DEVELOPING PRECIPITATION INTENSITY-DURATION-FREQUENCY (IDF) MODELS USING NONLINEAR MINIMIZATION IN R

by

Cristal Tay

Submitted in partial fulfillment
of the requirements for the degree
Masters of Science in Civil Engineering

Texas Tech University
October 2015

ABSTRACT

Development of rainfall coefficients from Intensity-Duration-Frequency (IDF) models has been used in the United States at least since the 1970s. Rainfall coefficient development methods involve regression analysis to best fit a line through an observed set of rainfall depths at varying durations and frequency. IDF equations provide an advantage to engineers who wish to calculate intensities at varying durations. IDF models are constructed based on rainfall patterns and thus vary by geographic location. The IDF model used in the state of Texas is found in the Texas Department of Transportation's (TxDOT) Hydraulic Design Manual [26]. The IDF model is composed of *e*, *b* and *d* variables known as the Texas rainfall coefficients. The most recent method for developing the *e*, *b* and *d* rainfall coefficients in Texas was completed through the 0-6824-1 TxDOT project that employed an ordinary least squares (OLS) regression. The OLS method required analyst time to linearize a nonlinear equation. Linearization within this thesis refers to transforming a nonlinear equation into linear form. The OLS method developed rainfall coefficients using linear regression analysis, one-dimensional optimization and a predicted residual sum of squares error (PRESS) statistic.

We decided to develop the coefficients directly through nonlinear minimization (NLM) to cut down on time and cost, and increase efficiency. In this thesis we present the background of intensity-duration-frequency models, the previous OLS method, the suggested NLM method, and a comparison of the results between the two methods in graphical and tabular form. We further discuss the adaptability, efficiency and regression fit of the two methods.

In addition, we present two **R** scripts, created to develop coefficients for various forms of IDF models. **R** is an open source, statistical programming language and software. The first code accepts data for single rainfall stations, and the second code accepts data for an entire state based on frequency. Both codes use the `nlm` package in **R** to develop the IDF model. This thesis presents the code with a general guidance but it is not meant to be a tutorial.

DEDICATION

To my beloved cat, Cali

ACKNOWLEDGEMENTS

I would like to express my gratitude to my advisor and committee chair, Dr. Theodore G. Cleveland, for his continual guidance, wisdom, and patience. Without his assistance, this thesis would not have been possible. Additionally, without him I would not have had the opportunity to obtain a Master's degree or be involved in wicked cool research projects.

I would also like to thank my other committee member, Dr. Ken A. Rainwater, for his knowledge, time and effort in providing helpful and constructive feedback of this thesis.

Lastly I want to thank my roommate, and my family for their continuous support.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1 – INTRODUCTION

This thesis documents an alternate analysis methodology for determining intensity-duration-frequency (IDF) rainfall coefficients. Previous methodology was implemented in the Texas Department of Transportation's (TxDOT) 0-6824-1 project, which will herein be referred to as the "0-6824 TxDOT project." The 0-6824 TxDOT project was completed August 2015 with collaboration of research teams from Texas Tech University's Center for Multidisciplinary Research in Transportation (TechMRT) and the United States Geological Survey's (USGS) Texas Water Science Center, Lubbock Field Office, Lubbock, Texas. The IDF model (Equation 1) from TxDOT's Hydraulic Design Manual [26] was used and its corresponding *e*, *b* and *d* coefficients were developed. The 2015 rainfall coefficients were developed from rainfall data retrieved from the USGS Depth-Duration-Frequency (DDF) Atlas of Texas [2]. Cleveland and others of the 0-6824 research project decided on developing the IDF model by linearizing the nonlinear IDF model [27]. The *ebd* coefficients were determined through linear regression using ordinary least squares (OLS).

After revision of the 0-6824 TxDOT project, we contemplated the use of a direct nonlinear programming approach, through nonlinear minimization (NLM). We incorporated the NLM method and evaluated the analysis and results against the OLS method. The *ebd*, depth and intensity values of the OLS and NLM methods were compared at the 2 year and 100 year annual recurrence intervals (ARI). The 2-year and 100 year ARI were chosen because they are commonly used for design.

The IDF model was constructed using the `nlm` package in the statistical programming language and environment **R**, prevalent amongst statisticians and engineers. The thesis includes two **R** codes that uses the `nlm` package to develop rainfall coefficients. The first code, titled *SingleLocation*, develops rainfall coefficients for a single rainfall station. The second code, titled *BatchLocation*, has the ability to develop rainfall coefficients at varying frequencies for multiple stations or an entire state. The **R** statistical language and software remains freely available under the GNU General Public License [22]. This thesis is not meant to be a tutorial for the codes but a presentation of the development and results of the codes.

1.1 Motivation

Although linearizing a nonlinear IDF model to develop rainfall coefficients is functional and robust, it requires additional steps and demands the model structure be linear (or log-linear). The code provided in the 0-6824 TxDOT project was created to specifically work with a particular nonlinear IDF power law model (Equation 1), and will be useless if the structure of the IDF model was to ever change in Texas. The IDF model structure in Equation 1 has been used in Texas at least since the 1970's, but it could be subject to change.[1]

Other forms of IDF equations are used elsewhere and vary geographically based on rainfall patterns. A direct nonlinear programming approach does not require the IDF model to be log-linearizable and can be solved with the `nlm` statistical package provided in **`R`** [22]. Thus a change in IDF model structure can be implemented easily with NLP without a need to rewrite the entire code base – instead just a rewrite of the model equation.

The `nlm` package in **`R`** allows for development of coefficients for other states with different IDF models. The ability to update rainfall coefficients more autonomously allows for rapid implementation within design manuals, that would benefit designers with up-to-date design tools. With an increase in rainfall gauges and accumulation of data, future updates will be necessary for adequate drainage design. Using a linear regression-based approach limits the 0-6824 TxDOT code to the specific IDF model structure for which it was written, that subsequently presents a challenge to any future updates regarding the IDF model because entirely new code will have to be created.

The developed `nlm` based code was written to adapt to IDF model changes easily, and function with a high degree of efficiency thereby requiring fewer user-interactive changes than the linearized 0-6824 TxDOT code. The **`R`** code can be edited and read through **`RStudio`** [39], a free and open-source integrated development environment (IDE) for the **`R`** language [22].

---

[1] Changes in the future could include more parameters than the current three or an entire structural change in the equation.

1.2 Rainfall Intensity Background

Intensity directly affects peak runoff estimates made using the rational method equation, which is used for hydraulic design in small rural and urban watersheds.[2] The IDF coefficients *e*, *b* and *d* are in the intensity equation. Equation 1 is the IDF equation from the Hydraulic Design Manual [26].

$$\boldsymbol{I_{AEP;\,COUNTY} = \frac{b}{(T_c+d)^e}} \qquad \textbf{Equation 1}$$

where I is the intensity in inches per hour, T_c is the time of concentration in minutes, b is a scaling value in inches per hour, d is an offset in minutes, and e is an exponent. The subscript on I is to imply that the coefficients are a function of annual exceedance probability (AEP) and location (county). The *ebd* coefficients are known as the rainfall coefficients within Texas

Intensity is defined as a depth of precipitation (covering an area) over a period of time. Intensity is expressed as Equation 2 when used in conjunction with the Texas Depth-Duration-Frequency Atlas [2] or with older National Weather Service (NWS) products, HYDRO-35 [11] and TP-40 [13], or even current NWS products, such as the NOAA Atlas 14 [20].

$$\boldsymbol{I_{ARI} = \frac{P_d}{T_c}} \qquad \textbf{Equation 2}$$

where P_d is the depth of rainfall in inches for a specified annual recurrence interval (ARI), for a design storm of duration T_c, where T_c is the time of concentration in hours.

Depth, P_d, for Texas is established from the 2004 Atlas of Depth-Duration Frequency of Precipitation Annual Maxima for Texas (DDF Atlas) [2] at various durations and frequencies. An engineer would typically interpolate a precipitation depth based on distance between contour lines and retrieve the precipitation depth at the location of interest. The DDF values from the 2004 DDF Atlas [2] is presented in Appendix A.

---

[2] The rational equation method, even with its limitations is a simple to implement tool to size hydraulic structures. A large percentage of hydraulic structures are sized based on the rational method and its use is anticipated to continue for decades.

To avoid graphical or tabular lookup for design rainfall intensities, Intensity-Duration-Frequency (IDF) relationships, often expressed as a simple algebraic equation (such as Equation 1), are used. IDF values are estimates of the average rate of precipitation that falls with a given period of time in a specific geographical region for a specified duration and frequency or recurrence interval. IDF values (inchers per hour) are essentially the associated DDF value (inches) divided by the duration (hour) of the storm event (Equation 2).

Because DDF analyses (DDF Atlas [2] and NOAA Atlas 14 [20]) of rainfall are restricted to a few discrete durations, equations for IDF curves provide a mechanism to estimate rainfall intensity for arbitrary durations. The ability to use equations instead of graphical lookup methods for any duration makes IDF models/equations attractive to practitioners because the documentation is simplified and subject to less individual subjectivity. The equations are "fit" to discrete depth or intensity values and thus produce a continuous model of rainfall intensity.[3] Many algebraic forms have been used to represent IDF curves. Equation 1 is the IDF model currently used by TxDOT [26].

IDF curves are a graphical representation of the annual probability that a given average rainfall intensity will occur over a specified duration at some location. The curves tend to differ based on geographic precipitation patterns. IDF curves are usually displayed on log-log plots with intensity (inches per hour) on the y-axis and duration (hours) on the x-axis. In log-log space, most IDF curves developed from Equation 1 display a quasi-parallel form. Figure 1 is a set of IDF curves for Hood County, Texas that are typical. Each curve on the plot represents an ARI; stacked from smaller to larger ARIs. At durations larger than an hour, the IDF curve typically displays zero curvature. At durations shorter than an hour, the IDF curve displays a concave curvature.

---

[3] Continuous in respect to duration variable. Generally, IDF equations are still for discrete risk levels (ARI).

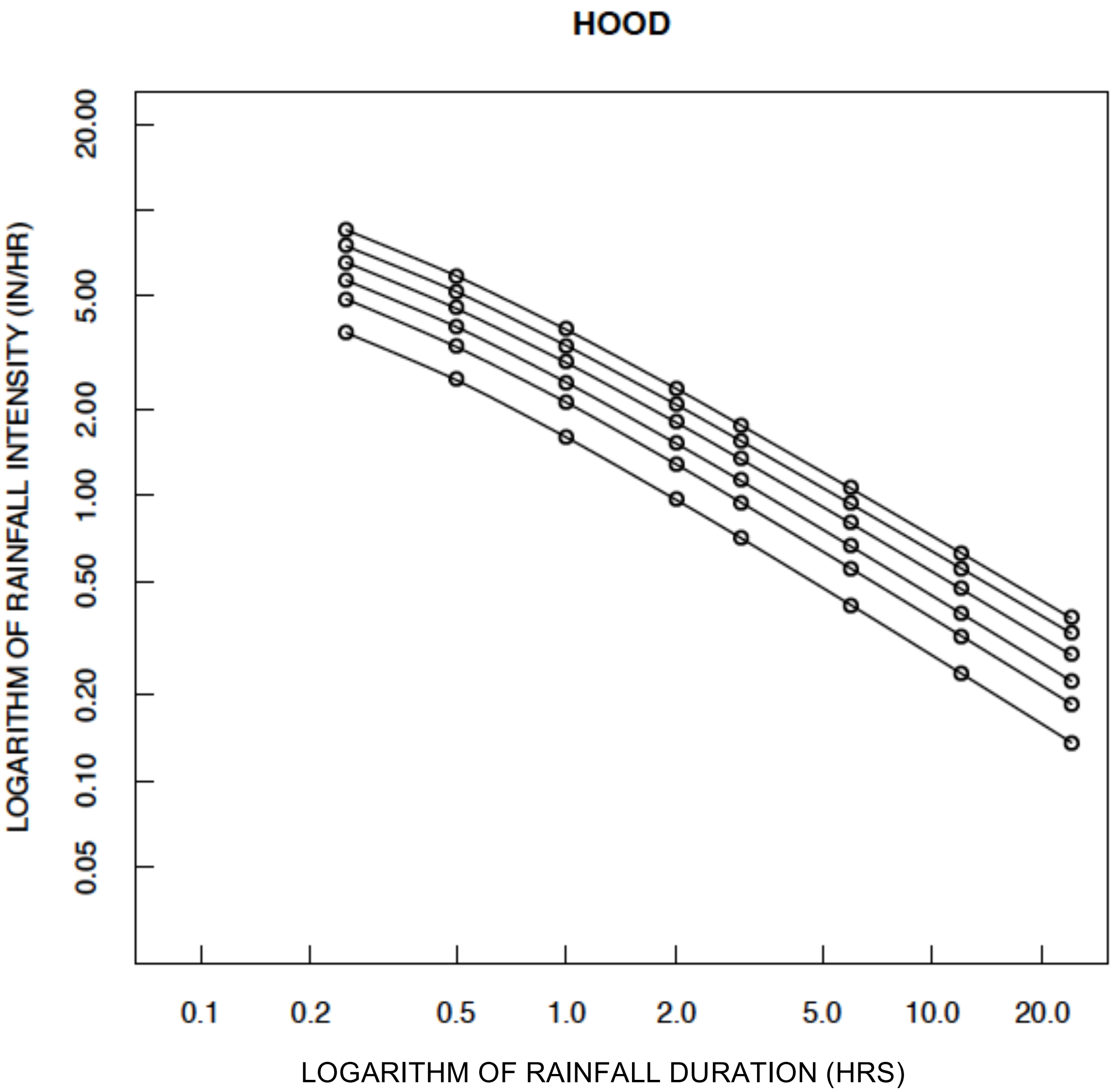
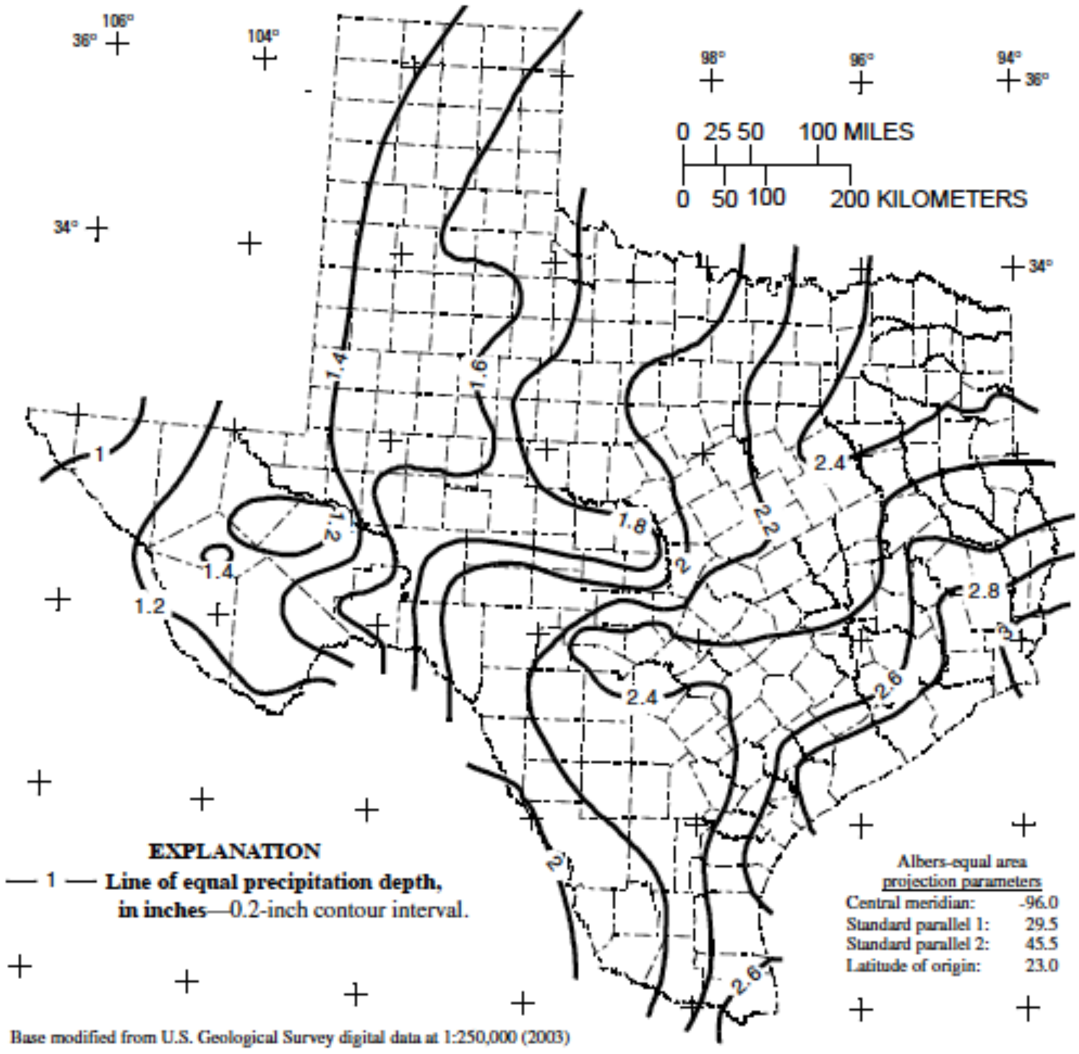


Figure 1: Intensity-duration-frequency curve of maximum rainfall in Hood County, Texas for the 2, 5, 10, 25, 50, and 100-year ARIs. Intensities developed by nonlinear minimization.

1.3 0-6824 TxDOT Research Background

The 2004 DDF Atlas [2] was used to generate the updated 2015 *ebd* coefficients. Using the 3D Analyst package in ArcGIS, the precipitation depths (at varying frequency and durations) at the centroid of each Texas county were extracted from the precipitation contour lines and placed into a database. The intensities for each county at varying frequencies were found by dividing the extrapolated depth over duration (Equation 2). The *ebd* values were subsequently obtained by fitting the IDF model (Equation 1) to these calculated intensity values.

The DDF Atlas [2] failed to provide isopluvial lines past the boundaries of the state, as seen in Figure 2. Thus precipitations for most border counties could not be extrapolated from the Atlas.



Base modified from U.S. Geological Survey digital data at 1:250,000 (2003)

Figure 2: Depth of precipitation for 2-year storm for 3-hour duration from the 2004 DDF Atlas [2].

Counties that bordered New Mexico, Oklahoma, Arkansas and Louisiana were interpolated from NOAA Atlas 14 values [20]. Counties that bordered Mexico were determined by engineering judgment because there are no products equivalent to TP-40 [13], HY-35 [11], DDF Atlas [2] or NOAA Atlas 14 [20] currently available.

Researchers, Cleveland and Asquith, of the 0-6824 TxDOT project [27] chose to develop the rainfall coefficients through a hybrid linear regression and single-variable minimization method instead of using a direct non-linear minimization method. Ordinary least squares (OLS) and a line search method were used through a sequential unconstrained minimization technique (SUMT) algorithm [32].[4] The team initially tried a nonlinear minimization (NLM) method. However, when run autonomously, the NLM method occasionally failed to find general solutions and produced unsatisfactory graphical fitting performance. Instead of attempting to fix these errors, the researchers proceeded onto the robust linear regression using the OLS method.[5] This decision increased the complexity of the regression and thus the **R** code. The code to compute the development of the coefficients was written in **R** language and was designed for only fitting IDF models with the structure of Equation 1.

The IDF curves were visually analyzed in log-log space, which revealed that durations shorter than an hour presented a departure from linearity. The shorter durations showed signs of curvature (logarithmic offset) for all counties. To solve for the *ebd* coefficients while accounting for the logarithmic offset, *d*, the IDF equation was log-linearized. First, the researchers used the *d* values provided from the 1985 *ebd* coefficients [25]. By pre-defining the value of *d*, the denominator of the IDF model could be simplified to a constant, η, and equation 1 could become a linear equation. Table 1 shows the equations used in the 0-6824 TxDOT algorithm.

Table 1: Linearized IDF Equations from 0-6824 TxDOT Project.

$\eta = T_c + d$
$\log_{10}(\mathrm{IDF_F}) = \log_{10}(b) + e\log_{10}(\eta)$

---

[4] The 0-6824 TxDOT project used the `optimize()` line search tool in **R**. `optimize` uses a golden section search also called Brent's line-search [32]. Brent's method is considered faster than bisection (Bolzano's method), regula falsi, secant method, or Fibonacci search.

[5] The term robust refers to the ability to always produce an optimal solution.

At each step of the SUMT algorithm, *d* was found by a line search (Brent's Method [32]) and always-optimal estimated values of *e* and *b* were produced.[6] The *e* and *b* coefficients were generated for each county in Texas at annual recurrence intervals (ARI) of 2, 5, 10, 25, 50, and 100 years. The algorithm then took the derivative of the linear function and through the line search method, solved for the optimal *d* coefficient. At every iteration, the predicted residual sum of squares error (PRESS) was minimized. Equation 1 is the PRESS statistic.

$$\boldsymbol{PRESS = \Sigma_{i=1}^{n} \left(\frac{y_i - \hat{y}_{i,-i}}{1 - h_{ii}}\right)^2} \qquad \textbf{Equation 3}$$

where *n* is the number of observations, *i* is the step count, y_i is the observed value, $\hat{y}_{i,-i}$ is the fitted value obtained from the remaining n-1 sample values, and h_{ii} is the leverage value. The double subscript *ii* refers to the diagonal of the hat matrix.[7]

When an observation was removed, a fitted regression function was used to obtain the predicted value for the *i*th observation. The PRESS statistic assessed the model's predictive ability (fitting methods); the smaller the PRESS value, the better the model's predictive ability.

In summary, the researchers of the TxDOT rainfall coefficient project created a SUMT algorithm for computing the *e, b,* and *d* coefficients centered on the IDF model in Equation 1. By pre-defining the values of d, the IDF equation (Equation 1) could be linearized. Using the 1985 rainfall values of *d* [25], the algorithm computed *e* and *b* by OLS regression. A line search then generated the values of *d*. Minimization of PRESS was used throughout to evaluate the error associated with the fitting method.

The team encountered IDF curvature inconsistencies with counties near the Mexican border of Texas. Three members of the team manually adjusted the DDF values to produce a quasi-parallel curve formation, as displayed in Figure 1 (Hood County). An **R** code was written to allow for adjustments of DDF values and re-fitting of IDF curves.

---

[6] In optimization, line search approach finds a descent direction in which the objective function will be reduced and computes a step size that determines how far to move along that direction [6]. Line search is available as a package in **R** [22].

[7] The hat matrix, sometimes referred to as the influence matrix, is a common term in regression analysis that describes the influence each observed value has on each fitted value [18]. The leverage value was obtained from regression packages in **R** **[22].**

The practical purposes of the adjustments were to correct inconsistencies in the few border counties that displayed radical behavior in curvature, such as El Paso County in Figure 2.

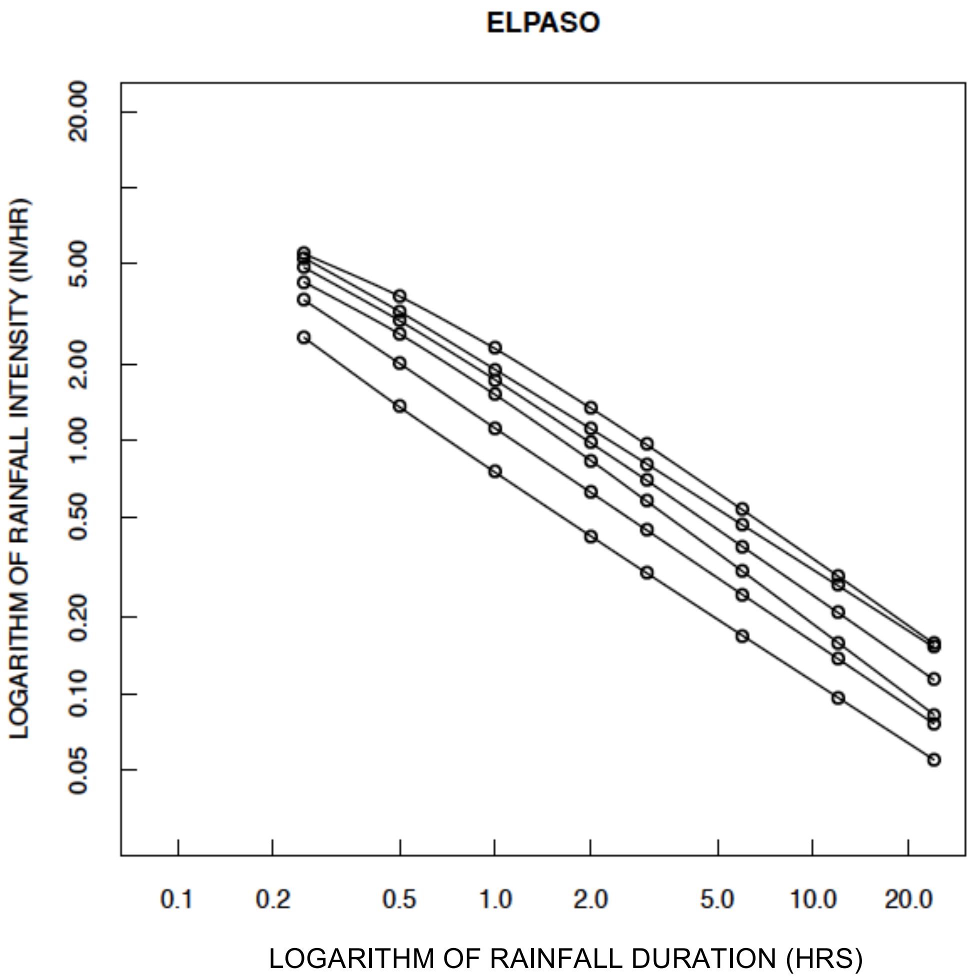


Figure 3: Intensity-duration-frequency curve of maximum rainfall in El Paso County, Texas for the 2, 5, 10, 25, 50, and 100-year ARIs. Intensities developed by nonlinear minimization.

The three members ultimately adjusted the IDF curves at every ARI for all 254 Texas counties. Depth values for each county were obtained from each analyst's adjusted IDF curves and averaged together resulting in the final 2015 DDF values that were delivered to TxDOT in August, 2015. The official 2015 DDF values can be found in Appendix B and the 2015 *ebd* coefficients (generated from the 2015 DDF values) can be found in Appendix C. The pre-adjusted depth and resulting *ebd* values (developed from the OLS method) can be found in Appendix D and E, respectively. The pre-adjusted results of the 0-6824 TxDOT research will be used to compare against the NLM method to avoid possible discrepancies from analyst adjustments.

CHAPTER 2 – LITERATURE REVIEW

Rainfall-runoff estimation is important for adequate design of storm water systems to minimize impacts within watersheds. Of the hydrologic methods used for determining peak discharge from basin runoff, the Rational Method is most utilized throughout Texas. Rainfall intensity (*I*) is one of the variables that directly affect the calculation of peak runoff. Equation 4 is the Rational Runoff Equation.

$$\boldsymbol{Q = \frac{CIA}{Z}} \qquad \textbf{Equation 4}$$

where Q is the maximum rate of runoff in cubic feet per second, C is a runoff coefficient, I is the rainfall intensity in inches per hour, A is the drainage area in acres, and Z is a conversion factor between U.S. Customary and S.I. units.

The methods and models for determining runoff and intensity have been established and used in Texas since at least 1970. In this section, precipitation-frequency data, methods for determining rainfall coefficients, and IDF relationships are investigated.

2.1 Precipitation Frequency Data

In the 1950s, the Weather Bureau began developing rainfall frequency data for the United States. The research was further extended in 1955 by the Weather Bureau and Soil Conservation Service to establish depth-area-duration-frequency values for the United States. The product of the research produced a series of Technical Papers that provided a background to rainfall methodology and concepts for the United States. Technical Paper No. 25 (TP-25) [29], published in 1955, presented the first set of IDF curves for selected cities and counties throughout the United States. The paper was available for purchase from the U.S. Government for 40 cents.

Technical Paper No. 40 (TP-40) [13], published in 1961, introduced the first depth-duration-frequency atlas for the United States. TP-40 published maps of the United States with isopluvial lines at varying durations ranging from 30 minutes to 24 hours and ARIs of 1 year to 100 years. TP-40 was a monumental feat that provided engineers with useful rainfall-frequency estimates for design.

In 1977, HYDRO-35 (HY-35) [11] published a supplemental DDF atlas for the eastern half of the United States. HY-35 displayed rainfall-frequency data at durations less than an hour that were not covered in TP-40, and further described ways to interpolate un-mapped durations.

In 1998, the USGS in cooperation with TxDOT developed a depth-duration frequency of precipitation analysis specifically for Texas [1]. The report was created to provide a method to develop accurate DDF estimates used for cost-effective structural hydraulic designs and reliable flood prediction. The 1998 report [1] provided equations for determining depths based on the location, scale, and shape parameters. These parameters were displayed on 37 contoured maps of Texas at various durations. The 1998 report was developed using the National Weather Service (NWS) cooperative rain gage data from New Mexico, Oklahoma, Arkansas, Louisiana, and Texas. The data included the same data as used in TP-40 [13] and HY-35 [11], but also included roughly 25 years more rain gage data than either of the NWS provide.

In 2004, the DDF Atlas [2] was created to depict an accurate DDF Atlas of precipitation annual maxima in Texas. Asquith and Roussel took the results of the 1998 report and created atlases based on precipitation depths instead of location, scale, and shape parameters. The 2004 DDF Atlas [2] was developed with precipitation-frequency data that replaced previous data from TP-40 [13] and HY-35 [11]. The 2004 DDF atlas consisted of 96 maps with isopluvial lines at frequencies of 2, 5, 10, 25, 50, 100, 250, and 500 years and at durations of 15 and 30 minutes, 1, 2, 3, 6, and 12 hours, and 1, 2, 3, 5, and 7 days.

Although newer depths for Texas were developed within the 0-6824 research project, a new DDF atlas for Texas has not and most likely will not be commissioned. The 2004 DDF Atlas [2] will most likely be replaced by the National Oceanic and Atmospheric Administration (NOAA) Atlas 14 [20] by mid 2018. NOAA has been working on this project since 2004 to provide updated precipitation frequency estimates for the entire United States. Referred to as NOAA Atlas 14 [20], the Atlas is divided into volumes based on geographic portions of the United States and is to supersede the estimates of TP-40 [13] and HY-35 [11]. NOAA Atlas 14 presents precipitations at more ARIs and durations than previous DDF atlases [1,2,11,13]. Precipitation data is provided at ARIs of 1, 2, 5, 10, 25, 50, 100, 200, 500, and 1000 years, and durations of 5, 10, 15, 30, and 60 minutes, 2, 3, 6, 12, and 24 hours, and 2, 3, 4, 7, 10, 20, 30, 45, and 60 days.

NOAA developed an online Precipitation Frequency Data Server (PFDS) [19], which acts as a point-and-click interface to present the NOAA Atlas 14 precipitation-frequency estimates. A user may enter a location, and select a rain station to view the partial duration series (PDS) or annual maximum series (AMS) based precipitation frequency estimates at 90% confidence intervals. At a single rainfall station, the spreadsheet (*.csv) of the precipitation values (in inches) may be retrieved from the PFDS for external use.

As of October 2015, the NOAA Atlas 14 [20] has covered the majority of the United States, except for the Pacific Northwest and Texas (six states total). The Atlas for Texas, Volume 11, is currently being created and will be completed by mid 2018 [20]. Review of multiple hydraulic design manuals from Departments of Transportation across the United States found that states with completed NOAA 14 Atlases have adopted NOAA's precipitation data. Additionally, the National Resources Conservation Service (NRCS) is adopting NOAA Atlas 14 for all states where it is applicable. Based on the majority of state implementations, it is very probable that NOAA Atlas 14 will eventually become the federal standard adopted throughout the United States.

Although the PFDS online tool [20] provides DDF estimates and curves, it currently does not provide an IDF model. Similar to the previous DDF Atlases [2,11], NOAA Atlas 14 produces DDF estimates at discrete durations and ARIs, which limits engineers to the defined durations. The development of IDF models will be needed for calculation of intensities at arbitrary durations. Without the IDF relationship, engineers will still be limited to the pre-defined durations of NOAA Atlas 14 [19]. Thus the IDF curve fitting process and the methods used in this thesis will be necessary to provide intensities at arbitrary times of concentration.

2.2 Rainfall Coefficients

The first set of *ebd* rainfall coefficients for Texas and the IDF model (Equation 1) were published in 1970 in the first edition of the Hydraulic Manual [28] produced by the Bridge Department of the Texas Highway Department (currently known as TxDOT). The manual provided the IDF model structure (Equation 1) still in use throughout Texas. In 1975, researchers produced a second set of *ebd* coefficients to replace the original table. These updates were incorporated by distribution of revised tables to the Texas districts.

At each Texas county, the *d* coefficient was kept constant across all frequencies while the *e* and *b* coefficients varied. Although the methodology behind the development of the coefficients was not documented, based on the unchanging and pre-set value of *d* (similar to 0-6824 TxDOT method), it can be assumed that the IDF model was linearized and the *e* and *b* coefficients were produced through linear regression.

The process of developing rainfall coefficients through an equation-fitting method for the IDF model (Equation 1) can be found in the Federal Highway Administration's Hydrologic Engineering Circular No. 12 (HEC 12) [15]. The HEC 12 manual was developed to provide design procedures dealing with drainage of highway pavements across the United States. Preceding the age of computational analysis of linear and non-linear programming, Appendix A of HEC 12 described a manual linear regression method of equation-fitting the rainfall coefficients into the power law intensity model (Equation 1). Instead of using the *b, d,* and *e* coefficients within the intensity equation, HEC 12 used the variables *a, b,* and *m*, respectively.

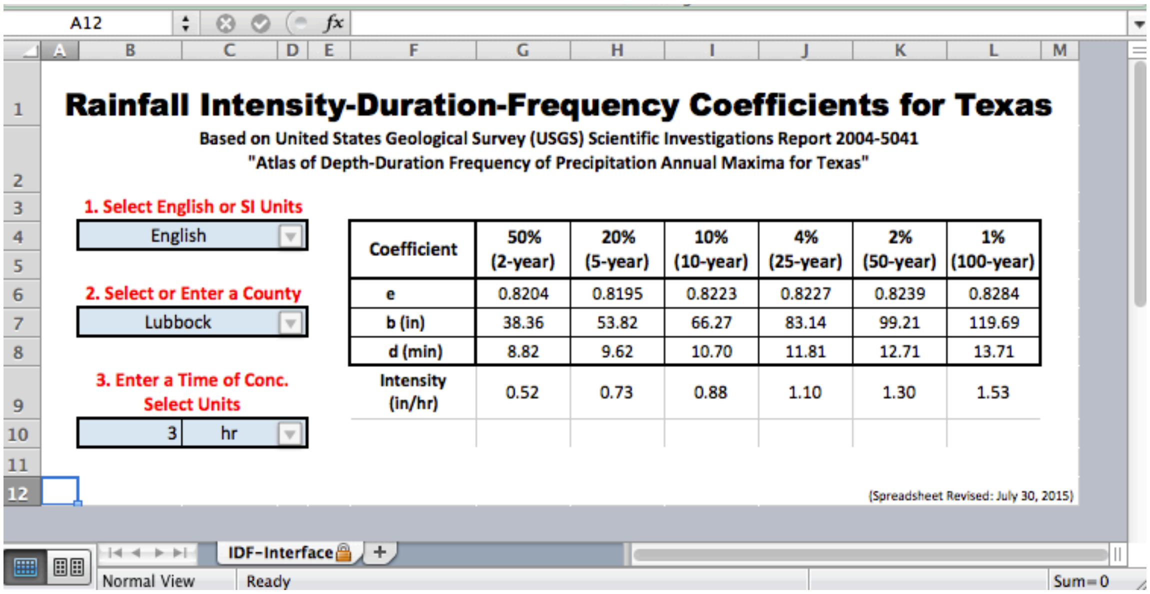
Given duration and intensity of a specified ARI, rainfall coefficients for a specific location within the United States could be determined through fitting the IDF model (Equation 1). Duration and rainfall intensity data were plotted on a 2-cycle logarithmic paper with a best-fit curve drawn through the data points (based on analyst judgment). The *b* (*d)* coefficient was estimated and adjusted until the plotted data points resembled a straight line. Based on the graph, the *a* (*b)* coefficient was determined at a 1 minute duration and the *m (e)* coefficient was equivalent to the slope of the line. The linear regression was then confirmed by checking the predicted values of intensity against the observed values of intensity. If the difference in intensity values were large, changes to the log plot would be necessary to develop a more accurate set of rainfall coefficients. These steps were repeated for all other frequencies and locations.

In 1985, the Bridge Department of the Texas Highway Department produced a Hydraulic Design of Culverts Manual [25] that published a new set of rainfall coefficients for Texas. The development methods for the 1985 coefficients were not documented within the Culvert Design Manual [25]. Contrasted with the table of *ebd* coefficients in 1975, the *d* coefficient varied across each frequency at each county. The *ebd* coefficients were developed based on rainfall frequency data found in TP-40 [13] and HY-35 [11].

Since 1985, the *ebd* coefficients from the Culvert Design Manual [25] have remained unchanged and are still used throughout Texas within the intensity equation (Equation 1). As of October 2015, the 1985 *ebd* coefficients are available in a computational spreadsheet tool for Texas, `EBDLKUP.xls`, which calculates intensity using Equation 1. The coefficients are also used in the drainage software, GEOPAK, by downloading the drainage library (*.dlb) for Texas [38].

In 2015, the 0-6824 TxDOT research project produced a revised set of *ebd* coefficients for the state of Texas based on the 2004 DDF Atlas for Texas [2]. Cleveland and others chose to fit the coefficients to the same IDF model (Equation 1) used for developing coefficients in Texas since at least 1970. The method of OLS regression was conducted through computational analysis to develop the revised 2015 *ebd* coefficients.

The 0-6824 TxDOT research team produced an updated IDF spreadsheet tool for Texas, `EBDLKUP-2015.xlsx`, that estimates intensity (Equation 1) based on the revised 2015 *ebd* coefficients (placed in an underlying database sheet). `EBDLKUP-2015.xlsx` is a very useful tool with a simple interface, as seen in Figure 4. When the *ebd* coefficients are updated in the future, the database in the spreadsheet can be replaced and the spreadsheet tool can continue being used. The 2015 *ebd* coefficients, updated depths, and spreadsheet tool will be implemented through TxDOT by the end of 2015.



Rainfall Intensity-Duration-Frequency Coefficients for Texas

Based on United States Geological Survey (USGS) Scientific Investigations Report 2004-5041
"Atlas of Depth-Duration Frequency of Precipitation Annual Maxima for Texas"

1. Select English or SI Units: English

2. Select or Enter a County: Lubbock

3. Enter a Time of Conc. Select Units: 3 hr

Coefficient	50% (2-year)	20% (5-year)	10% (10-year)	4% (25-year)	2% (50-year)	1% (100-year)
e	0.8204	0.8195	0.8223	0.8227	0.8239	0.8284
b (in)	38.36	53.82	66.27	83.14	99.21	119.69
d (min)	8.82	9.62	10.70	11.81	12.71	13.71
Intensity (in/hr)	0.52	0.73	0.88	1.10	1.30	1.53

(Spreadsheet Revised: July 30, 2015)

Figure 4: Display of `EBDLKUP-2015.xlsx` spreadsheet used for calculating intensity in Texas (adapted from Tay and others, 2015 [41].

The *e, b,* and *d* rainfall coefficients are a unique and significant part of hydraulic design in Texas. Although the IDF model (Equation 1) and its coefficients have existed and been used in Texas for at least 45 years, other states and countries use different IDF models with different rainfall coefficients.

2.3 Intensity-Duration-Frequency Relationships

IDF relationships differ due to varying rainfall patterns at different geographical locations. Unique rainfall observations generate a unique IDF curve, which thus requires a unique IDF model to fit the curve. Design engineers usually prefer IDF relationships because they provide the ability to calculate intensity at arbitrary durations. Some selected IDF models for the United States can be seen in Table 2; notice the wide variance in relationships and coefficients.

Table 2: Examples of intensity-duration-frequency models used in the United States.

Author	IDF Model	Cite
Bernard (1932)	$IDF(D,T;k,a,b) = \frac{kT^a}{D^b}$	[5]
Texas Highway Department (1970)	$IDF_{\mathcal{F}}^{county}(T_c;e,b,d) = K\left(\frac{b}{(T_c+d)^e}\right)$	[28]
Chow and others (1988)	$IDF_{\mathcal{F}}^{county}(T_c;c,E,f) = K\left(\frac{c}{(T_c{}^E+f)}\right)$	[7]
McCuen (1989)	$IDF_{\mathcal{F}}(D;a,b,c,d) = K\begin{cases}\frac{a}{(D+b)} & for\ T_c \leq 2\ hours \\ cD^d\ \ for\ T_c > 2\ hours\end{cases}$	[31]
Wanielista and Eaglin (1996)	$IDF_{\mathcal{F}}(T,D;c,s,d,t) = K\left(\frac{cT^s}{(d+D)^t}\right)$	[30]
Los Angeles Department of Public Works (2006)	$\frac{I_t}{I_{1440}} = \left(\frac{1440}{t_c}\right)^{0.47}$	[16]

In the equations presented in Table 2, *I* or *IDF* is the rainfall intensity in inches per hour, *K* is a unit conversion between U.S. Customary and S.I. units, *T* is the ARI, T_c or t_c is the duration in minutes, *D* is the duration in hours, and variables *a, b, c, d, E, f, k, s,* and *t* are rainfall coefficients. The subscript and superscript for *IDF* imply that the function and corresponding coefficients are a function of frequency ($\mathcal{F}$) and county.

Bernard [5] established the first accounts of an IDF model in 1932. Over the years, multiple IDF relationships have been developed and published in textbooks, and design manuals across the United States. In 1970, the Texas Highway Department published a Hydraulic Manual [28], which defined the IDF model for the state of Texas.[8] In 1988 Chow and others [7] provided an integrated approach to hydrology and provided an IDF model similar in form to Equation 1. Shortly after, in 1989 McCuen [31] published a general IDF model for the United States. McCuen suggested fitting the curve using least squares. In 1996, Wanielista and Eaglin [30] developed an IDF model for the State of Florida, supported by the Florida Department of Transportation. In 2006, the Los Angeles Department of Public Works [16] published a Hydrology Manual that presented their own unique IDF model. The equation is different from the others in that it scales all intensities based on the 24-hour intensity. The constant (1440) in the equation implies that t_c is in units of minutes.

Though the IDF models vary, all of the equations are nonlinear and most are multivariate. The included **R** code within this thesis develops intensity parameters through nonlinear programming. Without the need to linearize the IDF model, a user is able to adopt a different IDF model simply by changing a few lines of code that deal directly with the IDF equation and its corresponding parameters. The ability to quickly adapt the code to a new IDF model and develop corresponding coefficients increases the practicality of the proposed **R** code for use by engineers in and out of Texas.

Based on the code developed in the 0-6824 TxDOT project, if the IDF model structure changes for Texas, the development of new parameters will be time consuming because additional work will need to be done to linearize the new equation (if possible). In the future, new or updated IDF models for the state of Texas are very possible due to climatic change that causes variation

---

[8] The Texas Highway Department (1970) equation is identical to Equation 1, but includes a unit conversion factor that does not need estimation [28].

in rainfall intensity. Another limitation of the 0-6824 TxDOT code is that it is not applicable nationwide because it is only valid for states that use the same IDF model structure (Equation 1).

2.4 Linear Regression

Luenberger [17] stated that in most cases, a function possesses a certain degree of smoothness and thus there exist techniques to exploit that smoothness. Regression analysis is the process of best-fitting a function to a series of data points. Given a series of data, regression provides a method for determining the values of the parameters within the equation to best fit the data.

Linear least squares (or OLS) regression is a method of obtaining the best linear unbiased estimates (BLUE) of the rainfall coefficients. When using least squares, the predicted parameters are usually evaluated based on the PRESS statistic (Equation 3). The objective for the least squares method is to minimize the PRESS value. The TxDOT 0-6824 project used the principle of OLS and the PRESS statistic to develop *ebd* parameters.

Regression analysis is not a recent mathematical discovery. Gauss' work [cited in 12] contributed discoveries that led to the method of least squares, the earliest form of regression in the early 1800's. Galton [cited in 21], a mathematical scientist, recognized the phenomenon of regression toward the mean and coined the term "regression" in 1894. Pearson [21] later extended Galton's work to the general statistical context known today.

Linear regression is viewed as being a robust and efficient computational tool; if a solution exists, it will be found. Additionally linear regression produces a unique solution. Because of these advantages, many prefer linearizing nonlinear equations to be solved by linear regression analysis techniques [18].

The potential problem of the TxDOT 0-6824 code lies not with the method of linear programming but with the tediousness that arises by needing to transform the nonlinear IDF model into a linear model before regression analysis. The manual linearization process lowers the efficiency of the TxDOT code. The OLS method, though sufficient and robust, would be more ideal and efficient if the IDF model was already presented linearly. Almost all IDF models are nonlinear equations that can be solved through nonlinear minimization.

2.5 Non-Linear Regression

A regression model is defined as nonlinear when the functions and their derivatives of the model depend on one or nonlinear combinations of their parameters. Most IDF models are multivariate power-law models that contain multiple rainfall coefficients as products and quotients raised to some power. Methods for fitting non-linear equations are more complex than linear, though not necessarily more difficult.

The objective function for nonlinear regression used within the thesis is to minimize the sum of squared error (SSE). More commonly referred to as a sum of squared errors of prediction (SSE), the SSE is calculated as the sum of squared differences between the observed and predicted values, Equation 5. For developing IDF models, the SSE sums the squared differences between observed depth and the predicted depth at each duration of an ARI.

$$\boldsymbol{SSE = \Sigma_{i=1}^{n} (x_i - f(p_i))^2} \qquad \textbf{Equation 5}$$

where *n* is the number of observations, *i* is the step count, x_i is the the observed 2004 DDF depth [2], and $f(p_i)$ is the predicted depth based on the developed *ebd* parameters, p_i.

Avriel [3] stated that the two most typical unconstrained minimization algorithms included: Cauchy's steepest descent method and Newton's method. A derivative of Newton's method, referred to the Newton-Raphson method, is a popular iterative method used for approximating solutions to systems of nonlinear equations. It optimizes by finding roots of the differentiable function and requires an initial approximation to the solution. At each iteration, values of the parameters are adjusted to minimize the function. When an improvement is not possible, the solution is considered to have converged. If a solution does not converge, the process fails.

Deuflhard [9] stated that Newton's method was a faster method in comparison to gradient descent because it used additional information obtained from the second derivative. For comparison, a simple gradient can compute a tangent line (slope) of an objective function through the first derivative and perform a directional line search. Newton's method additionally accounts for the curvature through a second-order derivative and adjust its step size as appropriate in that direction (accelerating the line search), thus converging in fewer steps.

Luenberger [17] stated that quasi-Newton methods (Newton-based methods) were usually preferred to avoid the Jacobian search for zeroes and avoid the Hessian search for extrema, which required solving a system of linear equations that could be time consuming and costly.[9] The most common quasi-Newton algorithms include: symmetric rank-one (SR1), Berndt-Hall-Hall-Hausman (BHHH), and Broyden-Fletcher-Goldfarb-Shanno (BFGS).

The BFGS method was created in the 1970s and is named after its independent authors: Broyden [33], Fletcher [34], Goldfarb [35], and Shanno [36]. BFGS is considered to be the most effective quasi-Newton method and is commonly used for optimization of equations with a relatively small number (less than a 100) of variables. It is usually faster than other algorithms by requiring less iteration to reach convergence.

In 1985, Schnabel, Koontz, and Weiss [24] laid the foundation for multivariate nonlinear goal programming. Schnabel and others described two new packages, UNCMIN and REVMIN, developed in Fortran. UCMIN was used for finding the local minima of a real valued function with multiple variables. REVMIN used optimization algorithms identical to UNCMIS but was for obtaining values of a user-supplied function by reverse communication.[10] The two packages provided 18 possible algorithmic combinations for nonlinear programming.

The nonlinear minimization (`nlm`) package available in **R** uses a quasi-Newton algorithm for minimizing the objective function and references Schnabel's work [24]. The `nlm` package was created by DebRoy and the **R** Core Team [23]. DebRoy translated pre-existing FORTRAN code by Jones [14] who presented multivariate FORTRAN subroutine code in 1993. Jones [14] developed a nonlinear state space model for the purpose of modeling epidemics and used nonlinear optimization programming to obtain parameter estimates. The **R** manual [23] additionally references a numerical methods book [37] by Schnabel and Dennis, and an unconstrained minimization journal article by Schnabel and others [24].

---

[9] Most practical tools approximate the Jacobian using finite differences, and construct the Hessian during the directional line search – also by finite differences.

[10] Reverse communication is a software technique that has the calling method evaluate a function [40].

Nonlinear programming is highly used and advantageous because of its speed in solving the iterative process. Chong and Zak [6] stated that the most important part of nonlinear programming is the initial parameter estimates provided by the user. Lack of reasonable initial estimates can cause the algorithm to drift off into the inescapable parameter space that does not provide the best estimates. Newton and quasi-newton methods have superior convergence properties when the starting point is near the solution.

The advantages of using nonlinear regression compared to linear regression include faster speeds and smaller errors. The `nlm` algorithm is able to accept any nonlinear equation without the need for linearization and thus accelerates the process of developing rainfall coefficients. Linearization of the equation adds additional steps and time to the solution process. The use of a quasi-Newton method further improves the convergence rate of the function by avoiding the Hessian matrix. Additionally, analysis of the 2-year and 100-year ARI show that the SSE provided by the NLM method is consistently lower than the SEE provided by the OLS method (0-6824 TxDOT [27]).

2.6 Additional Research

For future research, we suggest evaluating other minimization algorithms (packages) offered in statistical programs to find the best available method (lowest SSE). The research should compare the SSEs between the predicted and observed depths (from NOAA Atlas 14).

CHAPTER 3 – METHOD OF ANALYSIS

This chapter presents the two codes designed to develop IDF models based on nonlinear programming. The resulting *ebd* coefficients developed through the NLM method will be used in comparison against OLS method. **R** was chosen for this thesis because of its advanced statistical packages. The `nlm` package was used to minimize the SSE of the predicted and observed depths. Two different **R** scripts were created to develop IDF models. The first code, *SingleLocation,* develops IDF models for a single location and is presented in Appendix F. The second code, *BatchLocation,* was created to develop IDF models for batch locations (such as an entire state) and is presented in Appendix G. Both codes are similar in structure and use the `nlm` package in R to generate rainfall coefficients.

3.1 `nlm` Package in R

In **R**, the `nlm` package is displayed:

```
nlm(f, p, ..., hessian = FALSE, typsize = rep(1, length(p)) fscale = 1, print.level = 0,
    ndigit = 12, gradtol = 1e-6, stepmax = max(1000 * sqrt(sum((p/typsize)^2)),1000),
    steptol = 1e-6,  iterlim = 100, check.analyticals = TRUE)
```

where `f` is the function to be minimized and `p` are the starting parameter values. The rest of the package allows the modeler to format the package to meet certain attributes. The `hessian,` if used provides the hessian of the function at the minimum. `typsize` adjusts the size of each parameter at the minimum. `fscale` is an estimate of the size of `f` at the minimum. `print.level` determines the level of printing. `ndigit` allows the user to set the significant digits of `f`. `gradtol` provides the tolerance to determine when the scaled gradient is close enough to zero to terminate the algorithm. `stepmax` defines the maximum allowable step length and is used to prevent steps that could cause the optimization function to overflow. `steptol` provides the minimum allowable relative step length. `interlim` specifies the maximum number of iterations to be performed before termination of algorithm. `check.analytics` helps to detect incorrectly formulated gradients of Hessians.

In both codes, default values were used except for the renamed function and parameter values, a `steptol` of 1e-10 and a `gradtol` of 1e-6. Research showed that the code worked best at a small step tolerance and thus the value of 1e-10 was implemented throughout. The gradient tolerance of 1e-6 was used to speed up the iteration process.

In using the `nlm` package, the most important attribute was the initial estimated set of the parameter values. Lack of reasonable initial parameter estimates can cause the algorithm to drift off the state space model and provide non-optimal estimates. Based on this knowledge, the code contains a short loop that reuses the last parameter values as the initial parameters within the `nlm` function. The loop provides the ability for an optimal convergence to achieve the smallest sum of squared error at each ARI. The number of iterations was chosen based on analysis that SSE values converged to the smallest number by four iterations. By adding this iteration, we did not have to be as concerned about the initial parameter estimates.

Several numerical experiments were done using different starting parameters in which all runs minimized to the same SSE value after being forced through the refinement loop. Based on this behavior, providing the same initial starting parameters for *e, b,* and *d* was deemed sufficient despite the knowledge of actual relative coefficient values – a majority of ARI data converges with starting parameters of 0.006. The code provides users the ability to change the parameter of each coefficient in prompt five of the “`USER DEFINED DATA`” section.

Two common warning messages may appear when running the `nlm` package. The warnings will appear in a bright red font in the console. The analyst should scan the console after each run. The first is a “`non-finite value supplied by ‘nlm’`” message, as seen below.

```
in nlm(sse, c(x[1], x[2], x[3]), duration, ddfdepth, steptol = 1e-10,  :
  non-finite value supplied by 'nlm'
```

The non-finite message warns the user that the optimization has terminated and failed to produce complete estimates. The warning usually occurs if the initial parameter values are too small. If the warning appears, a remedy is to iteratively change the starting parameters by differences of ±0.001. If the error persists, change the starting parameters by larger differences until convergence is achieved for that particular data set.

The second warning is a “`NA/Inf replaced by maximum positive value`” message, as seen below. The warning will be acknowledged by **`R`** but is not cause for distress or change and can be safely ignored. As the program approaches the minimization of the function and the derivative goes to zero, the division results in an infinite number. The `nlm` package recognizes the zero derivative as a solution and replaces the NA with the next maximum positive value.

```
In nlm(sse, c(X[1], X[2], X[3], X[4]), duration, ddfdepth, T2, steptol = 1e-10,  :
  NA/Inf replaced by maximum positive value
```

The results of the `nlm` package contain the following components accessible after the variable name: `minimum` contains the estimated minimum of the objective function (`f`), `estimate` provides the estimated parameter values within `f`, `gradient` provides the gradient at the estimated minimum `f`, and `code` informs why the optimization process terminated. A component is accessible from a variable by placing the symbol “`$`” followed by the component name. The symbol “`$`” in **`R`** simply takes the selected component from the output variable. The estimated components (fitted rainfall coefficients) was used to generate the output Excel sheet and IDF curve.

The `code` component indicates why an optimization process terminated based on integers from 1-5. Taken directly from the **`R`** `manual` [20], the code numbers and their description are listed in the table below.

Table 3: Code numbers and description available in the `nlm` package in R [20].

Code No.	Description
1	relative gradient is close to zero, current iterate is probably solution.
2	successive iterates within tolerance, current iterate is probably solution.
3	last global step failed to locate a point lower than estimate. Either estimate is an approximate local minimum of the function or steptol is too small.
4	iteration limit exceeded.
5	maximum step size stepmax exceeded five consecutive times. Either the function is unbounded below, becomes asymptotic to a finite value from above in some direction or stepmax is too small.

The code numbers and termination descriptions of the `nlm` package do not appear in either **`R`** scripts. The integers were used throughout research and testing to ensure that the parameters were being met at probable solutions. A code of 1 or 2 was deemed acceptable and a code of 3 instigated a change in `steptol` size. The user only has to be concerned with the red warning messages appearing in the console.

3.2 Model Development in `RStudio`

A brief overview of the main functions used within the *SingleLocation* and *BatchLocation* codes will be discussed within this section.[11] Both codes share similar structure and use the `nlm` package in **`R`** to minimize SSE and develop IDF models. The main differences between the two codes occur because of the format of the spreadsheet files containing rainfall data. *SingleLocation* develops coefficients for a single point location at varying durations for each ARI, and *BatchLocation* obtains coefficients for multiple locations at varying durations for a single ARI. Both codes were intentionally written to be overly detailed (by use of line feeds and comments) in an effort to increase maintainability for future changes; *SingleLocation* has approximately 300 lines of code. If the **`R`** scripts were designed to fit a single unique spreadsheet and IDF model, the script could be very simple and succinct.

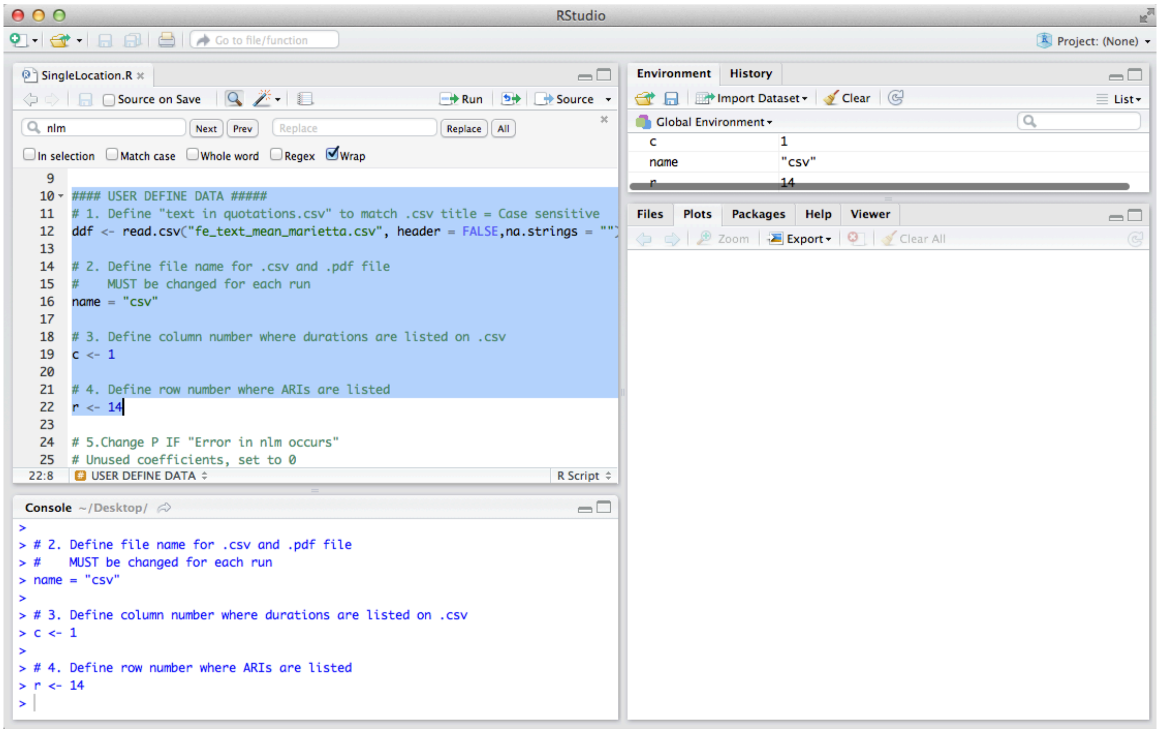
The codes account for variations in spreadsheet data format and contain functions that extract listed durations, and ARIs (*SingleLocation*) or locations (*BatchLocation*) from the data file. Both codes anticipate changes to the IDF model by providing additional rainfall coefficients (presented in various IDF models) to the functions; unused variables are set to zero and ignored until needed.

Although the code is designed to fit a majority of IDF models, it is possible that a newer IDF model will require substantial rewriting of the code. The code was written entirely with pointer variables (located at the beginning) for easy changes and maintenance.

---

[11] The programming details of the two proposed codes will not be discussed within this thesis to avoid any resemblance of a technical reference manual. The thesis is meant to present and provide a general guidance of how to use the codes.

`RStudio` [39] is an interface for using **`R`** [22] and is suggested for running the *SingleLocation* and *BatchLocation* codes. When initially opened, the program should look similar to Figure 5.



nlm

☐ In selection ☐ Match case ☐ Whole word ☐ Regex ☑ Wrap

```
#### USER DEFINE DATA #####
# 1. Define "text in quotations.csv" to match .csv title = Case sensitive
ddf <- read.csv("fe_text_mean_marietta.csv", header = FALSE,na.strings = ""

# 2. Define file name for .csv and .pdf file
#    MUST be changed for each run
name = "csv"

# 3. Define column number where durations are listed on .csv
c <- 1

# 4. Define row number where ARIs are listed
r <- 14

# 5.Change P IF "Error in nlm occurs"
# Unused coefficients, set to 0
```

```
>
> # 2. Define file name for .csv and .pdf file
> #    MUST be changed for each run
> name = "csv"
>
> # 3. Define column number where durations are listed on .csv
> c <- 1
>
> # 4. Define row number where ARIs are listed
> r <- 14
>
```

c	1
name	"csv"
r	14

Figure 5: Screenshot of `RStudio` program when initially opened.

The top left panel is referred to as the RScript; it is a color-coded text editor that sends commands to **`R`**. The bottom left panel is the console that displays code that has been ran through in `blue` and resulting warning messages from the code in `red`. Code can be sent from the RScript to the console by selecting and highlighting the code (Figure 5), and clicking the Run button: Run located above the Rscript panel. The top right panel is the Workspace where objects will be displayed. The bottom right panel serves many purposes, but will not be used for the presented codes. A user will need to interact directly with the RScript and console panels.

The *SingleLocation* and *BatchLocation* codes are presented in color in Appendix F and G. Code in `green`, followed by a `#` symbol, are comments used to describe the code or to provide prompts for a user to follow. Code in `blue` represents constants and objects reserved in **`R`**. The codes are sectioned off by `“CAPITALIZED HEADERS”` and surrounded by multiple `#` symbols; additional comments clarify what the section of code accomplishes.

In both codes, five numbered directions were provided for users to follow in the "`USER DEFINED DATA`" section. Additionally, to change the IDF model, four detailed prompts were given in the "`IDF MODEL STRUCTURE`" section. These two sections of code were intended to be the only user-interactive portion of the code. The remainder of the code should not need to be changed and should be left alone. After reading and responding to the prompts, to run the code, select all of the code (Ctrl+A) within the Rscript panel, and click the run button. When the script has finished the run, two files should appear in the working directory. The first output file is a .csv spreadsheet containing rainfall coefficients at various ARIs (*SingleLocation*) or various locations (*BatchLocation*). The second output file is a .pdf file that provides the IDF curves of the locations.

The IDF model (Equation 1) currently in the code is identical to the model presented within the TxDOT Design Manual [26]. The model is written as an intensity function:

```
intensity <- function(T1,tc,x1,x2,x3,x4) #IDF model
 {
  int <- x2/((x3+tc)^x1) #in/hr
  return(int)
  }
```

where *x1*, *x2*, and *x3* are the parameters *e*, *b*, and *d*, and t_c is time of concentration in minutes. The additional variables, *T1* (ARI) and *x4* (addition coefficient), are placeholders for other IDF models. Variables that are not used are simply set to zero and do not participate in the calculations.

With a defined IDF model, the variables, depths and SSE need to be calculated. The SSE function takes the difference squared of the observed 2004 DDF value [2] and the depth calculated by the predicted variables. The SSE function will be the minimized objective function to produce the best fit. The SSE is written as:

```
sse <- function(x,duration,ddfdepth,T1) #Sum of Errors Squared Function (SSE)
    {
    sum((ddfdepth - depth(T1,duration,x[1],x[2],x[3],x[4]))^2)
    }
```

The optimization of SSE is done through the `nlm` package provided by **R**:

```
nlm(SSE, c(x[1],x[2],x[3],x[4]),duration, ddfdepth, T1, hessian= FALSE, steptol= 1e-10, gradtol= 1e-6)
```

where SSE is the optimizing function and the *e, b,* and *d* parameters are represented as the subscripted variables, x[1], x[2], and x[3]. Variable x[4] is an additional variable for a different IDF model structure and is currently set to zero. Duration refers to the list of durations presented in the data file, ddfdepth refers to the list of depths from the 2004 DDF Atlas [2], and T1 is an additional variable for a different IDF model that refers to the of ARIs from the data file; it is unused in this IDF structure and is therefore set to zero.

The code then proceeds to iterate through each DDF depth from the provided data file and minimize the objective function to produce estimates of *ebd* at each ARI. At the end of the run, the produced coefficient values are written into a csv file in a single column. Additionally, a pdf of a produced IDF curve is created. Figure 6 is a flowchart of the major portions of the **R** code process. Table 4 describes the shapes used within the flowchart and their meanings.

Table 4: Flowchart shapes and designated descriptions.

Shape	Description
▢	Designates the entry and exit point of the flowchart
▱	Defines input or output
▭	Main process or task
◇	Asks a question and makes a yes or no decision
→	Determines flow through the chart

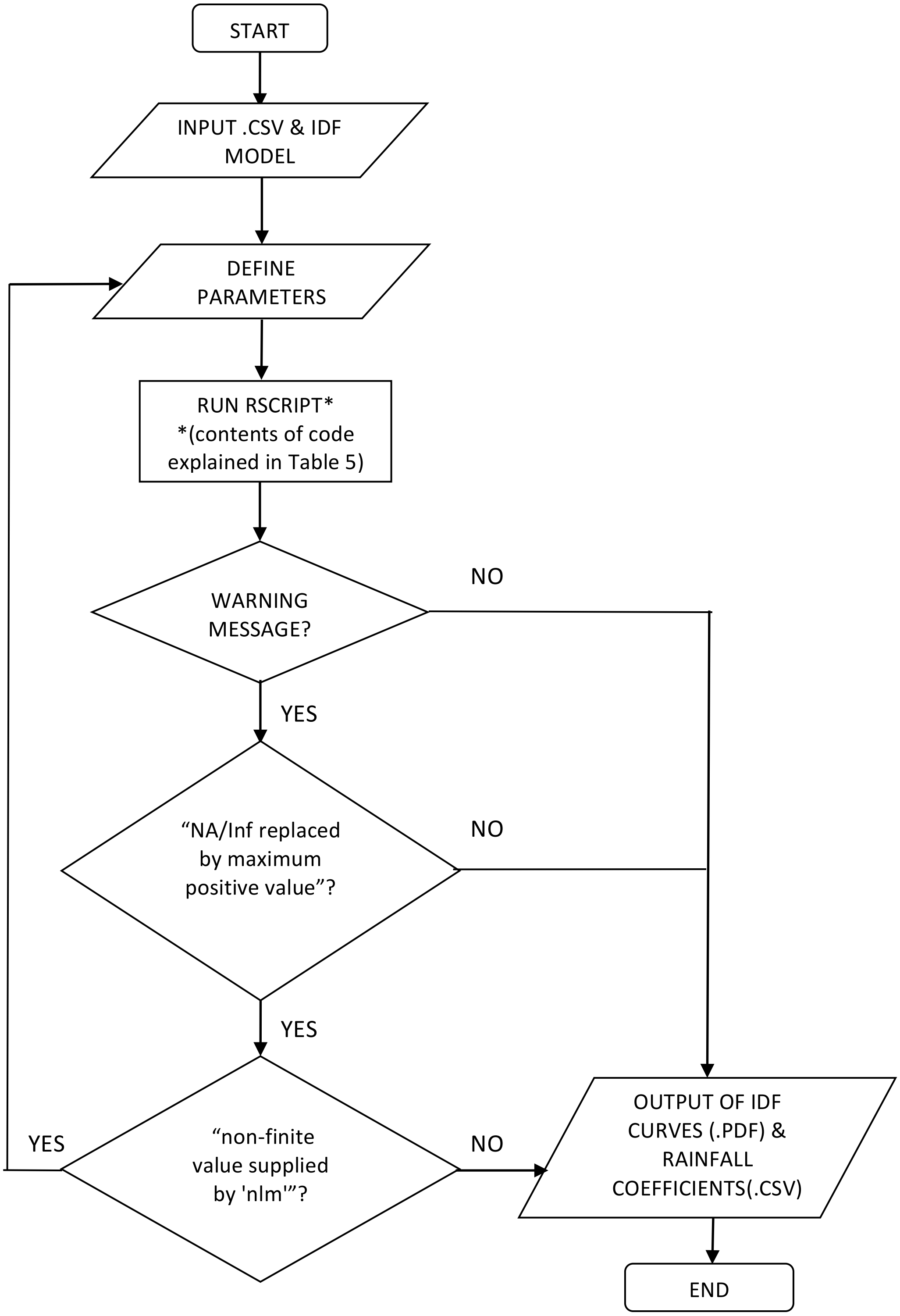


Figure 6: Process flowchart of R scripts for developing IDF models.

The simplified pseudo code of the main “RUN SCRIPT” task of the flow chart is described in order within Table 5.

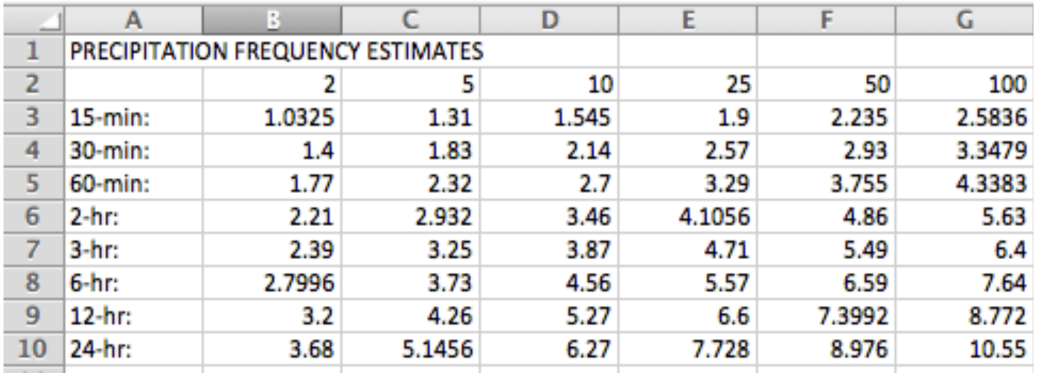
Table 5: Psuedo code of SingleLocation and BatchLocation scripts.

1	Load DDF data (.csv)
2	Initial estimates of rainfall parameters from user
3	Calls `nlm` — minimizes SSE
4	Recovers fitted rainfall coefficient values
5	Generates revised DDF and IDF at each duration (*Tc*)
6	Outputs coefficients (.csv) and IDF curves (.pdf)

The script loads the provided DDF data (observed values) as a .csv and defines the initial estimates of rainfall parameters. The `nlm` package is then used to minimize the SSE function and generate fitted rainfall coefficients. The coefficients are then plugged back into the depth and intensity equations to generate revised values. The revised intensity values are plotted onto the generated IDF curve.

3.3 *SingleLocation* Code

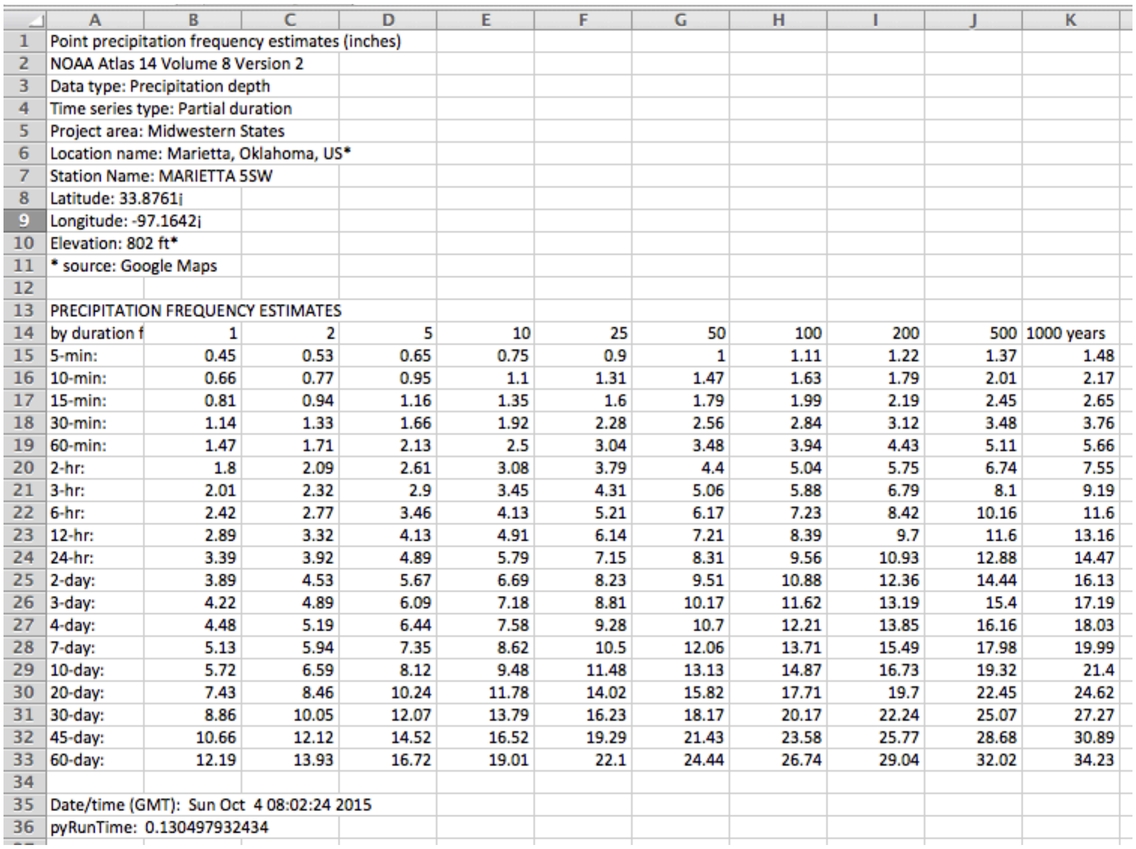
The *SingleLocation* code was written to be able to attach any DDF data file (.csv) following the format of labeling durations in a column and labeling ARIs across a row; the column and row numbers can vary. Additionally, the number of listed durations and ARIs can vary. User should ensure that durations in the data file should always have associated units of “min,” “hr,” or “day.” An example of an acceptable data file is seen in Figure 7.



	A	B	C	D	E	F	G
1	PRECIPITATION FREQUENCY ESTIMATES						
2		2	5	10	25	50	100
3	15-min:	1.0325	1.31	1.545	1.9	2.235	2.5836
4	30-min:	1.4	1.83	2.14	2.57	2.93	3.3479
5	60-min:	1.77	2.32	2.7	3.29	3.755	4.3383
6	2-hr:	2.21	2.932	3.46	4.1056	4.86	5.63
7	3-hr:	2.39	3.25	3.87	4.71	5.49	6.4
8	6-hr:	2.7996	3.73	4.56	5.57	6.59	7.64
9	12-hr:	3.2	4.26	5.27	6.6	7.3992	8.772
10	24-hr:	3.68	5.1456	6.27	7.728	8.976	10.55

Figure 7: Example of .csv file for *SingleLocation* code with duration in column 1 and ARI in row 2. DDF values are extracted from the 2004 DDF Atlas [2] for Anderson County, Texas.

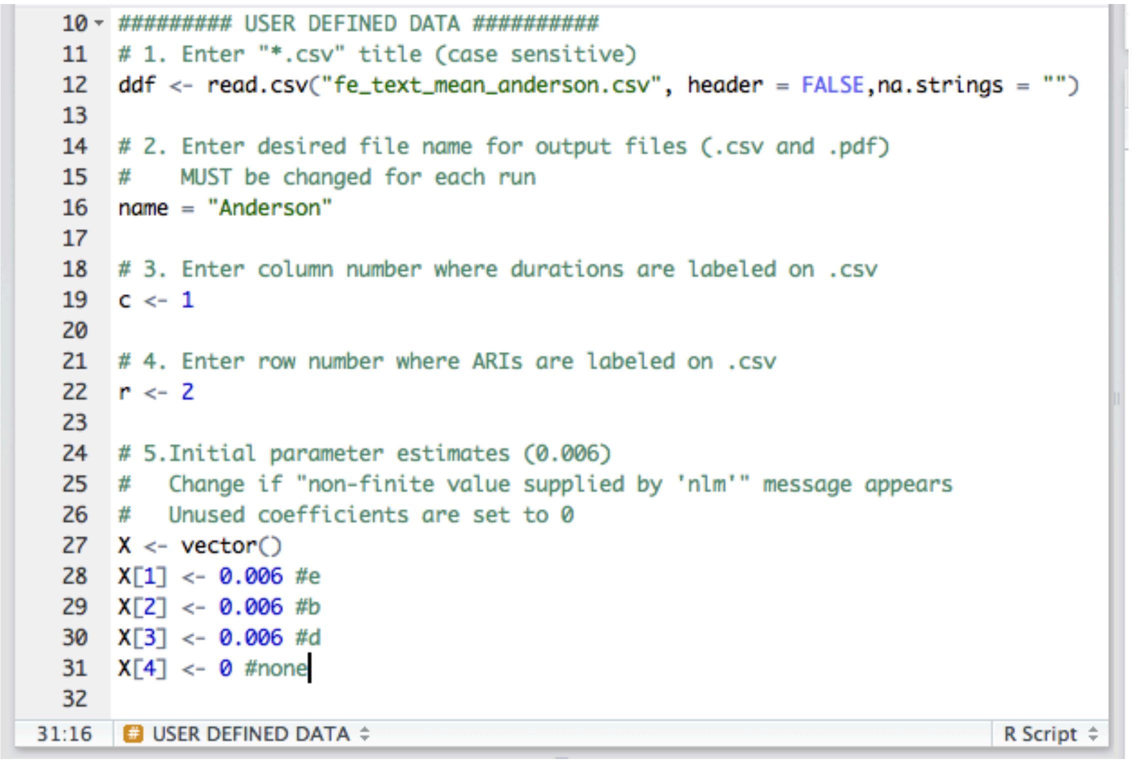
The *SingleLocation* code is preset to the format of the data file extracted from NOAA's PFDS online tool [20], as seen in Figure 8. The DDF data file should be kept in a .csv format and placed within the working directory of the code. We suggest renaming the data file with the appropriate location for distinction among other data files. The *SingleLocation* code uses the data from the data file to develop IDF rainfall coefficients for the specified location at each ARI.



	A	B	C	D	E	F	G	H	I	J	K
1	Point precipitation frequency estimates (inches)										
2	NOAA Atlas 14 Volume 8 Version 2										
3	Data type: Precipitation depth										
4	Time series type: Partial duration										
5	Project area: Midwestern States										
6	Location name: Marietta, Oklahoma, US*										
7	Station Name: MARIETTA 5SW										
8	Latitude: 33.8761¡										
9	Longitude: -97.1642¡										
10	Elevation: 802 ft*										
11	* source: Google Maps										
12											
13	PRECIPITATION FREQUENCY ESTIMATES										
14	by duration f	1	2	5	10	25	50	100	200	500	1000 years
15	5-min:	0.45	0.53	0.65	0.75	0.9	1	1.11	1.22	1.37	1.48
16	10-min:	0.66	0.77	0.95	1.1	1.31	1.47	1.63	1.79	2.01	2.17
17	15-min:	0.81	0.94	1.16	1.35	1.6	1.79	1.99	2.19	2.45	2.65
18	30-min:	1.14	1.33	1.66	1.92	2.28	2.56	2.84	3.12	3.48	3.76
19	60-min:	1.47	1.71	2.13	2.5	3.04	3.48	3.94	4.43	5.11	5.66
20	2-hr:	1.8	2.09	2.61	3.08	3.79	4.4	5.04	5.75	6.74	7.55
21	3-hr:	2.01	2.32	2.9	3.45	4.31	5.06	5.88	6.79	8.1	9.19
22	6-hr:	2.42	2.77	3.46	4.13	5.21	6.17	7.23	8.42	10.16	11.6
23	12-hr:	2.89	3.32	4.13	4.91	6.14	7.21	8.39	9.7	11.6	13.16
24	24-hr:	3.39	3.92	4.89	5.79	7.15	8.31	9.56	10.93	12.88	14.47
25	2-day:	3.89	4.53	5.67	6.69	8.23	9.51	10.88	12.36	14.44	16.13
26	3-day:	4.22	4.89	6.09	7.18	8.81	10.17	11.62	13.19	15.4	17.19
27	4-day:	4.48	5.19	6.44	7.58	9.28	10.7	12.21	13.85	16.16	18.03
28	7-day:	5.13	5.94	7.35	8.62	10.5	12.06	13.71	15.49	17.98	19.99
29	10-day:	5.72	6.59	8.12	9.48	11.48	13.13	14.87	16.73	19.32	21.4
30	20-day:	7.43	8.46	10.24	11.78	14.02	15.82	17.71	19.7	22.45	24.62
31	30-day:	8.86	10.05	12.07	13.79	16.23	18.17	20.17	22.24	25.07	27.27
32	45-day:	10.66	12.12	14.52	16.52	19.29	21.43	23.58	25.77	28.68	30.89
33	60-day:	12.19	13.93	16.72	19.01	22.1	24.44	26.74	29.04	32.02	34.23
34											
35	Date/time (GMT): Sun Oct 4 08:02:24 2015										
36	pyRunTime: 0.130497932434										

Figure 8: Example of .csv file for *SingleLocation* code with duration in column 1 and ARI on row 14. DDF values from NOAA PFDS online tool [20] for rainfall station: Marietta SSW in Marietta, Oklahoma.

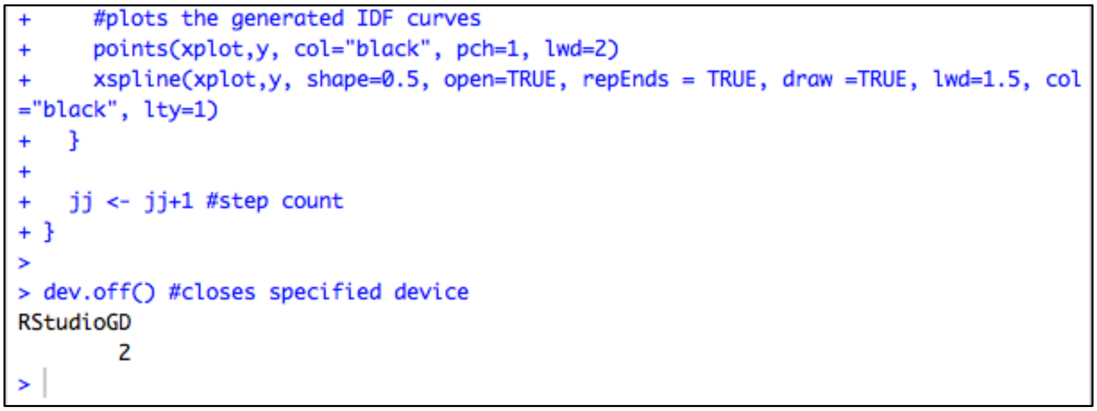
The code was written to accept multiple formats of data files based on the general guidelines. Within the "`USER DEFINED DATA`" section, the five prompts ask the user to enter the file name of the attached data file, provide the desired file name for the output files, enter the column number where durations are labeled, enter the row number where ARIs are labeled, and define the initial parameter estimates for the `nlm` package. The initial parameter estimates only need to be changed when the "`non-finite value`" warning message appears in the console after a run. An example of the filled "`USER DEFINED DATA`" section for the data file of Figure 7 (Anderson County, Texas) is seen in Figure 9.



```
########## USER DEFINED DATA ##########
# 1. Enter "*.csv" title (case sensitive)
ddf <- read.csv("fe_text_mean_anderson.csv", header = FALSE,na.strings = "")

# 2. Enter desired file name for output files (.csv and .pdf)
#    MUST be changed for each run
name = "Anderson"

# 3. Enter column number where durations are labeled on .csv
c <- 1

# 4. Enter row number where ARIs are labeled on .csv
r <- 2

# 5.Initial parameter estimates (0.006)
#   Change if "non-finite value supplied by 'nlm'" message appears
#   Unused coefficients are set to 0
X <- vector()
X[1] <- 0.006 #e
X[2] <- 0.006 #b
X[3] <- 0.006 #d
X[4] <- 0 #none

```

Figure 9: Screenshot of the "`USER DEFINED DATA`" section for the *SingleLocation* script in `Rstudio`. Prompts are filled out based on data file for Anderson County, Texas (Figure 7).

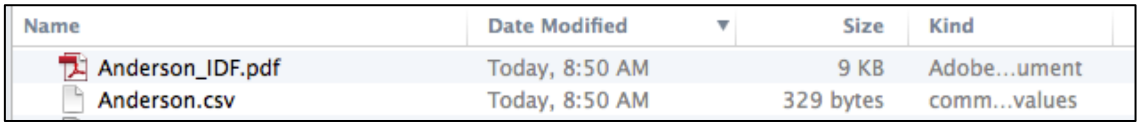
With the information defined by the user in the "`USER DEFINED DATA`" section, the script searches through the respective column and row to retrieve the exact values of durations and ARIs in the data file. Additionally, the script converts all durations into minutes based upon the labeled units of "min," "hr," or "day." When the code is run without warning messages, the console should look similar to Figure 10.



```
+     #plots the generated IDF curves
+     points(xplot,y, col="black", pch=1, lwd=2)
+     xspline(xplot,y, shape=0.5, open=TRUE, repEnds = TRUE, draw =TRUE, lwd=1.5, col
="black", lty=1)
+   }
+
+   jj <- jj+1 #step count
+ }
>
> dev.off() #closes specified device
RStudioGD
        2
> 
```

Figure 10: Screenshot of the `RStudio` console when the *SingleLocation* code is run without warnings.

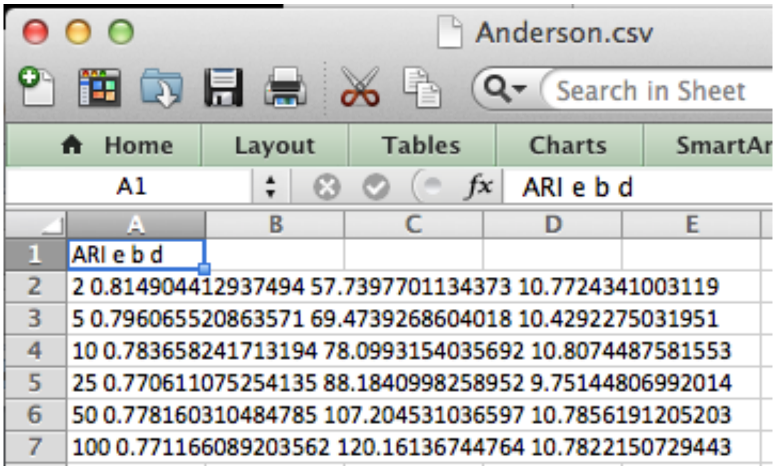
Figure 11 is a screen capture of the working directory showing that after the program is run, the generated output (pdf and csv) files are placed in the working directory.



Name	Date Modified	Size	Kind
Anderson_IDF.pdf	Today, 8:50 AM	9 KB	Adobe...ument
Anderson.csv	Today, 8:50 AM	329 bytes	comm...values

Figure 11: Screenshot of generated output files (.pdf and .csv) in the working directory.

The generated csv output file provides the rainfall coefficients generated from the `nlm` package. The output data is provided in a single column in Excel; to separate the data across multiple columns, use Excel's Text-To-Columns data tool. These coefficients can be used to generate intensities based on ARI for the respective location. Figure 12 displays the csv output file with *ebd* coefficients for Anderson County, Texas at varying ARIs.



```
1  ARI e b d
2  2 0.814904412937494 57.7397701134373 10.7724341003119
3  5 0.796065520863571 69.4739268604018 10.4292275031951
4  10 0.783658241713194 78.0993154035692 10.8074487581553
5  25 0.770611075254135 88.1840998258952 9.75144806992014
6  50 0.778160310484785 107.204531036597 10.7856191205203
7  100 0.771166089203562 120.16136744764 10.7822150729443
```

Figure 12: Screenshot of csv output file with predicted rainfall coefficients for Anderson County, Texas.

The generated pdf output file provides IDF curves for Anderson county at varying ARIs. The IDF curves are created based on the predicted coefficients from the `nlm` method. The x-axis displays the logarithm of duration in minutes and the y-axis displays the logarithm of intensities in inches per hour. The curves are stacked from smaller to larger ARIs. Figure 13 shows the generated IDF curve for Anderson County, Texas at the 2, 5, 10, 25, 50, and 100-year ARIs.

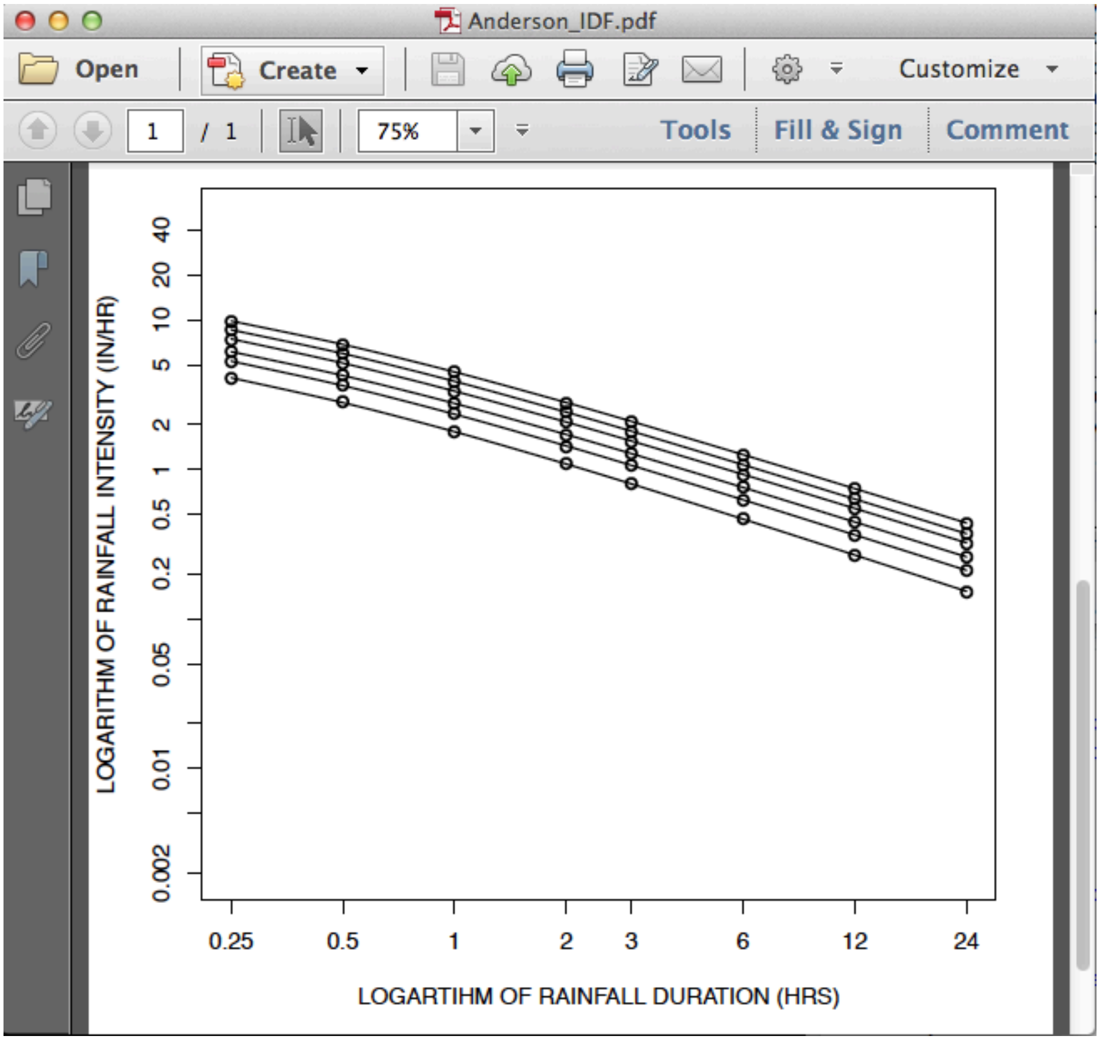
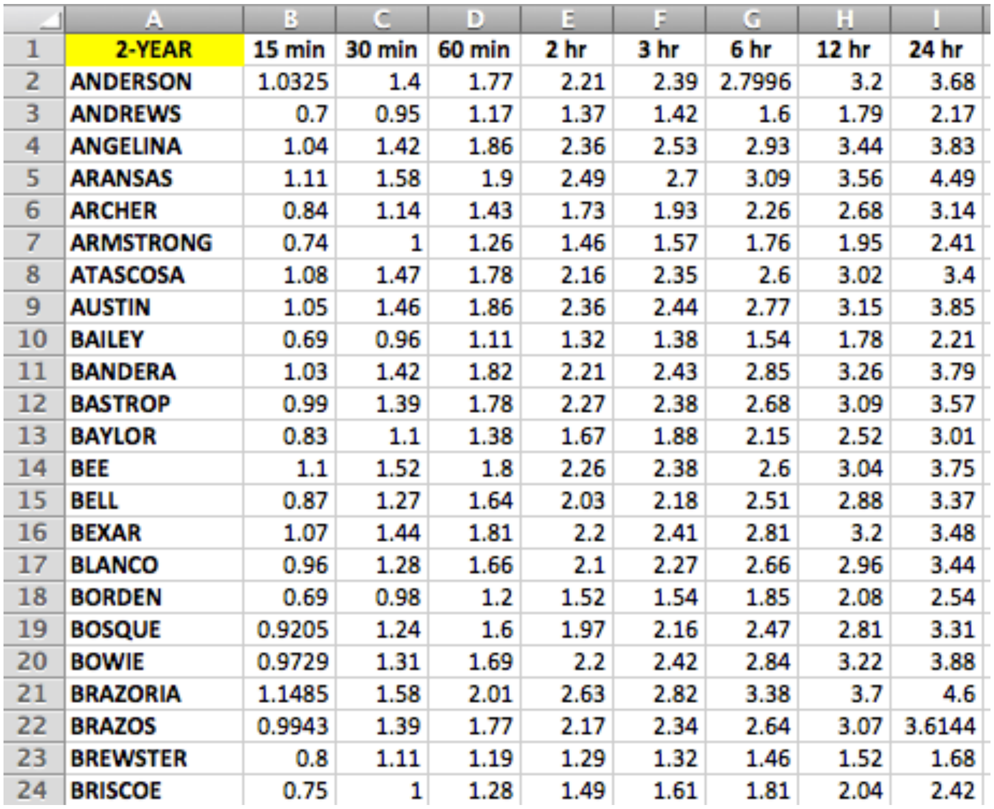


Figure 13: Screenshot of pdf output file with IDF curves for Anderson County, Texas.

3.4 BatchLocation Code

The *BatchLocation* code is almost identical in structure to the *SingleLocation* code. The main differences between the codes are caused by the difference in the input data file formats. *BatchLocation* was written to accept rainfall data for multiple rain stations or for an entire state. The code produces rainfall coefficients at various ARIs for multiple rainfall stations or counties. With the additional data from multiple locations, we decided to separate the data files by ARI instead of by location.

Figure 14 is a portion of an input file showing the data format for which the the code was designed. The code will accept data files with county or location labeled in a column and durations labeled across a row.



	A	B	C	D	E	F	G	H	I
1	2-YEAR	15 min	30 min	60 min	2 hr	3 hr	6 hr	12 hr	24 hr
2	ANDERSON	1.0325	1.4	1.77	2.21	2.39	2.7996	3.2	3.68
3	ANDREWS	0.7	0.95	1.17	1.37	1.42	1.6	1.79	2.17
4	ANGELINA	1.04	1.42	1.86	2.36	2.53	2.93	3.44	3.83
5	ARANSAS	1.11	1.58	1.9	2.49	2.7	3.09	3.56	4.49
6	ARCHER	0.84	1.14	1.43	1.73	1.93	2.26	2.68	3.14
7	ARMSTRONG	0.74	1	1.26	1.46	1.57	1.76	1.95	2.41
8	ATASCOSA	1.08	1.47	1.78	2.16	2.35	2.6	3.02	3.4
9	AUSTIN	1.05	1.46	1.86	2.36	2.44	2.77	3.15	3.85
10	BAILEY	0.69	0.96	1.11	1.32	1.38	1.54	1.78	2.21
11	BANDERA	1.03	1.42	1.82	2.21	2.43	2.85	3.26	3.79
12	BASTROP	0.99	1.39	1.78	2.27	2.38	2.68	3.09	3.57
13	BAYLOR	0.83	1.1	1.38	1.67	1.88	2.15	2.52	3.01
14	BEE	1.1	1.52	1.8	2.26	2.38	2.6	3.04	3.75
15	BELL	0.87	1.27	1.64	2.03	2.18	2.51	2.88	3.37
16	BEXAR	1.07	1.44	1.81	2.2	2.41	2.81	3.2	3.48
17	BLANCO	0.96	1.28	1.66	2.1	2.27	2.66	2.96	3.44
18	BORDEN	0.69	0.98	1.2	1.52	1.54	1.85	2.08	2.54
19	BOSQUE	0.9205	1.24	1.6	1.97	2.16	2.47	2.81	3.31
20	BOWIE	0.9729	1.31	1.69	2.2	2.42	2.84	3.22	3.88
21	BRAZORIA	1.1485	1.58	2.01	2.63	2.82	3.38	3.7	4.6
22	BRAZOS	0.9943	1.39	1.77	2.17	2.34	2.64	3.07	3.6144
23	BREWSTER	0.8	1.11	1.19	1.29	1.32	1.46	1.52	1.68
24	BRISCOE	0.75	1	1.28	1.49	1.61	1.81	2.04	2.42

Figure 14: Example of.csv file for *BatchLocation* code with county in column A and duration on row 1.

NOAA Atlas 14 currently does not offer statewide depth values on the PFDS tool [20]. To retrieve depth values for multiple locations or an entire state, ArcGIS software is needed to import, average, and extract gridded NOAA Atlas 14 depths by location centroids. NOAA Atlas 14 provides gridded Atlases that are broken down by ARI (1 to 1000 years) and duration (5-min to 60-day), for a total of 190 maps. Each map needs to be imported into ArcGIS to have its data extracted at the centroid of the desired locations. When the data is extracted, an analyst will need to transform the data file to match the acceptable data file format for the *BatchLocation* code(Figure 14).

At a completed run, the *BatchLocation* code provides a csv of rainfall coefficients at each location. The generated csv output files can be combined and placed into the database of a computational spreadsheet tool for estimating intensities, such as `EBDLKUP-2015.xlsx`.

CHAPTER 4 – RESULTS AND CONCLUSIONS

The OLS and NLM methods were compared by taking the difference between their SSEs for each county in Texas at the 2-year and 100-year ARIs (common ARIs used in design). If both methods were identical, these differences would be zero. A positive difference value showed that the NLM method produced a smaller SSE, and a negative difference value showed that OLS produced a smaller SSE. The generated *e, b,* and *d* values from each method were placed into the IDF equation (Equation 1) to generate intensities. The intensities at a 3-hour duration were converted to depths to further the difference in depths between each method and the observed DDF values [2].

To analyze the OLS method against the NLM method, the percent variance in *ebd* parameters, and difference in SSE, depth, intensity, and IDF curves developed from each method were compared at the 2-year and 100-year ARIs for all Texas counties. Both methods develop the IDF equation by fitting a line through the given DDF data series.

The optimized depths of both methods were computed against the original depth values from the 2004 Texas DDF Atlas [2]. The OLS method generated a greater SSE for both the 2-year and 100-year ARI for almost all Texas Counties. Table 1 below shows a portion of the completed counties with the difference in error between the OLS and NLM methods compared. The positive values reflect a higher error from the OLS method and the smaller values reflect minimal differences in error between the two methods.

Table 6: Error between OLS and NLM method for the 2-year and 100-year ARI.

COUNTY	ΔSSE (OLS-NLM)	
	2-YEAR	100-YEAR
ANDERSON	0.000669	0.019897
ANDREWS	0.001524	0.001837
ANGELINA	0.000835	0.068437
ARANSAS	0.002808	0.267218
ARCHER	0.000171	0.002073
ARMSTRONG	0.001445	0.000981
ATASCOSA	0.000160	0.158181
AUSTIN	0.001669	0.230319

BAILEY	0.003986	0.031629
BANDERA	0.000039	0.275041
BASTROP	0.000164	0.354357
BAYLOR	0.000054	0.168307
BEE	0.004549	0.190325
BELL	0.000453	0.351760
BEXAR	0.001079	0.311315
BLANCO	0.001488	0.150790
BORDEN	0.001546	0.083520
BOSQUE	0.000391	0.728069
BOWIE	0.006043	3.639957
BRAZORIA	0.000586	22.966024
BRAZOS	0.002533	0.441092
BREWSTER	0.001633	0.107387
BRISCOE	0.000452	0.003321
BROOKS	0.016455	1.281057
BROWN	0.002442	0.050799
BURLESON	0.001768	0.375674
BURNET	0.000008	0.228454

Out of the 254 Texas counties, `nlm` had a lower error approximately 98.4% of the time. Appendix C lists the entire table with all of the Texas counties. Nearly all of the differences in errors between the both methods are less than 1. The goal of the optimization was to produce the lowest error possible to produce a best-fit curve. An interesting thing to note is that the `ols` method for the 100-year ARI produced some errors significantly greater than 1; some upwards of 20. To compare, the maximum error for the `ols` and `nlm` method at the 2-year ARI were 0.159 and 0.091, respectively, and the maximum error for the `ols` and `nlm` method at the 100-year ARI were 23.178 and 1.444, respectively. The cause for such high error is attributed to the actual method used by the 0-6824 project's algorithm and regression method.

Based on the results and comparison, the `nlm` method produces smaller error than `ols`. By removing the need to linearize a nonlinear IDF model, the `nlm` package is a speedier algorithm for evaluating rainfall coefficients. In addition to efficiency, the `nlm` package increases generality of IDF evaluating codes by anticipating for future changes in IDF models within the code.

The **R** script provided by this thesis expects changes for the IDF model and nation-wide adoption of the NOAA Atlas 14. By anticipating for other IDF models, the code becomes more universal and easily adaptable for any future changes. The code itself was not written only for the state of Texas, but can be used worldwide. However if the IDF model is drastically different, the user will have to refactor portions of the code to accept the new model. Additionally, the code was written to receive data based on the data supplied by NOAA Atlas 14. We highly recommend adopting the precipitation frequencies for Texas when it becomes available in 2018.

Aside from generality and efficiency, the `nlm` package provides a lower sum of squared predictions error (SSE) than the linearization method and thus as far as this metric is a better fit for nonlinear IDF models. The efficiency, low error and generality allowed and provided by the `nlm` package in **R**, and the code provided in this thesis make it a robust and expandable tool for use in evaluating IDF models for rainfall coefficients.

REFERENCES

[1] Asquith, W.H., 1998, Depth-duration frequency of precipitation for Texas: U.S. Geological Survey Water-Resources Investigations Report 98–4044, 107 p., http://pubs.usgs.gov/wri/wri98-4044/.

[2] 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., http://pubs.usgs.gov/sir/2004/5041.

[3] Avriel, M. *Nonlinear Programming: Analysis and Methods*. Englewood Cliffs, NJ: Prentice-Hall, 1976. 287-298 p.

[4] Bates, D.M., and Watts, D.G., 2007, Nonlinear regression analysis and its applications: New York, Wiley, ISBN 978-047013900–4.

[5] Bernard, M. 1932. Formulas for rainfall intensities of long durations. TRANS. ASCE. Vol. 96. 592-624 p.

[6] Chong, E.K.P., and Zak, S.H., 1996, *An Introduction to Optimization*. New York: Wiley.

[7] Chow, V.T., Maidment, D.R., and Mays, L.W., 1988, Applied Hydrology: New York, McGraw-Hill.

[8] Dennis, J. E. and Schnabel, R. B. (1983) *Numerical Methods for Unconstrained Optimization and Nonlinear Equations.* Prentice-Hall, Englewood Cliffs, NJ.

[9] Deuflhard, P, 2010, Newton Methods for Nonlinear Problems. Affine Invariance and Adaptive Algorithms, Computational Mathematics, Vol. 35. Springer International, 2004.

[10] Fiacco, A.V., and McCormick, G.P., 1964. Computational Algorithm for the Sequential Unconstrained Minimization Technique for Nonlinear Programming: Journal of Management Science,Vol. 10, No. 4, 601-617 p.

[11] Frederick, R.H., Myers, V.A., and Auciello, E.P., Five to 60-minute precipitation frequency for the eastern and central United States, NOAA technical memo NWS HYDRO-35, National Weather Service, Silver Spring, Maryland, June 1977.

[12] Nievergelt, Y., 2001, A Tutorial History of Least Squares with Applications to Astronomy and Geodesy, Numerical Analysis: Historical Developments in the 20th Century, 77-112. http://www.sciencedirect.com/science/article/pii/S0377042700003435

[13] Hershfield, D.M., 1961, Rainfall Frequency Atlas of the United States. U.S. Department of Commerce, Weather Bureau, Technical Paper No. 40.

[14] Jones, R.H., 1993, Longitudinal Data with Serial Correlation: A State-Space Approach,

Portland, OR, Book News, Inc., 163-200 p.

[15] Johnson, F.L., and Chang F. F-M., 1984, Drainage of Highway Pavements. FHWA-TS-84-202, HEC 12, Federal Highway Administration, Virginia 22101, 151 p.

[16] Conkle, C., Moyer, J., Willardson, B., Walden, A., and Nasseri, I., January 2006, Hydrology Manual, Water Resources Division, Los Angeles County Department of Public Works, http://www.ladpw.org/wrd/publication/engineering/2006_Hydrology_Manual/2006%20Hydrology%20Manual-Divided.pdf.

[17] Luenberger, D.G., Linear and Nonlinear Programming. Reading, MA, Addison-Wesley, 1984.

[18] Montgomery, D.C., Peck, E.A., and Vining, G.G., 2012, Introduction to linear regression analysis (5th ed.): New York, Wiley, ISBN 978–047054281–1.

[19] NOAA, 2015, Hydrometeorological Design Studies Center Quarterly Report, National Oceanic and Atmospheric Administration, http://www.nws.noaa.gov/oh/hdsc/current-projects/progress/201507_HDSC_PR.pdf.

[20] NOAA, 2015, Precipitation Frequency Data Server (PFDS): Hydrometeorological Design Studies Center, National Oceanic and Atmospheric Administration, http://hdsc.nws.noaa.gov/hdsc/pfds/.

[21] Pearson, K., 1922, Francis Galton: A Centenary Appreciation, Cambridge University Press.

[22] **R** Development Core Team, 2015, **R**—A language and environment for statistical computing: Vienna, Austria, **R** Foundation for Statistical Computing, http://www.R-project.org.

[23] DebRoy, S., **R** Manual Documentation, Non-Linear Minimization, https://stat.ethz.ch/R-manual/R-devel/library/stats/html/nlm.html.

[24] Schnabel, R. B., Koontz, J. E. and Weiss, B. E. (1985) A modular system of algorithms for unconstrained minimization. *ACM Trans. Math. Software*, 11, 419–440 p.

[25] State Department of Highways and Public Transportation Department, 1985, Hydraulic Manual, Bridge Division, State of Texas, Austin, Texas.

[26] Texas Department of Transportation, 2014, Hydraulic Design Manual, rev. May 2014: Texas Department of Transportation, http://onlinemanuals.txdot.gov/txdotmanuals/hyd/hyd.pdf.

[27] Cleveland, T.G., Hermann, G.R., Tay, C.C., Neale, C.M., Schwarz, M.R., and Asquith, W.H., August 2015, Texas Department of Transportation, Research Report 0-6824-1: New Rainfall Coefficients – Including Tools for Estimations of Intensity and Hyetographs in Texas.

[28] Texas Highway Department, 1970, Hydraulic Manual, Bridge Division, Austin, Texas.

[29] Paulhus, J.L., Wilson, W.T., Hiatt, W.E., Hershfield, D.M., Kauffman, N.M., Weiss, L.L., and Pagan, A.R., December 1955, U. S. Weather Bureau, Technical Paper No. 25, Rainfall intensity-duration-frequency curves for selected stations in the United States, Alaska, Hawaiian Islands, and Puerto Rico.

[30] Wanielista, M., Eaglin, R. and Eaglin, E., 1996. Intensity-duration-frequency curves for the State of Florida, Florida Department of Transportation.

[31] McCuen, R.H., Johnson, P.A., and Ragan, R.M., 2002, Hydraulic Design Series No. 2 Highway Hydrology, 2nd ed., October 2002: U.S. Department of Transportation, Federal Highway Administration, National Highway Institute, FHWA Pub. No. NHI-02-001, http://www.fhwa.dot.gov/engineering/hydraulics/library_arc.cfm?pub_number=2&id=6andhttp://isddc.dot.gov/OLPFiles/FHWA/013248.pdf.

[32] Brent, R. 1973. Algorithms for Minimization without Derivatives, Prentice-Hall, Englewood Cliffs, New Jersey.

[33] Broyden, C. G., 1970. The convergence of a class of double-rank minimization algorithms, Journal of the Institute of Mathematics and Its Applications 6, 76–90 p.

[34] Fletcher, R., 1970. A New Approach to Variable Metric Algorithms, Computer Journal 13 (3), 317–322 p.

[35] Goldfarb, D., 1970, A Family of Variable Metric Updates Derived by Variational Means, Mathematics of Computation 24 (109), 23–26 p.

[36] Shanno, D.F., 1970, Conditioning of quasi-Newton methods for function minimization, Math. Comput. 24 (111), 647–656 p.

[37] Schnabel, R.B., Dennis, J.E., 1983, Numerical Methods for Unconstrained Optimization and Nonlinear Equations, Classics in Applied Mathematics (16).

[38] Texas Department of Transportation, 2012, GEOPAK Drainage: Training Course for GEOPAK V8i Civil Design Software. http://ftp.dot.state.tx.us/pub/txdot-info/isd/geopakv8i/geopak_drainage_v8i.pdf

[39] **RStudio** Team, 2015, RStudio: Integrated Development for R, RStudio, Inc., Boston, MA. http://www.rstudio.com/.

[40] Rice, J. R., 2014, *Numerical Methods in Software and Analysis*. Burlington: Elsevier Science, 301-310 p.

[41] Tay, C.C., Neale, C.M., Herrmann, G.R., and Cleveland, T.G. 2015. ``Updated Rainfall Coefficients for Texas -- The EBDLKUP-NEW.XLS Tool.'' *in* Proceedings of ASCE World Environmental and Water Resources Congress, ASCE-EWRI, Austin, TX. May 17-21, 2015. pp. 442-452.

APPENDIX A: 2004 DDF Values

APPENDIX B: 2015 DDF Values (0-6824 TxDOT)

APPENDIX C: 2015 *ebd* Values (0-6824 TxDOT)

APPENDIX D: Pre-Adjusted Depth Values (0-6824 TxDOT)

APPENDIX E: Pre-Adjusted *ebd* Values (0-6824 TxDOT)

APPENDIX F: *SingleLocation* Code

APPENDIX G: *BatchLocation* Code

APPENDIX H: