

## ABSTRACT

Frequentist, Bayesian, and Zero-One Inflated Beta Regression Models

Elysia A. Garcia

Director: Jeanne S. Hill, Ph.D.

The main objective of this paper is to introduce readers to the beta regression. The beta regression is unique in its ability to adapt to many data trends despite skewness and other factors. The beta regression is also unique in its use of proportions and percentiles as its dependent variable. The paper will look at the beta regression from different perspectives, consisting of frequentist and Bayesian, as well as adjusting for zero-one inflation. Finally, the paper will show the utilization of the beta regression in applications such as experimental studies concerning BMI percentiles and operational data on crude oil proportions after distillation. Statistical programs such as R and OpenBUGS will be used in this paper to give readers the tools needed to fit beta regression and interpret the output.

APPROVED BY DIRECTOR OF HONORS THESIS:

---

Dr. Jeanne S. Hill, Department of Statistical Science

APPROVED BY THE HONORS PROGRAM:

---

Dr. Andrew Wisely, Director

DATE: \_\_\_\_\_

FREQUENTIST, BAYESIAN, AND ZERO-ONE INFLATED BETA REGRESSION  
MODELS

A Thesis Submitted to the Faculty of  
Baylor University  
In Partial Fulfillment of the Requirements for the  
Honors Program

By  
Elysia A. Garcia

Waco, Texas

05 2014

## TABLE OF CONTENTS

Table of Figures .....	iv
Acknowledgments .....	v
Chapter One: Introduction .....	1
Regression Analysis (Linear, Logistic, and Poisson) .....	2
Chapter Two: Beta Regression .....	5
Beta Regression Model and Maximum Likelihood Estimator .....	5
Application .....	7
Chapter Three: Bayesian Beta Regression .....	15
Bayesian Beta Regression Model and Maximum Likelihood Estimator .....	16
Application .....	17
Chapter Four: Zero-One Inflated Beta Regression .....	23
Zero-One Inflated Beta Regression Model and Maximum Likelihood Estimator . .....	23
Application .....	24
Chapter Five: Conclusion. ....	26
Appendices .....	27
Appendix A: R code to create Figure 2. ....	28
Appendix B: R code to create Figure 15. ....	28
Appendix C: R Code to do beta regression on BMI data .....	29

Appendix D: Complete OpenBUGS code to Figure 12 .....	29
Appendix E: R code for Confidence Interval in Table 1. ....	31
Works Cited. ....	32

## TABLE OF FIGURES

Figure 1: Sample of “Gasoline Yield” . . . . .	7
Figure 2: Histogram and theoretical densities for “Gasoline Yield” data set. The two histograms display the same data fit to the beta distribution and the normal distribution..	8
Figure 3: R code for Beta regression using “Gasoline Yield” . . . . .	9
Figure 4: Summary statistics for "Gasoline Yield" data from R. . . . .	10
Figure 5: Likelihood ratio test for "Gasoline Yield" data in R . . . . .	11
Figure 6: Residual plot for “Gasoline Yield” in R . . . . .	11
Figure 7: Cook's distance plot for "Gasoline Yield" in R. . . . .	11
Figure 8: Generalized leverage vs predicted values plot for "Gasoline Yield" in R . . . .	12
Figure 9: Residuals vs linear predictor plot for "Gasoline Yield" in R . . . . .	12
Figure 10: Half-normal plot of residuals for “GasolineYield” data in R . . . . .	13
Figure 11: Sample of BMI data . . . . .	18
Figure 12: OpenBUGS code for Bayesian Beta Regression . . . . .	19
Figure 13: Summary statistics for BMI data from OpenBUGS . . . . .	20
Figure 14: Posterior Densities for BMI data for OpenBUGS . . . . .	20
Table 1: This table compares the frequentist and Bayesian outputs computed from using the BMI data. . . . .	21
Figure 15: Histogram and theoretical densities for BMI data set fit to the beta distribution . . . . .	25

## ACKNOWLEDGEMENTS

Thank you to my mentor, Dr. Jeanne Hill, for guiding me and assisting me through this process. We've had many setbacks in this thesis, but the final product is one that I am very proud of and one which was a joy to write. Thanks also goes to Dr. James Stamey, who helped in numerous occasions with odd questions and coding problems while always giving moral support. Thank you to my parents, who assisted me through hours of editing and mental breakdowns. Finally, thank you to Dr. Amy Maddox and Dr. Kelly Ylitalo for taking the time to analyze my thesis and help me critique it.

## CHAPTER ONE

### Introduction

The purpose of this paper is to provide an in-depth look into the beta regression. The paper will suffice as both a teaching tool for the concepts and methodology behind the regression, as well as a practical tool for applying the beta regression to real-life situations using various statistical programming packages. We will begin by introducing regression analysis and the beta regression's unique characteristics compared to other methods of regression analysis. As the paper moves on, we will begin to look at unique applications of the beta regression, specifically Bayesian modeling and zero-one inflation. By the end of this paper, it is hoped that readers not only learn a new and unique form of regression analysis, but can also grasp basic coding skills to apply said knowledge to better serve in their fields.

There are many reasons for focusing on the beta regression. As a relatively new concept, beta regression addresses the issue of how to analyze data that contain a dependent, continuous proportion. Other more common and popular regressions, such as linear or logistic, cannot accommodate this situation well. Therefore, based on the flexible beta distribution, the beta regression has become the focus of many researchers who wish to delve and expand on this unique concept.



### *Regression Analysis (Linear, Logistic, and Poisson)*

To analyze the results of any study, one must first understand the methodology behind the results. A common statistical tool which analyzes graphically the correlation between variables is regression modeling.

One of the most popular forms of regression analysis is the simple linear regression. Linear regression attempts to model the relationship between two variables by fitting a linear equation to observed data. One variable is considered to be an explanatory or predictor variable ( $x$ ), and the other is considered to be a dependent or response variable ( $y$ ). The model for this regression is as follows:

$$Y = \beta_0 + \beta_1 X + \epsilon \quad (1.1)$$

with  $\beta_0$  representing the intercept,  $\beta_1$  is the slope, and  $\epsilon$  is the random error in the model. The expected value for this regression is  $E(y|x) = \beta_0 + \beta_1 X$ .

In regression, we have one population for each value of  $x$ . We thus want to choose a line that best “fits” the data by minimizing the overall vertical difference between the line and the observed values, i.e. the residuals, thus allowing us to estimate the average value of  $Y$ . Following these concepts, if we want to do multiple linear regression, we would still have only our one dependent or response variable, but then have multiple predictor or explanatory variables which are independent, thus giving us the General Linear Model:

$$Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \cdots + \beta_n x_n + \epsilon. \quad (1.2)$$

Another popular form of regression analysis is the logistic regression. Logistic regression is used for predicting the outcome of a categorical dependent variable based

on one or more independent predictor variables - in other words, logistic regression is used to predict binary (dichotomous) outcomes based on one or more independent predictor variables. This is what makes the logistic regression unique; there are only two possible outcomes, and the job of logistic regression is to estimate the probabilities of the two possible outcomes or the odds of the outcome of interest.

The basic logistic function is given by

$$f(z) = \frac{1}{1 + e^{-z}} \quad (1.3)$$

where  $z_i = \beta_0 + \beta_1 x_i$  with an expected value  $E(x) = \frac{e^{(\beta_0 + \beta_1 x_i)}}{1 + e^{(\beta_0 + \beta_1 x_i)}}$ .

The final example of regression analysis before we move into the beta regression is the Poisson regression. The Poisson regression is based on the Poisson distribution which is used to calculate the probability of a given number of events ( $x$ ) occurring in some fixed unit of measurement (time or space, for example) if these events occur with a known average rate of occurrence per unit of measurement ( $\lambda$ ). The basic model, or probability model function (pmf), for the Poisson distribution is

$$f(x) = \frac{e^{-\lambda} \lambda^x}{x!}, x = 1, 2, 3, \dots, n \quad (1.4)$$

with an expected value  $E(x) = \lambda$ . For the Poisson regression, however, we have one more variable which must be taken into account.

In using the Poisson regression, one sees if the Poisson-distributed dependent variable ( $y$ ) can be predicted from or explained by the other variables ( $x_i$ ). To do this, we use the following:

$$\ln(\lambda) = \beta_0 + \beta_1 x \quad \text{or} \quad \lambda = e^{\beta_0} e^{\beta_1 x} \quad (1.5)$$

With this background knowledge, we can now move to the beta regression.

## CHAPTER 2

### Beta Regression

The beta regression model was initially introduced by Silvia L.P. Ferrari and Francisco Cribari-Neto in 2004. The model adapts to a continuous dependent variable which falls within a 0 to 1 interval. This distribution could model many different data types, such as proportions, fractions, or rates (Bayes, Bazán, García 2012). The reason why the beta distribution is used as the foundation for a regression model is basic; the beta distribution is a very flexible two-parameter family that can accommodate skewness and symmetry. Conditionally, that is, in a regression model, the beta distribution can accommodate many shapes (Smithson & Verkuilen 2006). The accommodating ability of the beta model addresses the problems of more variation around the mean and less around the upper and lower limits of the interval for the proportion, or extreme skewness and uniformity (Cribari-Neto & Zeileis 2010).

#### *Beta Regression Model and Maximum Likelihood Estimator*

The basic model for the beta distribution is as follows:

$$f(y; p, q) = \frac{\Gamma(p+q)}{\Gamma(p)\Gamma(q)} y^{p-1} (1-y)^{q-1} \quad (2.1)$$

where  $0 < y < 1$  and where  $p$  and  $q > 0$  and  $\Gamma(\cdot)$  is the gamma function which is defined as  $\Gamma(x) = \int_0^\infty t^x e^{-t} dt$ . The quantiles  $p$  and  $q$  are shape parameters, with  $p$  pulling density towards 0 and  $q$  pulling density towards 1. The expected value and variance for the beta distribution are  $E(y) = \frac{p}{p+q}$  and  $V(y) = \frac{pq}{(p+q)^2(p+q+1)}$  respectively.

Ferrari and Cribari-Neto (2004) introduced a different parameterization for their modeling of the beta regression by setting  $\mu = p/(p + q)$  and  $\varphi = p + q$ :

$$f(y; \mu, \varphi) = \frac{\Gamma(\varphi)}{\Gamma(\mu\varphi)\Gamma((1-\mu)\varphi)} y^{\mu\varphi-1} (1-y)^{(1-\mu)\varphi-1}, \quad 0 < y < 1 \quad (2.2)$$

with  $0 < \mu < 1$  and  $\varphi > 0$ . The notation for this distribution is  $y \sim B(\mu, \varphi)$  with an expected value  $E(y) = \mu$  and variance  $V(y) = \mu(1-\mu)/(1+\varphi)$ . The parameter  $\varphi$  is known as the precision parameter since the larger it becomes the smaller the variance of our dependent variable  $y$  for a set location parameter of  $\mu$  (Ferrari & Cribari-Neto 2004).

The reason for such a transformation is because shape parameters such as  $p$  and  $q$  are difficult to interpret in terms of conditional expectations (Smithson & Verkuilen 2006). Therefore, using Ferrari and Cribari-Neto's transformation, we adjust these parameters to create a location and precision parameters which are more easily interpreted within the context of regression analysis.

To estimate these parameters, we use maximum-likelihood. The log-likelihood for the  $i$ th observation  $y_i$  is

$$\begin{aligned} \ln L(p, q, y_i) &= \ln \Gamma(p + q) - \ln \Gamma(p) - \ln \Gamma(q) + (p - 1) \ln(y_i) \\ &\quad + (q - 1) \ln(1 - y_i) \end{aligned} \quad (2.3)$$

or, when undergone through Ferrari and Cribari-Neto's transformation, the log-likelihood function is  $l(\beta, \varphi) = \sum_{i=1}^n l_i(\mu_i, \varphi)$ , where

$$\begin{aligned} l_i(\mu_i, \varphi) &= \log \Gamma(\varphi) - \log \Gamma(\mu_i \varphi) - \log \Gamma((1 - \mu_i) \varphi) + (\mu_i \varphi - 1) \log(y_i) \\ &\quad + \{(1 - \mu_i) \varphi - 1\} \log(1 - y_i). \end{aligned} \quad (2.4)$$

This estimator was found by maximizing the sum of the log-likelihoods over the observations.

## Application

To show an example of the beta regression in use, we will use the statistical program R.

An experimental version of R was first developed in 1995 by Ross Ihaka and Robert Gentleman. The program has experienced immense popularity, and as an open environment program, it invites improvements (Ihaka 1998). The language of R is similar to other programming languages such as C++, and it allows more control over results and simulations compared to SAS or JMP, thus making it a rising program in academia.

Using the “Gasoline Yield” data set in R, we can recreate a beta regression analysis. The “Gasoline Yield” data set consists of operational data of the proportion of crude oil converted to gasoline after distillation and fractionation. We have six variables within this data set, with “yield”, or the proportion of crude oil to gasoline after distillation and fractionation, being our characteristic proportion needed for the analysis. The other two variables of interest for our analysis are “temp”, the temperature (degrees F) at which all gasoline has vaporized, and “batch”, the factor indicating unique batch of conditions. Other variables included in the data set are “gravity”, “pressure”, and “temp10”, which is the temperature (degrees F) at which 10 percent of crude oil has vaporized.

```
> head(Gasolineyield)
  yield gravity pressure temp10 temp batch
1 0.122   50.8     8.6    190   205     1
2 0.223   50.8     8.6    190   275     1
3 0.347   50.8     8.6    190   345     1
4 0.457   50.8     8.6    190   407     1
5 0.080   40.8     3.5    210   218     2
6 0.131   40.8     3.5    210   273     2
```

Figure 1: Sample of “Gasoline Yield”

The data set “Gasoline Yield”, apart from possessing the necessary presence of a dependent variable between 0 and 1, proves to be the perfect example for displaying the beta regression.

Figure 2 shows two histograms which both display the same data, but shows the data fit to two different distributions: the beta distribution, and the normal distribution. As mentioned before, one of the main objectives in regression analysis is to find a distribution which will best “fit” the data and thus reduce residual error. By doing so, one can get the best possible information from the data. In analyzing the two graphs, it is clear that the beta distribution best fits the data since it covers the majority of the density plot. The normal distribution, however, neglects to cover many of the data points near zero and in turn slightly inflates those points in the center of the graph. Thus with this knowledge, one can confidently apply the beta regression the “Gasoline Yield” data.

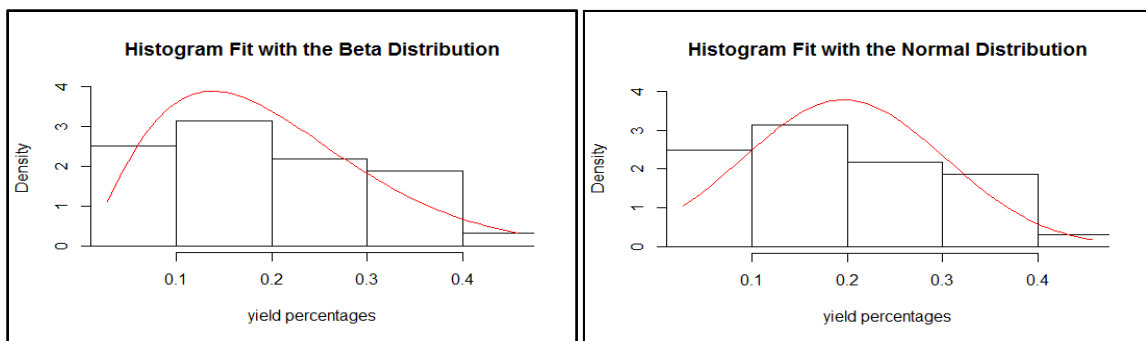


Figure 2: Histogram and theoretical densities for “Gasoline Yield” data set. The two histograms display the same data fit to the beta distribution and the normal distribution.

To use the beta regression, the “betareg” package has to be downloaded to R first. After this is completed, you can begin your regression analysis. The following code was used to create the regression analysis.

```
install.packages("betareg", "FILE LOCATION")
library(betareg)
library(lmtest)
data("GasolineYield", package = "betareg")
gy_logit <- betareg(yield ~ batch + temp, data = GasolineYield)
gy_logit2 <- betareg(yield ~ batch + temp | temp, data = GasolineYield)
summary(gy_logit)
summary(gy_logit2)
lrtest(gy_logit, gy_logit2)
set.seed(123)
plot(gy_logit, which = 1:4, type = "pearson")
plot(gy_logit, which = 5, type = "deviance", sub.caption = "")
```

Figure 3: R code for Beta regression using “Gasoline Yield”

A brief overview of the coding can be described in the following: the first three lines of code install the necessary packages needed to do beta regression in R, as well as opening the necessary libraries within the package. These libraries provide the tools and commands needed. The fourth line opens the data, “Gasoline Yield”, which is stored in the *betareg* package. We then create the functions for our regression with “yield” as our dependent variable and “batch” and “temp” as our independent predictor variables. The summary command then gives us basic statistics on the functions, and the *lrtest* gives the likelihood ratio tests. Finally, we output our descriptive plots.

The coding gives the following output.



```
> summary(gy_logit)
```

Call:  
betareg(formula = yield ~ batch + temp, data = Gasolineyield)

Standardized weighted residuals 2:

Min	1Q	Median	3Q	Max
-2.8750	-0.8149	0.1601	0.8384	2.0483

Coefficients (mean model with logit link):

	Estimate	Std. Error	z value	Pr(> z )
(Intercept)	-6.1595710	0.1823247	-33.784	< 2e-16 ***
batch1	1.7277289	0.1012294	17.067	< 2e-16 ***
batch2	1.3225969	0.1179020	11.218	< 2e-16 ***
batch3	1.5723099	0.1161045	13.542	< 2e-16 ***
batch4	1.0597141	0.1023598	10.353	< 2e-16 ***
batch5	1.1337518	0.1035232	10.952	< 2e-16 ***
batch6	1.0401618	0.1060365	9.809	< 2e-16 ***
batch7	0.5436922	0.1091275	4.982	6.29e-07 ***
batch8	0.4959007	0.1089257	4.553	5.30e-06 ***
batch9	0.3857930	0.1185933	3.253	0.00114 **
temp	0.0109669	0.0004126	26.577	< 2e-16 ***

Phi coefficients (precision model with identity link):

	Estimate	Std. Error	z value	Pr(> z )
(phi)	440.3	110.0	4.002	6.29e-05 ***

---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Type of estimator: ML (maximum likelihood)  
Log-likelihood: 84.8 on 12 Df  
Pseudo R-squared: 0.9617  
Number of iterations: 51 (BFGS) + 3 (Fisher scoring)

```
> summary(gy_logit2)
```

Call:  
betareg(formula = yield ~ batch + temp | temp, data = Gasolineyield)

Standardized weighted residuals 2:

Min	1Q	Median	3Q	Max
-2.5399	-0.7792	-0.1167	0.8621	2.9419

Coefficients (mean model with logit link):

	Estimate	Std. Error	z value	Pr(> z )
(Intercept)	-5.9232361	0.1835262	-32.275	< 2e-16 ***
batch1	1.6019877	0.0638561	25.087	< 2e-16 ***
batch2	1.2972663	0.0991001	13.090	< 2e-16 ***
batch3	1.5653383	0.0997392	15.694	< 2e-16 ***
batch4	1.0300720	0.0632882	16.276	< 2e-16 ***
batch5	1.1541630	0.0656427	17.582	< 2e-16 ***
batch6	1.0194446	0.0663510	15.364	< 2e-16 ***
batch7	0.6222591	0.0656325	9.481	< 2e-16 ***
batch8	0.5645830	0.0601846	9.381	< 2e-16 ***
batch9	0.3594390	0.0671406	5.354	8.63e-08 ***
temp	0.0103595	0.0004362	23.751	< 2e-16 ***

Phi coefficients (precision model with log link):

	Estimate	Std. Error	z value	Pr(> z )
(Intercept)	1.364089	1.225781	1.113	0.266
temp	0.014570	0.003618	4.027	5.65e-05 ***

---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Type of estimator: ML (maximum likelihood)  
Log-likelihood: 86.98 on 13 Df  
Pseudo R-squared: 0.9519  
Number of iterations: 33 (BFGS) + 28 (Fisher scoring)

Figure 4: Summary statistics for "Gasoline Yield" data from R.

```

> lrtest(gy_logit, gy_logit2)
Likelihood ratio test

Model 1: yield ~ batch + temp
Model 2: yield ~ batch + temp | temp
#Df LogLik Df Chisq Pr(>Chisq)
1 12 84.798
2 13 86.977 1 4.359 0.03681 *
---
signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

Figure 5: Likelihood ratio test for "Gasoline Yield" data in R

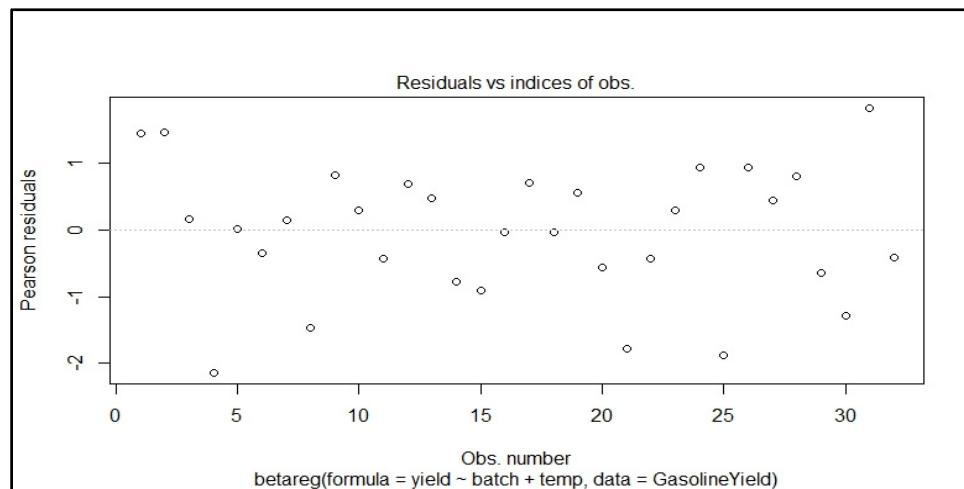


Figure 6: Residual plot for "Gasoline Yield" in R

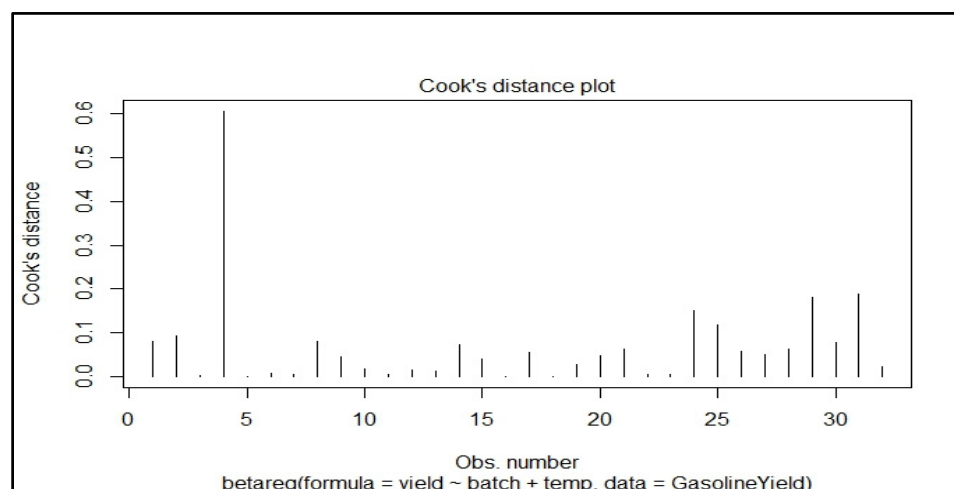


Figure 7: Cook's distance plot for "Gasoline Yield" in R

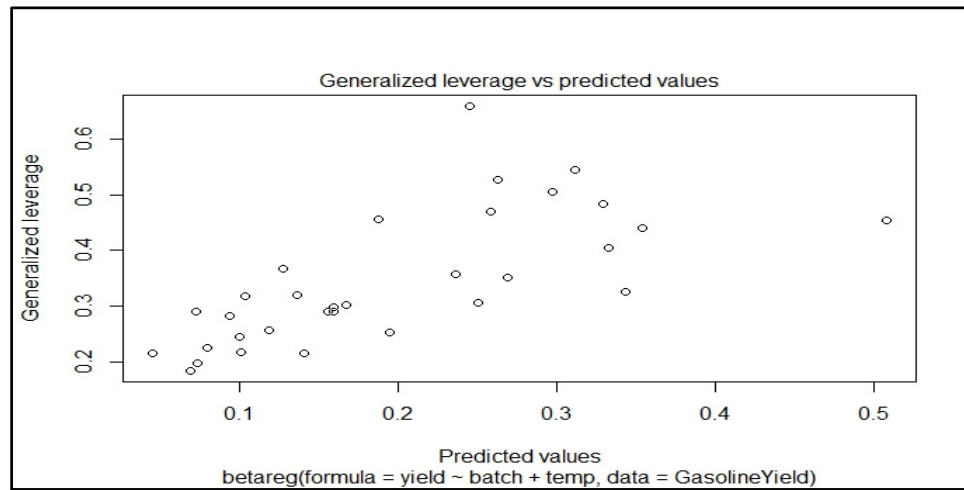


Figure 8: Generalized leverage vs predicted values plot for "Gasoline Yield" in R

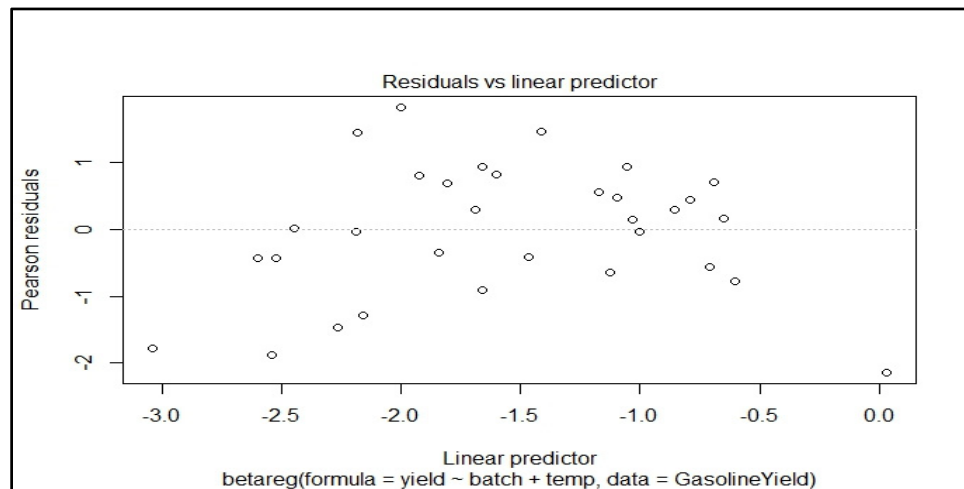


Figure 9: Residuals vs linear predictor plot for "Gasoline Yield" in R

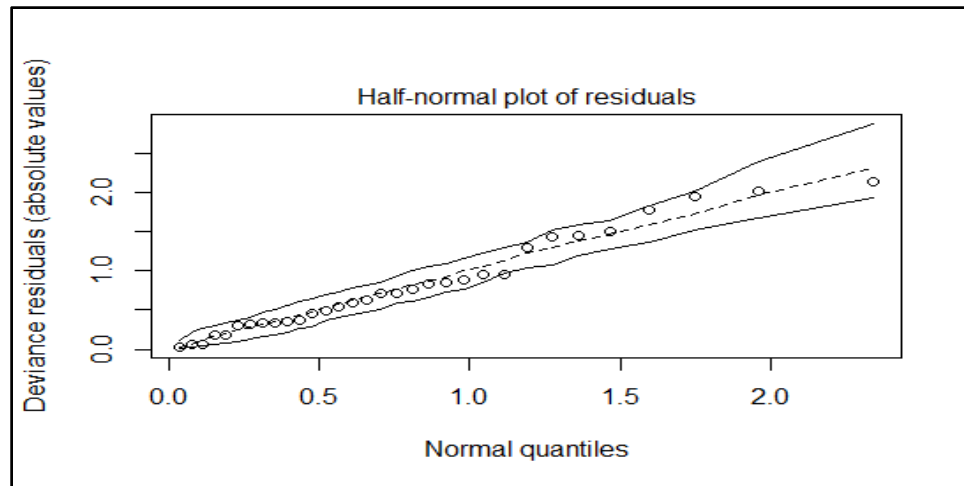


Figure 10: Half-normal plot of residuals for “Gasoline Yield” data in R

The R code utilized for beta regression outputs a milieu of graphics and data, with much of the outputs being beyond basic statistical understanding. The purpose and interpretation of the R outputs is as follows.

Figure 4 displays basic summary statistics for each variable and the overall model, giving such information as p-values, z-scores, and estimates of value. Figure 5 shows the results of the likelihood ratio test. The likelihood ratio test is used to compare the “fit” of a lesser model with fewer parameters and a general model which is nested, which in our example is *gy\_logit* and *gy\_logit2*. The nested model allows the variance or precision to change for each temperature. In laymen’s terms, the likelihood ratio test is used to compute the difference in the deviance of the two models. The test indicates that the second or nested model is preferred (p-value = 0.03681). Figure 6 shows the residual plot for the model, which shows if there is constant variance and can indicate if there is an interfering variable or side effect in the model, which will be apparent if there is an obvious pattern in the points. Figure 7 shows Cook’s Distance plot, which shows the overall influence of each observation on the model. On our plot, observation four is the

most obvious and only observation on the graph, thus indicating a high impact. In such a situation, it would probably be beneficial to analyze the model without this variable to see how it compares. Figure 8 shows the general leverage plot which displays the composition of the sum of squares for a hypothesis test. These plots reveal information regarding the degree of fit, the residuals, influential observations, non-fitting points, nonlinearities, and even colinearity (Sall 1990). Figure 9 shows the residuals plotted against the linear predictors. This plot gives the same information as Figure 6, but instead uses future values which are estimated as a function of our sample and model. Finally, Figure 10 shows the half-normal plot for the residuals. This plot takes the absolute value of the residuals and then arranges them in increasing order, thus easily indicating the outliers in the residuals (Friendly 2000).

## CHAPTER 3

### Bayesian Beta Regression

A Bayesian analysis on our already-established beta regression gives us another technique for statistical inference. Bayesian analysis takes a different perspective from the common frequentist ideas for statistical analysis. While a frequentist uses a sample to estimate the true value of a population parameter, Bayesian analysis uses a sample to better estimate a parameter modeled with an established prior distribution. This prior distribution can account for subjective information one may have about the parameter. The choice of priors will be considered later. However, one should note that the introduction of a prior constitutes an additional assumption (Hair 2010). By revising the prior with our observed sample data we obtain the posterior distribution using Bayes Theorem.

$$\frac{p(y|\theta)p(\theta)}{\int p(y|\theta)p(\theta)d\theta} \quad (3.1)$$

One of the most crucial components to Bayesian analysis is choosing the prior. Choosing a prior distribution for your analysis is not simple. There are three means in which one can choose a prior: subjective, objective (i.e. informative), or diffuse (i.e. noninformative). The first method allows the individual to choose the prior subjectively, or based on the experimenter's personal probability that  $\theta$  lies in any given subset of  $\Theta$ . The second method uses past data as a prior, either from historical data or past experiments. Essentially, one could use the posterior distributions of previous experiments or expert opinions and make it the prior. Finally, a noninformative approach

to choosing one's prior distribution expresses ignorance as to the value of  $\theta$ . Therefore, stated in a more vernacular fashion, the best prior is one which can fit the data and can assist in increasing the accuracy of the sample distribution. Based on what we know about beta regression and its properties, it is easy to understand why the beta regression is often used in tangent with Bayesian analysis (Seaman).

The benefits of using Bayesian analysis on our data is that, in some comparisons, Bayesian analysis is superior in its estimation precision (Hair 2010). Bayesian analysis also allows for conjoint models to be estimated at the individual level whereas previously only aggregate models were possible. However, to do Bayesian analysis, there may be the need for a large sample to keep the prior distribution from becoming too influential.

#### *Bayesian Beta Regression Model and Maximum Likelihood Estimator*

The Bayesian beta regression model we consider is

$$\begin{aligned}
 y_i | \mu_i, \varphi_i, x_i, \beta_i &\sim \text{Beta}(\varphi_i \mu_i, \varphi_i (1 - \mu_i)) \\
 \text{logit}(\mu_i) &= \text{logit}(\mu(x'_i)) = x'_i \beta_\mu \\
 p(\beta_\mu) &\propto 1
 \end{aligned} \tag{3.2}$$

where  $p(\beta_\mu)$  is an improper flat prior on the  $\beta_\mu$  vector.

With independent data, the likelihood function is given by

$$L(\beta, \varphi) = \prod_{i=1}^n \frac{\Gamma(\varphi)}{\Gamma(F(x_i^T \beta) \varphi) \Gamma(\varphi (1 - F(x_i^T \beta)))} y_i^{F(x_i^T \beta) \varphi - 1} (1 - y_i)^{(\varphi (1 - F(x_i^T \beta))) - 1} \tag{3.3}$$

and the posterior distribution is

$$p(\beta, \varphi | y) = \frac{L(\beta, \varphi) p(\beta, \varphi)}{\int L(\beta, \varphi) p(\beta, \varphi) d\beta d\varphi}$$

$$\propto \prod_{i=1}^n \Gamma(F(x_i^T \beta) \varphi) \Gamma(\varphi(1 - F(x_i^T \beta)))^{-1} y_i^{F(x_i^T \beta) \varphi} (1 - y_i)^{\varphi(1 - F(x_i^T \beta))} \times p(\beta, \varphi) \Gamma(\varphi)^n \quad (3.4)$$

(Branscum, Johnson, & Thurmond 2007).

### *Application*

For this application example, we will be using a statistical software called OpenBUGS. In the earliest stages, BUGS was a software package for performing Bayesian inference using Gibbs sampling, a technique using Monte Carlo Markov Chains to randomly select a sample (Lunn, Spiegelhalter, Thomas, & Best 2009). The Medical Research Council Biostatistics Unit in Cambridge first began the BUGS project in 1989, and the program has become a popular statistical modeling package (2009). The program has expanded to adapt with time, finally giving public users WinBUGS – the Microsoft Windows incarnation of the software – and finally now OpenBUGS which is a more accommodating and improved product.

BUGS is a package meant to perform Bayesian inference. The development of this package reflects an even larger trend within the statistical community towards the application of Bayesian ideas in the past few decades. As Lunn and co-authors mention in their paper, “The BUGS project: Evolution, critique and future directions”, BUGS was able to bring Bayesian modeling into the general awareness by making such applications easier and more accurate (2009). The application of this software goes beyond just that of a teaching tool for statistics students and researchers, but is also used in such settings as



disease mapping, ecology, genetics, coastal engineering, behavioral studies, and automated music transcription (2009).

To perform the tasks given to it, BUGS uses Gibbs Sampling which is linked to Monte Carlo Markov Chains (MCMC). The program gives outputs and creates a sample from the distribution and priors given for the Bayesian modelling by going through multiple simulations until convergence is achieved. The user tells the program how many simulations it should do, and with greater iterations the results become better. What we want to achieve is a result which best represents and fits our model. With MCMC, the results of the previous simulations are constantly taken into account and used for the readjustment of the samples.

Now that we understand what OpenBUGS is doing to create our regression analysis, we can look at the coding. The following coding includes data accumulated through a Baylor University project in conjunction with the “Straw to Bread” program by undergraduate student, Eric Goethe. The data, titled in this thesis as “BMI data”, includes various variables concerning the effects of water quality on the body mass index percentiles in children in rural western Kenya.

```
> head(data)
  muac waist wsource watloc      bmi tankyn
1 21.0  64.9      1    332 0.4230688      0
2 14.3  65.0      1    332 0.3859118      0
3 15.1  54.0      1    332 0.9946819      0
4 14.5  52.5      0    116 0.4760230      0
5 18.0  57.0      0    116 0.4499346      0
6 20.5  65.0      0    116 0.2776596      0
```

Figure 11: Sample of BMI data

The variables of interest for our model are “bmi”, which is body mass index percentiles, “wsource”, which indicates whether water consumption is from ground water only or tank water, and “watloc”, which is various numerical water location names. The variable “bmi” is used as the dependent proportion  $y$  and “wsource” and “watloc” as the independent  $x$  variables. The prior distribution we will be using for this regression analysis is the normal distribution. The code is as follows:

Model

```
{
for(i in 1:N) {
Y[i]~dbeta(a[i],b[i])
a[i]<-abs(((1-mu[i])*mu[i]*mu[i]-mu[i]*sg[i])/sg[i])
b[i]<-abs((1-mu[i])*(mu[i]-mu[i]*mu[i]-sg[i])/sg[i])
logit(mu[i])<-bo+b1*x1[i]+b2*x2[i]
sg[i]<-exp(co+c1*x1[i])
}
bo~dnorm(0,1)
b1~dnorm(0,0.1)
b2~dnorm(0,0.01)
co~dnorm(0,0.01)
c1~dnorm(0,0.2)
}
```

data

```
list(Y=c(0.423069, 0.385912, 0.994682,0.000000001, 0.476023, 0.449935,
0.27766, 0.073578, 0.026433, 0.933666, 0.940669, 0.978365, 0.999987,
0.993495, 0.999923, 0.998277, 0.667372, 0.848145, 0.746585, 0.249899,...,
0.589789),
x1=c(1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ..., 1),
x2=c(332, 332, 332, 10, 116, 116, 116, 338, 338, 338, 338, 338, 338, 338, 338,
338, 8, 8, 8, 8, ..., 6), N=156)
```

Inits

```
list(bo=0, b1=0, b2=0,co=3, c1=0)
```

Figure 12: OpenBUGS code for Bayesian Beta Regression

There are three parts to the coding: the model, the data, and the initials. The model outlines the beta regression as  $Y[i]$  and then goes on to define the beta parameters as “a” and “b”. These parameters are then reparameterized as a function of the mean ( $\mu$ ) and variance ( $\sigma^2$ ) (Cepeda-Cuervo 2012). Then a prior distribution is given to  $b_o$ ,  $b_1$ ,  $b_2$ ,  $c_o$ , and  $c_1$ . Under the data are included 156 observation with variables “bmi”= $Y$ , “wsorce”= $x_1$ , and “watloc”= $x_2$ . Finally, the initial values indicate where to start the MCMC.

The code outputs statistics for each variable, or node, as well as graphs of their distributions.

	mean	sd	MC_error	val2.5pc	median	val97.5pc	start	sample
b1	0.3917	0.124	0.0124	0.2182	0.3493	0.66	10001	10000
b2	-0.2192	0.01047	6.685E-4	-0.2451	-0.2165	-0.207	10001	10000
bo	1.288	0.1058	0.01057	1.043	1.313	1.457	10001	10000
c1	-0.7324	0.05578	0.005468	-0.8505	-0.7354	-0.616	10001	10000
co	2.732	0.1592	0.01594	2.347	2.786	2.904	10001	10000

Figure 13: Summary statistics for BMI data from OpenBUGS

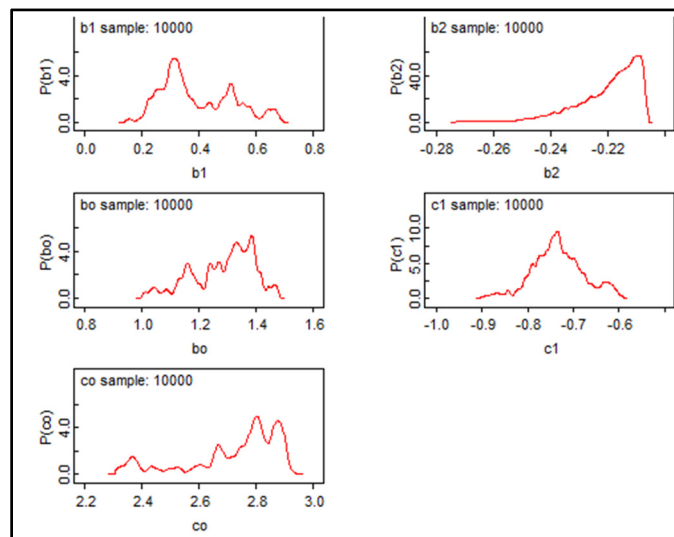


Figure 14: Posterior Densities for BMI data for OpenBUGS

Figure 13 shows the node statistics which gives the formal parameter estimates. The table gives the estimates for the means and standard deviations of each parameter, as well as the median and the 95% credible interval. The final estimate given in the table is the Markov Chain error, which gives the margin of error for the mean estimate. The smaller this error is, the more confidence we can have in our parameter estimates (Geyer). Figure 14 shows the posterior density estimates for each parameter in the model. The desired effect is to have a normal distribution, however, with our model, the densities lack this quality, thus indicating that the parameter estimates did not veer towards a certain range. The only posterior density which does seem to take on normal, though skewed, shape is  $c_1$ . The posterior density of  $b_2$  also has a distinguishable shape, though it appears more gamma than normal.

The final part of this chapter will compare the frequentist and Bayesian beta regression statistics. Using the BMI data, we look specifically at the summary statistics using R and OpenBUGS and compare the results.

<b>Frequentist and Bayesian Summary Statistics for BMI Data</b>						
	Frequentist			Bayesian		
	Mean	Standard Error	95% CI	Mean	Standard Error	95 % CI
Intercept (bo)	-0.4915	0.1736	(-0.921, -0.235)	1.288	0.1058	(1.043, 1.457)
Water source (b1)	0.1357	0.2138	(-0.283, 0.555)	0.3917	0.124	(0.218, 0.66)
Water location (b2)	0.0003	0.00040	(-0.0005, 0.001)	-0.2192	0.01047	(-0.245, -0.207)

Table 1: This table compares the frequentist and Bayesian outputs computed from using the BMI data.

When comparing the descriptive statistics from the two classes of thought, there is an obvious difference in the outputs. While the summary statistics from the frequentist beta regression reveal the variables water source and water location to be not significant at  $\alpha = 0.05$ , the Bayesian beta regression reveals all variables to be significant when considered at the same level of significance. This illustrates that Bayesian inference can be more expressive in regression analysis because of its use of prior distributions. This, of course, depends on the appropriateness of the prior as well.

## CHAPTER 4

### Zero-One Inflated Beta Regression

While beta regression is very useful because of its versatility, there are some limitations to the model. One such limitation is that observation of zero or one can cause extreme bias. Therefore, these observations must be scaled away from these values. Inflated beta regression takes this into account and allows complete modeling of the entire continuous interval from zero to one (Swearingen, Meliguizo Castro, & Bursac 2012).

#### *Zero-One Beta Regression Model and Maximum Likelihood Estimator*

For zero-inflation, a new parameter  $\pi_0$  is added to account for the probability of observations at zero. The resulting density function is

$$f(y; \pi_0, \mu, \varphi) = \begin{cases} \pi_0, & y = 0 \\ (1 - \pi_0)f(y; \mu, \varphi), & 0 < y < 1 \end{cases} \quad (4.1)$$

Where

$$f(y; \mu, \varphi) = \frac{\Gamma(\varphi)}{\Gamma(\mu\varphi)\Gamma(\varphi(1-\mu))} y^{\mu\varphi-1} (1-y)^{(1-\mu)\varphi-1}, 0 < y < 1. \quad (4.2)$$

Following this logic, one-inflation follows a similar pattern and a new parameter  $\pi_1$  is added to the beta distribution to account for the probability of observations at one. Thus, the density function is

$$f(y; \pi_1, \mu, \varphi) = \begin{cases} (1 - \pi_1)f(y; \mu, \varphi), & 0 < y < 1 \\ \pi_1, & y = 1 \end{cases} \quad (4.3)$$

Inflated beta distributions incorporate degenerate probability statements to produce a unique density function. For zero-one inflated beta regression, the two new parameters,  $\pi_0$  and  $\pi_1$ , are added to account for the probability of observations at both zero and at one.

$$\pi_0 = \text{logit}(x_i' \beta_0)^{-1} \text{ and } \pi_1 = \text{logit}(x_i' \beta_1)^{-1} \quad (4.4)$$

Thus we create the density function for the zero-one inflated beta regression:

$$f(y; \pi_0, \pi_1, \mu, \varphi) = \begin{cases} \pi_0, & y = 0 \\ (1 - \pi_0)(1 - \pi_1)f(y; \mu, \varphi), & 0 < y < 1 \\ \pi_1, & y = 1 \end{cases} \quad (4.5)$$

The expected value for  $y$  within the zero-one inflated beta regression is as follows:

$$E(y) = 0 * \pi_0 + 1 * \pi_1 + \mu(1 - \pi_0 - \pi_1) = \pi_{01}\mu_i + \pi_1 \quad (4.6)$$

where  $\pi_{01} = (1 - \pi_0 - \pi_1)$  (Ospina & Ferrari 2012, Wieczorek & Hawala 2011).

### *Application*

Finding a situation in which zero-one inflated beta regression can be applied is easy; however developing a coding specific for this purpose proves difficult. After intensive research, no coding could be found that directly applies to data which suffers from zero-one inflation with relation to beta regression modeling. Efforts are currently underway by myself in conjunction with departmental advisors within Baylor University's Department of Statistical Science to develop a OpenBUGS code which adjusts for inflated data. However, in the absence of coding, this paper can instead provide real life examples of data and situations where zero-one inflated beta regression

modeling can be applied. Referring back to former mentioned data sets, one in particular shows a prime example of inflation at both zero and one within the dependent variable.

The BMI dataset provides an example of a situation when zero-one inflated beta regression can be applied. In Figure 15, it can be clearly seen that the data contain more points towards zero and one, thus making any parameter estimation extremely biased.

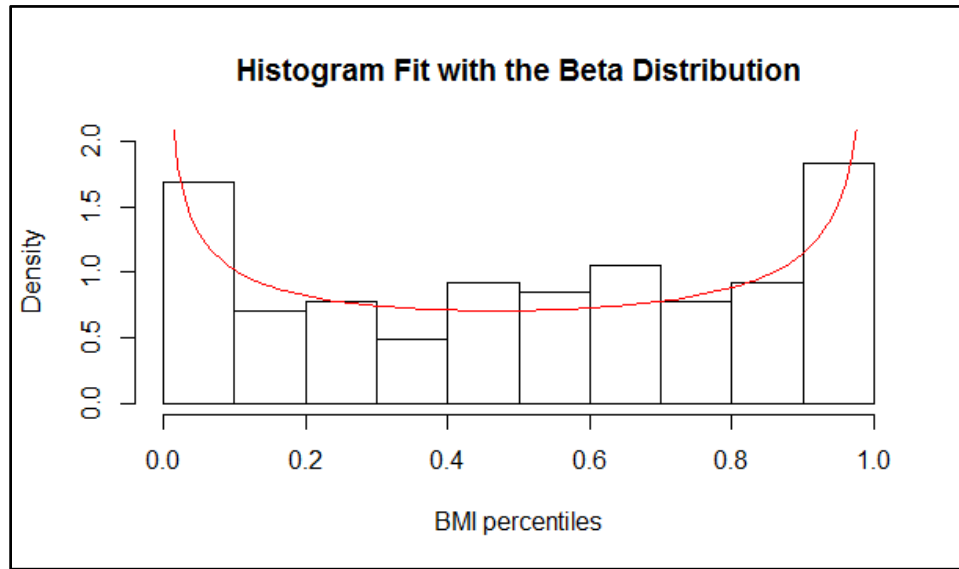


Figure 15: Histogram and theoretical densities for BMI data set fit to the beta distribution.



## CHAPTER 5

### Conclusion

With the conclusion of this paper, it is hoped that readers grasp the unique abilities of the beta regression in the face of other regression models. As a model that depends on a continuous dependent variable between 0 and 1, the beta regression can be used to model many data trends despite skewness, making it an invaluable tool in research. Building on this, it is hoped that readers can also feel comfortable in utilizing the beta regression in various coding settings, such as R and OpenBUGS, as well as with different perspectives such as frequentist and Bayesian, and also by adjusting for zero-one inflation. Finally, and most importantly, it is hoped that readers can interpret outputs in order to make educated and confident conclusions on their data.

As for the future of this thesis, I have plans to continue my work and research concerning the beta regression. As of now, it is my goal to develop an OpenBUGS coding which can accommodate zero-one inflated beta regression modeling. I also plan to continue applying and critiquing my work with the beta regression with various other data sets including another data set collected from the Straw to Bread program concerning lead levels in water and BMI percentiles as well as a multitude of childhood factors which can contribute to early on-set obesity in the local Waco demographic.

## APPENDICES

## APPENDIX A

### R code to create Figure 2

```
6 <<>>=
7 install.packages("betareg", "~/Baylor Classes/Fall 2013/Thesis",
8                 repos='http://cran.us.r-project.org')
9 library(betareg)
10
11 data("GasolineYield", package = "betareg")
12 head(GasolineYield)
13
14 library(MASS)
15 library(ggplot2)
16 library(e1071)
17 library(fitdistrplus)
18
19 x<-fitdist(GasolineYield$yield,dbeta,start=list(shape1=.5,shape2=.5))
20 y<-fitdist(GasolineYield$yield,dnorm)
21
22
23
24 denscomp(x,ylim=c(0,5),
25          main="Histogram Fit with the Beta Distribution",
26          xlab= "yield percentages",addlegend=FALSE)
27 denscomp(y,ylim=c(0,5),
28          main="Histogram Fit with the Normal Distribution",
29          xlab= "yield percentages",addlegend=FALSE)
30 @
```

## APPENDIX B

### R code to create Figure 15

```
6 <<>>=
7 install.packages("betareg", "~/Baylor Classes/Fall 2013/Thesis",
8                 repos='http://cran.us.r-project.org')
9 library(betareg)
10 library(MASS)
11 library(ggplot2)
12 library(e1071)
13 library(fitdistrplus)
14
15 data <- read.csv("blah.csv")
16 head(data)
17
18 x<-fitdist(data$bmi,dbeta,start=list(shape1=.5,shape2=.5))
19 y<-fitdist(data$bmi,dnorm)
20
21 denscomp(x,ylim=c(0,2),
22          main="Histogram Fit with the Beta Distribution",
23          xlab= "BMI percentiles",addlegend=FALSE)
24 denscomp(y,ylim=c(0,2),
25          main="Histogram Fit with the Normal Distribution",
26          xlab= "BMI percentiles",addlegend=FALSE)
27 @
```

## APPENDIX C

### R Code to do beta regression on BMI data

```
6 > <<>>=
7 install.packages("betareg", "~/Baylor Classes/Fall 2013/Thesis",
8                 , repos='http://cran.us.r-project.org')
9 library(betareg)
10 library(lmtest)
11
12 setwd("~/Thesis/data")
13 data <- read.csv("bmi.csv")
14
15 gy_logit <- betareg(bmi~ wsource + watloc, data = data)
16 gy_logit2 <- betareg(bmi ~ wsource + watloc|watloc, data = data)
17 summary(gy_logit)
18 summary(gy_logit2)
19 lrtest(gy_logit, gy_logit2)
20
21 set.seed(123)
22 plot(gy_logit, which = 1:4, type = "pearson")
23 plot(gy_logit, which = 5, type = "deviance", sub.caption = "")
24 plot(gy_logit, which = 1, type = "deviance", sub.caption = "")
25
26 @
```

## APPENDIX D

### Complete OpenBUGS code to Figure 12

```
model
{
  for(i in 1:N) {
    Y[i]~dbeta(a[i],b[i])
    a[i]<-abs(((1-mu[i])*mu[i]*mu[i]-mu[i]*sg[i])/sg[i])
    b[i]<-abs((1-mu[i])*(mu[i]-mu[i]*mu[i]-sg[i])/sg[i])
    logit(mu[i])<-bo+b1*x1[i]+b2*x2[i]
    sg[i]<-exp(co+c1*x1[i])
  }
  bo~dnorm(0,1)
  b1~dnorm(0,0.1)
  b2~dnorm(0,0.01)
  co~dnorm(0,0.01)
  c1~dnorm(0,0.2)
}
```

```
list(Y=c(0.423069, 0.385912, 0.994682,0.000000001, 0.476023, 0.449935, 0.27766,  
0.073578, 0.026433, 0.933666, 0.940669, 0.978365, 0.999987, 0.993495, 0.999923,  
0.998277, 0.667372, 0.848145, 0.746585, 0.249899, 0.47601, 0.565654, 0.288196,  
0.846573, 0.331747, 0.534659, 0.801769, 0.826486, 0.253655, 0.863, 0.637899,  
0.117,0.000000001, 0.177876, 0.570263, 0.913651, 0.10184, 0.053064, 0.088573, 0.36,  
0.940812, 0.432466, 0.63915, 0.000000001, 0.000000001, 0.000000001, 0.000000001,  
0.000000001, 0.856, 0.00476, 0.000000001, 0.091668, 0.445, 0.363732, 0.409038,  
0.751336,0.000000001, 0.626209, 0.283, 0.534, 0.64244, 0.073862, 0.015, 0.700635,  
0.040836, 0.594232, 0.969675, 0.009, 0.267746, 0.694049, 0.666329, 0.94, 0.979,  
0.908885, 0.965728, 0.946, 0.388161, 0.839488, 0.011747, 0.092726, 0.548237,  
0.173665, 0.7149, 0.001124, 0.002, 0.831953, 0.830909, 0.969, 0.724501, 0.479081,  
0.708137, 0.94, 0.905339, 0.72259, 0.338741, 0.04692, 0.996718, 0.50275, 0.652648,  
0.71318, 0.421708, 0.54, 0.30841, 0.001954, 0.008753,0.000000001, 0.068685,  
0.241527, 0.268616, 0.043973, 0.057917, 0.026516, 0.420656, 0.619609, 0.572666,  
0.679, 0.803234, 0.727357, 0.492, 0.981041, 0.257, 0.636, 0.966, 0.218266, 0.000494,  
0.663646, 0.993599, 0.197284, 0.179911, 0.099875, 0.547462, 0.209801, 0.995835,  
0.662246, 0.000000001,0.000000001, 0.000000001, 0.000000001, 0.981, 0.46517,  
0.162613, 0.671, 0.87484, 0.01231, 0.869269, 0.161747, 0.704, 0.866575, 0.998536,  
0.152747, 0.718642, 0.673, 0.533, 0.459589, 0.162, 0.589789),  
x1=c(1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0,  
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0,  
0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,  
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,  
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,  
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1),  
x2=c(332, 332, 332, 10, 116, 116, 116, 338, 338, 338, 338, 338, 338, 338, 338, 338, 8, 8,  
8, 8, 8, 8, 8, 333, 333, 333, 333, 333, 333, 332, 134, 134, 134, 134, 134, 134, 75, 75,  
75, 75, 75, 75, 84, 84, 84, 84, 84, 84, 116, 116, 116, 116, 116, 116, 121, 121, 121, 121,  
121, 121, 1471, 1471, 335, 19, 19, 19, 19, 6, 6, 6, 6, 6, 6, 8, 8, 8, 8, 8, 6, 6, 6, 6, 6, 6, 6, 6, 6,  
6, 6, 10, 10, 10, 75, 75, 75, 75, 75, 75, 336, 336, 336, 75, 75, 75, 75, 75, 75, 75, 10,  
10, 10, 10, 331, 331, 333, 337, 332, 8, 8, 8, 332, 134, 134, 6, 8, 1471, 1471, 1471, 75,  
75, 75, 75, 84, 84, 84, 84, 84, 335, 335, 335, 116, 116, 116, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6),  
N=156)
```

```
list(bo=0, b1=0, b2=0,co=3, c1=0)
```

## APPENDIX E

### R code for Confidence Interval in Table 1

```
6 > <<>>=
7 install.packages("betareg", "~/Baylor Classes/Fall 2013/Thesis")
8 library(betareg)
9
10 setwd("~/Thesis/data")
11 data <- read.csv("bmi.csv")
12
13 gy_logit <- betareg(bmi~ wsource + watloc, data = data)
14
15 confint(gy_logit)
16
17 @
```

## WORKS CITED

- Bayes, C. L., Bazán, J. L., & García, C. (2012), "A New Robust Regression Model for Proportions. *Bayesian Analysis*," 7(4), 841–866. doi:10.1214/12-BA728
- Branscum, A. J., Johnson, W. O., & Thurmond, M. C. (2007), "Bayesian Beta Regression: Applications to Household Expenditure Data and Genetic Distance Between Foot-and-Mouth Disease Viruses," *Australian & New Zealand Journal of Statistics*, 49(3), 287–301. doi:10.1111/j.1467-842X.2007.00481.x
- Cepeda Cuervo, E. (2012). "WinBugs CODE for Beta Regression Models," Universidad Nacional de Colombia.
- Cribari-Neto F, Zeileis A (2010), "Beta Regression in R," *Journal of Statistical Software*, 34(2), 1–24. URL <http://www.jstatsoft.org/v34/i02/>.
- Ferrari SLP, Cribari-Neto F (2004), "Beta Regression for Modelling Rates and Proportions," *Journal of Applied Statistics*, 31(7), 799–815.
- Friendly, Michael (2000), "*Visualizing Categorical Data*," SAS Institute.
- Geyer, Charles, "Introduction to Markov Chain Monte Carlo," University of Minnesota, URL <http://www.stat.umn.edu/geyer/mcmc/talk/mcmc.pdf>.
- Hair, J. F. (Ed.). (2010), *Multivariate data analysis* (7th ed.), Upper Saddle River, NJ: Prentice Hall.
- Ihaka, Ross. (1998) "R : Past and Future History," R Project, URL <http://cran.r-project.org/doc/html/interface98-paper/paper.html>.
- Lunn, D., Spiegelhalter, D., Thomas, A., & Best, N. (2009), "The BUGS project: Evolution, critique and future directions," *Statistics in Medicine*, 28(25), 3049–3067. doi:10.1002/sim.3680

- Ospina, R., & Ferrari, S. L. P. (2012), "A general class of zero-or-one inflated beta regression models," *Computational Statistics & Data Analysis*, 56(6), 1609–1623. doi:10.1016/j.csda.2011.10.005
- Resnik, P., & Hardisty, E. (n.d.), "GIBBS SAMPLING FOR THE UNINITIATED," *June 2010*, URL <http://www.umiacs.umd.edu/~resnik/pubs/gibbs.pdf>
- Sall, J. (1990). Leverage Plots for General Linear Hypotheses. *The American Statistician*, 44(4), 308–315. doi:10.1080/00031305.1990.10475750
- Seaman, John, Dr., "Bayesian Methods: Introduction," Lecture.
- Smithson, M., & Verkuilen, J. (2006), "A better lemon squeezer? Maximum-likelihood regression with beta-distributed dependent variables," *Psychological Methods*, 11(1), 54–71. doi:10.1037/1082-989X.11.1.54
- Swearingen C.J., Melguizo Castro M.S., & Bursac Z, "Inflated Beta Regression: Zero, One, and Everything in Between," SAS® Global Forum Proceedings 2012; Paper 325:1–11, URL <http://support.sas.com/resources/papers/proceedings12/325-2012.pdf>
- Wieczorek, J. and Hawala, S. (2011), "A Bayesian Zero-One Inflated Beta Model for Estimating Poverty in U.S. Counties," Proceedings of the American Statistical Association, Section on Survey Research Methods, Alexandria, VA: American Statistical Association.