smallest noncensored order statistic.

The **left-tail censoring fraction** (ζ) is a convenient parameter to accommodate data censoring in probability-weighted moment computations. The censoring fraction satisfies the relation $\zeta = F(T)$ for the CDF of random variable X with a QDF of $x(F)$. Differences between type I and type II censoring exist by definition and in sampling properties. These differences become less important as sample size becomes large. Values for ζ can be estimated by $\zeta = m/n$; this is not necessarily an optimal choice, but for convenience, it is all that is considered by Zafirakou-Koulouris and others (1998) as well as in this dissertation and the *lmomco* package.

12.3.1 Theoretical Probability-Weighted Moments for Left-Tail Censored Distributions

The theoretical probability-weighted moments of a left-tail censored distribution having a QDF of $x(F)$ are defined as two types (Zafirakou-Koulouris and others, 1998). The theoretical "A-prime"- and "B-prime"-type probability-weighted moments for a left-tail censored random variable $X_{1:n} = \cdots = X_{m:n} = T < X_{m+1:n} < \cdots < X_{n:n}$, where the censoring threshold remains denoted as T, are now defined.

The definition requires a conceptualization of two sample types. Consider first the uncensored values of random sample of size $n-m$ has a QDF expressed as

$$y^{A'}(F) = x[(1-\zeta)F + \zeta] \tag{12.10}$$

Whereas second, the complete random sample of size n has a QDF expressed as

$$y^{B'}(F) = \begin{cases} x(\zeta) = T & \text{for } 0 < F \le \zeta \\ x(F) & \text{for } \zeta < F < 1 \end{cases} \tag{12.11}$$

Using the two definitions for QDF, Zafirakou-Koulouris and others (1998) show that the probability-weighted moments for moment order r for $r \ge 0$, for the respective QDF are

$$\begin{aligned} \beta_r^{A'} &= \int_0^1 F^r y^{A'}(F)\, \mathrm{d}F \\ &= \frac{1}{(1-\zeta)^{r+1}} \int_\zeta^1 (F-\zeta)^r\, x(F)\, \mathrm{d}F \\ &= \frac{1}{(1-[F(T)])^{r+1}} \int_T^\infty [F(x) - F(T)]^r\, x\, \mathrm{d}F(x) \end{aligned} \tag{12.12}$$

for the "A-prime"-type probability-weighted moments and

$$\begin{aligned} \beta_r^{B'} &= \int_0^1 F^r\, y^{B'}(F)\, \mathrm{d}F \\ &= x(\zeta)\frac{\zeta^{r+1}}{r+1} + \int_\zeta^1 F^r\, x(F)\, \mathrm{d}F \\ &= T\frac{\zeta^{r+1}}{r+1} + \int_\zeta^1 F^r\, x(F)\, \mathrm{d}F \\ &= T\frac{[F(T)]^{r+1}}{r+1} + \int_T^\infty [F(x)]^r\, x\, \mathrm{d}F(x) \end{aligned} \tag{12.13}$$

for the "B-prime"-type probability-weighted moments.

12.3.2 Sample Probability-Weighted Moments for Left-Tail Censored Data

The sample A′- and B′-type probability-weighted moments (Zafirakou-Koulouris and others, 1998) are computed for a left-tail censored sample $x_{1:n} = \cdots = x_{m:n} = T < x_{m+1:n} < \cdots < x_{n:n}$, where the censoring threshold is denoted as T. The data possess $n - m$ values that are observed (noncensored, $> T$) out of a total of n samples. The ratio of m to n is defined as $\zeta = m/n$, which plays an important role in parameter estimation. The ζ is interpreted as the probability that x is greater than the QDF at $F = \zeta$: $\Pr[x > x(\zeta)]$. The sample A′-type probability-weighted moments are defined by

$$\hat{\beta}_r^{A'} = \frac{1}{n-m}\binom{n-m-1}{r}^{-1} \sum_{j=m+1}^{n} \binom{j-m-1}{r} x_{j:n} \tag{12.14}$$

which, to reiterate the definition, are the already familiar probability-weighted moments of the uncensored sample of Chapter 5 for k observed values.

The sample B′-type probability-weighted moments conversely are computed from the "complete" sample, in which the $n - m$ censored values are replaced by the T left-tail censoring threshold. The B′-type probability-weighted moments are defined by

$$\hat{\beta}_r^{B'} = \frac{1}{n}\binom{n-1}{r}^{-1} \left[\sum_{j=1}^{m} \binom{j-1}{r} T + \sum_{j=m+1}^{n} \binom{j-1}{r} x_{j:n} \right] \tag{12.15}$$

The A′- and B′-type probability-weighted moments are easily converted to A′- and B′-type L-moments by the usual linear methods (see eq. (6.32)), such as supported by the `pwm2lmom()` function.

USING R

Hosking (1995, table 29.2, p. 551) provides some right-tail censored data, which has prior use in the literature, for the lifetimes in weeks of 33 transistors.[1] These data are reproduced in example 12–3 in which the three values of 52 weeks are right-censored and the value 51.9999 is a numerical hack so that a threshold of 52 can be used in the function `pwmRC()` to compute the A- and B-type probability-weighted moments. The data are converted to left-tail censored by flipping and set into the `LC` variable (see Section 12.5 for full description of variable flipping). The example ends by reporting the right-censored L-moments. These can be compared to back-flipped, left-censored L-moments shown in the next example.

12–3

```
life.time <- c(3, 4, 5, 6, 6, 7, 8, 8, 9, 9, 9, 10, 10, 11, 11,
11, 13, 13, 13, 13, 13, 17, 19, 19, 25, 29, 33, 42, 42, 51.9999,
52, 52, 52) # last three are censored at 52
# 51.9999 was really 52, a real (noncensored) data point.
flip <- 100; T <- 52 # The flipping value and the threshold
LC <- flip - life.time # convert the data to left-censored
RCpwm <- pwmRC(life.time, threshold=T)
pwm2lmom(vec2pwm(RCpwm$Abetas)) # A-type PWM --> A,L-moments
```

[1] Hosking (1995) reports the count as 34 transistors in the title of table 29.2, but the 33 provided values from that table are reproduced here.

```
$lambdas
[1] 15.666663  6.202296  2.499668  1.513826  0.377672
$ratios
[1]         NA 0.39589129 0.40302308 0.24407516 0.06089229

pwm2lmom(vec2pwm(RCpwm$Bbetas)) # B-type PWM --> B,L-moments
$lambdas
[1] 18.9696939  8.2064369  3.0736178  1.0279813 -0.5654883
$ratios
[1]         NA  0.4326078  0.3745374  0.1252652 -0.0689079
```

The left-censored L-moments for the data in `LC` are computed in example 12–4 by the `pwmLC()` function, which implements eqs. (12.14) and (12.15). The `fliplmoms()` function provides the back flipping of the L-moments.

12–4

```
LCpwm  <- pwmLC(LC, threshold=(flip - T))
LCpwmA <- vec2pwm(LCpwm$Aprimebetas)
LCpwmB <- vec2pwm(LCpwm$Bprimebetas)
  # LClmrA <- pwm2lmom(LCpwmA) # These three commented out
  # LClmrA$flip <- 100 # steps would also work. These steps
  # fliplmoms(LClmrA)  # document subtle details of use.
fliplmoms(pwm2lmom(LCpwmA), flip=flip)
$lambdas
[1] 15.666663  6.202296  2.499668  1.513826  0.377672
$ratios
[1]         NA 0.39589129 0.40302308 0.24407516 0.06089229

fliplmoms(pwm2lmom(LCpwmB), flip=flip) # back-flip the L-moments
$lambdas
[1] 18.9696939  8.2064369  3.0736178  1.0279813 -0.5654883
$ratios
[1]         NA  0.4326078  0.3745374  0.1252652 -0.0689079
```

Examples 12–3 and 12–4 show the A- and B-type L-moments and back-flipped A′- and B′-type L-moments, respectively. These are congruent as judged by the equality of the moments on upon one-to-one comparison. The reliability of the `pwmLC()` function is demonstrated. ◀

12.4 L-moments of Right-Tail Censored Data by Indicator Variable

Section 12.2 considers the computation of L-moments in right-tail censoring circumstances involving a known or unknown, but constant, censoring threshold T. Wang and others

(2010) thoroughly describe a method to estimate L-moments based on a right-tail censoring indicator, which has application in survival or failure analysis.[2] For each of the sample order statistics $x_{1:n} \leq x_{2:n} \leq \cdots \leq x_{n:n}$ of random variable X, it is known that $x_j = \min(X_j, T)$ for a "noninformative" T (Wang and others, 2010). The noninformative nature of the censoring is very important and salient discussion is provided by Helsel (2005, pp. 30–33). The censoring threshold is unknown, is not explicitly needed, and T is itself possibly a random variable generated along side each realization of X: $x_j = \min(X_j, T_j)$. For the sample order statistics, let $\delta_{j:n}$ be indicators of right-tail censoring: $\delta_{j:n} = 0$ indicates that $x_{j:n}$ is uncensored, whereas, $\delta_{j:n} = 1$ indicates that $x_{j:n}$ is right-tail censored. Censoring that requires an indicator variable might occur as (1) right-tail censoring by patients leaving (no longer participating in) survival studies after medical procedures or as (2) left-tail censoring when multiple detection limits are used, which is common with environmental quality (chemical) data.

Wang and others (2010) describe an L-moment estimation method, which relies on the empirical survival function to determine weight factors on the observed (noncensored) values of the order statistics. These weight factors converge to those of the usual L-moments as the number of censored values goes to zero. The empirical survival function is defined as

$$\hat{S}_{j:n}(x) = \begin{cases} 1 & j = 0 \text{ (special condition, see text)} \\ \prod_{X_{j:n} \leq x} \left(\frac{n-j}{n-j+1}\right)^{1-\delta_{j:n}} & X_{1:n} \leq x < X_{n:n} \\ 0 & x \geq X_{n:n} \end{cases} \tag{12.16}$$

Using eq. (12.16) as the survival function in the role of a complemented[3] plotting position, the sample L-moments are computed by

$$\hat{\lambda}_r = \sum_{j=1}^{n} w_{j:n}(r) X_{j:n} \tag{12.17}$$

where $w_{j:n}(r)$ is a weight factor that is computed by

[2] Wang and others (2010) consider survival data, which is strictly greater than or equal to zero, but such a restriction is lifted here.

[3] Plotting positions are defined in this dissertation as nonexceedance probabilities. The survival function for the definitions in this section is an expression of exceedance probability; therefore, the complement is needed. The complement is seen in the $1 - \hat{S}$ in eqs. (12.19) and (12.20).

$$w_{j:n}(r) = \frac{1}{r}\sum_{k=0}^{r-1}(-1)^k \binom{r-1}{k}(B^\star - B_\star) \tag{12.18}$$

where B is the CDF of the Beta distribution $\mathrm{B}(q, a, b)$ for quantile q and parameters a and b. The two Beta distributions $B^\star$ and $B_\star$ are computed by

$$B^\star = \mathrm{B}(1 - \hat{S}_{j:n}(X_{j:n}), r-k, k+1) \tag{12.19}$$

$$B_\star = \mathrm{B}(1 - \hat{S}_{j-1:n}(X_{j-1:n}), r-k, k+1) \tag{12.20}$$

for parameters $r-k$ and $k+1$. Readers are asked to note that the $j-1$ term in eq. (12.20) takes on the value 0 for the first order statistic ($j = 1$). There obviously is no zeroth order statistic. Wang and others (2010) suggest $X_{0:n} = 0$, but such a condition, implies that $X \geq 0$. The result of $X_{0:n} = 0$ yields $\hat{S} = 1$. Therefore, the "special condition" in eq. (12.16) by the author (Asquith) has the same effect when $B_\star$ is computed for the first order statistic. The special condition lifts the $X \geq 0$ restriction and extends X to the real-number line $\mathbb{R}$.

Wang and others (2010) conduct, with an Exponential distribution censoring T_j, a simulation study of Generalized Extreme Value and two Weibull distributions and report that L-moment ratio diagrams still provide fair separation of $\hat{\tau}_3$ and $\hat{\tau}_4$. Wang and others also show that the method of L-moments generally performs better than the method of maximum likelihood for the Weibull distribution and strongly suggest use of L-moments for samples sizes less than 50 and less than 55 percent of right-tail censored data.

Helsel (2005, p. 77) recommends (quote follows) that the Kaplan-Meier method by Kaplan and Meier (1958) be used to compute summary statistics of right-tail censored data "for data with up to 50 [percent] censoring" because of "its predominant use in [nonenvironmental] disciplines" and "well-developed theory." Part of the Wang and others (2010) method is based on the Kaplan-Meier method. Helsel (2005, p. 67) reports that "estimates of standard deviation are even of less interest than the mean in traditional survival analysis [because of] the skewness found in most survival [and environmental[4]] data." The author of this dissertation advocates that such a statement should no longer be as applicable because of the support for L-moment computation on censored data because of the developments of Wang and others (2010).

[4] The author (Asquith) has added "environmental" as this data type is most certainly implied by Helsel.

The `lmomsRCmark()` function provides support for computation of $\hat{\lambda}_r$ and $\hat{\tau}_r$ for censored data by repeated calls to the `lmomRCmark()` function, which actually provides the implementation of eq. (12.17).

Efron (1988) provides survival-time data (these data also are used by Wang and others (2010) and thus utilized here in sequel) for 51 cancer patients in which 9 patients were lost (dropped out) from the study before death. These data are shown in example 12–5 as variable `Efron`. The time in days is to the left of the pairing comma and the right-tail censoring indicator is shown to the right of the comma. If the marking variable is 1, then the time is right-tail censored for a given sample.

12–5

```
Efron <-
c(7,0,  34,0,  42,0,  63,0,  64,0,  74,1,  83,0,  84,0,  91,0,
108,0,  112,0,  129,0,  133,0,  133,0,  139,0,  140,0,  140,0,
146,0,  149,0,  154,0,  157,0,  160,0,  160,0,  165,0,  173,0,
176,0,  185,1,  218,0,  225,0,  241,0,  248,0,  273,0,  277,0,
279,1,  297,0,  319,1,  405,0,  417,0,  420,0,  440,0,  523,1,
523,0,  583,0,  594,0,  1101,0,  1116,1,  1146,0,  1226,1,
1349,1,  1412,1, 1417,1)

# Break up the data,censor pairs into two vectors
ix <- seq(1,length(Efron), by=2) # create indexing variable
Efron.data   <- Efron[ix] # try repeating with a negation
Efron.rcmark <- Efron[(ix+1)]

# Ensure sorting and make sure to resort the indicator
# in case reader is experimenting with negation of the data
ix <- sort(Efron.data, index.return=TRUE)$ix
Efron.data   <- Efron.data[ix]
Efron.rcmark <- Efron.rcmark[ix]

# Distinguish between the data when graphing
# by changing the plotting character
my.pch <- Efron.rcmark
my.pch[Efron.rcmark == 0] <-  1  # open circle
my.pch[Efron.rcmark == 1] <- 16  # solid circle

ub   <- lmoms(Efron.data) # conventional sample L-moments
noRC <- lmomsRCmark(Efron.data) # ignore the censoring
RC   <- lmomsRCmark(Efron.data, rcmark=Efron.rcmark)
PP   <- pp(Efron.data) # plotting positions

ymax <- 1500
```

```
censored.data <- Efron.data[Efron.rcmark == 1]
n <- 3*length(censored.data)
ix <- seq(1,n, by=3) # create indexing variable
barsPP <- barsQ <- vector(mode="numeric", length=n)
barsPP[ix]     <- PP[Efron.rcmark == 1]
barsPP[(ix+1)] <- barsPP[ix]
barsPP[(ix+2)] <- NA
barsQ[ix]      <- censored.data
barsQ[(ix+1)]  <- ymax
barsQ[(ix+2)]  <- NA

#pdf("rcindicator.pdf")
plot(PP, Efron.data, ylim=c(0,ymax), type="n",
         xlab="NONEXCEEDANCE␣PROBABILIY", ylab="DATA")
lines(barsPP, barsQ, lty=3)
points(PP, Efron.data, pch=my.pch)
lines(PP, qlmomco(PP, lmom2par(noRC, type="kap")), lwd=3, col=8)
lines(PP, qlmomco(PP, lmom2par(ub,   type="kap")))
lines(PP, qlmomco(PP, lmom2par(RC,   type="kap")), lwd=2, lty=2)
legend(0,1000, c("Kappa␣by␣uncensored␣L-moments",
                 "Kappa␣by␣unbiased␣L-moments",
                 "Kappa␣by␣censored␣L-moments"),
             bty="n", lwd=c(3,1,2), col=c(8,1,1), lty=c(1,1,2))
legend(0.052,810, c("Uncensored␣data",
                    "Right-tail␣censored␣data"),
             bty="n", pch=c(1,16))
#dev.off()
```

The example computes three estimates of the sample L-moments: (1) the usual unbiased; (2) those by eq. (12.17), but ignoring the right-tail censoring indicator; and (3) those by eq. (12.17) using the right-tail censoring indicator. The Kappa distribution is fit to all three L-moment sets. The results are plotted on figure 12.2. The censored data are distinguished as solid circles. The two solid lines show very similar Kappa fits from uniquely different L-moment estimating functions—the reliability of the `lmomsRCmark()` function (for uncensored data) is demonstrated. (The reliability of `lmomsRCmark()` for censored data is evaluated in Section 12.5.) The Kappa distribution (dotted line) fit to the censored L-moments plots considerably to the left as anticipated. The censored values have dotted lines extending to the top of the plot from each in order to represent the interval in which the actual data value resides. This plotting style for censored data follows that of Helsel (2005, p. 52). ◀

For a demonstration of the generality of eq. (12.16) for $-\infty < X < \infty$ compared to the restriction by Wang and others (2010) that $X \geq 0$, readers are encouraged to repeat exam-

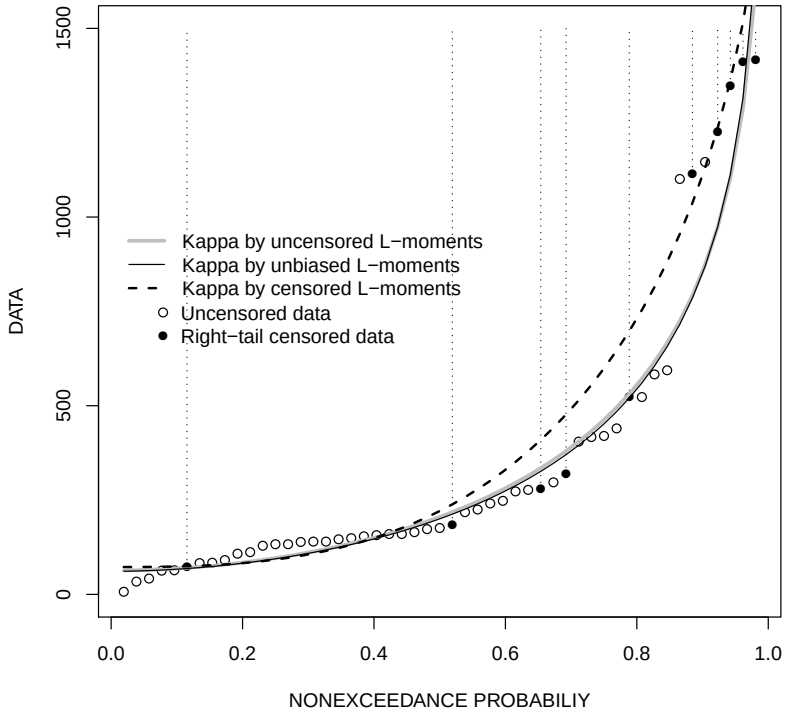

Figure 12.2. Comparison of three Kappa distribution fits to right-tail censored survival data (dotted lines extended from reported limit) from example 12–5

ple 12–5 with negated `Efron.data` by using the operation `Efron.data <-Efron[ix]*-1` (note the use of `*-1`). ◀

12.5 L-moments of Left-Tail Censored Data by Indicator Variable

Right-tail censoring is extremely common in survival analysis or failure analysis and the theory for accommodating such censoring is well developed (Helsel, 2005, p. 77) and select parts of the theory are discussed in Sections 12.2 and 12.4. Left-tail censoring is much more prevalent in the environmental and hydrologic sciences. Right-tail censoring theory is readily extended to left-tail censoring through the method of **flipping** the data to right-tail censored by subtraction of all data values from a constant M that must be greater than or equal to the maximum data values:

$$y_i = M - x_i \tag{12.21}$$

where y_i are the right-tail censored values and x_i are the original and left-tail censored. Statistical analysis, including computation L-moments, is made on the y_i. Location estimates of y_i such as mean, median, and quantiles "must be retransformed back" (Helsel, 2005, p. 65) by **back flipping** through subtraction of the constant M that was used to flip the data. Helsel (2005) and Lee (2009) provide arsenic concentration data in a data set called `Oahu`. These data are used here and results of L-moment computation compared, when possible, to parallel results from Helsel (2005, p. 65) or to the algorithms in the *NADA* package by Lee (2009) that have no connection to those in *lmomco*.

USING R ——— ——— USING R

Example 12–6 loads the `Oahu` dataset and provides a summary of the data values along with an indicator variable `AsCen`, which identifies, by a logical variable, those data that are left censored. For example, the second observation is <1.0, whereas the third observation is 1.7. The Kaplan-Meier nonparametric method is used in *NADA* to compute conventional summary statistics. These are computed and set into the `NADAfit` variable. The results show that the computed mean is about 0.949 milligrams per liter. The example ends with an output of selected quantiles of the data. The flipping of the data was performed automatically and retransformation (back flipping) is applied as necessary. These features of the *NADA* package will be more formally defined when L-moments are explained in a subsequent example (ex. 12–7).

12–6

```
library(NADA) # load the NADA package to get the Oahu dataset
data(Oahu) # load in arsenic data (left-tailed censored) in mg/L
print(as.list(Oahu)) # summarize these data
$As
 [1] 1.0 1.0 1.7 1.0 1.0 2.0 3.2 2.0 2.0 2.8
[11] 2.0 2.0 2.0 2.0 2.0 0.7 0.9 0.5 0.5 0.9
[21] 0.5 0.7 0.6 1.5

$AsCen
 [1]  TRUE  TRUE FALSE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE FALSE
[11]  TRUE  TRUE  TRUE  TRUE  TRUE FALSE FALSE FALSE FALSE  TRUE
[21] FALSE FALSE FALSE FALSE

# Now place data into shorthand variable names
A  <- Oahu$As    # the arsenic concentration
Ac <- Oahu$AsCen # logical as to left-tailed censored or not

# Kaplan-Meier nonparametric estimate of mean and standard dev.
```

```
NADAfit <- cenfit(Cen(A, Ac)) # cenfit and Cen from NADA package
print(NADAfit) # show the mean and standard deviation
         n      n.cen     median       mean         sd
24.0000000 13.0000000  0.7000000  0.9489583  0.8068068

quantile(NADAfit) # show some quantiles to be compared later
 5% 10% 25% 50% 75% 90% 95%
0.5 0.5 0.5 0.7 0.9 1.7 2.8
```

Example 12–7 continues to use the arsenic data in variable `A` and left-tail censoring indicator in variable `Ac`. The example opens with two uses of the `lmomsRCmark()` function to compute the L-moments by (1) ignoring the left-tail censoring and (2) using the left-tail censoring indicator. The example continues in parallel by fitting two Generalized Normal distributions by the `pargno()` function. For the remainder of the discussion, the censored L-moments in `lmr.cen` and the Generalized Normal fit in `lmomcofit.cen` are of interest. The purpose of showing how to ignore the censoring (not setting `rcmark` in `lmomsRCmark()`) is to provide a starting point for readers interested in further self study.[5] Although the flip was specified ($M = 5$ milligrams per liter), the flip used by the `lmomsRCmark()` function is explicitly extracted so as to hint that `lmomsRCmark()` also can automatically choose a flip for the user. The left-censored mean is set into `mean` and outputted. The result is 0.949, which precisely matches the left-censored mean computed by the independent algorithms of Lee (2009), which is shown in example 12–6.

12–7

```
lmr      <- lmomsRCmark(A, flip=5) # not fully used here
lmr.cen  <- lmomsRCmark(A, rcmark=Ac, flip=5) # used

lmomcofit      <- pargno(lmr) # fit GNO dist to lmr
lmomcofit.cen  <- pargno(lmr.cen) # fit the censored
# note, the L-moments and the GNO fit are RIGHT-TAIL CENSORED

# get the flip, in case not set in argument to lmomsRCmark()
flip <- lmr.cen$flip
mean <- flip - lmr.cen$lambdas[1] # back-flip
cat("# Mean is",mean,"\n") # this value matches earlier
# Mean is 0.9489583
```

The quantiles for the `Oahu` arsenic data were estimated nonparametrically in example 12–6. Example 12–8 estimates the quantiles via the distributional assumption and

[5] In other words, the author is trying to provide subtle details so as to show other twists to the distributional analysis that he thought would be neat to try for edification about censoring but decided not further explore in this dissertation. These data are left-censored; the remainder of the analysis is thus focused.

fit of the Generalized Normal made in example 12–7. The results in example 12–8 show that the estimated quantiles for the selected F (nonexceedance probability) are quite similar[6] to those in example 12–6. This example of Generalized Normal quantiles is explicitly chosen so as to show how the F must be mapped to S (exceedance probability, survival probability, or S, see page 27) then used in the QDF by the `qlmomco()` function and the result retransformed by back flipping (`-qlmomco(1-F)`, see Reflection Rule on page 36). The previous two examples thus show the mechanics of fitting a distribution in the usual fashion to the L-moments of left-censored data. The application to right-tail censored data is more straightforward because the $F \mapsto S$ mapping and back flipping is not required.

12–8

```
F <- c(0.05, 0.10, 0.25, 0.75, 0.90, 0.95) # !! nonexceedance !!
# carefully note the back-flipping and more subtle,
# the survival (exceedance) probability (1-F)
# back flip and F --> S transform
Q <- flip - qlmomco(1 - F, lmomcofit.cen)
# compare quantiles in example before last
print(round(Q, digits=1))
[1] 0.5 0.5 0.5 1.0 1.6 2.3
```

Finally to conclude this demonstration of distributional analysis of left-tail censoring, a graphical presentation of the results of the previous examples is highly informative. In example 12–9, another sequence of F is created. The `NADAfit` from example 12–6 is plotted (the stepped and solid line) and shown in figure 12.3. The empirical survival distribution is shown (after back flipping) in the thin line solid line, and the dashed lines represent 95-percent confidence bands. The example concludes by adding a thick solid line of the fitted Generalized Normal distribution in `lmomcofit.cen`. The author again emphasizes the two operations of $F \mapsto S$ and back flipping in the first argument to the `lines()` function call.[7] The thick line tracks through the stepped line, which confirms the implementation of the `lmomsRCmark()` function.

12–9

```
F <- seq(0,1, by=0.01) # !! nonexceedance probability !!
S <- 1 - F # exceedance probability
X <- flip - qlmomco(S, lmomcofit.cen) # back flip
```

[6] Equality is not anticipated, but if the fitted distribution is reasonable, then the computed nonparametric and estimated quantiles should be "similar."

[7] The author calls special attention to these two operations for treatment of left-tail censoring as more-than-cursory review of Helsel (2005) and Lee (2009) did not provide sufficient guidance and several iterations were needed before the figure looked and was correct.

```
#pdf("rcindicatorNADA.pdf")
plot(NADAfit) # creates plot with thin and dashed lines
lines(X, F, lwd=3)
#dev.off()
```

◀

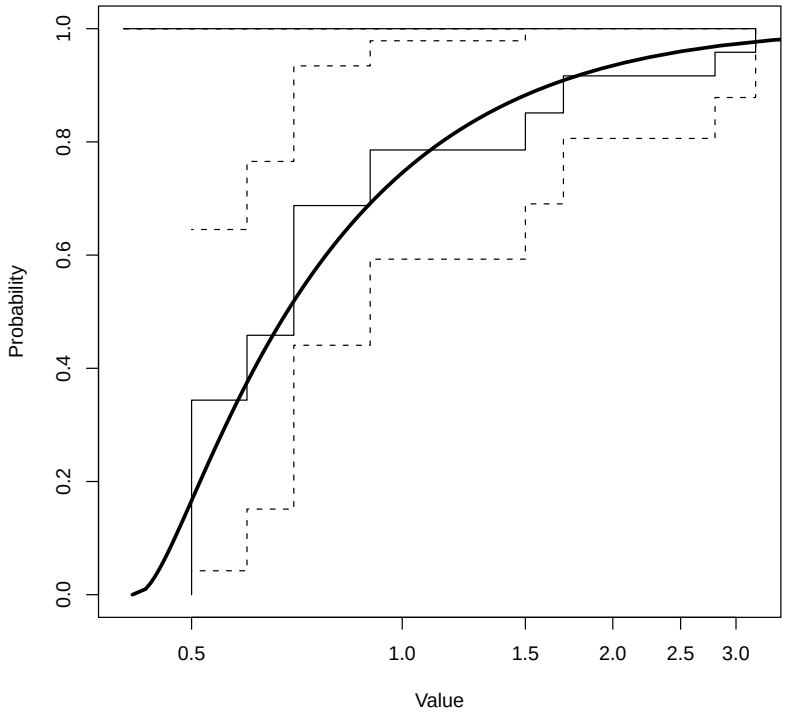

Figure 12.3. Empirical survival function (thin line and dashed 95-percent confidence bands) by Kaplan-Meier method from the *NADA* package to left-tail censored arsenic concentration in `Oahu` dataset compared to fit of Generalized Normal distribution (thick line) by flipped and right-censored L-moments by indicator variable from example 12–9

The flipping of the mean was shown in example 12–7. However, additional adjustments to the ensemble of L-moments are needed for higher-order distributional fit than two-parameter distributions. Odd-order L-moments, such as $\hat{\tau}_3$ and $\hat{\tau}_5$ require, a change of sign. Also $\hat{\tau}_2$ requires computation by the usual $\hat{\lambda}_2$ divided by the back-flipped mean. The *lmomco* package eases the hassle by providing the `fliplmoms()` function. This function receives the L-moments from `lmomsRCmark()`, queries the flip, and returns back-flipped L-moments. The process of selected quantile computation through a Generalized Nor-

mal distribution fit and the nonparametric method in the *NADA* package is shown in example 12–10.

12–10

```
# Create some data with **multiple detection limits**
# A left-tail censoring problem--flipping is required.
fakedat1 <- 10^rnorm(5000, mean=0.5, sd=0.25)
fake1.left.censor <- fakedat1 <  2
fakedat1[fake1.left.censor]   <- 2 # first limit

fakedat2 <- 10^rnorm(5000, mean=0.5, sd=0.25)
fake2.left.censor <- fakedat2 <  1
fakedat2[fake2.left.censor]   <- 1 # second limit

# combine the data sets
fakedat <- c(fakedat1, fakedat2)
fake.left.censor <- c(fake1.left.censor, fake2.left.censor)

lmr.flipped      <- lmomsRCmark(fakedat, flip=TRUE,
                                rcmark=fake.left.censor)
lmr.backflipped  <- fliplmoms(lmr.flipped)
F <- c(0.05, 0.10, 0.25, 0.50, 0.75, 0.90, 0.95)
library(NADA)
NADAfit <- cenfit(Cen(fakedat, fake.left.censor))
NADAqua <- quantile(NADAfit)
LMRqua  <- qlmomco(F, pargno(lmr.backflipped))
myquan  <- data.frame(F=F, NADA=NADAqua, LMRqua=LMRqua)
print(myquan)
       F     NADA   LMRqua
5%  0.05 1.220910 1.252923
10% 0.10 1.512826 1.530943
25% 0.25 2.147999 2.151226
50% 0.50 3.154967 3.157527
75% 0.75 4.670355 4.654464
90% 0.90 6.608210 6.616808
95% 0.95 8.168151 8.174190
```

The example simulated $n = 10{,}000$ values of a log-Normal distribution for which $5{,}000$ of the values are subject to a $T = 2$ left-tail censoring threshold and $5{,}000$ are subject to a $T = 1$ left-tail censoring threshold. The output shows close agreement between selected nonparametric quantiles of the *NADA* package to corresponding parametric quantiles of the fitted Generalized Normal distribution. The distribution is fit to the L-moments in the `lmr.backflipped` variable, which is derived from coupling the `lmomsRCmark()` and `fliplmoms()` functions for left-tail censored distributional analysis. Further, because

the `LMRqua` mimic those of the independent and non-L-moment algorithms of the *NADA* package; the reliability of the `lmomsRCmark()` function is demonstrated. ◀

12.6 Conditional Adjustment for Zero Values by Blipped-Distribution Modeling

Some data types can contain a substantial fraction of exactly zero values. For example, a time series of annual streamflow volume for a river located in a desert region might have relatively few nonzero values because for many years the river basin might receive little to no rainfall. In such circumstances, so-called **blipped distributions** (Gilchrist, 2000, p. 148) can prove useful as a means to implement conditional probability adjustment. Blipped distributions are a type of mixed distribution.

The use of blipped distributions can be used to accommodate zero values if particular attention to the fit of the lower tail is needed. For investigation of large quantiles (right tail) adjustments for zero values in the left tail can be of little consequence. For example, it might be perfectly reasonable to have a fitted distribution, which includes the zero values in the sample,[8] produce negative quantiles for a strictly positive phenomena for the lower quartile if the analyst requires quantile estimates at the $F \geq 0.90$ level (the opposite tail).

However, if preservation of correct sign (positive for the discussion here) in the left tail is needed, then **conditional probability** adjustment for zero values by blipped distributions is useful and is made by

$$F(x) = \begin{cases} 0 & \text{if } x \leq 0 \\ p + (1-p)G(x) & \text{if } x > 0 \end{cases} \tag{12.22}$$

and

$$x(F) = \begin{cases} 0 & \text{if } 0 \leq F \leq p \\ x_G[(F-p)/(1-p)] & \text{if } F > p \end{cases} \tag{12.23}$$

where p is the probability of a zero value and $G(x)$ is the CDF of the nonzero values, and $x_G(G)$ is the QDF of the nonzero values, and G is a nonexceedance probability. The distribution $x_G(G)$ does not have to be constrained to a lower bound of exactly zero, and

[8] That is, the zero values are left in the sample when the L-moments are computed.

the zero for x in eqs. (12.22) and (12.23) can be other constant lower bounds. For the discussion here, the focus is on a zero lower bound without a loss of generality. The value for p can be estimated from a sample as the ratio of the number of zero values to the total sample size.

USING R ——— ——— USING R

The conditional adjustment for zero values is demonstrated by the sequence of examples and concomitant discussion that follow. To begin, example 12–11 generates an $x > 0$ sample of size $n = 30$ and an arbitrary fraction of zero values `tp=0.20` (20 percent) to control the number of zero values synthetically added to the sample. The Generalized Pareto is selected for the example because the distribution has readily set lower bounds. The parameters of the true GPA(150, 700, 0.03) are set by `vec2par()` and a random sample `fake.dat.nz` of size `n` from this Generalized Pareto is drawn and sorted by `rlmomco()` and `sort()` functions, respectively. The `sapply()` function truncates the `fake.dat.nz` sample to positive values if any are present. However, for the example, the lower bound of the Generalized Pareto is `quagpa(0,tpga)` $= 150$, so explicit truncation is not needed as shown here. The last two steps in the final two lines: (1) generate the fake data in `fake.dat` (the complete sample) by adding `tp*n` zero values using the `rep()` function and (2) select the sample of greater-than-zero values.

12–11

```
n <- 30; tp <- 0.20
tgpa <- vec2par(c(150,700,0.03), type="gpa")
fake.dat.nz <- sort(rlmomco(n,tgpa))
fake.dat.nz <- sapply(fake.dat.nz,
         function(x) { if(x < 0) return(0); return(x) })
fake.dat    <- c(rep(0,tp*n), fake.dat.nz)
fake.dat.nz <- fake.dat.nz[fake.dat.nz > 0]
```

The distributional analysis of `fake.dat` continues in the following example 12–12 by (1) computing, for later plotting purposes, the Weibull plotting positions using `pp()`, and (2) computing the L-moments using the `lmoms()` function on the complete sample (variable `lmr`) and the sample values greater than zero ("`nz`", nonzero; variable `lmr.nz`).

12–12

```
PP  <- pp(fake.dat)
lmr    <- lmoms(fake.dat)
lmr.nz <- lmoms(fake.dat.nz)
```

The discussion continues in example 12–13 with the estimation of Generalized Pareto parameters from the sample L-moments using the `pargpa()` function for both the complete sample L-moments in `lmr` and the sample L-moments for the partial sample of values greater than zero in `lmr.nz`. Finally, the fraction of zero values for the sample is computed and set into the `p` variable.

12–13

```
PARgpa <- pargpa(lmr); PARgpa.nz <- pargpa(lmr.nz)
     p <- length(fake.dat[fake.dat <= 0])/length(fake.dat)
```

Based on the previous three examples (12–11 – 12–13), a visual representation of the blipped Generalized Pareto distribution is produced in example 12–14 and shown in figure 12.4. The plotting position values in `PP` of the complete sample provide values for F. These values also will be used for drawing QDFs of the distributions. The `quagpa()` function returns the Generalized Pareto quantiles, and the `z.par2qua()` function adheres to eq. (12.23) and performs as a blipped-distribution implementation of the `par2qua()` function. The `par2qua()` function internally dispatches the parameter lists `PARgpa` or `PARgpa.nz` to the `quagpa()` function to compute Generalized Pareto quantiles.

12–14

```
#pdf("zero1.pdf")
plot(qnorm(PP), fake.dat,
     xlab="STANDARD␣NORMAL␣DEVIATE", ylab="QUANTILE")
lines(qnorm(PP),quagpa(PP,PARgpa), lty=2) # dashed line
F <- PP # set nonexceedances to those in PP
Q <- z.par2qua(F,p,PARgpa.nz)
lines(qnorm(F),Q, lwd=2) # solid and thicker line
legend(-1.5, 1500, lty=c(2,1),
       c("GPA␣by␣complete␣sample",
         "GPA␣by␣blipped␣distribution"))
#dev.off()
```

As shown in figure 12.4, the use of blipped-distribution modeling of the Generalized Pareto provides an inherently better fit to the simulated data than the Generalized Pareto fit to the complete sample. The better fit in the left tail is obvious, but it is important to remark that the two fitted distributions are indistinguishable from each other in the right tail. Thus, if the analyst's interest is restricted to right-tail estimation, then little benefit would be gained using blipped-distribution modeling. In conclusion, blipped-distribution modeling builds a more complex distribution model, but the process of implementation is relatively straightforward and might be a useful tool for some circumstances. ◀

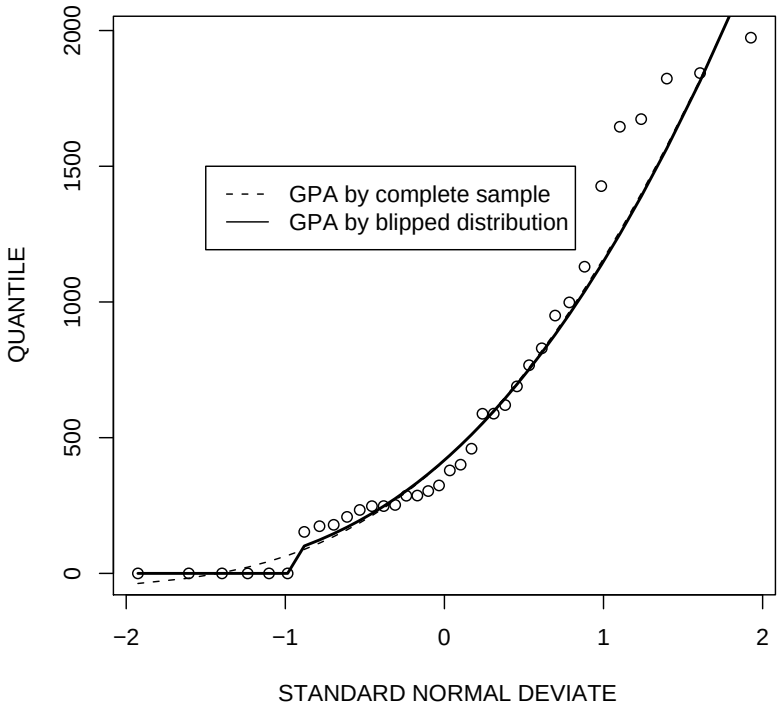

Figure 12.4. Conditional adjustment for zero values by blipped-distribution modeling of the Generalized Pareto from example 12–14

12.7 Exploration of Quantile Uncertainty

Assuming that one has at their disposal unbiased and generally small sampling variance estimators, the uncertainty in a prediction of individual quantiles from a given data set can be thought of as being produced by two components: sampling error and model-selection error, which is the error induced by having to choose or select a distribution that adequately represents the unknown parent distribution.

Assuming that the proper distribution has been chosen for a particular data set, sampling error is the uncertainty associated with sample size—more accurate quantile estimates are acquired as sample size increases. It requires little emphasis that smaller samples contain less information than larger samples. On the other hand, for a given sample size, model-selection error is the error associated with the choice of distribution. The next three sections provide informative explorations of quantile uncertainty by these two sources of error. These sections do not by any means attempt an exhaustive analysis of quantile uncertainty, but in the context of readily implemented statistical simulation, the

sections should provide readers with a look and feel of how much is unknown (uncertain) in distal tail estimates of distributions fit to samples. For purposes here, the distal tail begins at about a standard-normal quantile (deviate) or `qnorm(pnorm(1))` or $F \approx 0.84$.

12.7.1 Exploration of Sampling Error for a Single Data Set

An exploration of sampling error is initiated in example 12–15 by loading in the annual peak streamflow data for U.S. Geological Survey streamflow-gaging station 08151500 Llano River at Llano, Texas using the `data()` function. The streamflow data are placed into the `Qdat` variable, and the data are shown in figure 12.5.

12–15

```
#pdf("llano1.pdf")
data(USGSsta08151500peaks) # from lmomco package
Qdat <- USGSsta08151500peaks$Streamflow # a smaller variable name
plot(Qdat, xlab="YEAR␣NUMBER", ylab="PEAK␣STREAMFLOW,␣IN␣CFS")
#dev.off()
```

Next, in example 12–16, the data are sorted into the variable `Qs`, the Weibull plotting positions are computed by `pp()`, the sample L-moments are computed by `lmoms()`, and Wakeby distribution parameters by `parwak()` are placed into the variable `PARwak`. The `str()` function is used to report the L-moments and Wakeby parameters. The results are listed in table 12.1.

12–16

```
Qs <- sort(Qdat); PP <- pp(Qdat); lmr <- lmoms(Qdat)
PARwak <- parwak(lmr)
str(lmr); str(PARwak) # results shown in body of text
```

Table 12.1. L-moments of annual peak streamflows for Llano River at Llano, Texas (1940–2006) and Wakeby distribution parameters

$\hat{\lambda}_1$	$\hat{\lambda}_2$	$\hat{\tau}_3$	$\hat{\tau}_4$	$\hat{\tau}_5$	ξ	α	γ	β	δ
51,160	28,900	0.3925	0.1701	0.0962	972.0	−64,170	1.616	81,800	−0.09491

A by-now-familiar plot of the empirical distribution and a fitted Wakeby distribution is generated by example 12–17 and shown in figure 12.6. For this particular data set, the Wakeby distribution provides a generally acceptable fit to the empirical distribution.

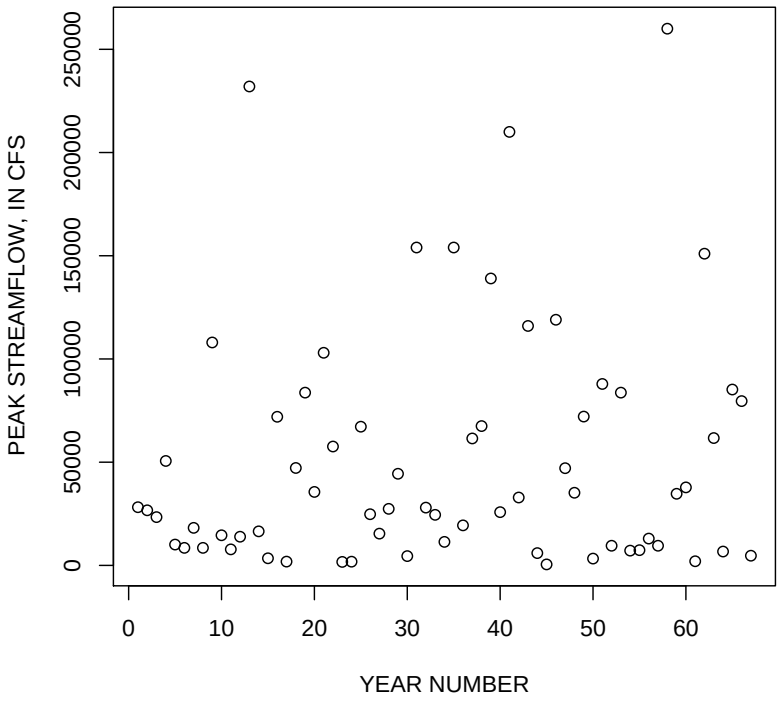

Figure 12.5. Time series of annual peak streamflows for Llano River at Llano, Texas (1940–2006) from example 12–15

The data have a nearly 2.5-order of magnitude range, yet the distribution is fit in the untransformed units of the data—logarithmic transformation is not used in the analysis.

12–17

```
#pdf("llano2.pdf")
plot(qnorm(PP),log10(Qs),
     xlab="STANDARD␣NORMAL␣DEVIATE",
     ylab="LOG10␣STREAMFLOW,␣IN␣FT^3/S")
lines(qnorm(PP),log10(quawak(PP,PARwak)), lwd=3, lty=1)
legend(-2,5.5, c("Wakeby␣by␣L-moments"),
       lwd=c(3), lty=c(1), box.lty=0, bty="n")
#dev.off()
```

◀

The `gen.freq.curves()` function is a high-level function that drives simulation by the `genci()` function (called internally) for a specified sample size and a given parent distribution. The distribution is specified by an *lmomco* parameter list (see page 163 and ex. 7–1). The `gen.freq.curves()` function collects intermediate results and provides

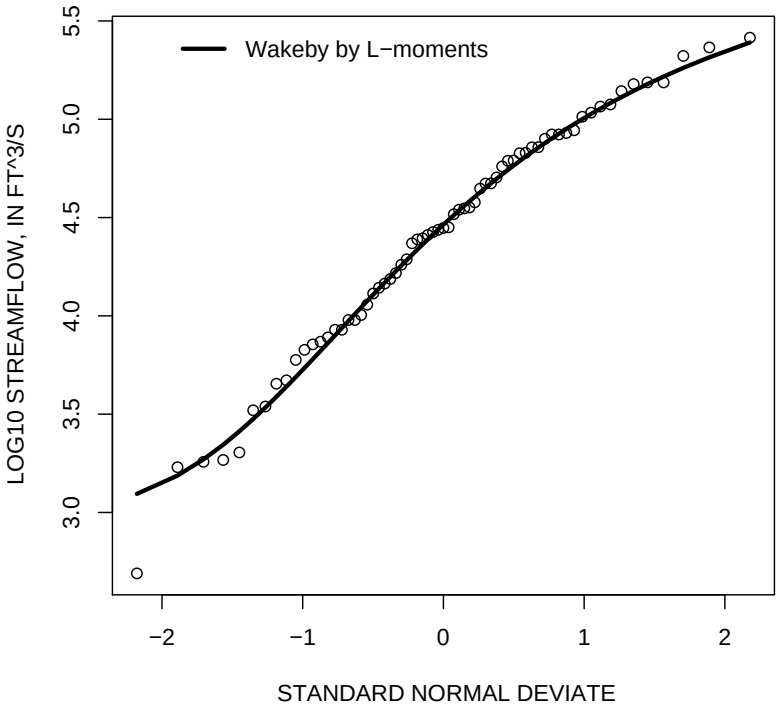

Figure 12.6. Empirical distribution and fitted Wakeby distribution to annual peak streamflows for Llano River at Llano, Texas from example 12–17

options for graphical visualization. The number of simulations and other features that generally control graphical output are set by named arguments.

Example 12–18 demonstrates the `gen.freq.curves()` function using a sample size of `n=67` for 100 distinct simulations (`nsim=100`) from the Wakeby parent. With each drawing, the sample L-moments and estimated Wakeby parameters of the simulated sample are computed and each resulting ith Wakeby for $1 \leq i \leq$ `nsim` is depicted on the plot in figure 12.7. The `nonexceeds()` function is used to generate a convenient vector of F values for drawing of the Wakeby parent by the `quawak()` function. The example ends by superimposing the true parent (dashed line) on the 100 simulated Wakeby distributions.

12–18

```
F <- nonexceeds()
n <- length(Qdat) # 67 years of record

#pdf("llano3.pdf", version="1.4")
gen.freq.curves(n, PARwak, nsim=100,
                asprob=TRUE, col=rgb(0,0,0,0.08))

lines(qnorm(F), quawak(F,PARwak), lty=2, lwd=3)
```

```
legend(-2.5,350000, c("Wakeby␣by␣L-moments"),
       lwd=c(3), lty=c(2),
       box.lty=0, bty="n")
#dev.off()
```

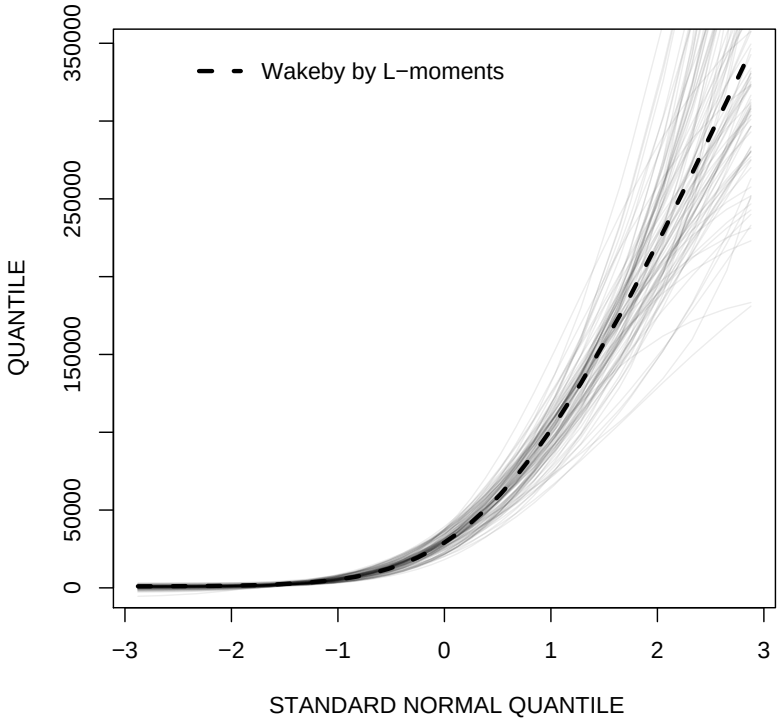

Figure 12.7. Study of 100 simulations of sample size of $n = 67$ for indicated Wakeby parent from example 12–18

Example 12–18 makes use of semi-transparency, which can be provided by the portable document format (PDF) device `pdf()`. The transparency is accessed through specification of a fourth parameter to the `rgb()` color function. The "fuzziness" or grayness of the simulated distributions in figure 12.8 is a graphical depiction of sampling error. ◀

For a demonstration of the influence of sample size, it is informative to repeat the example 12–18 in 12–19 for a sample size of $n = 20$ (fig. 12.8) and then again in example 12–20 for $n = 200$ (fig. 12.9). The dramatic increase in variability at a given F in the distribution of the simulated distributions between figures 12.8 and 12.9 exist because of the different sample sizes. In fact at $n = 20$, radically different curvatures of a few simulated distributions compared to the curvature of the Wakeby parent distribution are visible. Some distributions have upper limits much less, and conversely much larger, than the parent

distribution. It must be remarked that the algorithm used to fit the Wakeby includes three solution styles: (1) the ξ parameter is estimated, (2) the ξ parameter is set to $\xi = 0$, or (3) a Generalized Pareto distribution is fit instead if either of the other two solutions are not viable.

12–19

```
#pdf("llano4.pdf", version="1.4")
gen.freq.curves(20, PARwak, nsim=100,
                asprob=TRUE, col=rgb(0,0,0,0.08))
lines(qnorm(F), quawak(F,PARwak), lty=2, lwd=3)
legend(-2.5,350000, c("Wakeby␣by␣L-moments"),
       lwd=c(3), lty=c(2), box.lty=0, bty="n")
#dev.off()
```

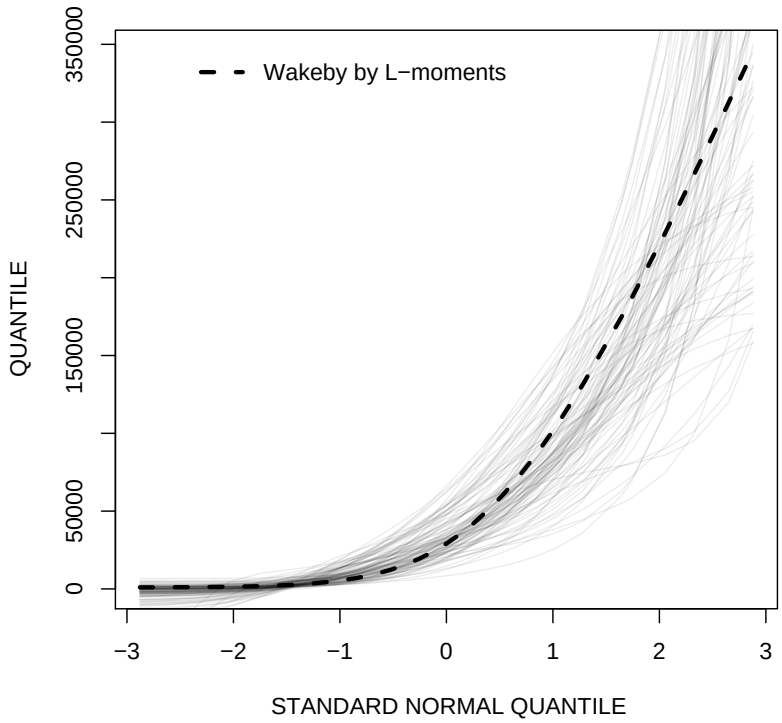

Figure 12.8. Study of 100 simulations of sample size $n = 20$ for indicated Wakeby parent from example 12–19

12–20

```
#pdf("llano5.pdf", version="1.4")
gen.freq.curves(200, PARwak, nsim=100,
                asprob=TRUE, col=rgb(0,0,0,0.08))
lines(qnorm(F), quawak(F,PARwak), lty=2, lwd=3)
legend(-2.5, 350000, c("Wakeby␣by␣L-moments"),
```

```
       lwd=c(3), lty=c(2), box.lty=0, bty="n")
#dev.off()
```

◀

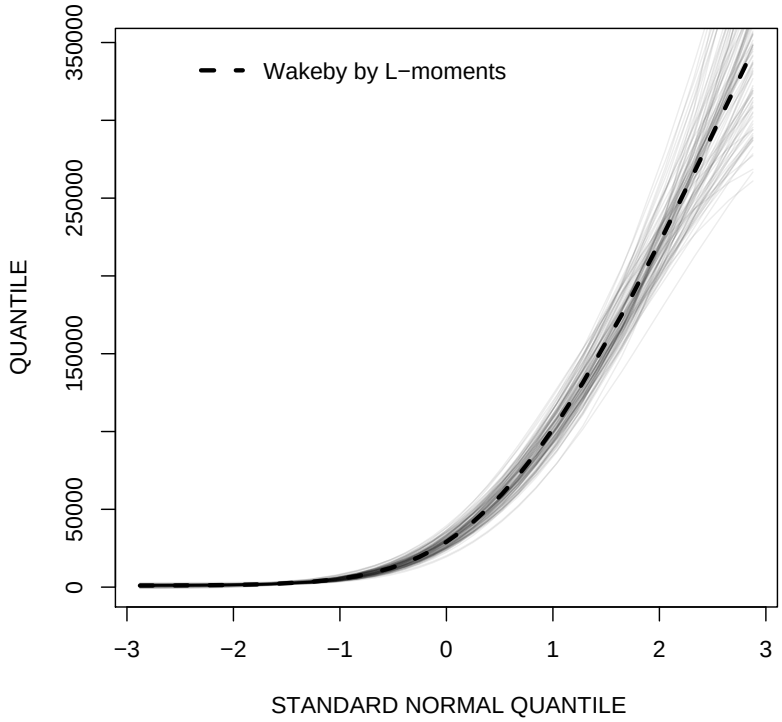

Figure 12.9. Study of 100 simulations of sample size $n = 200$ for indicated Wakeby parent from example 12–20

For examples 12–18 – 12–20, the emphasis is on visualization of sampling error as a function of sample size. It might be more useful to acquire an actual metric of sampling error for a range of sample sizes for a given quantile. These quantile sampling error metrics, be they $\hat{\sigma}^2$ (sample variance), $\hat{\sigma}$ (standard deviation), $\hat{\lambda}_2$ (L-scale), or measures of relative variability such as $\hat{CV}$ and $\hat{\tau}_2$, could be used to determine a sample size sufficiently large to meet some specified tolerance.

Governed by the theme of this dissertation, the L-moments of the distribution of a selected quantile are considered. Example 12–21 computes $\hat{\tau}_2$ of the $F = 0.99$ quantile from the fitted Wakeby distribution in variable `PARwak`. The output shows that $\hat{\tau}_2 = 0.116$ (far right entry in 12–21) for the $x_{0.99}$ quantile or $\hat{\lambda}_2 = 30{,}900$ cubic feet per second.

12–21

```
genci(PARwak,n=67,F=0.99,nsim=100) # nsim should really be larger
  nonexceed_prob    lower     true    upper   lscale       lcv
1           0.99 183377.9 266462.1 363678.4 30921.80 0.1160457
```

The $\hat{\tau}_2 = 0.116$ for $x_{0.99}$ will be used again in the next section.

12.7.2 Exploration of Sampling Error for a Regional Data Set

Approximately 670 U.S. Geological Survey streamflow-gaging stations provide 8 or more years of annual peak streamflow data for undeveloped watersheds in a study area encompassing Texas, eastern New Mexico, and part of the bordering areas near Texas for Oklahoma and Louisiana. The $\hat{\lambda}_1$, $\hat{\lambda}_2$, $\hat{\tau}_3$, $\hat{\tau}_4$, and $\hat{\tau}_5$ for each station were computed by Asquith and Roussel (2009). By taking $\hat{\lambda}_1 = 1$ and $\hat{\lambda}_2 = \hat{\tau}_2$, a dimensionless Wakeby distribution or regional growth curve for the study area can be estimated from `weighted.mean()` values for the sample L-moments. The number of years of record or data for each station constitute weight factors as also done in example 11–7 within a different context. Although intermediate computations are not shown, the regional L-moments and corresponding parameters of the Wakeby are listed in table 12.2 in which the listed parameter values are computed in example 12–22.

12–22

```
L <- vec2lmom(c(1,0.505,0.394,0.250,0.159))
W <- parwak(L) # compute Wakeby parameters
```

Table 12.2. Regional L-moments and equivalent Wakeby parameters for dimensionless distribution of annual peak streamflow in Texas

$\hat{\lambda}_1$	$\hat{\tau}_2$	$\hat{\tau}_3$	$\hat{\tau}_4$	$\hat{\tau}_5$	ξ	α	β	γ	δ
1	0.505	0.394	0.250	0.159	−0.0266	1.100	6.105	0.692	0.206

Continuing in example 12–23 and for purposes of illustration, the variability of quantile estimates at the $F = 0.50$ and 0.99 levels are of interest and are set into `F`. A sequence of sample sizes from 10 to 100 by increments of 2 is set into `needed.n`. The `LCV` data frame is created by the `data.frame()` function, and the data frame will be used to hold the sample size and $\hat{\tau}_2$ for each of the F values. The core function of the example is to iterate through each sample size, call the `genci()` function, and retrieve the results into `LCV`.

12–23

```
F <- c(0.50,0.99); needed.n <- seq(10,14, by=2); nsim <- 20
LCV <- data.frame(SampleSize = vector(mode="numeric"),
                      Q50lcv = vector(mode="numeric"),
                      Q99lcv = vector(mode="numeric"))
for(n in needed.n) {
   cat(c("SAMPLESIZE=",n,
         "␣␣SIMULATIONSIZE=",nsim,"\n"))
   CI <- genci(W, n, F=F, nsim=nsim)
   LCV[n,] <- c(n, CI$lcv[1:2])
}
```

The primary purpose of the `genci()` function is to estimate, for a specified distribution and sample size, the lower and upper limits of a specified **confidence interval** for specific quantile values using simulation. These computations are shown in example 12–24.

12–24

```
genci(W,n=16,nsim=200,F=(16/(16+1)))
  nonexceed_prob    lower     true   upper    lscale       lcv
1      0.9411765 1.039137 2.816574 3.33673 0.3940406 0.1399006
```

In the example and by default, the `genci()` function also returns the $\hat{\lambda}_2$ and $\hat{\tau}_2$ values. The quantile values are specified by a vector of F values. Although in the example, only a single $F = 0.941$ is used. The parameters of the parent distribution (the Wakeby distribution in this case) are provided as the first argument. The `genci()` function is a wrapper on `qua2ci()` function, which is not shown in the example. The returned contents of the `genci()` function are shown in the last line of example 12–24.

12–25

```
my.min <- min(LCV$Q50lcv, na.rm=TRUE)
my.max <- max(LCV$Q99lcv, na.rm=TRUE)
#pdf("regwak_nsim20.pdf")
plot(LCV$SampleSize, LCV$Q99lcv, ylim = c(my.min, my.max),
     xlab = "SAMPLE␣SIZE",
     ylab = "L-CV␣AT␣INDICATED␣QUANTILE")
points(LCV$SampleSize, LCV$Q50lcv,pch=16)
#dev.off()
```

12.7.3 Exploration of Model-Selection Error

If a single distribution is fit to some data, then model-selection error at a given $x(F)^{\text{model}}$ is a bias computed as the difference between the selected distribution and the parent. How-

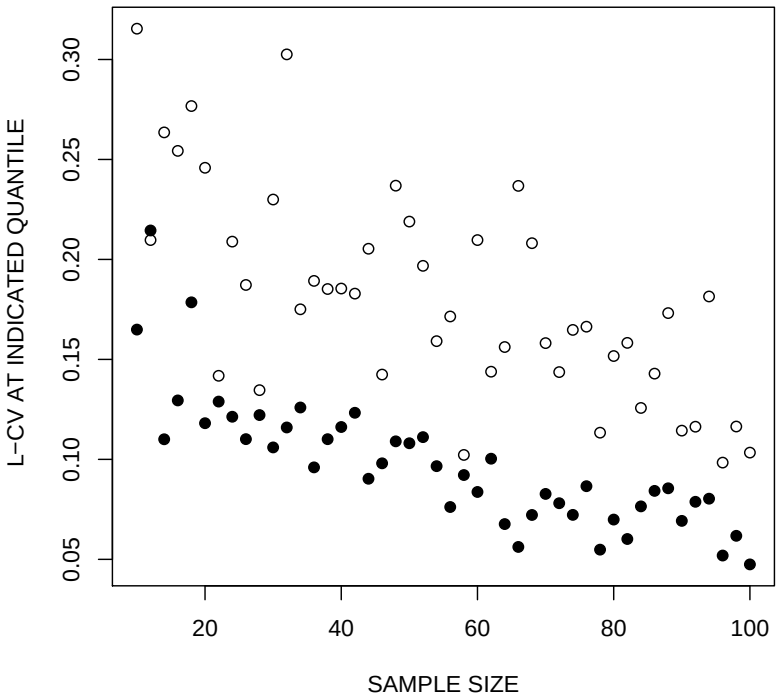

Figure 12.10. Comparison of simulated τ_2 values for 50th (open circles) and 90th (closed circles) percentiles of regional Wakeby parent using 20 simulations for indicated sample size from example 12–25

ever, this bias can be quite difficult to assess in most circumstances because the parent distribution is unknown. If multiple distributions, which could each arguably be appropriate, are fit, then it is possible to compute measures of variability from the multiple fits at each $x(F)$ of interest. This variability subsequently can be compared to the sampling variability. The variability of $x(0.99)$ is the subject of this section.

Wallis (1988, pp. 304–305) refers to the topic of this section "differences in $x(F)$ as a function of choice of distribution." Wallis proceeds to summarize a study of the $T = 10^6$ year annual maximum wind speed event for Corpus Chrisi, Texas in which the Extreme Value Type I (EV I, Gumbel in this dissertation) and Extreme Value Type II (EV II, Fréchet, which is special case of Generalized Extreme Value in this dissertation) distributions are each used. Wallis states "The EV I estimate for $T = 10^6$ event equals the commonly observed maximum wind speed for large hurricanes, while the comparable value for the EV II distribution is almost half the velocity needed to escape from the Earth's field of gravity!" (The explanation point is Wallis'.) Wallis concludes that "neither estimate appears particularly reasonable," and other analysis could be done. The author of this dissertation cites this

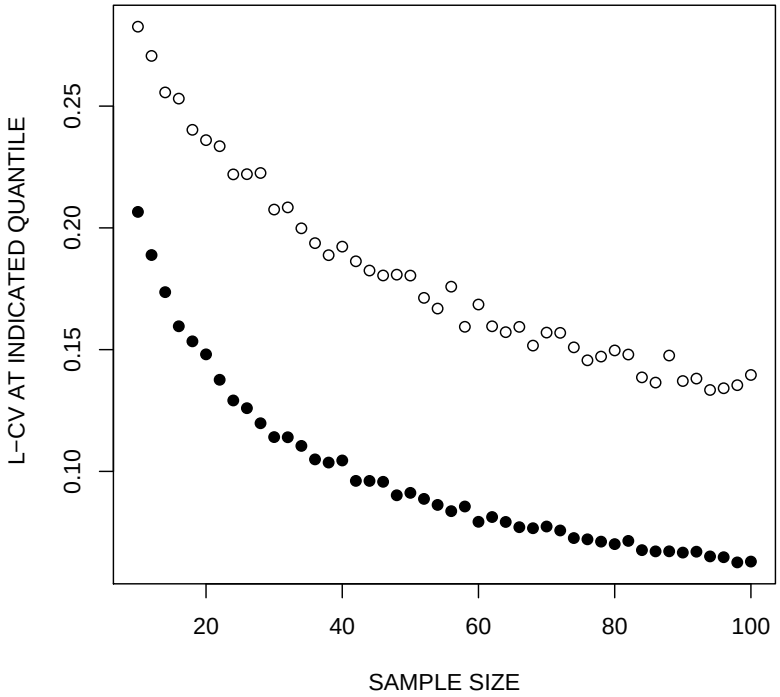

Figure 12.11. Comparison of simulated τ_2 values for 50th (open circles) and 90th (closed circles) percentiles of regional Wakeby parent using 2,000 simulations for indicated sample size from repeating of examples 12–23 and 12–25 using `nsim=2000`

example as a case where two reasonably chosen distributions yield radically divergent far-tail quantile estimates.

In example 12–26, an exploration of model-selection error is made using the five three-parameter distributions supported by *lmomco*. The data chosen are the annual peak streamflow for for U.S. Geological Survey streamflow-gaging station 08151500 Llano River at Llano, Texas are computed. These data are shown in figures 12.5 and 12.6.

Example 12–26 begins by the computation of the sample L-moments, computation of Weibull plotting positions, and creation of a list of distribution abbreviations in variable `dist`. The example continues with creation of a `grv()` function to compute **Gumbel Reduced Variates** and setting of a familiar sequence of F. The Gumbel Reduced Variates are used to dilate the horizontal axis. A five-element vector `QsG` is initialized for the five distributions to store estimates of the 99th-percentile annual peak streamflow.

12–26

```
data(USGSsta08151500peaks) # from lmomco package
Qdat  <- USGSsta08151500peaks$Streamflow # a smaller name
```

```
lmr   <- lmoms(Qdat); weibullpp <- pp(Qdat)
dist  <- c("gev", "gno", "glo", "gpa", "pe3")

G   <- 0.99 # for dotted vertical line in a plot
grv <- function(x) return(-log(-log(x))) # Gumbel RV
F   <- nonexceeds()
QsG <- vector(mode="numeric", length=length(dist))
#pdf("modelselection.pdf")
plot(grv(weibullpp), sort(Qdat), log="y",
     xlim=c(0,5), ylim=c(1e4,4e5),
     xlab="GUMBEL␣REDUCED␣VARIATE,␣-log(-log(F))",
     ylab="STREAMFLOW,␣CFS")
for(i in 1:length(dist)) {
  ifelse(dist[i] == "gpa", lty <- 1, lty <- 2)
  QDF <- qlmomco(F, lmom2par(lmr, type=dist[i]))
  lines(grv(F), QDF, lty=lty)
  QsG[i] <- QDF[F == G]
}
lines(c(grv(G),grv(G)), c(1e4,4e5), lty=3)
#dev.off()
```

The example continues by plotting the data and the five fitted distributions as shown in figure 12.12. The fitted distributions are shown as dashed lines and a solid line (the Generalized Pareto). The Generalized Pareto distribution is plotted differently because the L-moment ratio diagram (but not shown here), but which can be created by the nested functions `plotlmrdia(lmrdia())` and `points(lmr$ratios[3], lmr $ratios[4])`, shows that $\{\hat{\tau}_3, \hat{\tau}_4\}$ from the L-moment ratios in variable `lmr` plot closest to the Generalized Pareto trajectory. This distribution thus is specifically singled out for the plot. Finally, a dotted vertical line is drawn at $F = 0.99$. It is the variation in the solid and dashed lines at this vertical that will be used to express model-selection error.

The values stored in the `QsG` variable created in example 12–26 represent five separate estimates of $x(0.99)$ from five different distributions. The basic summary statistics of `QsG` are shown in example 12–27 along with the standard deviation. In the example, the L-moment estimate of σ by multiplication of $\sqrt{\pi}$ also is shown. These two estimates suggest that the standard error of the $x(0.99)$ estimate, which is attributed to choice of an unknown, but three-parameter, distribution model, is about $17{,}000$ cubic feet per second. The example ends with the reporting of ${\tau_2}^{\text{model}} = 0.034$, which is an expression of the relative error.

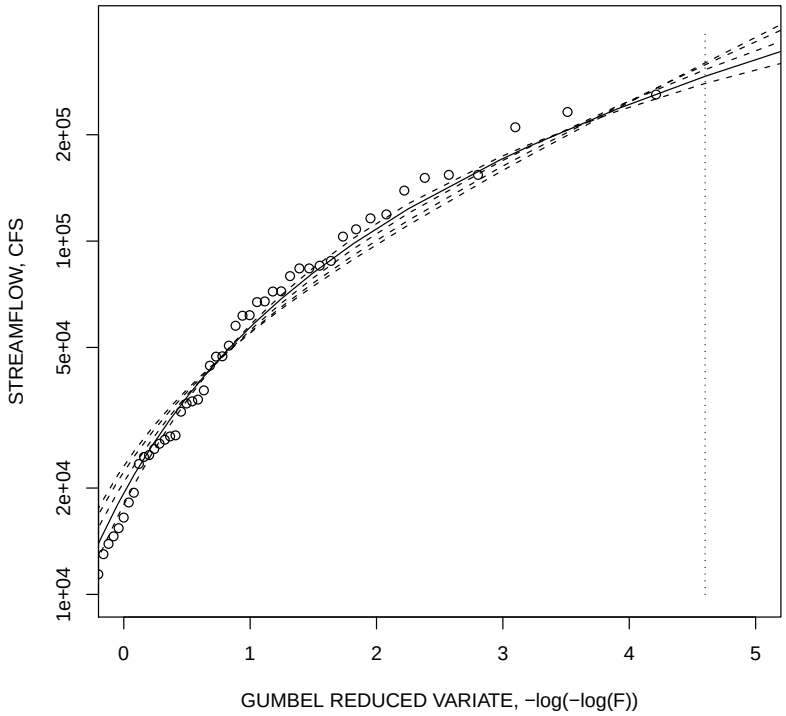

Figure 12.12. Empirical distribution and five fitted distributions to annual peak streamflows for Llano River at Llano, Texas from example 12–26. The dotted vertical line is drawn at $F = 0.99$ (the 100-year recurrence interval).

12–27

```
print(summary(Qs))
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 279400  292700  305600  302500  315500  319400

print(sd(Qs)) # standard deviation
[1] 16548.20

QsGlmr <- lmoms(QsG)
print(sqrt(pi)*QsGlmr$lambdas[2]) # sigma by L-moments
[1] 18230.22

print(QsGlmr$ratios[2]) # L-CV
[1] 0.03399913
```

◀

Finally, in example 12–21, the relative sampling variability of the five-parameter Wakeby distribution, based on limited simulations, is reported as ${\tau_2}^{\text{sampling}} = 0.116$. This can be compared and combined with ${\tau_2}^{\text{model}} = 0.034$. Comparing the two, it is seen

that the relative variation because of sampling in the $x(0.99)$ estimate is about 3.4 times larger than the relative variation from deciding amongst the 5 three-parameter distributions. Combining the two variations, the relative variation in the $x(0.99)$ estimate is about $\tau_2 = \sqrt{(0.116)^2 + (0.034)^2} = 0.121$.

If the best estimate of the analyst doing magnitude and frequency work for this river, including the distributional work in this section and Section 12.7.1 along with other engineering analyses (perhaps interpretations of a rainfall and runoff model), is say $300{,}000$ cubic feet per second, then the estimated uncertainty in the context of the more familiar standard deviation is $\sigma = 300{,}000 \times 0.121 \times \sqrt{\pi} = 64{,}000$ cubic feet per second. The analyst then might write the $x(0.99)$ estimate as $300{,}000 \pm 64{,}000$ cubic feet per second.

12.8 Product Moments versus L-moments for Estimation of Pearson Type III Distributions

The three-parameter Pearson Type III distribution is a popular distribution for analysis of hydrologic data (U.S. Water Resources Council, 1981). Ding and Yang (1988) consider the Pearson Type III distribution in the context of probability-weighted moments. The distribution is attractive for making product moment and L-moment comparisons because the distribution is skewed and the product moments are the parameters of the distribution. In turn, the first three L-moments can be represented as product moment equivalents through the Pearson Type III distribution. The comparison of product moments and L-moment computations, thus, is readily made using the methods of product moments and L-moments. Readers are asked to recall that the method of product moments uses conventional sample mean, standard deviation, and skew as direct estimates for Pearson Type III parameters. Whereas, the method of L-moments uses the sample mean, L-scale, L-skew, and numerical methods to estimate the Pearson Type III parameters.

After additional commentary (additional commentary for this dissertation) concerning logarithmic transformation, this section describes a simulation study (Section 12.8.2) of a Pearson Type III having zero mean and ranges of standard deviation (0.1 to 10) and skew (-5 to $+5$). These ranges provide for a comparison of product moments and L-moments for Pearson Type III parameter estimation. For the simulations, sample sizes of 10, 20, 40, 60, and 100 are used. The results show that L-moments outperform, in a Pearson Type III context, the product moments in terms of bias.

An example application of product moment and L-moment parameter estimation for Pearson Type III and log-Pearson Type III for a small sample for a hypothetical right-tail heavy distribution also is provided (Section 12.8.3). Additional joint simulations of right-tail heavy Pearson Type III and log-Pearson Type III distributions show preference to a Pearson Type III fit by L-moments whether the parent distribution is Pearson Type III or log-Pearson Type III.

12.8.1 Logarithmic Transformation

The limitations of the method of product moments for considerably skewed distributions are substantial. As a result, $\log_{10}$ transformations of the data often are recommended to reduce skewness and increase data symmetry, and as a result, increase the utility of the method of moments. In a widely distributed book, Stedinger and others (1993, chap. 18, p. 5) succinctly comment on logarithmic transformation and state

> A logarithmic transformation is an effective vehicle for normalizing values which vary by orders of magnitude, and also for keeping occasionally large values [high outliers] from dominating the calculation of product-moment estimators. However, the danger with use of logarithmic transformations is that unusually small observations [low outliers] are given greatly increased weight. This is a concern if it is the large events that are of interest, small values are poorly measured, small values reflect rounding, or small values are reported as zero if they fall below some threshold.

When the Pearson Type III is fit to the product moments of $\log_{10}$ transformed data, it is referred to as the log-Pearson Type III. In particular, for many applied hydrologists and engineers (at least in the United States), the log-Pearson Type III is understandably considered as the default (often only) distribution to use for the skewness of hydrologic data. A contributing reason is the log-Pearson Type III role in many analyses of annual peak streamflow (U.S. Water Resources Council, 1981); the techniques therein are well known within the discipline of flood magnitude analysis. Further, prepackaged "frequency" supporting software, such as PeakFQ by U.S. Geological Survey (2007a) or SWSTAT by U.S. Geological Survey (2007b), emphasize log-Pearson Type III and product moment usage. A common practice, therefore, is for the analyst to logarithmically transform ($\log_{10}$ transform) the phenomena being investigated and fit the log-Pearson Type III using product moments.

When $\log_{10}$ transformation is used, analysis of data containing zero or negative values becomes more complicated. A more philosophical drawback is that analysis is based in $\log_{10}$ space, but real-world implementation of statistical results, such as flood volume or rainfall magnitude, is needed in linear space. The philosophical topic of transformation is briefly discussed by Vogel and Fennessey (1993, p. 1750) in the context of goodness-of-fit of probability distributions and L-moment ratio diagrams. Their discussion can be summarized by this author (Asquith), "why transform if transformation is not needed or does logarithmic transformation obscure otherwise salient features of the data?" A benefit of $\log_{10}$ transformation, however, is that logarithmic transformation can simplify analysis of strictly positive data as untransformation of a distribution of logarithms does not produce negative values. However, the blipped distribution modeling described in Section 12.6 could mitigate for negative values as well as zero values.

12.8.2 Simulation Study of Pearson Type III Parameter Estimation

A comparison of product and L-moment parameter estimation for a wide range of product moment σ (standard deviation) and γ (skew) for a Pearson Type III parent with $\mu = 0$ (mean), without a loss of generality, is made by simulation in this section. The simulation study, for which core logic is seen in example 12–28, considered 21 population values for σ ($\sigma = 0.1, 0.5, 1.0, \ldots, 10$) and 21 population values for γ ($\gamma = -5.0, -4.5, \ldots, 4.5, 5.0$). For each unique pairing of σ and γ and $\mu = 0$, sample sizes n ($n = 10, 20, 40, 100$) were randomly drawn from the Pearson Type III parent. For each simulated sample, the sample product moments ($\hat{\mu}_{pm}, \hat{\sigma}_{pm}, \hat{\gamma}_{pm}$) and sample L-moments ($\hat{\lambda}_1, \hat{\lambda}_2, \hat{\lambda}_3$) were computed. The sample L-moments then were converted to Pearson Type III parameters to form L-moment-based "product moments" ($\hat{\mu}_\lambda, \hat{\sigma}_\lambda, \hat{\gamma}_\lambda$). The computation of $\hat{\mu}_{pm}, \hat{\sigma}_{pm}, \hat{\gamma}_{pm}$ and $\hat{\mu}_\lambda, \hat{\sigma}_\lambda, \hat{\gamma}_\lambda$ for each σ and γ pairing was repeated $10{,}000$ times. The mean value for the $10{,}000$ values of each of the six statistics were computed. These mean values provide the coordinates necessary to render the arrows in figure 12.13–12.16.[9]

12–28

```
# mu=0, sigma=10, gamma=5
pe3pars <- vec2par(c(0,10,5), type="pe3")
n       <- 10    # sample size
```

[9] These figures were rendered in METAPOST using a custom software program in Perl written by the author.

```
nsim    <- 10000 # number of simulations
pm.est.sd <- vector(mode = "numeric") # pm estimated sigma
pm.est.g  <- vector(mode = "numeric") # pm estimated gamma
lm.est.sd <- vector(mode = "numeric") # L-moment estimated sigma
lm.est.g  <- vector(mode = "numeric") # L-moment estimated gamma
for(i in seq(1,nsim)) { # loop the number of simulations
  Q   <- quape3(runif(n),pe3pars) # draw n samples from parent
  lmr <- lmoms(Q) # compute L-moments
  pmr <- pmoms(Q) # compute product moments
  estpars <- parpe3(lmr,checklmom=FALSE) # est. PE3 parameters
  # store the four sample values into the vectors
  pm.est.sd[i] <- pmr$sd
  pm.est.g[i]  <- pmr$skew
  lm.est.sd[i] <- estpars$para[2]
  lm.est.g[i]  <- estpars$para[3]
}
# compute sample means of the sample statistics
pm.sd <- mean(pm.est.sd); pm.g <- mean(pm.est.g)
lm.sd <- mean(lm.est.sd); lm.g <- mean(lm.est.g)
# display the results
cat(c("SD = ",10,"  pmSD = ",pm.sd,"  lmSD = ",lm.sd,"\n"))
cat(c(" G = ",5, "   pmG = ",pm.g, "   lmG = ",lm.g,"\n"))

# The author's computer, after rounding, produces:
SD = 10   pm.estSD = 7.64   lm.estSD = 12.27
 G = 5     pm.estG = 2.18    lm.estG = 6.03
```

Figures 12.13–12.16 depict the simulated σ and γ parameter space, and each figure represents a different sample size. The graph on the left of each figure represents the results using product moments and the right graph represents the results using L-moments. The arrows lead from the population values to the means of the $10{,}000$ sample statistics. The arrow lengths (arrow head plus shaft) represent bias; long arrows represent large bias and short line segments represent small bias. For a given estimation technique (product moment or L-moment), the arrow lengths systematically shorten as sample size increases. If the length is less than the arrow-head length, then the arrow head is not shown; for this dissertation then the condition of "unbiased" is represented by the absence of the arrow head.

Drawing attention to the left graph of figure 12.13 for $n = 10$, and by generality, the product moment (left) graphs in figures 12.14–12.16, the arrows are oriented in the σ^- direction (horizontal to the left) and increasingly angled toward the left as σ increases. The left graphs show that σ is reliably estimated for small σ and γ, but σ is substantially underestimated by the product moments as σ increases. The arrows are symmetrically oriented

toward zero in the γ direction (vertical), which shows that γ is systematically underestimated by the product moments. The opposite situation is shown for the L-moment case. The arrow lengths for the L-moments indicate that the magnitude of σ and γ are overestimated. However, the arrow lengths for the L-moments generally are much shorter, which demonstrates superior small sample performance of the sample L-moments.

For $n = 10$ (fig. 12.13), use of either product moments or L-moments can be questioned, except in the near symmetrical situation ($\gamma = 0$) for product moments and approximately $\sigma \leq 8$ and $|\gamma| < 3$ for the L-moments. By $n = 20$ (fig. 12.14), the L-moments provide essentially unbiased estimation of Pearson Type III parameters for approximately $\sigma \leq 9$ and $|\gamma| < 4$. Only in the central region ($|\gamma|$ close to 0) do the product moments perform well. By $n = 40$ (fig. 12.15), the sample L-moments are effectively unbiased for much of the parameter space. However, the sample product moments continue to be substantially biased for approximately $|\gamma| > 2$. By $n = 100$ (fig. 12.16), which often would be considered a large sample size in many hydrologic data sets, the product moments continue to show substantial bias.

Several observations are made for the σ and γ parameter space. L-moments for Pearson Type III parameter estimation appear superior to product moments. The author acknowledges that this conclusion is not a particularly new contribution in the sense that L-moments already are documented to have more robust sampling properties than the product moments. However, the simulations, in particular, dramatically demonstrate that the skewness of the data in a Pearson Type III context is more reliably estimated using L-moments. As a general judgement, sample sizes of at least 40 (60 would be better) are sufficient for reliable estimation of the variability and skewness for Pearson Type III-distributed data. The sample size judgement compares favorably with Guttman (1994) who concluded that the measure of "dispersion" ($\hat{\lambda}_2$) for monthly rainfall data required 40 to 50 samples and L-skew of $\hat{\tau}_3$ required 60 to 70 samples.

12.8.3 Further Evaluation of Pearson Type III Parameter Estimation

Asquith and others (2006, table 5) in an extensive analysis of storm statistics for hourly rainfall data in Texas concluded that the L-moments of rainfall depth for storms defined by a 72-hour minimum interevent time are $\lambda_1 = 24.5$ millimeters (mm), $\lambda_2 = 14.2$ mm, and $\tau_3 = 0.452$. These L-moments were derived from data recorded by 533 rainfall stations

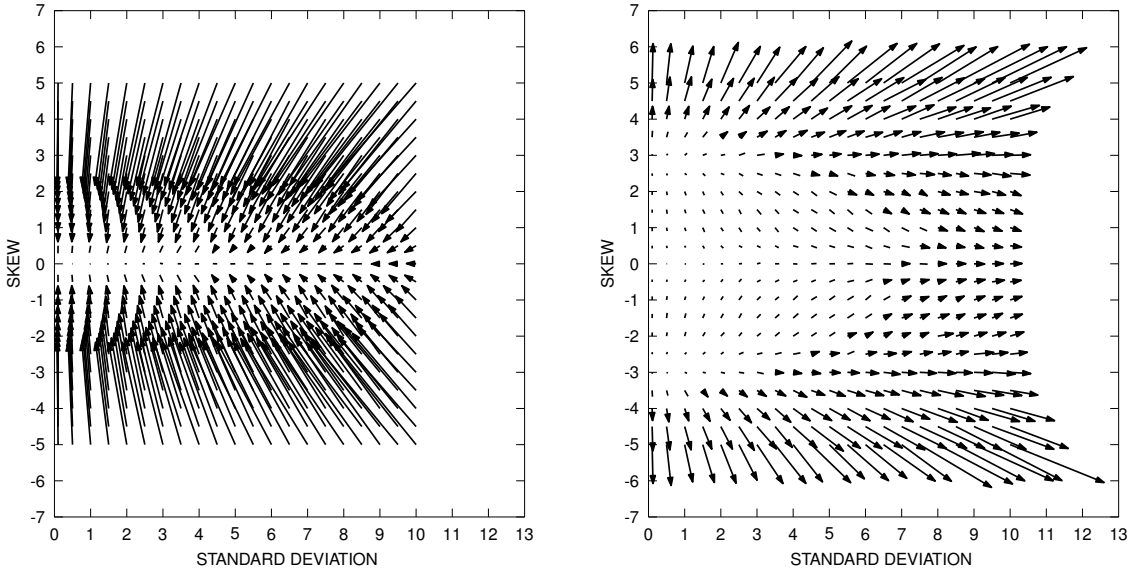

Figure 12.13. Bias of sample standard deviation and skew statistics for a Pearson Type III parent and sample size 10. Left graph is for product moment estimation. Right graph is for L-moment estimation. Arrows lead from the population values to the means of 10,000 sample statistics.

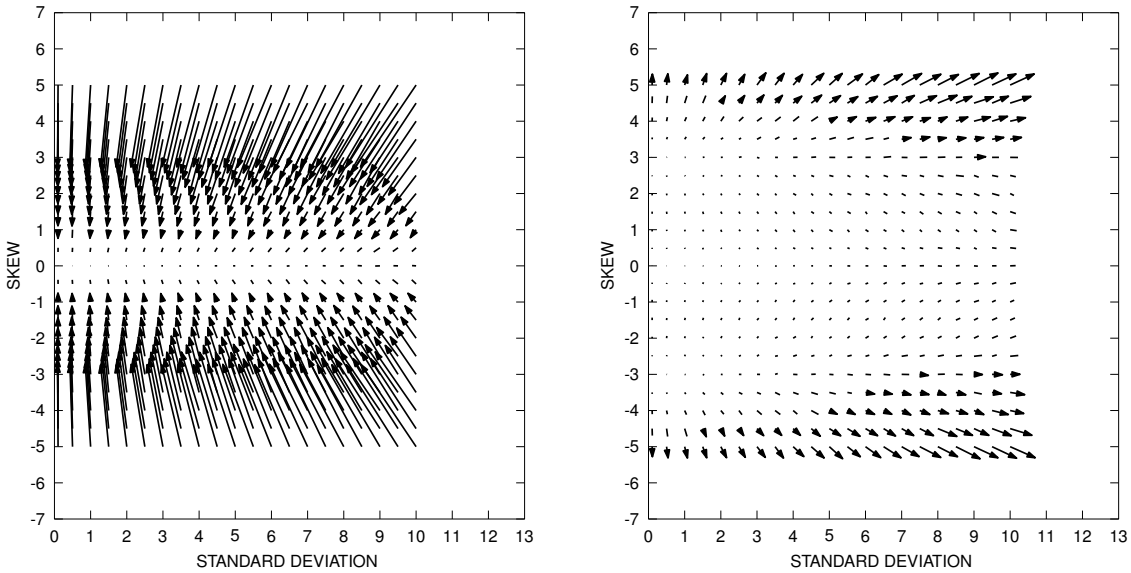

Figure 12.14. Bias of sample standard deviation and skew statistics for a Pearson Type III parent and sample size 20. Left graph is for product moment estimation. Right graph is for L-moment estimation. Arrows lead from the population values to the means of 10,000 sample statistics.

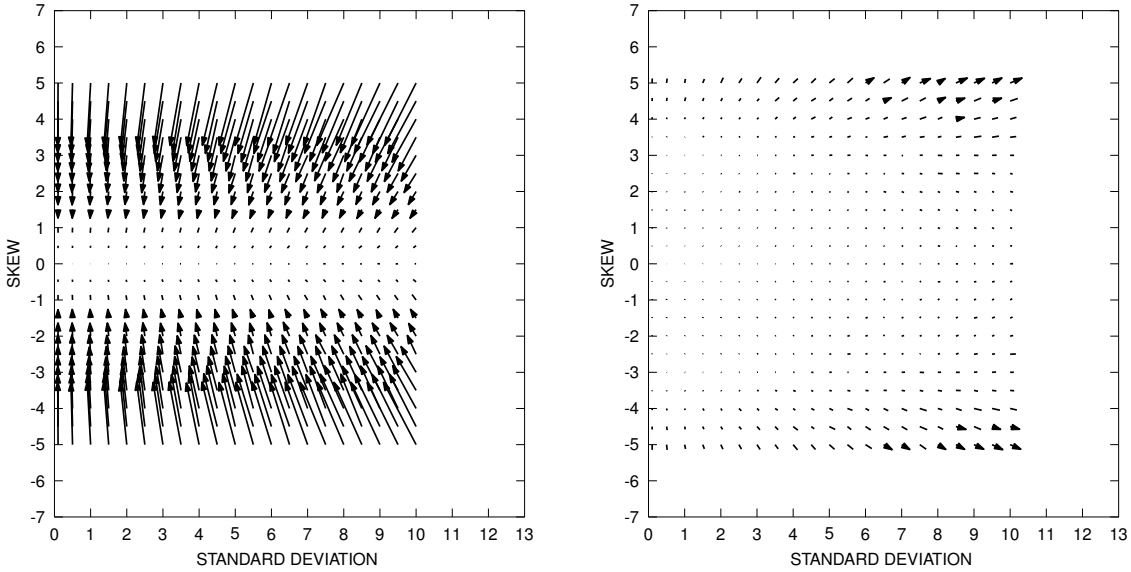

Figure 12.15. Bias of sample standard deviation and skew statistics for a Pearson Type III parent and sample size 40. Left graph is for product moment estimation. Right graph is for L-moment estimation. Arrows lead from the population values to the means of 10,000 sample statistics.

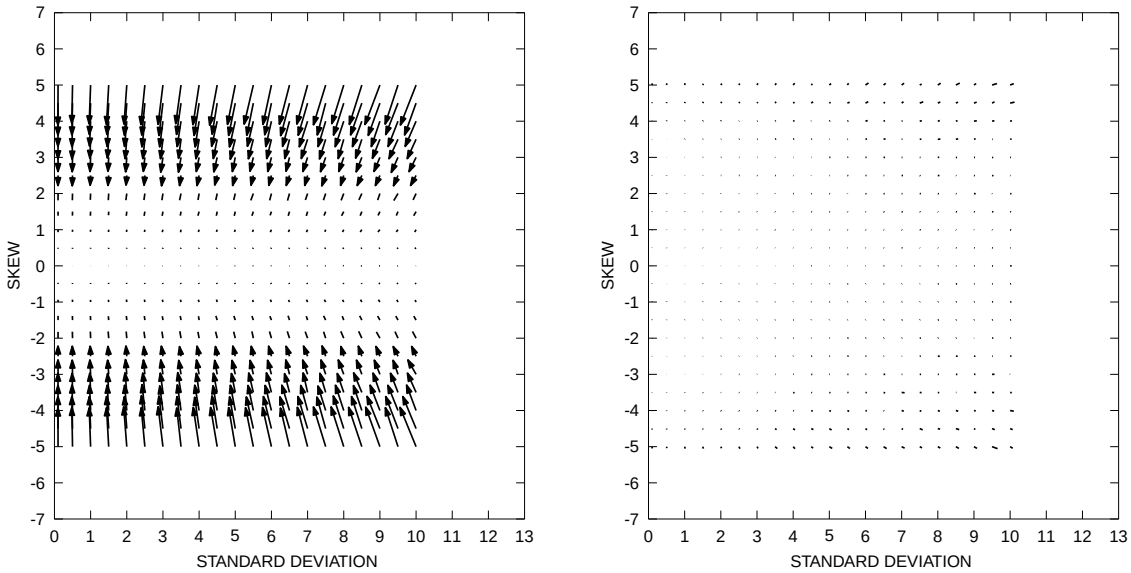

Figure 12.16. Bias of sample standard deviation and skew statistics for a Pearson Type III parent and sample size 100. Left graph is for product moment estimation. Right graph is for L-moment estimation. Arrows lead from the population values to the means of 10,000 sample statistics.

throughout Texas with a combined data record in excess of 1.03 million values. The equivalent Pearson Type III parameters for these L-moments are $\mu = 24.5$ mm, $\sigma = 31.2$ mm, and $\gamma = 2.75$.

These parameters define a $\mathrm{PE3}'$ parent, $\mathrm{PE3}'(24.5$ mm, 31.2 mm, $2.75)$, of storm depth. A random sample of $n = 20$ was drawn from this parent to represent somewhat heavy-tailed hydrologic data for example purposes. The random sample is shown in figure 12.17 along with the parent $\mathrm{PE3}'$distribution.[10] The sample product moments are $\hat{\mu}_{pm} = 29.0$, $\hat{\sigma}_{pm} = 35.0$, $\hat{\gamma}_{pm} = 1.89$, and the sample L-moments, expressed as Pearson Type III parameters, are $\hat{\mu}_{\lambda} = 29.0$, $\hat{\sigma}_{\lambda} = 38.4$, $\hat{\gamma}_{\lambda} = 2.70$. These two Pearson Type III distributions [$\mathrm{PE3}_{pm}(29.0, 35.0, 1.89)$ and $\mathrm{PE3}_{\lambda}(29.0, 38.4, 2.70)$] are shown in figure 12.17 by the black curves.

The parent distribution is heavy tailed with $\gamma = 2.75$. The results in Section 12.8.2 show that the sample product moments have considerable bias for data having this much skew. Therefore, following a step that a practitioner might (should?) do, and to facilitate comparison purposes, $\log_{10}$ transformation of the data was made, the sample product moments and L-moments again were computed, and log-Pearson Type III fit to both moment types. The two log-Pearson Type III distributions [$\mathrm{LP3}_{pm}(1.13, 0.599, -0.103)$ and $\mathrm{LP3}_{\lambda}(1.13, 0.623, -0.157)$] are shown in figure 12.17 by the thin grey curves.

Several observations of the two sample Pearson Type III and two sample log-Pearson Type III curves are made. $\mathrm{PE3}_{pm}$ is truncated at about $F < 0.2$ (horizontal axis), which reflects negative quantiles. In general, the focus of distributional analysis is on the right or high magnitude tail ($F \geq 0.5$) of the distribution. For the remainder of this discussion, therefore, indifference to the left tail is made, and the presence of negative quantiles is ignored.

From the figure, an immediately apparent difference between the two Pearson Type III and two log-Pearson Type III for the sample is that the log-Pearson Type III curves are straighter (less skewed) in the logarithmic axis than the Pearson Type III curves. As a result, the quantiles for $F > 0.90$ relative to the parent are overestimated by the log-Pearson Type IIIas judged by the thin lines plotting above the thick grey line of the parent. The overestimation is considerable. Whether or not this observation is a vagary of sampling can be explored by simulation. Natural questions are: (1) on average, does log-Pearson Type III overestimate (to clarify, the use of an log-Pearson Type III distribution) for $F > 0.90$

[10] This figure was generated by the author's TKG2 graphics package and annotated in Adobe Illustrator CS3.

as hinted at by the figure or (2) does log-Pearson Type III underestimate for $F > 0.90$? These questions and several others are explored in the next section.

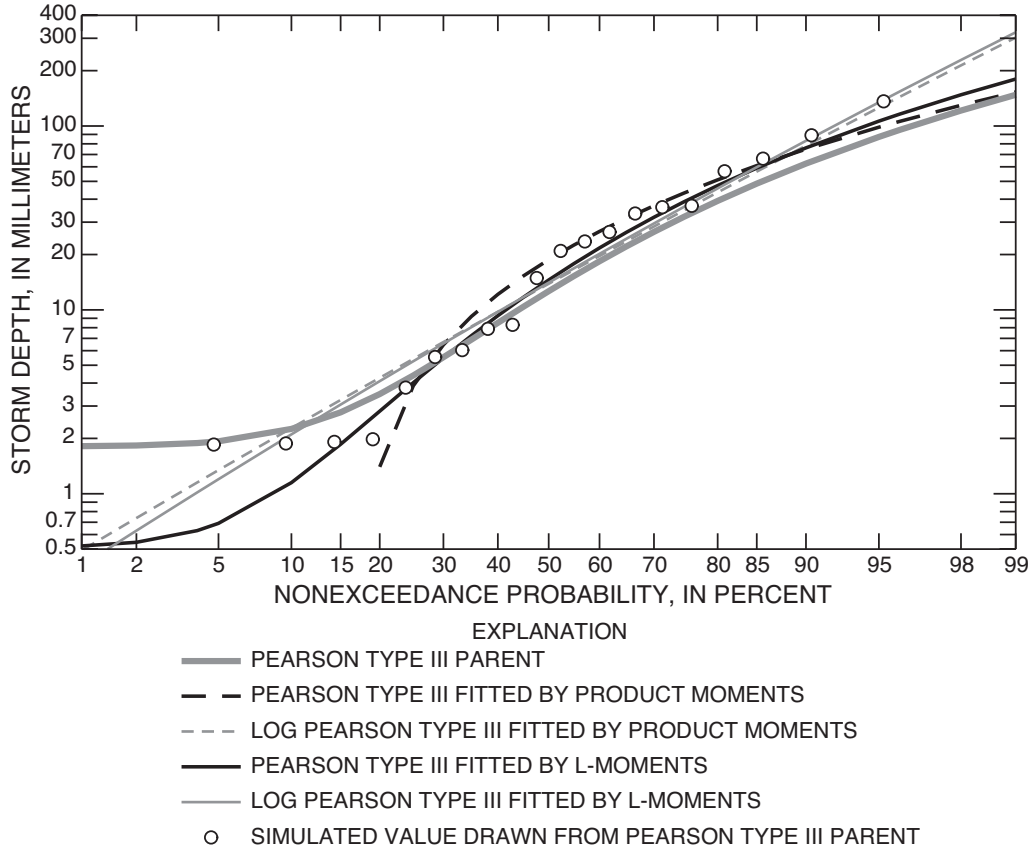

Figure 12.17. Comparison of product moment and L-moment fits of Pearson Type III and log Pearson Type III to 20 samples drawn from a Pearson Type III parent

12.8.4 Thought Experiment—To Product Moment or L-moment and To Transform Data?

For a thought experiment, suppose that only the $n = 20$ sample of figure 12.17 for this rainfall phenomena was available; the actual parent distribution is unknown. In particular, it is unknown whether the parent is in $\log_{10}$ space or not. In practical circumstances,

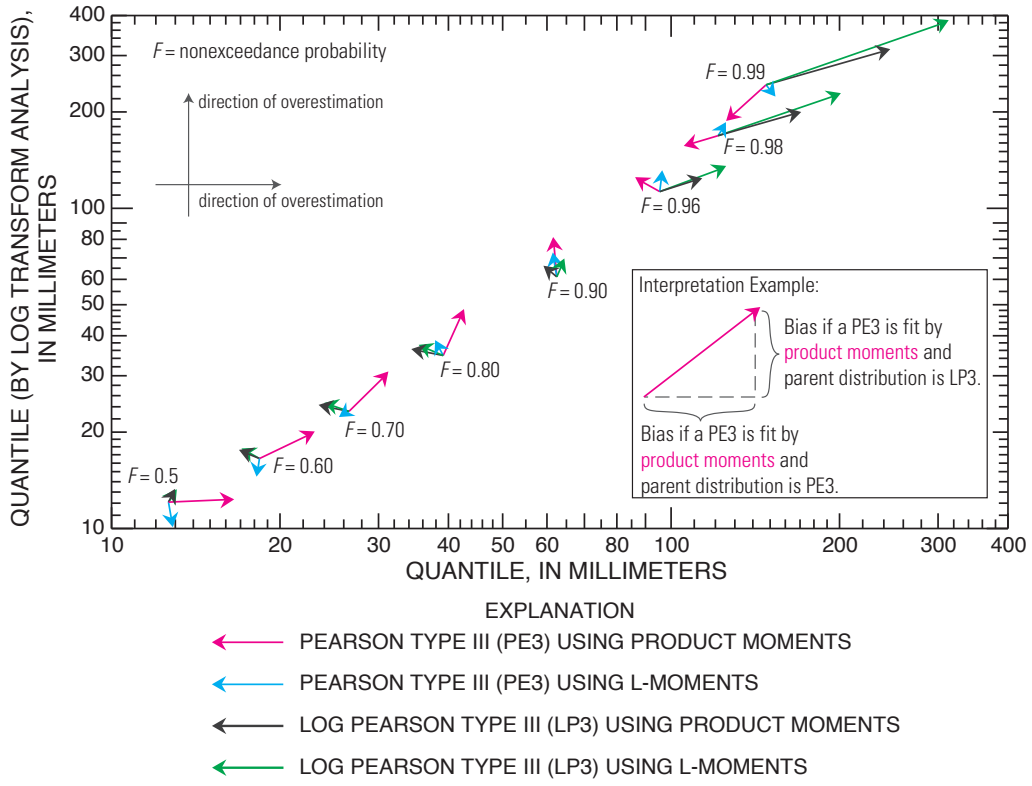

Figure 12.18. Comparison of product moment and L-moment estimation of the 0.99 quantile of Pearson Type III and log-Pearson Type III parents using both nontransformed and $\log_{10}$ transformed data for a sample size of 20. Arrows lead from the population value to the means of 10,000 sample statistics.

should the fitted log-Pearson Type III or Pearson Type III be preferred and how should either be estimated?

The $\hat{\gamma}$ value of the sample is greater than about 1.5, which from the left graph in figure 12.14 suggests that skew is large enough that $\log_{10}$ transformation might be warranted to increase the effectiveness of the product moments. The left graph in figure 12.14 shows that the sample product moments are expected to underestimate γ and in fact $\hat{\gamma} < \hat{\gamma}_\lambda$. The right graph in the figure shows that the L-moment estimate of γ is unbiased. Based on visual or graphical comparison of the data points to the fitted distributions shown in figure 12.17, one might conclude that the log-Pearson Type III is appropriate; however, much caution is advised in judging fit for such a small sample in this way.

When analyzing a sample such as in figure 12.17, the analyst has a serious quandary. Which of the four fitted curves to the $n = 20$ data given application of a Pearson Type III

family is most appropriate? (For this thought experiment, the fact that the actual parent is Pearson Type III is not known.) The quandary is that four options exist, and collectively, these options are termed the "four methods" and are:

1. Compute sample product moments and fit the Pearson Type III, which is abbreviated as $\mathrm{PE3}_{pm}$;
2. Compute sample L-moments and fit the Pearson Type III, which is abbreviated as $\mathrm{PE3}_{\lambda}$;
3. Perform $\log_{10}$ transform, compute sample product moments, and fit the log-Pearson Type III, which is abbreviated as $\mathrm{LP3}_{pm}$; and
4. Perform $\log_{10}$ transform, compute sample L-moments, and fit the log-Pearson Type III, which is abbreviated as $\mathrm{LP3}_{\lambda}$.

The $\mathrm{PE3}'$ parent is in real space. As will become evident, a somewhat equivalent version of $\mathrm{PE3}'$ as a log-Pearson Type III parent ($\mathrm{LP3}'$) will be useful. A giant sample size of $n = 1{,}000{,}000$ was drawn from $\mathrm{PE3}'$, the L-moments of $\log_{10}$ transformed values were computed, and the log-Pearson Type III parameters estimated. The estimated $\mathrm{LP3}'$ is $\mathrm{LP3}'(1.0929, 0.3021, 0.01935)$.

The $\mathrm{PE3}'$ and $\mathrm{LP3}'$ distributions are used in a simulation experiment involving selected quantiles $X(F)$ for $F = (0.5, 0.6, 0.7, 0.8, 0.9, 0.96, 0.98, 0.99)$. The results of the experiment are shown in figure 12.18. Similar to earlier figures, the arrows lead from the population value to the sample values for each quantile, more precisely for figure 12.18 alone, because of limitations of the interactive-graphical editing software,[11] the arrow-head centers are at the coordinates of the sample values. (This rendering is in contrast to the arrow heads in figs. 12.13–12.16.)

For each of the selected F values, the eight true values $\mathrm{PE3}'(F)$ and $\mathrm{LP3}'(F)$ were computed. These values are identified in figure 12.18 at the nexus of the arrow clusters and the corresponding F value label. The horizontal axis represents the $\mathrm{PE3}'(F)$ values, and the vertical axis represents the $\mathrm{LP3}'(F)$ values.

The simulation experiment was conducted as follows. In a process that was repeated $10{,}000$ times for each F value, samples $n = 20$ were drawn from $\mathrm{PE3}'(24.5, 31.2, 2.75)$ and separately from $\mathrm{LP3}'(1.0929, 0.3021, 0.01935)$. For each sample, $\hat{\mu}_{pm}$, $\hat{\sigma}_{pm}$, and $\hat{\gamma}_{pm}$ were computed and $\hat{\mu}^{\log}_{pm}$, $\hat{\sigma}^{\log}_{pm}$, and $\hat{\gamma}^{\log}_{pm}$ of a $\log_{10}$ transformation of the sample. Using

[11] Adobe Illustrator CS3, frustrating limitations of the arrow rendering features.

these values, the quantiles of the fitted $\mathrm{PE3}_{pm}$ and $\mathrm{LP3}_{pm}$ were computed. Finally, the means of the $10{,}000$ quantiles of fitted $\mathrm{PE3}_{pm}$ and $\mathrm{LP3}_{pm}$ for each F were computed. These sixteen mean values provide the coordinates at the center of the magenta and black arrow heads in figure 12.18.

Similarly, for the same $n = 20$ samples, the L-moments were computed and converted to Pearson Type III parameters ($\hat{\mu}_\lambda$, $\hat{\sigma}_\lambda$, $\hat{\gamma}_\lambda$) and log-Pearson Type III parameters ($\hat{\mu}_\lambda^{\log}$, $\hat{\sigma}_\lambda^{\log}$, $\hat{\gamma}_\lambda^{\log}$). Again, using these values, the quantiles of the fitted $\mathrm{PE3}_\lambda$ and $\mathrm{LP3}_\lambda$ were computed. Finally, the means of the $10{,}000$ quantiles of fitted $\mathrm{PE3}_\lambda$ and $\mathrm{LP3}_\lambda$ for each F were computed. These 16 mean values provide the coordinates at the center of the cyan and green arrow heads infigure 12.18.

The lengths of the arrows in the figure represent bias. The choice of $\log_{10}$ for both axis scales is intentional so relative bias for each F is represented. Arrows oriented toward the right indicate that overestimation of $\mathrm{PE3}$ occurs, and arrows oriented toward the top indicate that overestimation of $\mathrm{LP3}$ occurs. The cyan arrows (estimation by L-moments without $\log_{10}$ transformation of the data) are almost all *systematically shorter* than the others. This indicates that use of a Pearson Type III fitted by L-moments is preferred whether the parent is Pearson Type III or log-Pearson Type III. The magenta arrows (estimation by $\mathrm{PE3}_{pm}$) are generally the longest, which indicates the poorest performance of the four methods.

The description of the $\mathrm{LP3}_{pm}$ and $\mathrm{LP3}_\lambda$ methods is more complex. Each method apparently outperforms $\mathrm{PE3}_{pm}$ for $F \leq 0.90$ but appears to dramatically underperform for $F > 0.90$. The $\mathrm{LP3}_{pm}$ and $\mathrm{LP3}_\lambda$ perform similarly and for a given F are oriented in the same direction (unlike $\mathrm{PE3}_{pm}$ and $\mathrm{PE3}_\lambda$). As F increases above $F \geq 0.96$, $\mathrm{LP3}_\lambda$ appears to dramatically underperform and, in particular, underperforms (by overestimation) for $\mathrm{PE3}'$. This implies that use of L-moments on $\log_{10}$ transformed data, whether the true parent form was $\mathrm{PE3}'$or $\mathrm{LP3}'$, provides little and perhaps even a harmful parameter estimation benefit for large F.

12.8.5 Some Conclusions

If a parent distribution is Pearson Type III, then the sample L-moments appear to systematically outperform the sample product moments for parameter estimation. Specifically, as measured by bias, L-moments outperform the product moments as variability and skew-

ness of the Pearson Type III parent becomes large; the product moments underestimate both variability and skewness. Under conditions of near zero skewness, sample product moments and L-moments have similar performance. The use of any moment statistic for small samples requires caution. The L-moments overestimate the variability and skewness for the Pearson Type III for small samples, but by a sample size of 40, the sample L-moments can be considered reasonably unbiased.

Logarithmic transformation of the data decreases skewness and is a useful and important tool for increasing the performance of product moments. The comparison of Pearson Type III and log-Pearson Type III parameter estimation for a hypothetical right-tail heavy sample of size 20 suggests that a $\mathrm{PE3}_{\lambda}$ performs better whether the actual parent distribution is either Pearson Type III or log-Pearson Type III. Finally, the author concludes that $\mathrm{PE3}_{\lambda}$ estimation generally should be preferred over $\mathrm{PE3}_{pm}$ estimation in applied circumstances. This conclusion is complementary to that of Wallis (1988, p. 311) who concludes at a minimum that $\mathrm{LP3}_{\lambda}$ should be preferred over $\mathrm{LP3}_{pm}$.

12.9 L-comoments—Multivariate Extensions of L-moments

This dissertation is focused on univariate distributional analysis using L-moments. Fortunately, about 17 years after the ground breaking work of Hosking (1990), Serfling and Xiao (2007) established a coherent extension of L-moments into multivariate space. It seems therefore fitting to end this dissertation with an introduction to L-comoments and show some original contributions of the author to L-moment theory that involves L-comoments and copulas. In particular, this section summarizes multivariate L-moments and support provided by the *lmomco* package.

Serfling and Xiao (2007) introduced **multivariate L-moments** or **L-comoments** and provide considerable discussion of their properties and computation. In brief, L-comoments have a representation in terms of **concomitants**. Serfling and Xiao (2007, p. 1772) consider, through an adapted quotation, "a sample $\{(X_i^{[1]}, X_i^{[2]}),\ 1 \le i \le n\}$ from [bivariate distribution] $G(x^{[1]}, x^{[2]})$ with marginals $F_1(x)$ and $F_2(x)$." For the ascending ordered values of $X^{[2]}$, Serfling and Xiao refer to the "element of $\{X_i^{[1]}, \cdots, X_n^{[1]}\}$ that is paired with $X_{j:n}^{[2]}$ the concomitant $X_{j:n}^{[12]}$ of $X_{j:n}^{[2]}$." The authors Serfling and Xiao show that the rth L-comoment has a representation as the expected value of a concomitant in precisely "the same way as L-moments are defined in terms of expected values of order statistics"

as shown in eq. (3.4), and the representation is

$$\lambda_r^{[12]} = \frac{1}{r}\sum_{j=0}^{r-1}(-1)^j \binom{r-1}{j} \mathrm{E}[X_{r-j:n}^{[12]}] \tag{12.24}$$

Serfling and Xiao, using (12.24), subsequently provide an unbiased sample estimator

$$\widehat{\lambda}_r^{[12]} = \frac{1}{n}\sum_{j=1}^{n} w_{j:n}^{(r)} X_{j:n}^{[12]} \tag{12.25}$$

where the weights $w_{j:n}^{(r)}$ are computed as in eq. (6.50) and $\widehat{\lambda}_r^{[12]}$ is defined as the rth L-comoment of $X^{[1]}$ with respect to $X^{[2]}$. Likewise the respective estimator for the rth L-comoment of $X^{[2]}$ with respect to $X^{[1]}$ is

$$\widehat{\lambda}_r^{[21]} = \frac{1}{n}\sum_{j=1}^{n} w_{j:n}^{(r)} X_{j:n}^{[21]} \tag{12.26}$$

An important characteristic of L-comoments is that they need not be symmetric (and usually are not), that is, $\widehat{\lambda}_r^{[12]} \neq \widehat{\lambda}_r^{[21]}$ or the expected co-movements of $X^{[1]}$ with respect to $X^{[2]}$ are not necessarily the same as the expected co-movements of $X^{[2]}$ with respect to $X^{[1]}$. Like Serfling and Xiao, the author embraces the asymmetry as a feature of these statistics. The asymmetry is counter to the symmetry defined into conventional **measures of association** (Nelson, 2006, p. 169), such as the **measures of concordance** statistics of Kendall's Tau and Spearman's Rho (see `help(cor)`).

The L-comoment ratios $\tau_r^{[12]} = \lambda_r^{[12]}/\lambda_2^{[1]}$ or $\tau_r^{[21]} = \lambda_r^{[21]}/\lambda_2^{[2]}$ for $r \geq 2$ are analogs to τ_r and have sample counterparts $\widehat{\tau}_r^{[12]} = \widehat{\lambda}_r^{[12]}/\widehat{\lambda}_2^{[1]}$. The ratios are interpreted as follows: for $r = 2$, the notation $\tau_2^{[12]}$ is to be read "the **L-correlation** of $X^{[1]}$ with respect to $X^{[2]}$," and for $r = 3$, $\tau_3^{[21]}$ is to be read "the **L-coskew** of $X^{[2]}$ with respect to $X^{[1]}$." The L-comoment ratios of $r = 4$ are known as **L-cokurtosis**.

USING R ——————————————————————————— USING R

The L-comoments are readily demonstrated with several functions of the *lmomco* package. Starting in example 12–29, a bivariate random sample of $n = 500$ for a standard Normal distributed `X` with `Y` being computed as shown along with a standard Normal error term. The bivariate sample is stored in the data frame `D`. The example continues by plotting the simulated data in figure 12.19 along with the two **rug plots**, which are created

by the `rug()` function in a semi-transparent red color. The rug plots show the marginal distribution of each variable or "drape" the values onto the respective axis of the variable.

12–29

```
X <- rnorm(500); Y <- X^2 + rnorm(500)
D <- data.frame(X=X, Y=Y)
#pdf("lcomomentA.pdf", version="1.4")
plot(D)
rug(D$X, side=1, col=rgb(1,0,0,0.4))
rug(D$Y, side=2, col=rgb(1,0,0,0.4))
#dev.off()
```

The plot in figure 12.19 shows that the bivariate sample has a somewhat complex dependency structure between `X` and `Y`. The horizontal-axis rug plot shows symmetrically distributed values with tapering tails and is obviously standard Normal. The distribution of `Y` that is shown on the vertical-axis rug plot shows that the distribution has positive skewness. These differences in symmetry are quantified in example 12–32.

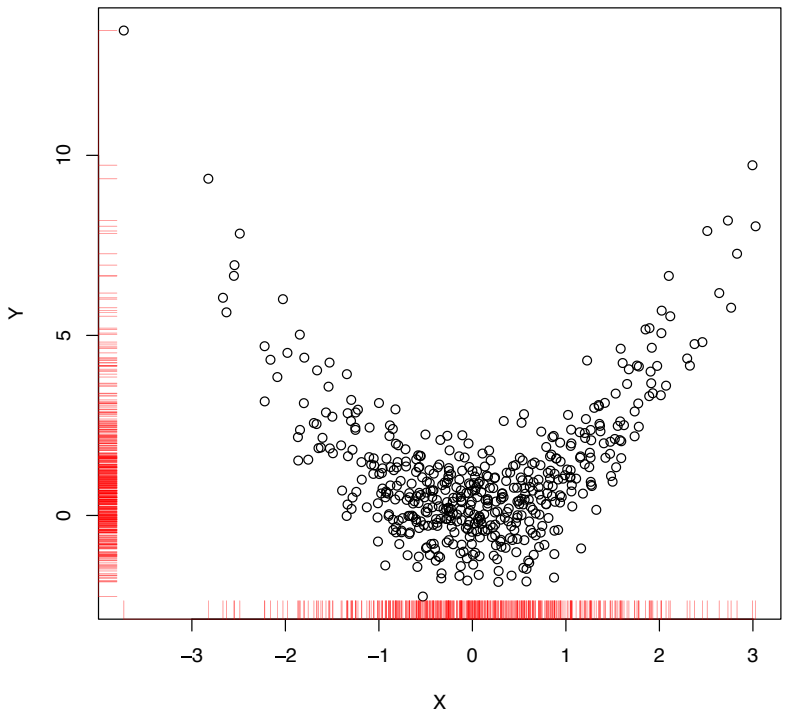

Figure 12.19. Simulated bivariate data for computation of L-comoments from example 12–29

Example 12–30 continues the discussion by computing the first L-comoment using the `Lcomoment.matrix()` function for `k=1`. The results are shown. In particular, the

content of the `$matrix` attribute holds the L-comoment matrix of order 1. This matrix contains each arithmetic mean as the subsequent call to the `mean()` function confirms at the end of the example.

12–30

```
Lcomoment.matrix(D,k=1)
$type
[1] "Lcomoment.matrix"
$order
[1] 1
$matrix
           [,1]     [,2]
[1,] 0.07685957       NA
[2,]         NA 1.122278


# Now finally, for comparison, compute means
mean(D)
         X          Y
0.07685957 1.12227798
```

◀

Continuing with the bivariate random sample in variable `D` from example 12–29, the L-comoment-matrix of order 2 is computed in example 12–31 and set into variable `L2`. Using the second order matrix, the L-correlation between the two variables is computed by the `Lcomoment.correlation()` function, and the results are shown.

12–31

```
L2 <- Lcomoment.matrix(D, k=2) # order 2 matrix
Lcomoment.correlation(L2) # compute L-correlation
$type
[1] "Lcomoment.coefficients"
$order
[1] 2
$matrix
          [,1]      [,2]
[1,] 1.0000000 0.1173538
[2,] 0.1222056 1.0000000

# Just for comparison, compute Spearman's Rho
cor(D$X,D$Y, method="spearman")
[1] 0.1153683
```

The results show that the respective L-correlations are $\tau_2^{[12]} = 0.117$ and $\tau_2^{[21]} = 0.122$. These are small values, so the association between the variables (note the lack of numerical

equality in the statistics) is weak. This conclusion is confirmed using the `cor()` function to compute a Spearman's Rho of about 0.115 as shown in example 12–31. Many measures of association, such as Kendall's Tau and Spearman's Rho, are symmetric statistics (Nelson, 2006, p. 169), that is, the equality `cor(X,Y)=cor(Y,X)` exists. This is not true of the L-comoments. ◀

Continuing the presentation and using the bivariate random sample in variable `D` from example 12–29, the L-comoment-matrix of order 3 is computed in example 12–32 and set into variable `L3`. For subsequent comparison, the familiar L-moments of variable `X` (contained in `D$X`) are computed by the `lmoms()` function and set into variable `LMRx`. The third L-moment is $\lambda_3 = 0.0086$ as shown in example 12–32.

12–32

```
L3 <- Lcomoment.matrix(D, k=3)
LMRx <- lmoms(D$X)
cat(c("# OUTPUT: Lcomoment M[1,1] =",
          round(L3$matrix[1,1], digits=5),  "\n",
      "# and 3rd L-moment by lmoms=",
          round(LMRx$lambdas[3], digits=5), "\n"))
# OUTPUT: Lcomoment M[1,1] = 0.0086
# and 3rd L-moment by lmoms= 0.0086

Lcomoment.coefficients(L3,L2) # compute L-coskew
$type
[1] "Lcomoment.coefficients"
$order
[1] 3
$matrix
           [,1]       [,2]
[1,] 0.01499681 0.06300544
[2,] 0.64066883 0.23312682

round(lmoms(D$X)$ratios, digits=5)
[1]       NA  7.46231  0.01500  0.15098 -0.01338

round(lmoms(D$Y)$ratios, digits=5)
[1]      NA 0.85248 0.23313 0.22161 0.09630
```

Example 12–32 continues by computing the L-coskew between the two variables using the `Lcomoment.coefficients()` function. This function requires the L-comoment matrix of order 2 (`L2`) from example 12–31 as the second argument. The results show that $\tau_3^{[12]} = 0.06$ and $\tau_3^{[21]} = 0.64$.

The two sample L-coskews attain different values. Can the differences ($0.06 \ll 0.64$) be interpreted? The $\tau_3^{[12]} = 0.06$ or L-coskew of `X` with respect to `Y` is near zero—in other words, symmetry of sorts exists in the co-movement of `X` with respect to `Y`. ◀

Further consideration of L-coskew is needed. In example 12–33, a trivariate random sample is created and set into the `Trivar` variable. Readers are alerted that the primary extension from the previous examples is the negation of the `X^2` term between `Y` and `Z`. The L-coskew matrix then is computed and shown.

12–33

```
X <- rnorm(500); Y <- X^2 + rnorm(500); Z <- -X^2 + rnorm(500)
Trivar <- data.frame(X=X, Y=Y, Z=Z) # Trivariate random sample
L2 <- Lcomoment.matrix(Trivar, k=2) # 2nd L-comoment matrix
L3 <- Lcomoment.matrix(Trivar, k=3) # 3rd L-comoment matrix
Lcomoment.coefficients(L3,L2) # compute L-coskew
$type
[1] "Lcomoment.coefficients"
$order
[1] 3
$matrix
             [,1]          [,2]         [,3]
[1,] -0.006193864 -0.006558694 -0.06360398
[2,]  0.598668642  0.206718534  0.32379560
[3,] -0.627044762 -0.342615780 -0.19173351
```

In the output, $\tau_3^{[21]} = 0.60$ (differs from example 12–32 because a new sample is created) and $\tau_3^{[31]} = -0.63$. Readers are asked to notice that these two L-coskew values are of similar magnitude as anticipated but differ in sign because of the negation of `Z` relative to `Y`.

Example 12–33 shows that the L-comoments are readily computed for >2-dimensional data using the `Lcomoment.coefficients()` function. Because bivariate data are so common, the *lmomco* package provides the `lcomoms2()` function (the `2` in the function name reflects "2 dimensional") to provide potentially more accessible L-comoment data structures. ◀

Some setup of new mathematics is needed before the `lcomoms2()` function is used for an example of multivariate distributional analysis.

Nelson (2006) provides a comprehensive introduction to **copulas**, which are special multivariate distributions having marginal distributions that are Uniform. The copula $\mathbf{C}(u, v)$ (bivariate) is the expression of the joint nonexceedance probability of random variable U and random variable V. The quantities u and v are the respective nonexceedance probabilities.

Solely for the purpose of illustration, the Marshall-Olkin copula is used. The copula is set by parameters α and β, and the copula possesses some interesting L-comoments. The Marshall-Olkin copula is

$$\mathbf{C}(u, v) = \min(vu^{1-\alpha}, uv^{1-\beta}) \tag{12.27}$$

and this bivariate copula is created in example 12–34 by the `MOcop()` function. The function receives the two probabilities u and v and a vector of parameters in the `para` argument.

12–34

```
"MOcop" <-
function(u,v, para=NULL) {
  alpha <- para[1]
  beta  <- para[2]
  return(min(v*u^(1-alpha), u*v^(1-beta)))
}
```

The method of conditional simulation (Nelson, 2006, pp. 40–42) can be used to create random pairs u and v as jointly distributed by a copula. The method requires a function to compute the inverse of the derivative of a copula. In example 12–35, the `derCOPinv()` function is created to numerically compute the inverse of the derivative of a copula for a given u.

12–35

```
"derCOPinv" <-
function(cop=NULL, u, t, delu=.Machine$double.eps^0.5, para=NULL)
{
    "func" <-
    function(v, u=NULL, LHS=NULL, cop=NULL,
                delu=delu, para=para) {
             dc <- (cop(u+delu,v, para=para) -
                    cop(u,     v, para=para))/delu
             return(LHS - dc)
    }
    try(rt <- uniroot(func, interval=c(0,1), u=u, LHS=t,
                      cop=cop, delu=delu, para=para))
    ifelse(length(rt$root) != 0, return(rt$root), return(NA))
}
```

Conditional simulation is actually implemented by the `simulateCopula()` function, which is created in example 12–36. The number of simulations is specified by the `n` argument. The copula function to simulate and its parameters are set by the `cop` and `para` arguments, respectively.

12–36

```
"simulateCopula" <-
function(n, cop=NULL, para=NULL) {
  U <- V <- vector(mode="numeric")
  for(i in 1:n) {
    u <- runif(1); t <- runif(1) # two uniformly distributed vars
    v <- derCOPinv(cop=cop, u, t, para=para)
    if(is.na(v)) {
       warning("could␣not␣uniroot␣in␣derCOPinv,␣skipping␣sample")
       warning(para); next
    }
    U[i] <- u; V[i] <- v
  }
  return(data.frame(U=U, V=V))
}
```

Finally, in example 12–37, the Marshall-Olkin copula of eq. (12.27) is simulated for $n = 1{,}000$, and the results are shown in figure 12.20. The figure shows a complex and asymmetrically dependent joint distribution that also has both continuous and singular components. By inspection of the plot, the data have positive association, and therefore, the data should have positive L-correlation. The asymmetry of the plot suggests that the data should have non-zero L-coskew.

12–37

```
simA <- simulateCopula(1000, cop=MOcop, para=c(0.4,0.9))
#pdf("mocopA.pdf")
plot(simA$U, simA$V,
     xlab="RANDOM␣VARIABLE␣U␣NONEXCEEDANCE␣PROBABILITY",
     ylab="RANDOM␣VARIABLE␣V␣NONEXCEEDANCE␣PROBABILITY")
#dev.off()

lcomoms2(simA, nmom=4)
$L1
          [,1]      [,2]
[1,] 0.4821094        NA
[2,]        NA 0.4898829

$L2
           [,1]       [,2]
[1,] 0.16851096 0.08525086
[2,] 0.08598543 0.16825552

$T2
          [,1]      [,2]
[1,] 1.0000000 0.5059069
[2,] 0.5110408 1.0000000
```

```
$T3
             [,1]       [,2]
[1,]  0.02575336 0.19724727
[2,] -0.02013093 0.01465324

$T4
              [,1]         [,2]
[1,] -0.004583203  0.014791149
[2,]  0.051722854 -0.001361602
```

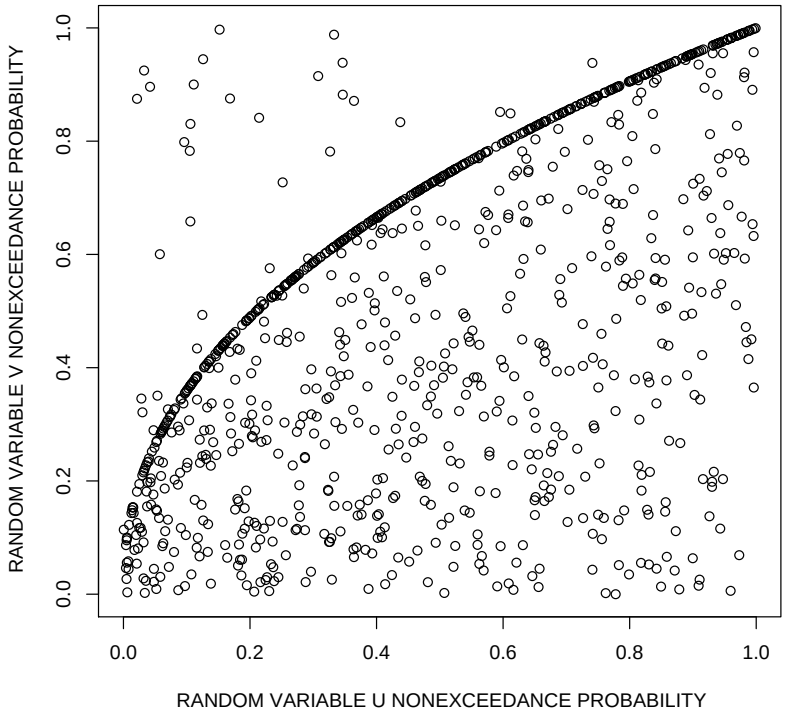

Figure 12.20. Simulated bivariate data from Marshall-Olkin copula $\alpha = 0.4$ and $\beta = 0.9$ (open circles) from example 12–37

Example 12–37 continues after the plotting operations by computing the first four L-comoments using the `lcomoms2()` function. The L-moments (the matrix diagonals) and L-comoments (the off diagonals) for U and V for the low moment orders ($r = 1, 2$) are the two matrices

$$\widehat{\lambda}_1^{[1,2]} = \begin{bmatrix} 0.482 & \text{--} \\ \text{--} & 0.490 \end{bmatrix} \equiv \text{ means} \tag{12.28}$$

$$\widehat{\lambda}_2^{[1\leftrightarrow 2]} = \begin{bmatrix} 0.169 & 0.085 \\ 0.086 & 0.168 \end{bmatrix} \equiv \text{ L-scales and L-coscales} \tag{12.29}$$

and the τ_2 equivalent matrix is

$$\widehat{\tau}_2^{[1\leftrightarrow 2]} = \begin{bmatrix} 1 & 0.506 \\ 0.511 & 1 \end{bmatrix} \equiv \text{ L-correlations} \tag{12.30}$$

and for the higher moment orders ($r = 3, 4$), the L-moments and L-comoments are the two matrices

$$\widehat{\tau}_3^{[1\leftrightarrow 2]} = \begin{bmatrix} 0.026 & 0.197 \\ -0.020 & 0.015 \end{bmatrix} \equiv \text{ L-skews and L-coskews} \tag{12.31}$$

$$\widehat{\tau}_4^{[1\leftrightarrow 2]} = \begin{bmatrix} -0.005 & 0.015 \\ 0.052 & -0.001 \end{bmatrix} \equiv \text{ L-kurtosis's and L-cokurtosis} \tag{12.32}$$

To summarize the part of the notation, the quantity $[1\leftrightarrow 2]$ implies the row-major order of matrix entries of $\{[1], [12], [21], [2]\}$. ◀

The preceding example involves the Marshall-Olkin copula with $\alpha = 0.4$ and $\beta = 0.9$ and shows that the L-comoments are capable of measuring asymetrical skew (L-coskew). One of the L-coskews in $\widehat{\tau}_3^{[2\leftrightarrow 1]}$ is near zero ($\widehat{\tau}_3^{[21]} = -0.020$), whereas the other $\widehat{\tau}_3^{[12]} = 0.20$. Now, the parameters of the Marshall-Olkin copula are reversed in example 12–38, and an $n = 1{,}000$ simulation is performed. The results are set into variable `simB`. Finally, the simulated copula from example 12–37 again is plotted in figure 12.21 with the new simulated copula superimposed.

It is seen in figure 12.21 that the two copulas are stochastically-reflected images of each other. Therefore, there is the expectation that the L-coskew values will have different signs than those seen in example 12–37. In fact, example 12–38 shows this to be the case.

12–38

```
simB <- simulateCopula(1000, cop=MOcop, para=c(0.9,0.4))
#pdf("mocopB.pdf")
plot(simA$U, simA$V,
     xlab="RANDOM␣VARIABLE␣U␣NONEXCEEDANCE␣PROBABILITY",
     ylab="RANDOM␣VARIABLE␣V␣NONEXCEEDANCE␣PROBABILITY")
```

```
points(simB$U, simB$V, cex=0.75, pch=16) # filled circles
#dev.off()
lcomoms2(simB, nmom=4)

$L1
          [,1]     [,2]
[1,] 0.5056422       NA
[2,]        NA 0.503327

$L2
           [,1]       [,2]
[1,] 0.16598985 0.07705152
[2,] 0.07686484 0.16414484

$T2
          [,1]      [,2]
[1,] 1.0000000 0.4641942
[2,] 0.4682745 1.0000000

$T3
             [,1]         [,2]
[1,] -0.01561889 -0.080550864
[2,]  0.19063895  0.001365165

$T4
             [,1]       [,2]
[1,] 0.007412991 0.05057821
[2,] 0.032911336 0.01198380
```

Example 12–38 continues by using the `lcomoms2()` function to compute the first four L-comoments. In particular, the $\widehat{\tau}_3^{[1\leftrightarrow 2]}$ for U and V are

$$\widehat{\tau}_3^{[1\leftrightarrow 2]} = \begin{bmatrix} -0.016 & \underbrace{-0.081} \\ \overbrace{0.191} & 0.001 \end{bmatrix} \equiv \text{Marshall-Olkin copula } (\alpha{=}0.9, \beta{=}0.4) \tag{12.33}$$

$$\widehat{\tau}_3^{[1\leftrightarrow 2]} = \begin{bmatrix} 0.026 & \overbrace{0.197} \\ \underbrace{-0.020} & 0.015 \end{bmatrix} \equiv \text{Marshall-Olkin copula } (\alpha{=}0.4, \beta{=}0.9) \tag{12.34}$$

It is especially informative to compare the L-coskew values according to the $\overbrace{\text{overbrace}}$ and $\underbrace{\text{underbrace}}$. The paired values are effectively the same, but differ in position within the $\widehat{\tau}_3^{[1\leftrightarrow 2]}$ L-comoment matrix.

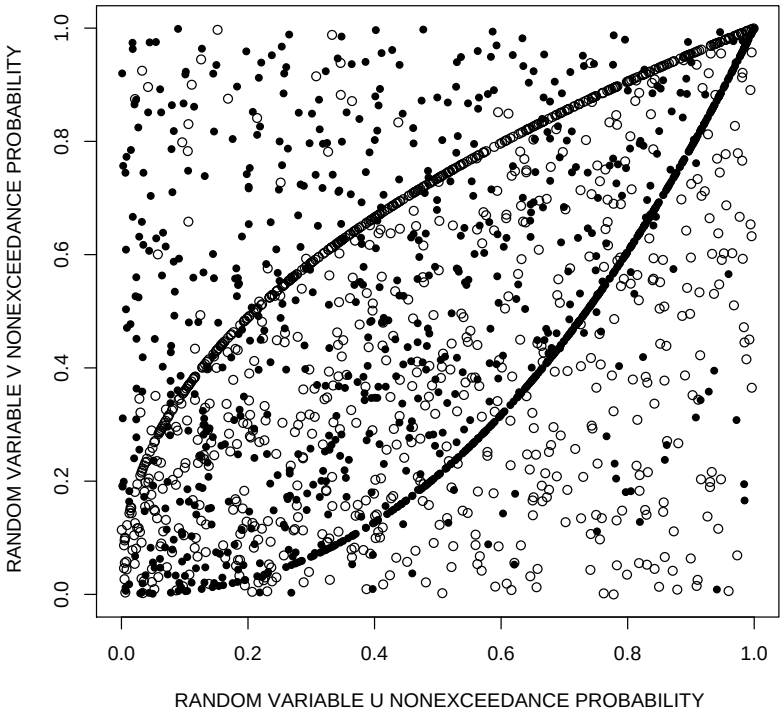

Figure 12.21. Simulated bivariate data from Marshall-Olkin copulas with $\alpha = 0.4$ and $\beta = 0.9$ (open circles) and $\alpha = 0.9$ and $\beta = 0.4$ (filled circles) from example 12–38

To further examine this observation, the two Marshall-Oklin copulas have the same parameters that only are exchanged with each other. Either copula has the same Spearman's Rho because Rho is defined for the Marshall-Olkin copula as

$$\text{Spearman's Rho } \rho = \frac{\alpha\beta}{\alpha + \beta - \alpha\beta} \tag{12.35}$$

This formula obviously results in the same numerical value for exchanged parameter values. Yet figure 12.21 clearly shows that the associative structure of the two bivariate distributions are distinct. The L-comoments quantify the asymmetry in the bivariate relation, whereas Spearman's Rho is incapable of capturing such asymmetry.

In conclusion, L-comoments have obvious applications in evaluation of multivariate dependency structure and for parameter estimation for multivariate distributions, including parameter estimation for copulas. However, further elucidation and pursuit of a **method of L-comoments** for parameter estimation of copulas is beyond the scope of this dissertation. ◀

12.10 Summary

This chapter presents more advanced demonstrations of L-moment-based distributional analysis than seen in the other chapters. The 38 examples provided readily followed and extendable examples of right-tail and left-tail censoring distributional analysis. The former is common in survival and lifetime analysis, whereas, the later is common in hydrologic or environmental data involving detection limits. Subsequently, the censoring discussion expands to include right-tail censoring by indicator variable and left-tail censoring by indicator variable through variable flipping. Following the censoring material, conditional probability adjustment for the presence of zero values by blipped-distribution modeling is shown. Although zero values are likely the most common application, the examples should be readily extendable to other lower thresholds. An extended discussion and demonstration of quantile uncertainty involving simulated sampling error (error related to sample size) and model-selection error (error related to choice of distribution) is provided. An extensive comparison of the performance of product moments and L-moments for parameter estimation for a wide range of skewness within Pearson Type III and log-Pearson Type III distributions is made. The results show that L-moments can significantly outperform product moments in terms of bias and that L-moments are preferred whether the parent distribution is Pearson Type III or log-Pearson Type III. Finally, this chapter and this dissertation ends with an introduction to multivariate L-moments or L-comoments, their sample computation, and applications that they might have in the context of applied statistical analysis using copulas.

Epilogue

This dissertation concerns distributional analysis with L-moment statistics using R. The breadth of the text is ambitious, complex, and encompasses the background, mathematics, algorithms, techniques, interpretations, references, and indexing needed for thorough documentation of L-moments and related statistics for the R environment for statistical computing. These elements are needed by beginners and many are useful to experts involved in distributional analysis of Normal to non-Normal, symmetrical to asymmetrical, and thin to heavy-tailed distributions. A wide range of disciplines are anticipated to be, or are already, impacted by material in this dissertation, the *lmomco* package, and other L-moment-related packages for R. Such judgement is made because statistical analysis of distributions touches investigations and research in all scientific, engineering, medical, and financial endeavors. It therefore is fitting to end this dissertation with a summary of the impact of the *lmomco* package (Asquith, 2011) and commentary on "where to go from here."

Impact of the *lmomco* Package

Close to the date of publication of this dissertation, there are several recognizable citations of the *lmomco* package that are found through Internet searches:

- General Statistical Programming—Cohen and Cohen (2008) provide a substantial book on statistical programming using R that references the *lmomco* package, although a traditional citation (end of chapter or end of text) to Asquith (2011) seems to be missing. The *lmomco* package is suggested for initial parameter guessing of a distribution for handoff into the numerical methods of the method of maximum likelihood.
- Agriculture—Liou and others (2007) cite the *lmomco* package; although the authors identify the author of *lmomco* as "William, H.A." The article appears in a Chinese agricul-

tural engineering journal; however, the abstract (only part in English) suggests that the authors' purpose (seemingly more general than agriculture engineering) is to evaluate the power of a goodness-of-fit test concerning L-moment ratio diagrams to established goodness-of-fit tests (Komogorov-Smirnov and Chi-squared).

- Biology—The *asbio* package (Aho, 2010) in R reverse suggests the *lmomco* package.
- Bioinformatics—The L-moment ratio diagram functions of *lmomco* are used to produce figures in Thomas and others (2010, p. 6); although the authors identify the package as "*lmomc*."
- Finance—The *lmomco* and *Lmoments* (Karvanen, 2009) packages are used by Kerstens and others (2010) for computation of L-moments.
- Geophysics—Thompson and others (2007, p. 3) use *lmomco* to compute at least the L-moments. Although distributions are used and in particular the Gumbel is used, the authors do not seem to identify the algorithmic source for their distributional support.
- Hydrology—Many hydrologic articles and reports on water resources with emphasis towards regionalization of floods or streamflow exist for which reference to *lmomco* is made (Cobo and Verbist, 2010; Neykov and others, 2007; Rustomji, 2009, 2010; Rustomji and others, 2009; Roudier and Mahe, 2009; van Nooijen and Kolechkina, 2010a,b). In particular, Rustomji and others (2009) credit *lmomco* in their Acknowledgements, "Statistical analyses were undertaken using the 'lmomco' package."
- Meteorology—Morgan and others (2011) use *lmomco* support of the Wakeby distribution in research into the distributions of offshore wind speed.

The author personally thanks these investigators and researchers for crediting the *lmomco* package.

Extensions of L-moments and the *lmomco* Package

It is natural for this epilogue to consider or suggest extensions to L-moment theory, in general, and the *lmomco* package, in particular. Extension of the L-comoments into left- and right-tail censored multivariate data would be fascinating. The author has dabbled in parameter estimation of copulas (bivariate versions) using the L-comoments and promise is shown for the particular problem of fitting copulas that model particularly asymmetric multivariate probabilities. Extension of the L-moments to simultaneous left- and right-censoring by indicator variable might be useful in some disciplines.

The *lmomco* package is an open-ended library available to the global community. The author would like to see audits and enhancements to the *lmomco* user's manual (Asquith, 2011) to bring the manual, as needed, into concordance with this dissertation. Invitation is extended to readers to communicate with the author suggestions and contributions in pursuit of continual enhancement and extension of both documents (or their derivatives). Specific enhancements to *lmomco* are now identified.

The author would like additional distributions, such as the L-moments of the **asymmetric exponential power distribution** by Delicado and Goria (2008), added to *lmomco* as well as the distributions of Section 8.3. Another example is the **truncated exponential distribution** that is considered by Vogel and others (2008) in the application of L-moments for distributional analysis and goodness-of-fit of species extinction time based on sightings. Other distributions most certainly exist in which the L-moments (or equivalently the probability-weighted moments) have been or will be derived in the future. It would be exciting to have these added to *lmomco*.

The *lmomco* package would be substantially enhanced by the inclusion of the A- and B-type probability-weighted moments and the A′- and B′-type probability-weighted moments for more distributions than just the Reverse Gumbel as currently (May 2011) implemented. Hosking (1995) provides the A- for the Weibull distribution and B- for the Generalized Pareto and Gamma distributions. These three apparently are not yet implemented in any R package.

As of May 2011, the *lmomco* package has a heavily procedural language structure with hints of object-oriented design. Perhaps one day, a fully object-oriented version will emerge and new features and flexibility will result. An object-oriented code base might facilitate the adaption of quantile function algebra and parameter estimation into the package. The incorporation of quantile function algebra would facilitate the construction of even more complex distributions than those shown in this dissertation.

References

The author refrains from providing a padded bibliography in the age of readily available topical and bibliographical searches on the Internet. Bibliographic entries with a right justified † on the terminal line are those which were not acquired by the author, although abstracts or summaries may have been consulted, prior to completion of this dissertation. Their citation in the text is sourced by one or more other entries. In most cases, the †-entry should be understood to have historical significance to L-moments or provide additional topical context for which citation might be useful to some readers. All other entries have been acquired and reviewed at various times by the author during the period 1995–2011.

Adler, D., 2005, *vioplot*—Violin plot: R package version 0.2, dated October 29, 2005, initial package release July 16, 2004, `http://www.cran.r-project.org/package=vioplot`.

Adler, J., 2010, R in a nutshell: Sebastopol, California, Florida, O'Reilly Media, Inc., ISBN 978–0–596–80170–0, 611 p.

Aho, K., 2010, *asbio*—A collection of statistical tools for biologists: R package version 0.3-28, dated December 09, 2010, initial package release July 16, 2004, `http://www.cran.r-project.org/package=asbio`.

Alkasasbeh, M.R., and Raqab, M.Z., 2008, Estimation of the generalized logistic distribution parameters—Comparative study: Statistical Methodology, v. 6, pp. 262–279.

Ahmad, U.M., Shabri, A., and Zakaria, Z.A., 2011, Flood frequency analysis of annual maximum stream flows using L-moments and TL-moments: Applied Mathematical Sciences, v. 5, no. 5, pp. 243–253.

Asquith, W.H., 1998, Depth-duration frequency of precipitation for Texas: U.S. Geological Survey Water-Resources Investigations Report 98–4044, 107 p.

Asquith, W.H., 2001, Effects of regulation on L-moments of annual peak streamflow in Texas: U.S. Geological Survey Water-Resources Investigations Report 01–4243, 66 p.

Asquith, W.H., 2003, Modeling of runoff-producing rainfall hyetographs in Texas using L-moment statistics: Ph.D. dissertation, Jackson School of Geosciences, University of Texas at Austin, 386 p.

Asquith, W.H., 2007, L-moments and TL-moments of the generalized lambda distribution: Computational Statistics and Data Analysis, v. 51, no. 9, pp. 4484–4496.

Asquith, W.H., 2011, *lmomco*—L-moments, censored L-moments, trimmed L-moments, L-comoments, and many distributions: R package version 1.3.4, dated April 15, 2011, initial package release January 31, 2006, `http://www.cran.r-project.org/package=lmomco`.

Asquith, W.H., and Roussel, M.C., 2004, Atlas of depth-duration frequency of precipitation annual maxima for Texas: U.S. Geological Survey Scientific Investigations Report 2004–5041, 106 p.

Asquith, W.H., Roussel, M.C., Cleveland, T.G., Fang, X., and Thompson, D.B., 2006, Statistical characteristics of storm interevent time, depth, and duration for eastern New Mexico, Oklahoma, and Texas: U.S. Geological Survey Professional Paper 1725, ISBN 1–411–31041–1, 299 p.

Asquith, W.H., and Roussel, M.C., 2007, An initial-abstraction, constant-loss model for unit hydrograph modeling for applicable watersheds in Texas: U.S. Geological Survey Scientific Investigations Report 2007–5243, 82 p.

Asquith, W.H., and Roussel, M.C., 2009, Regression equations for estimation of annual peak-streamflow frequency for undeveloped watersheds in Texas using an L-moment-based, press-minimized, residual-adjusted approach: U.S. Geological Survey Scientific Investigations Report 2009–5087, 48 p.

Baclawski, K., 2008, Introduction to probability with R: Boca Raton, Florida, Chapman and Hall/CRC, ISBN 978–1–4200–6521–3, 363 p.

Barnett, V., 2004, Environmental statistics—Methods and applications: Chichester, West Sussex, England, John Wiley, ISBN 0–471–48971–9, 293 p.

Barnett, V., and Lewis, T., 1995, Outliers in statistical data, 3rd ed.: New York, John Wiley, ISBN 0–471–93094–6, 584 p.

Benson, C.H., 1993, Probability distributions for hydraulic conductivity of compacted soil liners: Journal of Geotechnical Engineering, v. 119, no. 3, pp. 471–486.

Braun, W.J., and Murdoch, D.J., 2007, A first course in statistical programming with R: Cambridge, Cambridge University Press, ISBN 978–0–521–87265, 163 p.

Chambers, J.M., Cleveland, W.S., Kleiner, B., and Tukey, P.A., 1983, Graphical methods for data analysis: Pacific Grove, California, Wadsworth and Brooks/Cole, ISBN 0–87150–413–8, 395 p. †

Balakrishnan, N., and Chen, G., 1995, The infeasibility of probability-weighted moments estimation of some generalized distribution: *in* Balakrishnan, N. (ed.) Recent Advances in Life-Testing and Reliability: Boca Raton, Florida, CRC Press, ISBN 0–8493–8972–0, pp. 565–573.

Clarke, R.T., and Terrazas, L.E.M., 1990, The use of L-moments for regionalizing flow records in the Rio Uruguai basin—A case study: Regionalization in Hydrology, International Association of Hydrologic Sciences (IAHS), Publication no. 191, pp. 179–185.

Cobo, J.N., and Verbist, K., 2010, Guía metodológica para la aplicación del Análisis Regional de Frecuencia de Sequías basado en L-momentos y resultados de aplicación en América Latina: Programa Hidrológico Internacional (PHI) de la Oficina Regional de Ciencia para América Latina y el Caribe de la Organización de las naciones Unidas para la Educación, la Ciencia y la Cultura (UNESCO), Documentos Técnios del PHI-LAC, no. 27, Montevideo, Uruguay, ISBN 978–92–9089–157–4, 77 p.

Cohen, Y., and Cohen, J.Y., 2008, Statistics and data in R—An applied approach through examples: New York, John Wiley, ISBN 978–0–470–75805–2, 599 p.

Cunnane, C., 1989, Statistical distributions for flood frequency analysis: World Meteorological Organization Operational Hydrology Report No. 33, Geneva, Secretariat of the World Meteorological Organization.

Dalen, J., 1987, Algebraic bounds on standardized sample moments: Statistics and Probability Letters, v. 5, pp. 329–331.

Dalgaard, P., 2002, Introductory statistics with R: New York, Springer, ISBN 0–387–95475–9, 267 p.

David, H.A., 1981, Order statistics, 2nd ed.: New York, John Wiley, ISBN 0–471–02723–5, 360 p.

Delicado, P., and Goria, M.N., 2008, A small sample comparison of maximum likelihood, moments and L-moments methods for the asymmetric exponential power distribution: Computational Statistics and Data Analysis, v. 52, no. 3, pp. 1661–1673.

Ding, J., and Yang, R., 1988, The determination of probability weighted moments with the incorporation of extraordinary values into sample data and their application to estimation parameters for the Pearson type three distribution: Journal of Hydrology, v. 101, pp. 63–81. †

Dingman S.L., 2002, Physical hydrology, 2nd ed.: Upper Saddle River, New Jersey, Prentice-Hall, ISBN 0–13–099695–5, 646 p.

Delicado, P., and Goria, M.N., 2008, A small sample comparison of maximum likelihood, moments, and L-moments methods for the asymmetric exponential power distribution: Computational Statistics and Data Analysis, v. 52, pp. 1661–1673.

Dupuis, D.J., and Winchester, C., 2001, More on the four-parameter kappa distribution: Journal of Statistical Computation and Simulation, v. 71, no. 2, pp. 99–113.

Durrans, S.R., 1992, Distributions of fractional order statistics in hydrology: Water Resources Research, v. 28, no. 6, pp. 1649–1655.

Efron, B., 1988, Logistic regression, survival analysis, and the Kaplan-Meier curve: Journal of the American Statistical Association, v. 83, no. 402, pp. 414–425. †

Elamir, E.A.H, and Seheult, A.H., 2003, Trimmed L-moments: Computational Statistics and Data Analysis, v. 43, pp. 299–314.

Elamir, E.A.H, and Seheult, A.H., 2004, Exact variance structure of sample L-moments: Journal of Statistical Planning and Inference, v. 124, pp. 337–359.

Evans, M., Hastings, N.A.J., and Peacock, B.J., 2000, Statistical distributions, 3rd ed.: New York, John Wiley, ISBN 0–471–37124–6, 221 p.

Everitt, B.S., 2005, An R and S-PLUS companion to multivariate analysis: London, Springer, ISBN 1–85233–882–2, 221 p.

Everitt, B.S., and Hothorn, T., 2005, A handbook of statistical analyses using R: Boca Raton, Florida, Chapman and Hall/CRC, ISBN 1–58488–539–4, 275 p.

Faraway, J.J., 2005, Linear models with R: Boca Raton, Florida, Chapman and Hall/CRC, ISBN 1–58488–425–8, 229 p.

Faraway, J.J., 2006, Extending the linear model with R—Generalized linear, mixed effects and nonparametric regression models: Boca Raton, Florida, Chapman and Hall/CRC, ISBN 1–58488–424–X, 301 p.

Gilchrist, W.G., 2000, Statistical modelling with quantile functions: Boca Raton, Florida, Chapman and Hall/CRC, ISBN 1–58488–174–7, 320 p.

Gilleland, E., Katz, R., and Young, G., 2010, extRemes—Extreme value toolkit: R package version 1.62, dated April 25, 2010, initial package release March 19, 2004, `http://www.cran.r-project.org/package=extRemes`.

Gini, C., 1912, Variabilità e mutabilità, contributo allo studio delle distribuzioni e delle relazione statistiche: Studi Economico-Giuridici della Reale Università di Cagliari, v. 3, pp. 3–159. †

Greenwood, J.A., Landwehr, J.M., Matalas, N.C., and Wallis, J.R., 1979, Probability weighted moments—Definition and relation to parameters of several distributions expressible in inverse form: Water Resources Research, v. 15, pp. 1049–1054.

Guttman, N.B., 1993, The use of L-moments in the determination of regional precipitation climates: Journal of Climate, v. 6, no. 12, pp. 2309–2325.

Guttman, N.B., 1994, On the sensitivity of sample L moments to sample size: Journal of Climate, v. 7, pp. 1026–1029.

Guttman, N.B., Hosking, J.R.M., and Wallis, J.R., 1993, Regional precipitation quantile values for the continental United States computed from L-moments: Journal of Climate, v. 6, no. 12, pp. 2309–2325.

Haktanir, T., 1997, Self-determined probability-weighted moments method and its application to various distributions: Journal of Hydrology, v. 194, pp. 180–200.

Hald, A., 1998, A history of mathematical statistics from 1750 to 1930: New York, John Wiley, ISBN 0–471–17912–4, 795 p.

Hamada, M., 1995, Analysis of experiments for reliability improvement and robust reliability: *in* Balakrishnan, N. (ed.) Recent Advances in Life-Testing and Reliability: Boca Raton, Florida, CRC Press, ISBN 0–8493–8972–0, pp. 155–172.

Hansen, W.R., 1991, Suggestions to authors of the reports of the United States Geological Survey: U.S. Government Printing Office, Washington, D.C., 7th ed., 289 p.

Heiberger, R.M., and Holland, B., 2005, Statistical analysis and data display—An intermediate course with examples in S-Plus, R, and SAS: New York, Springer, ISBN 0–387–40270–5, 729 p.

Helmert, F.R., 1876, Die Berechnung des wahrscheinlichen Beobachtungsfehlers aus den ersten Potenzen der Differenzen gleichgenauer director Beobachtungen: Astronomische Nachrichten, v. 88, pp. 127–132. [*citation by David (1981), but Wiley Interscience reports:* Helmert, 1876, Die Genauigkeit der Formel von Peters zur Berechnung des wahrscheinlichen Beobachtungsfehlers directer Beobachtungen gleicher Genauigkeit: Astronomische Nachrichten, v. 88, no. 8–9, pp. 113–131.] †

Helsel, D.R., 2005, Nondetects and data analysis—Statistics for censored environmental data: Hoboken, New Jersey, John Wiley, ISBN 0–471–67173–8, 250 p.

Helsel, D.R., and Hirsch, R.M., 1992, Statistical methods in water resources: New York, Elsevier, ISBN 0–444–88528–5, 529 p.

Helsel, D.R., and Hirsch, R.M., 2002, Statistical methods in water resources: U.S. Geological Survey Techniques of Water-Resources Investigations, book 4, chap. A3, 510 p., http://pubs.usgs.gov/twri/twri4a3/

Hershfield, D.B., 1961, Rainfall frequency atlas of the United States for durations from 30 minutes to 24 hours and return periods from 1 to 100 years: Washington, D.C., U.S. Weather Bureau Technical Paper 40, 61 p.

Hollander, M., and Wolfe, D.A., 1973, Nonparametric statistics: New York, John Wiley, ISBN 0–471–40635–X, 503 p.

Hosking, J.R.M., 1985, Maximum-likelihood estimation of the parameters of the generalized extreme-value distribution: Applied Statistics, v. 34, pp. 301–10.

Hosking, J.R.M., 1986, The theory of probability weighted moments: Research Report RC12210, IBM Research Division, Yorktown Heights, New York, reissued with corrections April 1989, 160 p.

Hosking, J.R.M., 1990, L-moments—Analysis and estimation of distributions using linear combinations or order statistics: Journal of Royal Statistical Society, series B, v. 52, no. 1, pp. 105–124.

Hosking, J.R.M., 1992, Moments or L moments?—An example comparing two measures of distributional shape: American Statistician, v. 46, no. 3, pp. 186–189.

Hosking, J.R.M., 1994, The four-parameter kappa distribution: IBM Journal of Research and Development, Yorktown Heights, New York, v. 38, pp. 251–258.

Hosking, J.R.M., 1995, The use of L-moments in the analysis of censored data: *in* Balakrishnan, N. (ed.) Recent Advances in Life-Testing and Reliability: Boca Raton, Florida, CRC Press, ISBN 0–8493–8972–0, pp. 545–564.

Hosking, J.R.M., 1996a, Some theoretical results concerning L-moments: Research Report RC14492, IBM Research Division, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M., 1996b, FORTRAN routines for use with the method of L-moments, version 3: Research Report RC20525, IBM Research Division, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M., 1998, L-moments, *in* Kotz, S., Read, D.L., (eds.): Encyclopedia of Statistical Sciences, v. 2, John Wiley, New York, pp. 357–362.

Hosking, J.R.M., 1999, L-moments and their applications in the analysis of financial data: Research Report RC21466, IBM Research Division, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M., 2000, Maximum-entropy characterization of the logistic distribution using L-moments: Research Report RC21691, IBM Research Division, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M., 2006, On the characterization of distributions by their L-moments: Journal of Statistical Planning and Inference, v. 136, no. 1, pp. 193–198.

Hosking, J.R.M., 2007a, Distributions with maximum entropy subject to constraints on their L-moments or expected order statistics: Journal of Statistical Planning and Inference, v. 137, no. 9, pp. 2870–2891.

Hosking, J.R.M., 2007b, Some theory and practical uses of trimmed L-moments: Journal of Statistical Planning and Inference, v. 137, no. 9, pp. 3024–3039.

Hosking, J.R.M., 2007c, Supplement to "Distributions with maximum entropy subject to constraints on their L-moments": Research Report RC24177, IBM Research Division, T.J. Watson Research Center, Yorktown Heights, New York.

Hosking, J.R.M., 2009a, *lmom*—L-moments: R package version 1.5, dated November 29, 2009, initial package release July 3, 2008, `http://www.cran.r-project.org/package=lmom`.

Hosking, J.R.M., 2009b, *lmomRFA*—Regional frequency analysis using L-moments: R package version 2.3, dated August 22, 2010, initial package release March 3, 2009, `http://www.cran.r-project.org/package=lmomRFA`.

Hosking, J.R.M., Bonti, G., and Siegel, D., 2000, Beyond the lognormal—Accurate estimation of the frequency of rare events in VaR calculations: Risk, v. 13, no. 5, pp. 59–62.

Hosking, J.R.M., and Wallis, J.R., 1987, Parameter and quantile estimation for the generalized Pareto distribution: Technometrics, v. 29, pp. 339–349.

Hosking, J.R.M., and Wallis, J.R., 1993, Some statistics useful in regional frequency analysis: Water Resources Research, v. 29, no. 2, pp. 271–281.

Hosking, J.R.M., and Wallis, J.R., 1995, A comparison of unbiased and plotting-position estimators of L moments: Water Resources Research, v. 31, no. 8, pp. 2019–2025.

Hosking, J.R.M., and Wallis, J.R., 1997, Regional frequency analysis—An approach based on L-moments: Cambridge, Cambridge University Press, ISBN 0–521–43045–3, 224 p.

Hosking, J.R.M., Wallis, J.R., and Wood, E.F., 1985, Estimation of the generalized extreme-value distribution by the method of probability-weighted moments: Technometrics, v. 27, pp. 251–261.

Houghton, J.C., 1978, Birth of a parent—The Wakeby distribution for modeling flood flows: Water Resources Research, v. 15, pp. 1055–1064.

Huber, P.J., 1981, Robust statistics: New York, John Wiley, ISBN 0–471–41805–6, 308 p.

Hyndman, R.J., and Fan, Y., 1996, Sample quantiles in statistical packages, American Statistician, v. 50, pp. 361–365. †

Jeffrey, A., 2004, Handbook of mathematical formulas and integrals, 3rd ed.: Amsterdam, Elsevier, ISBN 0–12–382256–4, 453 p.

Jensen, J.L., Lake, L.W., Corbett, P.W.M., and Goggin, D.J., 1997, Statistics for petroleum engineers and geoscientists: Upper Saddle River, New Jersey, Prentice Hall, ISBN 0–13–131855–1, 390 p.

Jones, M.C., 2002, Student's simplest distribution: Journal of the Royal Statistical Society, series D (*The Statistician*), v. 51, no. 1, pp. 41–49.

Jones, M.C., 2004, On some expressions for variance, covariance, skewness and L-moments: Journal of Statistical Planning and Inference, v. 126, pp. 97–106.

Jones, M.C., 2009, Kumaraswamy's distribution—A beta-type distribution with some tractability advantages: Statistical Methodology, v. 6, pp. 70–81.

Jurczenko, E.F., Maillet, B.B., and Merlin, P.M., 2008, Efficient frontier for robust higher-order moment portfolio selection: Documents de Travail du Centre d'Economie de la Sorbonne, Centre National de La Recherche Scientifique Working Papers 2008.62, ISSN 1955–611X, 68 p.

Jurečková, J., and Picek, J., 2006, Robust statistical methods with R: Boca Raton, Florida, Chapman and Hall/CRC, ISBN 1–58488–454–1, 197 p.

Kaigh, W.C., and Driscoll, M.F., 1987, Numerical and graphical data summary using O-statistics: American Statistician, v. 41, no. 1, pp. 25–32.

Kampstra, P., 2008a, *beanplot*—Visualization via beanplots: R package version 1.1, dated November 6, 2008, `http://www.cran.r-project.org/package=beanplot`.

Kampstra, P., 2008b, Beanplot—A boxplot alternative for visual comparison of distributions: Journal of Statistical Software, Code Snippets, v. 28, no. 1, pp. 1–9, `http://www.jstatsoft.org/v28/c01/`.

Kaplan, E.L., and Meier, P., 1958, Nonparametric estimation of incomplete observations: Journal of the American Statistical Association, v. 53, pp. 457–481. †

Karian, Z.A., Dudewicz, E.J., 2000, Fitting statistical distribution—The generalized Lambda distribution and generalized bootstrap methods: Boca Raton, Florida, CRC Press, ISBN 1–58488–069–4, 438 p.

Karvanen, J., 2006, Estimation of quantile mixtures via L-moments and Trimmed L-moments: Computational Statistics and Data Analysis, v. 51, no. 2, pp. 947–959.

Karvanen, J., and Nuutinen, A., 2008, Characterizing the generalized lambda distribution by L-moments: Computational Statistics and Data Analysis, v. 52, no. 4, pp. 1971–1983.

Karvanen, J., 2009, *Lmoments*—L-moments and quantile mixtures: R package version 1.1–4, dated January 19, 2011, initial package release October 12, 2005, `http://www.cran.r-project.org/package=Lmoments`.

Katz, R.W., and Parlange, M.B., and Naveau, P., 2002, Statistics of extremes in hydrology: Advances in Water Resources, v. 25, pp. 1287–1304.

Keen, K.J., 2010, Graphics for statistics and data analysis with R: Boca Raton, Florida, Chapman and Hall/CRC, ISBN 978–1–58488–087–5, 447 p.

Kerstens, K., Mounir, A., and Van de Woestyne, I., 2010, Non-parametric frontier estimates of mutual fund performance using C- and L-moments—Some specification tests: IESEG School of Management, Brussels, Belgium, CNRS-LEM (UMR 8179), Document de travail du LEM, 2010-05, 29 p.

Kirby, W., 1974, Algebraic boundedness of sample statistics: Water Resources Research, v. 10, pp. 220–222.

Klemeš, V., 2000a, Tall tales about tails of hydrological distributions I: Journal of Hydrologic Engineering, v. 5, no. 3, pp. 227–231.

Klemeš, V., 2000b, Tall tales about tails of hydrological distributions II: Journal of Hydrologic Engineering, v. 5, no. 3, pp. 232–239.

Kliche, D.V., Smith, P.L., and Johnson, R.W., 2008, L-moment estimators as applied to gamma drop size distributions: Journal of Applied Meteorology and Climatology, v. 47, pp. 3117–3130.

Kottegoda, N.T., and Rosso, R., 2008, Applied statistics for civil and environmental engineers: Blackwell Publishing (acquired by John Wiley), ISBN 978–1–4051–7917–1, 718 p.

Kroll, C.N., and Stedinger, J.R., 1996, Estimation of moments and quantiles using censored data: Water Resources Research, v. 32, no. 4, pp. 1005–1012.

Kumaraswamy, P., 1980, A generalized probability density function for double-bounded random processes: Journal of Hydrology, v. 46, pp. 79–88. †

Landwehr J.M., Matalas, N.C., and Wallis, J.R., 1979a, Estimation of parameters and quantiles of Wakeby distributions: Water Resources Research, v. 15, no. 5, pp. 1362–1379.

Landwehr, J.M., Matalas, N.C., and Wallis, J.R., 1979b, Probability weighted moments compared with some traditional techniques in estimating Gumbel parameters and quantiles: Water Resources Research, v. 15, no. 5, pp. 1055–1064.

Landwehr, J.M., Matalas, N.C., and Wallis, J.R., 1980, Quantile estimation with more or less floodlike distributions: Water Resources Research, v. 16, no. 3, pp. 547–555.

Lee, L., 2009, *NADA*—Nondetects and data analysis for environmental data: R package version 1.5-3, dated December 22, 2010, initial package release June 24, 2004, `http://www.cran.r-project.org/package=NADA`.

Liou, Jun-Jih, Wu, Yii-Chen, Chiang, Jie-Lun, and Cheng, Ke-Sheng, 2007, Assessing power of test for goodness-of-fit test using L-moment-ratios diagram: Journal of Chinese Agricultural Engineering, v. 53, no. 4, pp. 80–91.

Liou, Jun-Jih, Wu, Yii-Chen, and Cheng, Ke-Sheng, 2008, Establishing acceptance regions for L-moments based goodness-of-fit tests by stochastic simulation: Journal of Hydrology, v. 355, pp. 49–62.

Maindonald, J.H., and Braun, J., 2003, Data analysis and graphics using R—An example-based approach: Cambridge, Cambridge University Press, ISBN 0–521–81336–0, 362 p.

Mays, L.W., 2005, Water resources engineering: Hoboken, New Jersey, John Wiley, ISBN 0–471–70524–1, 842 p.

Mercy, J., and Kumaran, M., 2010, Estimation of the generalized lambda distribution from censored data: Brazilian Journal of Probability and Statistics, v. 24, no. 1, pp. 42–56.

Mielke, P.W., 1973, Another family of distributions describing and analyzing precipitation data: Journal of Applied Meteorology, v. 12, no. 1, pp. 275–280.

Morgan, E.C., Lackner, M., Vogel, R.M., and Baise, L.G., 2011, Probability distributions for offshore wind speeds: Energy Conversion and Management, v. 52, no. 1, pp. 15–26.

Murrell, P., 2006, R graphics: Boca Raton, Florida, Chapman and Hall/CRC, ISBN 1–58488–486–X, 301 p.

Nelson, R.B., 2006, An introduction to copulas: New York, Springer, ISBN 0–387–28659–4, 269 p.

Neykov, N.M., Neytchev, P.N., Van Gelder, P.H.A.J.M., Todorov, V.K., 2007, Robust detection of discordant sites in regional frequency analysis: Water Resources Research, v. 43, W06417, 10 p.

Parida, B.P., 1999, Modeling of Indian summer monsoon rainfall using a four-parameter kappa distribution: International Journal of Climatology, v. 19, pp. 1389–1398. †

Park, J.S., and Park, B.J., 2002, Maximum likelihood estimation of the four-parameter kappa distribution using the penalty method: Computers and Geosciences, v. 28, pp. 65–68.

Peel, M., Wang, Q.J., Vogel, R.M., and McMahon, T.A., 2001, The utility of L-moment ratio diagrams for selecting a regional probability distribution: Hydrological Sciences, v. 46, no. 1, pp. 147–155.

Qian, S.S., 2010, Environmental and ecological statistics with R: Boca Raton, Florida, Chapman and Hall/CRC, ISBN 978–1–4200–6206–9, 421 p.

R Development Core Team, 2009, Writing R Extensions: R Foundation for Statistical Computing, Vienna, Austria, version 2.10.1 (2009-12-14), `http://www.R-project.org`, PDF file `R-exts.pdf`.

R Development Core Team, 2010, R—A language and environment for statistical computing: R Foundation for Statistical Computing, Vienna, Austria, version 2.12.2 (2010-12-16), ISBN 3–900051–07–0, `http://www.R-project.org`.

Raju, B.I., and Srinivasan, M.A., 2002, Statistics of envelope of high-frequency ultrasonic backscatter from human skin *in vivo*: IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control, v. 49, no. 7, pp. 871–882.

Reimann, C., Filzmoser, P., Garrett, R., and Dutter, R., 2008, Statistical data analysis explained—Applied environmental statistics with R: West Sussex, England, John Wiley, ISBN 978–0–470–98581–6, 343 p.

Ribatet, M., 2009, *POT*—Generalized Pareto distribution and peaks over threshold: R package version 1.1–0, dated October 16, 2009, initial package release September 6, 2005, `http://www.cran.r-project.org/package=POT`.

Ribatet, M., 2010, *RFA*—Regional Frequency Analysis: R package version 0.0–9, dated January 14, 2010, initial package release September 14, 2005, `http://www.cran.r-project.org/package=RFA`.

Rinne, H., 2008, The Weibull distribution—A handbook: Boca Raton, Florida, Chapman and Hall/CRC, ISBN 978–1–42008–743–7, 808 p. †

Rizzo, M.L., 2008, Statistical computing with R: Boca Raton, Florida, Chapman and Hall/CRC, ISBN 978–1–58488–545–0, 399 p.

Ross, S., 1994, A first course in probability, 4th ed.: New York, MacMillan College Publishing Company, ISBN 0–02–403872–5, 473 p.

Royston, P., 1992, Which measures of skewness and kurtosis are best?: Statistics in Medicine, v. 11, no. 3, pp. 333–343.

Roudier, P., and Mahe, G., 2009, Study of water stress and droughts with indicators using daily data on the Bani river (Niger basin, Mali): International Journal of Climatology, v. 30, no. 11, pp. 1689–1705.

Rustomji, P., 2009, A statistical analysis of flood hydrology and bankfull discharge for the Daly River catchment, Northern Territory, Australia: CSIRO Water for a Healthy Country National Research Flagship (09/2009), 59 p.

Rustomji, P., 2010, A statistical analysis of flood hydrology and bankfull discharge for the Mitchell River catchment, Queensland, Australia: CSIRO Water for a Healthy Country National Research Flagship (01/2010), 108 p.

Rustomji, P., Bennett, N., and Chiew, F., 2009, Flood variability east of Australia's Great Dividing Range: Journal of Hydrology, v. 374, no. 3–4, pp. 169-208.

Sawitzki, G., 2009, Computational statistics—An introduction to R: Boca Raton, Florida, Chapman and Hall/CRC, ISBN 978–1–4200–8678–2, 251 p.

Sen, P.K., 1964, On some properties of the rank-weighted means: Journal Indian Society of Agricultural Statistics: v. 16, pp. 51–61. †

Serfling, R.J., 1980, Approximation theorems of mathematical statistics: New York, John Wiley, ISBN 0–471–21927–4, 371 p.

Serfling, R.J., and Xiao, P., 2007, A contribution to multivariate L-moments—L-comoment matrices: Journal of Multivariate Analysis, v. 98, pp. 1765–1781.

Schaefer, M.G., 1990, Regional analysis of precipitation annual maxima in Washington State: Water Resources Research, v. 26, no. 1, pp. 119–131.

Short, T., 2004, R reference card: dated November 7, 2004, Granted to the public domain, *see* `http://www.Rpad.org` for the source and latest version, *includes* material from "R for Beginners" by Emmanuel Paradis (with permission), `http://cran.r-project.org/doc/contrib/Short-refcard.pdf`

Sijbers, J., den Dekker, A.J., Scheunders, P., Van Dyck, D., 1998, Maximum likelihood estimation of Rician distribution parameters: IEEE Transactions on Medical Imaging, v. 17, no. 3, pp. 357–361.

Sillitto, G., 1951, Interrelations between certain linear systematic statistics of samples from any continuous population: Biometrika, v. 38, no. 3–4, pp. 377–382. †

Sillitto, G., 1969, Derivation of approximants to the inverse distribution function of a continuous univariate population from the order statistcs of a sample: Biometrika, v. 56, no. 3, pp. 641–650. †

Singh, V.P., and Deng, Z.Q., 2003, Entropy-based parameter estimation for kappa distribution: Journal of Hydrologic Engineering, v. 8, no. 2, pp. 81–92.

Spatz, C., 1996, Basic statistics—Tales of distributions: Pacific Grove, California, Brooks/Cole Publishing Company, ISBN 0–534–26424–7, 488 p.

Spector, P., 2008, Data manipulation with R: New York, Springer, ISBN 978–0–389–74730–9, 152 p.

Stedinger, J.R., Vogel, R.M., and Foufoula-Georgiou, E., 1993, Frequency analysis of extreme events, *in* Handbook of Hydrology, chapter 18, editor-in-chief D.A. Maidment: New York, McGraw-Hill, ISBN 0–07–039732–5.

Su, S., 2010, *GLDEX*—Fitting single and mixture generalized lambda distributions (RS and FMKL) using various methods: R package version 1.0.4.1, dated January 20, 2010, initial package release October 11, 2007, `http://www.cran.r-project.org/package=GLDEX`.

Thomas, R., de la Torre, L., Chang, X., and Mehrotra, S., 2010, Validation and characterization of DNA microarray gene expression data distribution and associated moments: BMC Bioinformatics, v. 11, no. 576, 14 p.

Thompson, E.M., Baise, L.G., and Vogel, R.M., 2007, A global index earthquake approach to probabilistic assessment of extremes: Journal of Geophysical Research, v. 112, B06314, 12 p.

Ugarte, M.D., Militino, A.F., and Arnhold, A.T., 2008, Probability and statistics with R: Boca Raton, Florida, Chapman and Hall/CRC, ISBN 978–1–58488–891–8, 700 p.

Ulrych, T.J., Velis, D.R., Woodbury, A.D., and Sacchi, M.D., 2000, L-moments and C-moments: Stochastic Environmental Research and Risk Assessment, v. 14, pp. 50–68.

Unnikrishnan, N., and Vineshkumar, B., 2010, L-moments of residual life: Journal of Statistical Planning and Inference, v. 140, no. 9, pp. 2618–2631.

U.S. Geological Survey, 2007a, PeakFQ—Flood frequency analysis based on Bulletin 17B: U.S. Geological Survey, Water Resources Applications Software, Reston, VA., http://water.usgs.gov/software/peakfq.html

U.S. Geological Survey, 2007b, SWSTAT—Surface-water statistics: U.S. Geological Survey, Water Resources Applications Software, Reston, VA., http://water.usgs.gov/software/swstat.html

U.S. Water Resources Council, 1981, Guidelines for determining flood flow frequency: U.S. Water Resources Council, Bulletin 17B, Washington, D.C., variously paged.

van Nooijen, R.R.P., and Kolechkina, A.G., 2010a, A comparison of fitting methods and tests for several distributions on computer generated samples: Advances in Statistical Hydrology, International Workshop, May 22–25, 2010, Taormina, Italy, 10 p.

van Nooijen, R.R.P., and Kolechkina, A.G., 2010b, A comparison of fitting methods and tests for several distributions on hydrological data: Advances in Statistical Hydrology, International Workshop, May 22–25, 2010, Taormina, Italy, 9 p.

Venables, W.N., Smith, D.M., and the R Development Core Team, 2008, An introduction to R; Notes on R—A programming environment for data analysis and graphics: version 2.7.2 (2008-08-25), ISBN 3–900051–12–7, 94 p., http://www.R-project.org, PDF file R-intro.pdf

Verzani, J., 2005, Using R for introductory statistics: Boca Raton, Florida, Chapman and Hall/CRC, ISBN 1–58488–4509, 414 p.

Vogel, R.M., 1986, The probability plot correlation coefficient test for the normal, lognormal, and Gumbel distributional hypotheses: Water Resources Research, v. 22, no. 4, pp. 587–590.

Vogel, R.M., 1995, Recent advances and themes in hydrology: Reviews of Geophysics, v. 33, supplement, http://www.agu.org/journals/rg/rg9504S/95RG00935/index.html

Vogel, R.M., Hosking, J.R.M., Elphick, C.S., Roberts, D.L., and Reed, J.M., 2008, Goodness of fit of probability distributions for sightings as species approach extinction: Bulletin of Mathematical Biology, v. 71, no. 3, pp. 701–719.

Vogel, R.M., and Fennessey, N.M., 1993, L moment diagrams should replace product moment diagrams: Water Resources Research, v. 29, no. 6, pp. 1745–1752.

Walker, H.M., 1940, Degrees of freedom: Journal of Educational Psychology, v. 31, pp. 253–269. †

Wallis, J.R., Matalas, N.C., and Slack, J.R., 1974, Just a moment!: Water Resources Research, v. 10, pp. 211–219.

Wallis, J.R., 1988, Catastrophes, computing and containment—Living in our restless habitat: Speculation in Science and Technology, v. 11, no. 4, pp. 295–315.

Wang, Q.J., 1990a, Estimation of the GEV [Generalized Extreme Value] distribution from censored samples by method of partial probability weighted moments: Journal of Hydrology, v. 120, pp. 103–114.

Wang, Q.J., 1990b, Unbiased estimation of probability weighted moments and partial probability weighted moments from systematic and historical flood information and their application to estimating the GEV [Generalized Extreme Value] distribution: Journal of Hydrology, v. 120, pp. 115–124.

Wang, Q.J., 1996a, Using partial probability weighted moments to fit the extreme value distributions to censored samples: Water Resources Research, v. 32, no. 6, pp. 1767–1771.

Wang, Q.J., 1996b, Direct sample estimators of L-moments: Water Resources Research, v. 32, no. 12., pp. 3617–3619.

Wang, D., Hutson, A.D., Miecznikowski, J.C., 2010, L-moment estimation for parametric survival models given censored data: Statistical Methodology, v. 7, no. 6, pp. 655–667.

Weiss, L.L., 1964, Ratio of true to fixed-interval maximum rainfall: American Society of Civil Engineers, Journal of the Hydraulics Division, v. 90, HY–1, pp. 77–82.

Whalen, T.M., Savage, G.T., and Jeong, G.D., 2002, The method of self-determined probability weighted moments revisited: Journal of Hydrology, v. 268, pp. 177–191.

Yatracos, Y.G., 1998, Variance and clustering: Proceedings American Mathematical Society, v. 126, no. 4, pp. 1177–1179. †

Zafirakou-Koulouris, A., Vogel, R.M., Craig, S.M., and Habermeier, J., 1998, L-moment diagrams for censored observations: Water Resources Research, v. 34, no. 5, pp. 1241–1249.

Index

For this topical index, page numbers on which a term is defined or the primary discussion exists are typeset in bold.

Symbols

LaTeX iii, 6
METAPOST xxiv, 376
TeX iii

A

Agency
  Internal Revenue Service 248, 250, 251
  National Weather Service 315
  U.S. Geological Survey xxiii, xxiv, 56, 219, 222, 231, 245, 263–265, 270, 272, 324, 325, 335, 362, 368, 371

B

Baron Münchhausen 152
bean plots 318, **318**, 319
Bessel function 205, **205**, 206
Beta distribution 67, **67**, 69, 196, 244, 255, 349
beta function 64, **64**, 67, 197
  incomplete ratio **244**
  regularized **244**
  regularized incomplete **244**
binomial coefficients **106**, 127
  relation to beta function 64
Binomial distribution 106
binomial distribution 339
biology
  species extinction time 401
bootstrap 152
box plots **18**, 19–21, 24, 51, 58, 316–318, 325, 326

C

Cauchy distribution 116, 139, 144, 165, 167, 168, 177, 183, **183**, 184, 185, 214, 275, 279, 280, 291, 296
censoring 267, 337, 338
  by indicator variable 337, 398
  distributions 13
  Kaplan-Meier 349, 353
  left tail 337, **337**, 338, 343, 344, 346, 348, 352–355, 357, 398
    back flipping 346, 347, 353, **353**, 355, 356
    by indicator variable 354, 400
    censoring fraction **344**
    censoring threshold **344**
    detection limits **344**, 398
    flipping 337, 346, **352**, 353, 354, 356, 398
    L-moments 346, 347
    mean 354
    probability-weighted moments 344, 345, 401
    threshold 343–346, 357
    type I 344, **344**
    type II 344, **344**
  multiple detection limits 348

right tail 190, 191, 239, 337, **337**, 338, 339, 342, 346–350, 352, 353, 355, 398
by indicator variable 348, 350, 351, 400
censoring fraction 191, 192, 239, **339**, 340
censoring threshold **338**
L-moments 346, 347
probability-weighted moments 339–342, 401
threshold 338, 339, 341, 342, 347, 348
type I 191, **338**, 339
type II 191, **338**, 339
City
Amarillo 316
Baraboo 57, 59, 231, 233
Brady 245–247
Brownsville 218, 219
Bruce 325, 335
Canyon 316
Claude 316
Clifton 325
Comfort 219, 221, 222
Corpus Chrisi 370
Corpus Christi 218, 219
Dallas 11
Denver ii
Elkton 325
Hereford 316
Laguna 263–265, 270, 272
Llano 325, 362–364, 371, 373
Lubbock xxiv, xxv
Port Arthur 218, 219
Tulia 316
Vega 316
Waverly 325
coefficient of L-variation *see* L-CV
coefficient of variation 84, **84**, 87, 88, 90–92, 98, 148
sample **85**, 87, 90, 91, 148
Comprehensive R Archive Network *see* CRAN
conditional probability *see* probability, conditional
confidence interval **369**
copula
bivariate 393, 394
copulas 9, 386, 391, **391**, 392, 395, 397, 398, 400
bivariate 391, 392, 395–397, 400
conditional simulation 392
derivative of 392
inverse of derivative 392
Marshall-Olkin 392–397
County
State of Texas
Ector 54
Travis 224
CRAN 4, 7, 10, 11
cumulative distribution function 24, 25, **25**, 26–30, 32, 43, 44, 48, 59, 64–67, 101, 118, 125, 126, 140, 159–162, 165, 170, 173, 174, 205, 208–211, 214, 215, 220, 221, 224, 225, 229, 243, 244, 248, 249, 255–257, 339, 344, 349, 358
complementary *see* survival function
empirical **52**
cumulative percentile *see* probability, nonexceedance
cumulative probability *see* probability, nonexceedance
cumulative probability function *see* cumulative distribution function

D

degrees of freedom 88
distribution
blipped 358, **358**, 360, 376
modeling **338**, 398
bounds 24
continuous 158
lower bound 358, 359
marginal 388
parameters 79
distribution metric or parameter
location 41, 45, **45**, 47, 49, 62, 63, 68, 70, 71, 83, 87, 118, 144, 172, 173, 176, 178, 183, 187, 192, 200, 201, 204, 215, 217, 222, 228, 236, 239, 240, 243, 247, 252, 253, 255, 261, 268, 279–281, 299, 300, 317
scale 29, 41, **45**, 47, 49, 63, 68, 71, 73, 74, 87, 91, 111, 118, 138, 148, 172, 173, 176, 179, 183, 187, 192, 196, 200, 204, 216, 217, 221,

222, 228, 236, 239, 240, 243, 247, 252, 253, 255, 259, 261, 268, 279–281, 299, 300
shape 25, 86, 91, 96, 104, 111, 113, 114, 138, 148, 173, 177, 179, 200, 201, 204, 216, 217, 221, 222, 228, 236, 239, 240, 243, 247, 252, 253, 255, 259, 261, 268, 280, 281, 300, 302, 313, 330
distribution type
beta *see* Beta distribution
Cauchy *see* Cauchy distribution
censored
right-censored generalized Pareto *see* Right-Censored Generalized Pareto distribution
right-censored reverse Gumbel *see* Reverse Gumbel distribution
exponential *see* Exponential distribution
extreme value type I *see* Gumbel distribution
extreme value type II *see* Generalized Extreme Value distribution
extreme value type III *see* Generalized Extreme Value distribution
Fréchet *see* Generalized Extreme Value distribution
gamma *see* Gamma distribution
generalized extreme value *see* Generalized Extreme Value distribution
generalized lambda *see* Generalized Lambda distribution
generalized log-logistic *see* Generalized Logistic distribution
generalized logistic *see* Generalized Logistic distribution
generalized normal *see* Generalized Normal distribution
generalized Pareto *see* Generalized Pareto distribution
geometric *see* geometric distribution
Gumbel *see* Gumbel distribution
kappa *see* Kappa distribution
log-normal *see* Generalized Normal distribution and log-Normal distribution
three parameter *see* log-Normal3
log-Pearson type III *see* log-Pearson Type III distribution
normal *see* Normal distribution
Pearson type III *see* Pearson Type III distribution
polynomial density-quantile3 *see* Polynomial Density-Quantile3 distribution
polynomial density-quantile4 *see* Polynomial Density-Quantile4 distribution
Rayleigh *see* Rayleigh distribution
reverse Gumbel *see* Reverse Gumbel distribution
reversed generalized extreme value *see* Weibull distribution
Rice *see* Rice distribution
Rician *see* Rice distribution
right-censored reverse Gumbel *see* Reverse Gumbel distribution
shifted log-logistic *see* Generalized Logistic distribution
Student t (3-parameter) *see* Student t (3-parameter) distribution
trimmed
generalized lambda *see* Trimmed Generalized Lambda distribution
generalized Pareto *see* Trimmed Generalized Pareto distribution
uniform *see* Uniform distribution
Wakeby *see* Wakeby distribution
Weibull *see* Weibull distribution
drainage infrastructure 331
bridges 56, 324, 335
culverts 315
levees 2, 30
parking lots 315
storm drains 315
drought 16, 245

E

ecological data 8
empirical distribution 50, 51, 53, 55, 58, 60, 151, 170, 222, 232, 245, 249–251, 263, 264, 270, 271, 321, 326, 330, 331, 334, 362
cumulative distribution function **52**
survival
left-tail censoring 355
engineering

civil 42
earthquake 24, 42, 48
structural 42
entropy
method of 262
principle of maximum 114
environmental data 8
arsenic 353, 354, 356
Eulerian 133, 134
exceedance probability *see* probability, exceedance
exploratory data analysis 1, 2
Exponential distribution 29–31, 34–37, 65, 68, 116, 135, 143, 160, 162, 165, 167, 168, 173, 175, **175**, 176–180, 182, 214, 238, 248, 257, 300, 349
asymmetric exponential distribution **401**
stretched exponential distribution **176**
truncated exponential distribution **401**
exponential integral 192, **192**
Extreme Value Type I distribution **186**, 370, *see* Gumbel distribution
Extreme Value Type II distribution **217**, 370, *see* Generalized Extreme Value distribution
Extreme Value Type III distribution *see* Generalized Extreme Value distribution

F

factorial 64
as defined by complete gamma function 64
failure analysis 348, 352
failure rate function **28**
FORTRAN ii, xxiii–xxv, 127, 172, 324
Fréchet distribution **217**, 370, *see* Generalized Extreme Value distribution
Free Software Foundation 3
FreeBSD 3

G

Gamma distribution 31, 91, 93, 109–111, 114, 116, 138, 148, 150, 158, 160, 162, 165, 167, 168, 172, 173, 175, 179, **179**, 180–183, 201–204, 212, 214, 215, 300, 303–305, 308, 401
gamma function
complete 64, 65, 85, 109, 180, 218, 243, 244, **244**, 245, 262
relation to factorial 64
incomplete 243, **243**
Generalized Extreme Value distribution 101, 104, 107, 116, 125, 129, 159, 160, 165, 167, 168, 170, 172, 187, 217, **217**, 218–228, 230, 235, 248–251, 256–258, 260, 261, 280, 300, 302, 306–308, 323, 349, 370
Generalized Lambda distribution xxv, 10, 116, 139, 165, 167, 168, 174, 189, 238, 260, 261, 267, **267**, 268–276, 278–281, 291, 295, 296, 300, 309, 327, 331, 333–336
censored 267
trimmed *see* Trimmed Generalized Lambda distribution
Generalized Logistic distribution 104, 114, 116, 160, 165, 167–169, 221, **221**, 222–228, 253, 256, 261, 262, 266, 267, 272, 300, 302–305, 307–309, 313, 320, 321, 327
Generalized Normal distribution 116, 159, 160, 163, 165, 167, 168, 227, **227**, 228–236, 256, 257, 261, 264, 280, 296, 307, 308, 354–357
Generalized Pareto distribution 10, 24, 39, 41, 69, 104, 116, 142, 160, 165, 167, 168, 213, 235, **235**, 236, 238–240, 257, 261, 270, 274, 281, 285–290, 292, 293, 297, 300, 302, 308, 342, 343, 359–361, 366, 372, 401
blipped example 360
lower bound 359
geology
earthquake 16, 24, 42, 48
geophysics 42
Permian
Clear Fork formation 54
volcanology 42
Geometric distribution 43, 44
geophysics 16
Gini mean difference 61, 71, **73**, 74–77, 118
object of 74
GNU iii, 3
Google 205

Gumbel distribution 52, 116, 160, 165, 167, 168, 170–172, 186, **186**, 187–194, 215, 217, 261, 306, 308, 370, 400
Gumbel Reduced Variate 371, **371**

H

hazard function 28, **28**, 59
histograms 18, **18**, 19, 21, 22
hydraulic conductivity 230
hydrologic data 8
  common condition of positive skew 244
hydrology 16, 30, 42, 97, 99, 374, 375
hydrometeorological data 93, 97, 314, 320, 336

I

incomplete Beta function ratio 255
Internal Revenue Service 248, 250, 251
Internet 4, 7, 11, 15
interquartile range *see* range, interquartile
inverse distribution function *see* quantile function

J

Jacobi polynomials
  shifted **140**

K

Kappa distribution ii, 116, 158, 160, 165, 167, 168, 260, **260**, 261–264, 266, 267, 270–272, 280, 281, 292, 294–296, 300, 302, 310, 311, 313, 320, 321, 323, 326, 327, 336, 351, 352
Kendall's Tau 120, 387, 390
Kohlrausch function *see* Exponential distribution
Kumaraswamy distribution 116, 121, 158, 165, 167, 168, 196, **196**, 197–199, 215, 291
kurtosis 10, 79, 84, **84**, 98, 113
  sample 86, **86**
  unbounded unlike L-kurtosis 121
kurtosis, concept of 63, 118, 121, 146, 259, 299, 302

L

L-comoments xxv, 9, 10, 337, 338, 386, **386**, 387–392, 394–398, 400
  as concomitants 386
  censoring
    a vision 400
  feature of asymmetry 387
  matrices 389–391, 396
  method of **397**
  random variable co-movement
    symmetric 391
  ratios 387
    L-cokurtosis **387**
    L-correlation **387**, 389, 393
    L-coskew **387**, 390, 391, 393, 395, 396
L-CV **120**, 124
  relative L-variation 148
  sample **126**
L-estimators 70, **70**, 71, 73, 75
  type I 70
  type II 70
L-kurtosis **120**, 157, 213, 307, 327
  hyper 272, 326
  in L-moment ratio diagram 213, 301, 306, 307, 309, 312
  more bounded than kurtosis 121
  sample **126**
L-moment ratio diagrams 13, 184, 212, 213, 215, 298, 299, **299**, 300, 302–305, 308, 310, 311, 313, 314, 321, 322, 327, 328, 336, 338, 349, 372, 376, 400
L-moment ratios 120, 121, 123, 126, 128, 140, 141, 200, 298, 372
  boundedness 157
  bounds of 120
  sample **126**, 145
  theoretical **120**
L-moments ii, iii, xxii–xxv, 4, 8, 9, **9**, 10–14, 16, 23, 46, 55, 58, 61–63, 65, 71, 74–79, 81, 86–88, 92, 97–100, 103, 105, 108, 110–115, 117–125, 127–143, 145, 146, 148, 149, 152, 156–159, 161, 164–167, 169–171, 174–181, 183, 184, 186, 187, 191–194, 196, 197, 200, 201, 206, 214–218, 220, 223–231, 233, 235–238, 241, 243–245, 249, 251–263, 267–272, 278, 280–284, 286, 291, 293–296, 298–300, 303, 306, 308, 311, 313–316, 318, 320, 321, 323–326, 333, 335, 337, 338, 340–342, 346–349, 351, 353–357, 359, 362, 367, 368,

372, 374–378, 381, 383–386, 390, 394, 395, 398–402
A′-type **346**
A-type **341**
as derivatives of quantile functions 118
as statistics of jumps 134
B′-type **346**
B-type 191, 192, **341**
back flipped 356
books concerning 9
censored xxv, 10, 351
in terms of probability-weighted moments 121, 123
method of 135, **135**, 136, 138, 164, 171, 206, 214, 222, 233, 245, 262–264, 266, 270, 313, 318, 325, 349, 374
multivariate 386, **386**, 398, *see* L-comoments
ratios of *see* L-moment ratios
right-censored 115
sample 55, 74, 75, 91, 114–116, 119, 124–126, **126**, 127, 129, 130, 132, 135, 138, 145, 157, 169, 171, 177, 182, 213, 214, 219, 231–233, 248, 259, 301, 313, 318–320, 324, 325, 327, 336, 348, 351, 360, 362, 364, 368, 371, 376, 378, 381, 384–386
as weighted-linear combinations **130**
unbiased estimator 124
theoretical 114, 117, **117**
theory of **8**, 9, 400
trimmed *see* TL-moments
with statistical literature 113
L-scale 74, 75, **118**, 157, 171, 214, 374
L-skew **120**, 157, 214, 307, 374, 378
in L-moment ratio diagram 213, 301, 306, 307, 309, 312
more bounded than skew 121
sample **126**
Lagrangian 133, 134
Laguerre polynomial 206, **206**, 209, 211, 215
landfill 230
least-squares 189, 222, 271, 331
Legendre polynomial
rth-shifted **117**
lifetime analysis 1, 338, 346, 398
linear interpolation 52
Linux iii, xxiv, 3
lmomco list
L-moment **127**, 128, 142, 143, 164, 224, 238
parameter 69, 102, 123, 142, 143, 148, 163, **163**, 164, 166, 170, 177, 193, 224, 248, 279, 363
probability-weighted moment 108, **108**, 111, 142, 164
product moment **90**
TL-moment **143**
log-logistic distribution *see* Generalized Logistic distribution, **221**
log-Normal distribution 2, 57–59, 97, 150, 151, 153–156, **174**, 181, 227–229, 232–235, 257, 264, 296, 357
log-Normal3 distribution 116, 159, 160, 165, 167, 168, 229, **229**, 230, 235, 236, 257, 291
log-Pearson Type III distribution 97, **245**, 375, 381–386, 398
logarithmic transformation 57, 97, 98, 156, 157, 181, 191, 228, 229, 232, 233, 363, 374–376, 381, 385, 386

M

MacOSX iii, 3
Marcum Q function 205, **205**, 207, 208, 215
Matlab 11
maximum 9, 21, 22, 44, 45, 47, 49, 58, 61, **62**, 66–68, 77, 115, 318, 352
maximum likelihood 179, 262
method of 2, 222, 262, 266, 349, 399
mean 9, 14, 20, 22, 27, 33, 41, 43, 44, 48, 51, 60, 65, 67, 68, 71, 79, 82, 83, **83**, 85, 87, 88, 92, 98, 101, 102, 129, 134, 135, 137, 144, 157, 166, 171, 173, 202, 204, 206, 210, 211, 213, 215, 216, 221, 229, 243, 256, 259, 278, 304, 308, 349, 353, 356, 374, 376, 385, 389
back flipped 356
interpretation of 65
left-tail censored 354
sample **84**
Sen *see* Sen weighted mean
trimmed 73, 83, 141
weighted 315, 316, 319, 320, 322, 336
mean square error 72, 73, **80**, 83
measures of association **387**, 390

measures of concordance **387**
Kendall's Tau *see* Kendall's Tau
Spearman's Rho *see* Spearman's Rho
median 9, 14, 21, 22, 29, 34, 41, 44, 45, 61, **62**, 63, 69–73, 77, 82, 83, 137, 167, 178, 189, 211, 318, 353
meteorology 16, 42
drought *see* drought
frost-free days 42
rainfall *see* rainfall
temperature 19, 21, 24
coldest daily 42
wind *see* wind
method of L-moments *see* L-moments, method of
method of moments *see* product moments, method of
method of percentiles 189, **189**, 190, 215, 222
method of probability-weighted moments *see* probability-weighted moments, method of
minimum 9, 21, 22, 44, 45, 47, 61, **62**, 67, 68, 77, 210, 318
mixed distribution 358
blipped *see* blipped distribution
mode 180, **180**, 197, 201–203, 215
antimode 197
multivariate L-moments *see* L-comoments

N

National Weather Service 315
nonexceedance probability *see* probability, nonexceedance
Normal distribution 15, 27, 28, 34, 45, 48–50, 52, 57–59, 72, 75, 83, 94, 97, 102, 106, 107, 114, 116, 123, 126, 145, 147, 148, 156, 158, 160, 162–168, 173, **173**, 174, 175, 177, 210–215, 227, 229, 232, 243, 244, 254, 257, 260, 300
standard 27, 41, 45, 46, 53, 73, 81, 102, 125, 146, 147, 166, 174, **174**, 254, 272, 275, 283, 285, 286, 288–290, 296, 297, 387, 388

O

O-statistics 63
order statistics 9, 10, 12, 51, 52, 61, **61**, 62–67, 69–71, 73, 74, 77, 100, 106, 112, 115, 117, 122, 126, 132, 139, 141, 157, 339, 344, 348, 349, 386
expectation of 63, **63**, 65, 77, 117, 141
extreme value 62, 65, 122, 144
quantile function of 67
interpretations of 62
linear combinations of 117
maxima 58, 122, 129, 130
maximum *see* maximum
median *see* median
minimum *see* minimum
quartile *see* quartile
sample 30, **61**, 85, 103, 126, 131, 143, 348

P

parameter estimation
entropy *see* entropy, method of
L-moments *see* L-moments, method of
maximum likelihood *see* maximum likelihood, method of
percentiles *see* method of percentiles
probability-weighted moments *see* probability-weighted moments, method of
product moments *see* product moments, method of
TL-moments *see* TL-moments, method of
parent distribution 2, **2**, 18, 96, 159, 175, 302, 313, 315, 317, 321, 336, 361, 365, 366
PeakFQ 375
Pearson Type III distribution 32, 33, 94–96, 116, 145, 160, 165, 167, 168, 216, 242, **242**, 243–245, 257, 280, 300, 302–305, 307–309, 337, 338, 374–376, 378–381, 383–386, 398
percentile *see* probability, nonexceedance
Perl iii, xxiv, 376
plot types *see* bean plots, *see* box plots, *see* histograms, *see* rug plots, *see* violin plots
plotting positions 51, **51**, 53–55, 60, 170, 245, 248, 250, 348, 360
California 51
Cunnane 51, 53, 54, 193
Hazen 53–55

plotting-position coefficient 51, **51**, 52, 103, 107
plotting-position formula 51, **51**, 53, 55, 107
Weibull 51, 53–55, 57, 220, 263, 270, 318, 325, 342, 359, 362, 371
Polynomial Density-Quantile3 distribution 252, **252**, 253
Polynomial Density-Quantile4 distribution 253, **253**, 254
porosity 53, 54
portable document format 6, 7, 91, 196, 220, 365
probability
annual nonexceedance 42
annual recurrence interval 60
as percentiles 78, 79
conditional 24, **358**
adjustment for 337, 338, 358, 359, 398
cumulative **25**, 26
density 24, **24**, 25, 26, 32, 318
exceedance **26**, 27, 42, 60, 355
graph paper 51
nonexceedance 1, 25, **25**, 30, 32, 37, 51, 52, 60, 67, 102, 136, 137, 220, 239, 245, 249, 257, 321, 340, 355, 358, 391
probability density function 24, **24**, 25, 26, 28–30, 32, 48, 49, 59, 64, 65, 67, 83, 106, 160–162, 165, 170, 173, 176, 180, 184, 185, 201–203, 205, 209, 214, 215, 226, 228, 230, 231, 243, 249, 254, 256, 257, 267, 279, 280, 286–297, 330, 332
probability mass function **106**
probability-weighted moment
sample
plotting-position estimator 105
probability-weighted moments xxii, xxiii, 8, 9, 12, 13, 62, 63, 78, 79, 98, 99, **99**, 100–105, 108–115, 121, 123, 127, 142, 157–159, 164, 170, 191, 197, 239, 267, 337–346, 374, 401
A′-type
sample 345, **345**, 346
theoretical **344**, 345
A-type 401
sample 341, **341**, 342
theoretical **339**, 340
B′-type
sample **345**, 346
theoretical **344**, 345
B-type 191, 239, 401
sample 341, **341**, 342, 346
theoretical **339**, 340
method of 109, **109**, 111, 135
partial 338
prior 105, **105**
sample 100, 101, **103**, 105–107, 109, 111, 125, 127, 343
plotting-position estimator 101, 103, 104, 107, 111, 125
unbiased estimator 103–105, 111
self-determined **100**
theoretical **101**
product moment ratios
coefficient of variation *see* coefficient of variation
kurtosis *see* kurtosis
skew *see* skew
product moments xxii, 9, 10, 12, 14, 33, 46–48, 60, 67, 78, 79, **79**, 81, 83, 84, 86, 87, 91, 92, 94, 97–102, 109–112, 114, 120, 133–135, 138, 145, 146, 148, 152, 156, 159, 173, 175, 179, 180, 183, 206, 231, 234, 235, 242, 243, 256, 257, 267, 337, 338, 374–378, 383, 385, 386, 398
as statistics of moment arms 134
C-moments 114
coefficient of variation *see* coefficient of variation
estimators 86
kurtosis *see* kurtosis
L-moments preferable to 113
method of 2, **47**, 48, 49, 57, 58, 60, 135, 222, 264, 374, 375
moment ratio diagram 298, **298**
noncentral 101
ratios **84**
sample 47, 60, **84**, 86, 87, 97, 98, 102, 145, 157, 376, 378, 381, 383–385
sample size boundedness of 78, 87, 90, 92, 93, 150, 157
coefficient of variation 90, 91, 98
skew 90, 93, 94, 96
skew *see* skew

standard deviation *see* standard deviation
theoretical **83**
variance *see* variance
Pythagorean
distance 197

Q

quantile **29**, 32, 48, 50–52, 55, 68, 69, 94, 167, 178, 182, 184, 189, 219, 226, 234, 239, 249, 256, 257, 263, 264, 270, 279, 291, 296, 311, 321, 340, 349, 353–358, 360, 361, 367–369, 371, 381, 384, 385
extreme-tail estimation 324
mixtures 309
uncertainty **338**, 361, 398, *see* model-selection error, *see* sampling, error
model-selection error **338**, 361, 369, 371, 372, 398
sampling error **338**, 361, 362, 365, 367, 398
quantile distribution function *see* quantile function
quantile function 24, 29, **29**, 30, 32, 33, 35, 38, 39, 45, 48, 55, 59, 63, 65, 67, 94, 99, 101, 117–119, 121–123, 126, 136, 139–141, 160–162, 165, 170, 173, 174, 178, 181, 183, 188, 190, 214, 215, 224, 225, 227, 235, 254, 256–258, 267, 271, 274, 288–291, 296, 329, 330, 339–341, 344, 345, 355, 358, 360
algebra of 33, 401
as inverse distribution function 30
by inverse transform method **31**
by recursion 32
derivatives of 118, 119
sample 30
quantile-quantile plot 256
quartile 22, 44, 46
interquartile range *see* range, interquartile
lower 44, 46, 358
upper 44, 46, 163

R

R environment ii, iii, xxii–xxv, 2, 3, **3**, 4–18, 22–25, 27, 30–33, 36, 37, 43, 44, 46, 49, 52, 57, 61, 64, 68, 70, 73, 76, 78, 79, 81, 85, 87, 90, 91, 98, 99, 105, 110, 112, 114, 125, 127, 137, 142, 158, 161–164, 166, 169, 171, 172, 175–179, 182, 183, 195–197, 205, 208, 214, 216, 230, 247, 248, 259, 272, 298, 309, 314, 318, 337, 338, 399–401, 429
as freedom 3, 4, 30, 183
packages
GLDEX 10, 272–275, 434
Lmoments xxv, 10–13, 15, 115, 116, 127, 143, 272, 400, 434
NADA 338, 353, 356–358
POT 10, 11
RFA 10, 11
asbio 400
beanplot 318
extRemes 10
lattice 198
lmomRFA 10, 11
lmomco ii, iii, xxiv, xxv, 7, 10–13, 15, 18, 29, 53, 55, 57, 69–71, 74, 75, 87, 90, 94, 98, 100–102, 107–111, 114–116, 123, 124, 127–129, 142, 143, 148, 158, 159, 161–172, 176–179, 182–184, 186, 187, 191–193, 195, 197, 204, 207–211, 214–220, 224, 230, 238, 240, 247, 248, 251, 256, 258–260, 271, 273–275, 279, 281, 291, 295, 296, 303, 305, 306, 310, 311, 313–316, 323, 324, 337, 339, 342, 344, 353, 356, 363, 371, 386, 387, 391, 399–401, 422, 431
lmom 10–13, 15, 115, 119, 127, 143, 159, 160, 164, 183, 214, 256, 295, 434
vioplot 318
RData format 18
workspace 5, 18, 20
rainfall xxiii, 16, 48, 181, 186, 235, 302, 315, 316, 324, 358, 376, 378, 382
annual maximum 223, 224, 314–317, 319, 321–323, 336
depth 381
depth-duration frequency of 315
drought 235
extreme 113
hourly 378
inception of 201
minimum interevent time 378
monitoring 315
monthly 378

raindrop size 179
recording bias 319
storm statistics 378
random sample 2, 23, 41, 53, 61, 84, 130, 134, 135, 152, 188, 193, 279, 280, 339, 344, 359, 381, 387, 389–391
random variable 9, 23–26, 28, 30, 36, 37, 51, 59–61, 64–66, 77–79, 83, 94, 101, 105, 117, 120, 137, 139, 174, 204, 209, 245, 310, 338, 339, 343, 344, 348, 391
bivariate 386–392, 397, 400
continuous **23**, 59
discrete **23**
multivariate 9, 156, 338, 386, 391, 397
trivariate 391
univariate 1, 2, 8, 9, 11, 12, 15, 23, 113, 156, 158
range 30, 38, 47, **47**, 50, 93, 374
interquartile 21, 46, **46**, 47, 58, 317
midrange **70**
sample **71**, 78
Rayleigh distribution 116, 165, 167, 168, 198, **198**, 200–205, 210–213, 215, 291
recurrence interval 42, **42**, 43, 44, 219, 220, 314
return period **42**
relative efficiency 76
reliability analysis 338
reliability function **26**
Reverse Generalized Extreme Value distribution 248, *see* Weibull distribution, **248**
Reverse Gumbel distribution 116, 165, 167, 168, 186, 190, **190**, 191–193, 195, 196, 215, 291, 401
Rice distribution 116, 165, 167, 168, 204, **204**, 205, 206, 208, 210–213, 215, 256, 291
mean 210
variance 206, 210, 211
Right-Censored Generalized Pareto distribution 116, 165, 167, 168, 239, **239**, 240, 257, 342, 343
River
Baraboo 57–59, 231, 233
Choctawhatchee 325–327, 331, 335
Gila 325, 328
Guadalupe 219, 221, 222
Llano 325, 331, 362–364, 371, 373
Nueces 263–265, 270, 272
Platte 245–247
Rio Uruguai 187
Susquehanna 325, 328
Umpqua 325, 330
root mean square error 80, **80**
rug plots **387**, 388

S

S language 3
S-Plus xxiv
sample distribution **2**, 18, 216
sampling
accuracy **79**
bias 46, 78, 79, **79**, 80, 81, 214
precision **79**
variance 46, 78, 79, **79**, 80–83, 98, 361
minimum 89
uniformly-minimum of standard deviation 85
Sen weighted mean 61, 71, **71**, 77, 143
object of 71
serial correlation **24**
skew 9, 10, 79, 84, **84**, 93, 94, 96, 98, 156, 374, 383
bias 98
boundedness 98
sample **86**, 90, 93, 94, 96
unbounded unlike L-skew 121
skewness, concept of 9, 63, 86, 93, 94, 96, 97, 118, 121, 145, 152, 154, 156, 187, 200, 204, 216, 256, 259, 280, 296, 299, 374, 375, 381, 386, 398
distribution asymmetry 9, 93, 96, 113, 123, 283, 302, 330, 399
distribution shape *see* distribution parameter, shape
distribution symmetry 9, 19, 21, 86, 97, 120, 123, 144, 185, 225, 295, 313, 375, 378, 388, 399
negative 63, 186, 236
positive 19, 21, 63, 150, 181, 186, 302, 310, 330, 388
zero 386

soil liner 230
Spearman's Rho 387, 390, 397
standard deviation 9, 14, 27, 41, 44, 48, 51, 60, 75, 76, 79, 83, **83**, 84–90, 98, 148, 171, 173, 204, 214, 229, 243, 250, 263, 349, 367, 372, 374, 376
  bias of 89
  sample 85, **85**, 86, 88–90, 92
standard normal deviate 58, 249, 328, 331
standard normal variate *see* standard normal deviate
Stata 11
State
  Arizona 325
  Colorado ii
  Florida 325, 335
  Illinois 188
  Indiana 188
  Louisiana 368
  Montana xxiii
  Nebraska 245–247, 257
  New Mexico 301, 368
  New York 325
  Oklahoma 301, 368
  Oregon 325
  Texas xxiii–xxv, 11, 53, 54, 97, 181, 218, 219, 221–224, 256, 263–265, 270, 272, 281, 296, 301, 314, 315, 317, 319, 322, 323, 325, 362–364, 368, 370, 371, 373, 378, 381
    Panhandle region 314, 315, 317, 319, 322, 323
  Wisconsin 57, 59, 231, 233
statistical
  accuracy 80, 244
  bias **79**, 80, 82, 84, 87, 89, 90, 92–94, 96, 98, 145, 146, 148, 174, 296, 369, 370, 374, 377, 378, 381, 385
  bias ratio 146, 149
  concomitants 386, **386**
  consistency 114, 150, **150**, 152, 155, 156
  contamination 150, 151, 155, 156
  efficiency 62, 75, 76, **80**, 134
  estimator 45, 80–82, 84–86, 89
  goodness-of-fit 13, 298, 336, 376, 400, 401
    Chi-squared 400
    Kolmogorov-Smirnov test 13
    Komogorov-Smirnov 400
    outside scope of text 13, 299
    Shapiro-Wilk test for normality 13
  high outlier 150, 151, 153, 242, 375
  inconsistency 155, 156
  low outlier 97, 228, 242, 375
  moments 47, 78, 79, 99
    central product 83
  outlier 16, 22, 80, 113, 241, 242
  performance 80, **80**
  population 45, 46, **46**, 49
  precision 80
  relative efficiency 76, 80, **80**, 81, 235
  robustness 8, 9, 71, 74, **80**, 114, 139, 150
  sample **46**
  simulation 12, 16, 45, **45**, 46, 49, 65, 68, 73, 76, 80, 82, 89, 91–94, 104, 110, 137, 144, 145, 147, 149, 153, 159, 175, 209, 215, 241, 257, 261, 279, 280, 303, 304, 311–313, 337, 349, 361, 363, 364, 369, 373–376, 378, 381, 384, 392, 395
    bootstrap 130, 152, **152**
    pseudo-random numbers 45
  unbias 9, 80, **80**, 82, 84–86, 89, 101, 106, 114–116, 119, 124, 125, 145, 146, 351, 361, 377, 378, 383, 386, 387
streamflow 24, 48, 245, 336, 400
  annual maximum 336
  annual peak 56–58, 60, 97, 219, 222, 231, 256, 257, 263–265, 270–272, 296, 324, 326, 328, 330–332, 335, 362, 368, 371, 375
  annual volume 358
  daily mean 245, 257
  flood 16, 42, 43, 97, 99, 113, 186, 187, 235, 315, 336, 375, 400
  flood control levees 2
  flood plains 56, 324
  flood risk 335
  flood volume 376
  flow-duration curve 245, 257
  hydrograph 179, 180, 201, 203
    unit 179
  peak of hydrograph 180, 201
  river-flow modeling 315
  time of peak 201
  water quality 48

Student 3t distribution **254**
Student t (3-parameter) distribution 254, 255
survival analysis 338, 348, 352, 398
survival function **26**, 177, 348
  empirical 348
survivor function *see* survival function
SWSTAT 375

T

TKG2 xxiv, 381
TL-moments xxv, 8, 72, 77, 79, 112, 115, 128, 139, **139**, 140–143, 157, 183–186, 214, 240–242, 257, 275–278, 280, 291
  alternative version 141, **141**
  asymmetrical trimming 139, 143
  method of 275
  ratios 140
  recurrence relations of 141
  robustness 241, 242
  sample 115, 116, 143, **143**, 144, 279
  symmetrical trimming 72, 139, 142–144, 183, 185, 240, 275
  theoretical **139**, 140
  TL-mean 143, 144
  TL-mean, as Sen weighted mean 143
transistor data 346
Trimmed Generalized Lambda distribution 116, 165, 167, 168, 275, **275**, 276–279, 296
Trimmed Generalized Pareto distribution 116, 165, 167, 168, 240, **240**, 241, 257
trimmed L-moments *see* TL-moments

U

U.S. Geological Survey xxiii, xxiv, 56, 219, 222, 231, 245, 263–265, 270, 272, 324, 325, 335, 362, 368, 371
Uniform distribution 45, 68, 136, **136**, 137, 138, 197, 198, 238, 391
United States 188, 324, 326, 328, 330, 332
UNIX 3

V

variance 67, 80, 83, **83**, 84, 85, 88, 98, 148, 206
  sample **84**, 85, 86, 88, 367
violin plots 318, **318**, 319

W

Wakeby distribution 99, 114, 116, 160, 165, 167, 168, 260, 280, **280**, 281, 283–290, 292, 295–297, 300, 309, 327–332, 334–336, 362, 364–371, 373, 400
Weibull distribution 2, 25, 26, 35, 55, 116, 160, 165, 167, 168, 170–172, 177, 191, 195, 196, 247, **247**, 248–251, 257, 258, 349, 401
Weibull plotting positions *see* plotting positions, Weibull
wind
  speed
    annual maximum 218, 256, 370
    hurricane maximum 370
    offshore 400
    risk 219
  storm surge 42
  wave height 186
Windows 3

Index of R Functions

This index lists in alphabetical order the R functions used in the text. The functions are cataloged by heredity, whether built-in to R, listed by package, or other. The page for which functions are discussed within paragraphs are typeset in the normal font. The beginning page the code example for which functions are used only within the example and not discussed in the text are typeset in an italic font.

Built-in to R

`IQR()` 46, 58
`Sys.sleep()` 196, *284*
`abline()` *146*, 220
`abs()` *32*, *72*, *75*, *135*
`any()` 38
`as.character()` 36
`as.list()` *353*
`attach()` 20, *53*, 57, *231*, 248, *263*, *270*
`attributes()` 46, 47
`besselI()` 205, *207*
`boxplot()` 19, *59*, *317*, *325*
`c()` *17*, *35*, *36*, *49*, *54*, *57*, *66*, *69*, *74*, *81*, *82*, 87, *102*, *108*, *110*, *123*, *124*, *128*, *129*, *132*, *135*, *142*, *144*, *146*, *148*, *149*, *151*, *163*, *164*, *166*, *169–171*, *175*, *177–179*, *181*, *182*, *184*, *185*, *188*, *190*, *193*, *198*, *202*, *209*, *210*, *212*, *219*, *224*, *226*, *227*, *234*, *238*, *241*, *242*, *246*, *263*, *265*, *266*, *271*, *274*, *279*, *284*, *286*, *291*, *292*, *303*, *304*, *310*, *311*, *318*, *320–322*, *327*, *329*, *346*, *350*, *355*, *357*, *359*, *360*, *363*, *364*, *366*, *368*, *369*, *371*, *376*
`cat()` *57*, *66*, *81*, *82*, 87, 102, *102*, *110*, 130, *132*, *135*, *138*, *144*, *164*, *175*, 177, *179*, *181*, *182*, *209*, *226*, *242*, *266*, *284*, *369*, *376*, *390*
`cbind()` 34
`choose()` 105, 106, *125*, *126*, 127, *129*
`cor()` *387*, *389*, 390, *390*
`data()` 18, *20*, 53, *53*, 57, *231*, *245*, *249*, *263*, *270*, *316*, *325*, *342*, *353*, 362, *371*
`data.frame()` *17*, *18*, 57, *219*, *357*, 368, *388*, *391*, *393*
`dbinom()` *103*, *106*
`detach()` 20, 57, *263*
`dev.off()` 6, *6*, 196
`diff()` *47*
`dnorm()` 162
`do.call()` *318*
`dweibull()` 25
`ecdf()` 52
`exp()` *66*, *85*, *195*, *198*, *207*, *208*
`factorial()` 106
`file.choose()` 219, 220
`floor()` 30
`for()` *89*, *90*, 91, *107*, *110*, *131*, *132*, *144*, *146*, *149*, *152*, *175*, *195*, *198*, *208*, *210*, *212*, 234, *242*, *266*, 284, 291, *303*, 311, *311*, 313, *369*, *371*, *376*, *393*
`function()` *32*, *38*, *39*, *66*, 69, *73*, *76*, *81*, *125*, *126*, *129*, *188*, *207*, *208*, *286*, *320*, *359*, *371*, *392*, *393*
`gamma()` 64, 85, *85*

gsub() 220
help() 5, *15*, 19, 91, *91*, *387*
hist() 19
if() *32*, *39*, *81*, *152*, *169*, *207*, *208*, *266*, *284*, *286*, *291*, *359*, *392*, *393*
ifelse() *210*, *266*, *371*, *392*
integrate() *66*, 102, 125, *125*, *126*, 185, *207*, 208
is.na() *393*
is.null() 38
ks.test() 13
layout() 19, *131*, *201*, *208*, *224*, 230, *231*, *235*, 248, *286*, *292*, *294*, *329*, *331*
legend() 35, *149*, *193*, *210*, *226*, *227*, *246*, *263*, *265*, *279*, *284*, *327*, *350*, *360*, *363*, *364*, *366*
length() *17*, *71*, *73*, *86*, *107*, *132*, *207*, *208*, *210*, *292*, *350*, *360*, *371*, *392*
lgamma() 64, *66*, *85*
library() 15, *15*, *198*, 272, *273*, *318*, *353*, *357*
lines() 35, 39, *49*, 55, *57*, *146*, *149*, 171, *171*, *193*, *195*, *202*, *208*, *210*, *212*, *226*, *227*, *231*, *246*, *249*, *263*, *265*, *271*, *274*, *279*, *284*, *286*, *308*, *322*, *329*, *334*, *342*, *350*, 355, *360*, *363*, *364*, *366*, *371*
list() *81*, *107*, *124*, *198*, *317*, *318*, *325*
log() *188*, *198*, *371*
log10() *57*, *151*, *152*, *231*, *246*, *249*, 263, *265*, *271*, *334*, *363*
ls() 18
matrix() 19, *131*, *201*, *208*, *224*, *231*, *235*, *286*, *292*, *294*, *329*, *331*
max() *35*, 45, 49, 68, *68*, *69*, *202*, *266*, *291*, *318*, *369*
mean() 5, *20*, 49, *57*, *66*, *68*, *69*, *71*, 81, *81*, 82, *86*, 87, *89*, 98, *110*, *129*, *135*, *144*, *146*, *149*, *155*, *175*, *209*, *212*, *234*, *242*, *265*, *311*, 312, *376*, 389
median() 72, 73, 98
min() *35*, 45, 68, *202*, *266*, *291*, *318*, *369*, *392*
mtext() *59*, 249, *286*, *292*, *329*
names() 18, *32*, 46, *53*, *138*, *318*
next() *284*, *291*, *292*, *393*
optim() 188, 197, 333
options() 284
par() *318*
paste() *286*
pbinom() 106
pdf() 6, *6*, 91, 196, 365
pdffile() 220
pgamma() 31, *170*
pgeom() 44
plot() 25, 35, 36, 39, *39*, *49*, *54*, *56*, *57*, 58, 91, *94*, *131*, *135*, *146*, *149*, *151*, *152*, *171*, *181*, *184*, *188*, *190*, *193*, *195*, *202*, *208*, *210*, 220–222, *224*, *226*, *227*, *231*, *245*, *246*, *263*, 264, *265*, *271*, *274*, *279*, *284*, *286*, *292*, *304*, *322*, *329*, *334*, *342*, *350*, *355*, *360*, *362*, *363*, *369*, *371*, *388*, *393*, *395*
pnorm() 27, *151*, 162, *263*
points() *54*, 92, *131*, *135*, *188*, *190*, *210*, *212*, *220*, 250, *304*, 311, *321*, *322*, *327*, *342*, *350*, *369*, *372*, *395*
pp() 362
print() 6, 31, *46*, *47*, *69*, *71*, *72*, 75, *76*, *106*, *127*, *142*, *143*, *146*, *148*, *169*, 177, *219*, *234*, *241*, *270*, *273*, *323*, 326, 331, *353–355*, *357*, *373*
pweibull() 248
q() 5
qbeta() 69
qexp() 30, *34–36*, 65, 68, *68*, *69*, 178
qgamma() 31, 182
qnorm() *32*, *34*, 45, *46*, *49*, *54*, *57*, 58, *151*, 162, *164*, *210*, 211, *220*, *231*, 249, 263, *265*, *271*, *284*, 328, *329*, *334*, *360*, *363*, *364*, *366*
quantile() 52–55, *353*, *357*
qweibull() *35*
range() *47*
rbind() 34
rcauchy() *144*, 279
read.csv() 17, 53
read.cvs() *17*
read.cvs2() *17*
read.delim() *17*
read.delim2() *17*
read.table() *17*, 219
rep() *107*, 359, *359*
replicate() *66*, *68*, *73*, *76*, *82*, *129*, 145, *155*

return() *32*, *38*, *39*, *69*, *73*, *76*, *81*, *86*, *107*, *125*, *126*, *188*, *207*, *208*, *286*, *320*, *359*, *371*, *392*, *393*
rexp() 65, 68, *135*, *143*
rgamma() 91, 138, *149*
rgb() 91, *152*, *284*, *304*, *322*, *327*, *364*, 365, *366*, *388*
rm() *17*
rnorm() 45, 49, 53, *72*, *76*, 81, *89*, *90*, *106*, *107*, *127*, *151*, *154*, *155*, 174, *175*, *273*, *357*, *388*, *391*
round() 34, *34*, *54*, *57*, *66*, *82*, 87, *102*, *110*, *129*, *132*, *135*, *144*, *175*, 178, *181*, *182*, *209*, *219*, *226*, *234*, *242*, *266*, *354*, *355*, *389*, *390*
rug() 388
runif() 45, *46*, 68, *68*, *69*, 137, *209*, *376*, *393*
rweibull() 55, 196
sample() *129*, 130, 152, 153
sapply() *86*, *110*, *125*, *126*, *129*, *164*, 178, *317*, *318*, 359
save() *18*
sd() 44, 49, *57*, 58, 81, 85, 87, 89, 98, *149*, *175*, *234*, *265*, *373*
segments() 304, *304*, *311*
seq() 25, 31, *34*, *35*, 36, *49*, *89*, *90*, *107*, *131*, *144*, *146*, *149*, *152*, *175*, *202*, *208*, *234*, *242*, *249*, *274*, *279*, *284*, *286*, *303*, *311*, *322*, *350*, *355*, *369*, *376*
set.seed() *66*, 72, *273*, *279*
shapiro.test() 13
sin() *207*
sort() *53*, 55, 57, *66*, *73*, *86*, *107*, *132*, *135*, *151*, 170, *171*, *193*, *195*, *220*, 231, *246*, 248, *249*, *263*, *270*, *274*, 316, *329*, *334*, *342*, *350*, 359, *371*
sqrt() *76*, *81*, *82*, *86*, *89*, *102*, *209*, *284*, *286*, *294*, *373*
stop() *207*
str() *74*, 108, *108*, 128, *163*, *166*, 181, *181*, 185, *224*, *232*, *238*, *320*, 326, 362
sum() *73*, *86*, *125*, *126*, *132*, *207*, *208*, *234*
summary() 44, 46, 58, 81, 89, *89*, *90*, 138, 145, 155, *234*, *373*
system.time() 69
try() *392*
uniroot() *392*
var() 76, *76*
vector() *89*, *90*, *110*, *131*, *132*, *146*, *149*, *152*, *175*, *208*, *212*, *234*, *242*, *303*, *350*, *369*, *376*, *393*
warning() *39*, *286*, *393*
weighted.mean() 320, 368
while() *266*
write.table() 17

Other Functions

Barnett() *76*
Ftrans() 38, 39
HF() 38
MOcop() 392
MarcumQ1() *208*
NADA:Cen() *353*, *357*
NADA:cenfit() *353*, *357*
beanplot:beanplot() *318*
derCOPinv() 392, *393*
func() *392*
grv() 371
lambda.by.cdf() 125, *125*, 126
lattice:contourplot() *198*
maxOstat.system() 129
myWAK() 286, 287
myquagum() *188*
ostat.sd() *86*
qua.by.recursion() *32*, 33
qua.ostat() *69*
sam.biasvar() 73, 81, 82, *82*, 83
simulateCopula() 392, *393*, *395*
test.pwm.pp() 107, 108
trim.mean() 72, 73
vioplot:vioplot() *318*

Package: *lmomco*

LaguerreHalf() 209
Lcomoment.Wk() 131, 132, *132*
Lcomoment.coefficients() 390, 391, *391*
Lcomoment.correlation() 389
Lcomoment.matrix() 388, *389–391*
T2prob() 43, 323
TLmom() 115
TLmoms() *72*, 115, 143, 157, 279

are.lmom.valid() 115, 124, 170, 284, *284*, *286*
are.par.valid() 166, 168, 170
are.parTLgld.valid() 168
are.parTLgpa.valid() 168
are.parcau.valid() 168
are.parexp.valid() 168, *169*
are.pargam.valid() 168
are.pargev.valid() 168
are.pargld.valid() 168
are.parglo.valid() 168
are.pargno.valid() 168
are.pargpa.valid() 168
are.pargum.valid() 168, *169*, 188
are.parkap.valid() 168
are.parkur.valid() 168
are.parln3.valid() 168
are.parnor.valid() 168
are.parpe3.valid() 168
are.parray.valid() 168
are.parrevgum.valid() 168
are.parrice.valid() 168
are.parwak.valid() 168
are.parwei.valid() 168
cdfcau() 165
cdfexp() 165, *178*
cdfgam() 165, *170*
cdfgev() *159*, 165, 220, 249
cdfgld() 165
cdfglo() 165, 224
cdfgno() 165
cdfgpa() 165
cdfgum() 15, 165
cdfkap() 165
cdfkur() 165
cdfln3() 165
cdfnor() *126*, 162, 165
cdfpe3() 165
cdfray() 165
cdfrevgum() 165
cdfrice() 165, 208, 210, *210*
cdfwak() 165, 286
cdfwei() 165, 249
check.fs() 38
check.pdf() *201*, 230, 249, 286, 287, 292, *292*, *294*, 330
dist.list() 148, 164, 291
dlmomco() 29, *32*, 33, 162, 171, 291
expect.max.ostat() *66*
fliplmoms() 347, 356, 357
gen.freq.curves() 363, 364, *366*
genci() 363, 368, 369
gini.mean.diff() 74, *76*
hlmomco() 29, 171
is.TLgld() 168
is.TLgpa() 168
is.cau() 168
is.exp() 168
is.gam() 168
is.gev() 168
is.gld() 168
is.glo() 168, 169
is.gno() 168
is.gpa() 168
is.gum() 168
is.kap() 168
is.kur() 168
is.ln3() 168
is.nor() 168
is.pe3() 168
is.ray() 168
is.revgum() 168
is.rice() 168
is.wak() 168
is.wei() 168
lcomoms2() 391, 394, 396
lmom.ub() 55, *169*
lmom2par() 138, 170, 171, *171*, 230, *291*, *292*, *350*, *371*
lmom2pwm() *108*, 111, 115, 123, 157
lmom2vec() 115, 164, 170
lmomRCmark() 115, 350
lmomTLgld() 116, 167
lmomTLgpa() 116, 167, 241
lmomcau() 116, 167, 186
lmomexp() 116, 167, 177, *182*
lmomgam() 116, 148, 167, 181
lmomgev() 116, 167, 226
lmomgld() 116, 167
lmomglo() 116, 167, 224, 226
lmomgno() 116, 167, 231
lmomgpa() 116, 167, 238, *238*

lmomgpaRC() 116, 167
lmomgum() 116, 167
lmomkap() 116, 167
lmomkur() 116, 167, *197*, *198*
lmomln3() 116, 167
lmomnor() 116, 166, 167
lmompe3() 116, 167
lmomray() 116, 167, 202, *202*
lmomrevgum() 116, 167
lmomrice() 116, 167, 210, *210*, 213
lmoms() 75, 115, 120, 124, 127, 132, *132*, 138, *149*, *152*, *154*, 157, 174, *177*, *178*, *220*, 231, 248, *249*, *263*, *266*, *270*, *273*, 303, *303*, 311, *318*, *326*, *350*, 359, 362, *371*, *373*, *376*, *389*, 390
lmoms.ub() 115
lmomsRCmark() 115, 350, 351, 354–358
lmomwak() 116, 167
lmomwei() 116, 167
lmorph() 115, 124, 128, *177*, 178, 197, 224, 231, 238
lmrdia() 212, *212*, 306, *306*, 308, *308*, 309, *309*, *321*, *372*
nonexceeds() *39*, 94, 181, 184, *190*, *210*, *224*, *226*, 264, *291*, *328*, 364
par2cdf() 170, *170*
par2lmom() 145, 170, 224, 238
par2pdf() 170
par2qua() 69, 170, 184, *329*, *334*, 360
par2vec() 164, 170
parTLgld() 167, 271, 279
parTLgpa() 167
parcau() 167, 279
parexp() 167, 177, *178*
pargam() 110, 167, 181, 303
pargev() 167, *220*, 226, 248
pargld() 167, 270, 271, 273, 274, 331, 333, 334
parglo() 167, 169, 226, 303
pargno() 167, 230, 231, *265*, 354, *357*
pargpa() 167, 342, 360
pargpaRC() 167, 342
pargum() 167, *169*, 193, *193*
parkap() 167, 263, *266*, *294*, 311, *320*, *326*
parkur() 167, 197
parln3() 167
parnor() 165, 167, 232
parpe3() 167, 303
parray() 167, *201*
parrevgum() 167, *193*, 196
parrice() 167
parwak() 167, *284*, 286, *286*, *328*, 362, *368*
parwei() 55, 167, 248
pdfcau() 165, 184, 279
pdfexp() 165
pdfgam() 165, *202*
pdfgev() 165, 226
pdfgld() 165, 279
pdfglo() 165, 226
pdfgno() 165
pdfgpa() 165
pdfgum() 165
pdfkap() 165
pdfkur() 165
pdfln3() 165
pdfnor() 162, 165
pdfpe3() 165
pdfray() 165, *201*, *202*
pdfrevgum() 165
pdfrice() 165, 208, *208*
pdfwak() 165
pdfwei() 165
plmomco() 29, *32*, 33, 162, 169, *170*, 171
plotlmrdia() 212, 308, 309, *310*, *311*, *321*, *327*, *372*
pmoms() *15*, 87, 89, 90, 94, 98, *152*, *155*, *376*
pp() 53–55, 57, *151*, 170, *193*, *220*, 231, 248, *249*, *263*, *270*, *318*, *326*, 342, *350*, 359
prettydist() 291
prob2T() 43, *151*, 220
pwm() 101, 110, *342*
pwm.gev() 101, 107, 108, 111, *125*
pwm.pp() 101
pwm.ub() 101, 106, 111
pwm2lmom() *108*, 110, 111, 115, *123*, *125*, 341, *342*, 346, *346*, *347*
pwm2vec() 101, 164, 170
pwmLC() 347
pwmRC() 342, 346
qlmomco() 39, 162, 171, 264, *274*, 291, *291*, *342*, *350*, *354*, 355, *355*, *357*, *371*

qua.ostat() 69, 70
qua2ci() 369
quacau() 165, 184, 279
quaexp() 165, *169*, 178, *178*, 179
quagam() 165, 181, 182, *202*
quagev() 165, *220*, 226, 249
quagld() 165, 270, *271*, 279
quaglo() 165, *169*, 224, 226, *226*
quagno() 165
quagpa() 165, 360
quagum() 165, 167, *169*, 188, *190*, *193*
quakap() 165, 263, *271*
quakur() 165
qualn3() 165
quanor() 162, 165, *231*
quape3() 94, 165
quaray() 165, *202*
quarevgum() 165, *193*
quarice() 165, *209*, 210, *210*
quawak() 165, *284*, 286, *363*, 364, *366*
quawei() 55, 165, 249
rlmomco() 94, 110, 130, 138, *146*, 162, 171, *171*, 193, *212*, *266*, 303, 311, 359
sen.mean() 71, 72, *72*, 73
theoLmoms() 115, 123, 125, 166, 184, 186, 194, *194*
theoLmoms.max.ostat() 115
theoTLmoms() 115, 142, 166, 185, 186, 241
theopwms() 101, 102, 111
vec2TLmom() 115
vec2lmom() 115, 120, *123*, 124, *128*, 164, 170, 181, *193*, *197*, *201*, 225, 230, *284*, *286*, 291, *294*, 303, 311, *320*, *368*
vec2par() *29*, 33, 39, 69, 94, 102, 110, 123, *126*, 142, *146*, 148, *163*, 164, *164*, *166*, 170, *170*, 178, *179*, 184, *185*, *188*, *190*, *202*, *208–210*, *212*, *224*, *238*, *241*, *266*, 270, *273*, *274*, *333*, 359, *376*
vec2pwm() 101, *108*, 164, 170, *346*, *347*
z.par2qua() 360

Package: *GLDEX*
Lcoefs() 272
Lmomcov() 272
Lmomcov_calc() 272
Lmoments() 272
Lmoments_calc() 272
fun.RMFMKL.lm() *273*
fun.RPRS.lm() 273, *273*, 274
fun.data.fit.lm() *273*
t1lmoments() 272

Package: *Lmoments*
Lmoments() 116, 127
t1lmoments() 143
t1moments() 116

Package: *lmom*
cdfexp() 160
cdfgam() 160
cdfgev() *159*, 160
cdfglo() 160
cdfgno() 160
cdfgpa() 160
cdfgum() 160
cdfkap() 160
cdfln3() 160
cdfnor() 160
cdfpe3() 160
cdfwak() 160
cdfwei() 160
lmrexp() 119, 160
lmrgam() 119, 160
lmrgev() 119, 160
lmrglo() 119, 160
lmrgno() 119, 160
lmrgpa() 119, 160
lmrgum() 119, 160
lmrkap() 119, 160
lmrln3() 119, 160
lmrnor() 119, 160
lmrpe3() 119, 160
lmrwak() 119, 160
lmrwei() 119, 160
pelexp() 160
pelgam() 160
pelgev() 160
pelglo() 160
pelgno() 160
pelgpa() 160
pelgum() 160

pelkap() 160
pelln3() 160
pelnor() 160
pelpe3() 160
pelwak() 160
pelwei() 160
quaexp() 160
quagam() 160
quagev() 160
quaglo() 160
quagno() 160
quagpa() 160
quagum() 160
quakap() 160
qualn3() 160
quanor() 160
quape3() 160
quawak() 160
quawei() 160
samlmu() 119, 127, 157