2010-01-01

Bayesian Nonparametric Regression with a Flexible Error Term Distribution

Courtney Marie Barnes
University of Texas at El Paso, cmbarnes@miners.utep.edu

BAYESIAN NONPARAMETRIC REGRESSION

WITH A FLEXIBLE ERROR

TERM DISTRIBUTION

COURTNEY BARNES

Department of Mathematics

APPROVED:

Ori Rosen, Chair, Ph.D.

Joan Staniswalis, Ph.D.

Martine Ceberio, Ph.D.

Patricia D. Witherspoon, Ph.D.
Dean of the Graduate School

to my children

Brendan and Liam

with love

BAYESIAN NONPARAMETRIC REGRESSION WITH A FLEXIBLE ERROR TERM DISTRIBUTION

by

COURTNEY BARNES

THESIS

Presented to the Faculty of the Graduate School of

The University of Texas at El Paso

in Partial Fulfillment

of the Requirements

for the Degree of

MASTER OF SCIENCE

Department of Mathematics

THE UNIVERSITY OF TEXAS AT EL PASO

May 2010

Table of Contents

List of Tables

List of Figures

Chapter 1

Literature Review

1.1 Smoothing

More often than not the shape of a conditional mean of a response variable, given a covariate is unknown, leading to the use of nonparametric regression. In this section we will explain the basis function approach to scatterplot smoothing. The discussion in this section is based on Ruppert et al. (2003) and Rosen and Thompson (2009). First, consider the simple linear regression model

$$y_i = \beta_0 + \beta_1 x_i + \epsilon_i. \tag{1.1}$$

The vector of fitted values, $\widehat{\mathbf{y}}$, can be found using

$$\widehat{\mathbf{y}} = X(X'X)^{-1}X'\mathbf{y}, \tag{1.2}$$

where,

$$X = \begin{pmatrix} 1 & x_1 \\ \vdots & \vdots \\ 1 & x_n \end{pmatrix}.$$

The design matrix X can be viewed as a matrix whose columns are the basis functions 1 and x, evaluated at the data points x_i, $i = 1, \ldots, n$. For more general regression functions consider the model

$$y_i = f(x_i) + \epsilon_i, \tag{1.3}$$

where f is an unknown smooth function. In order to handle such a general function we will look at a more complex basis, which uses truncated lines, $(x - \kappa)_+ = \max(0, x - \kappa)$, where κ is called a knot. We note that any linear combination of 1, x, $(x - \kappa_1)_+$, $(x - \kappa_2)_+$, ...,

$(x - \kappa_K)_+$ is a piecewise linear function with knots at $\kappa_1, \ldots, \kappa_K$. This type of function is called a linear spline. The linear spline model for f is

$$f(x) = \beta_0 + \beta_1 x + \sum_{i=1}^{K} u_i (x - \kappa_i)_+, \tag{1.4}$$

where the u_i's are the coefficients of the truncated lines. Let X and Ψ be the design matrices corresponding to the linear and nonlinear parts in equation (1.4),

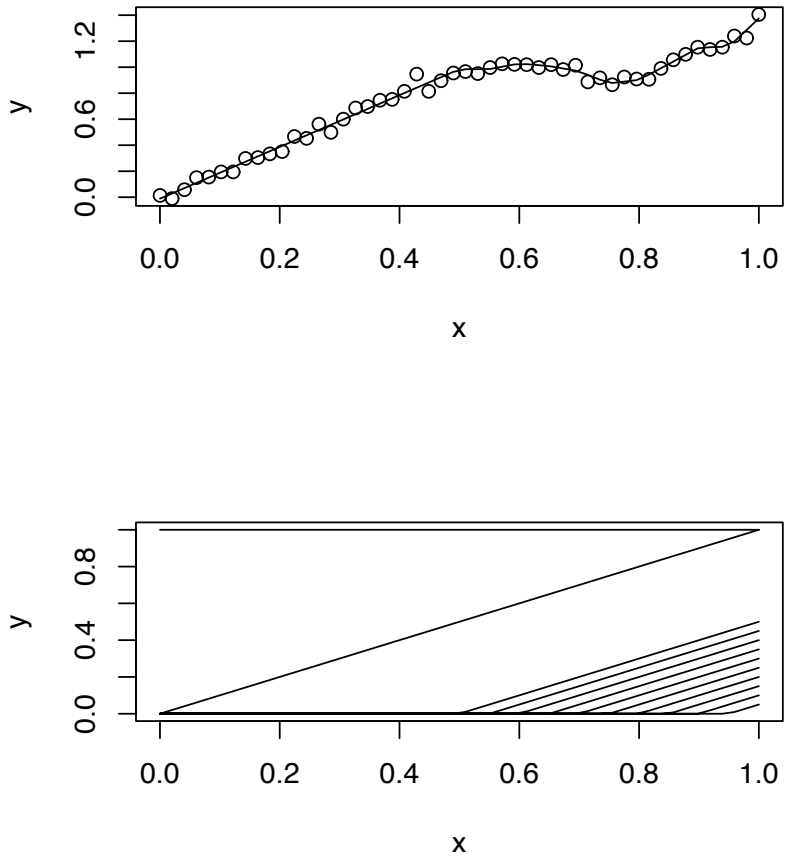

Figure 1.1: Top: data with a fitted linear spline. Bottom: basis functions

$$X = \begin{pmatrix} 1 & x_1 \\ \vdots & \vdots \\ 1 & x_n \end{pmatrix} \quad \text{and} \quad \Psi = \begin{pmatrix} (x_1 - \kappa_1)_+ & \cdots & (x_1 - \kappa_K)_+ \\ \vdots & \ddots & \vdots \\ (x_n - \kappa_1)_+ & \cdots & (x_n - \kappa_K)_+ \end{pmatrix}.$$

To illustrate how the linear spline basis works, Figure 1.1 shows data with a "whip" appearance similar to the examples in Rosen and Thompson (2009) and Ruppert et al. (2003). Equidistant knots are placed at .50, .55, ..., .90, .95. When comparing the panels of Figure 1.1, one can see that the basis functions on the bottom should be able to fully capture the shape of the data on the top panel. Any structure can be fitted by placing basis functions at additional knots.

Different approaches can be used to avoid overfitting. One approach is to use penalized spline regression, in which all knots are retained but their coefficients are constrained to control their influence. Choosing 30-40 knots for a medium-sized dataset is usually sufficient. The penalized spline approach prevents overfitting, leading to a less wiggly fit, by adding a roughness penalty.

The minimization criterion for penalized spline regression is

$$\|\boldsymbol{y} - X\boldsymbol{\beta} - \Psi\boldsymbol{u}\|^2 = (\boldsymbol{y} - X\boldsymbol{\beta} - \Psi\boldsymbol{u})'(\boldsymbol{y} - X\boldsymbol{\beta} - \Psi\boldsymbol{u}),$$

subject to some constraints on the coefficients u_i's. Possible constraints on the u_i's are

1. $\max|u_i| < \mathrm{C}$
2. $\sum |u_i| < \mathrm{C}$
3. $\sum u_i^2 < \mathrm{C}$.

The third constraint is the easiest to implement. First define a $(K+2) \times (K+2)$ matrix, D, by

$$D = \begin{pmatrix} 0_{2\times 2} & 0_{2\times K} \\ 0_{K\times 2} & I_{K\times K} \end{pmatrix}.$$

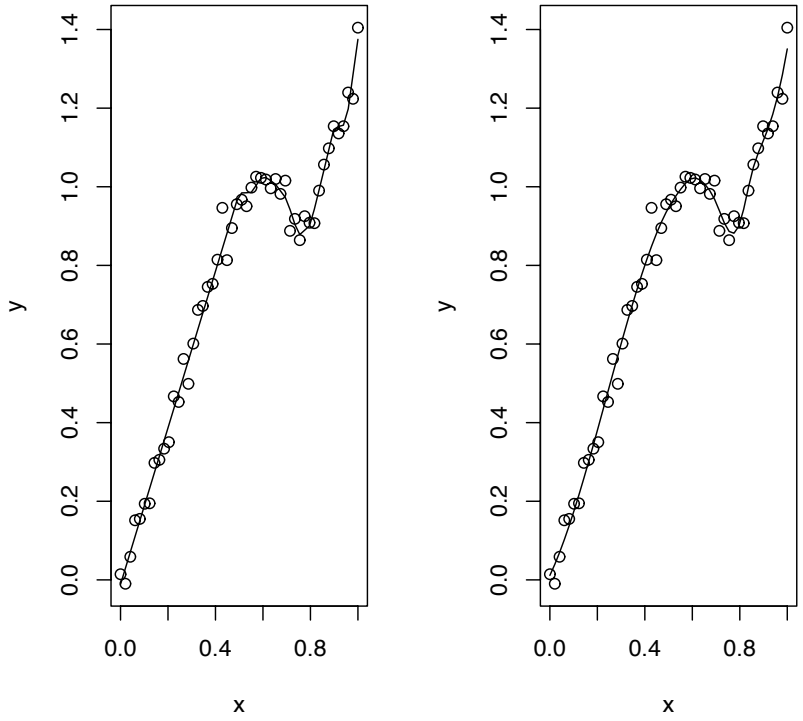

Figure 1.2: Left: fit using truncated lines basis. Right: fit using radial cubic basis

This leads to the minimization criterion

$$\min_{\boldsymbol{\gamma}} \|\boldsymbol{y} - X\boldsymbol{\beta} - \Psi\boldsymbol{u}\|^2,\ \boldsymbol{\gamma}' D\boldsymbol{\gamma} \leq C,$$

where

$$\boldsymbol{\gamma} = (\boldsymbol{\beta}'\ \boldsymbol{u}')',$$

which is equivalent to choosing $\boldsymbol{\gamma}$ minimizing

$$\|\boldsymbol{y} - X\boldsymbol{\beta} - \Psi\boldsymbol{u}\|^2 + \lambda^2 \boldsymbol{\gamma}' D\boldsymbol{\gamma}, \tag{1.5}$$

where $\lambda \geq 0$ and depends on C. Equation (1.5) is equivalent to

$$\|\boldsymbol{y} - X\boldsymbol{\beta} - \Psi\boldsymbol{u}\|^2 + \lambda^2 \|\boldsymbol{u}\|^2.$$

Dividing the last equation by σ_ε^2 results in

$$\frac{1}{\sigma_\varepsilon^2}\|\boldsymbol{y} - X\boldsymbol{\beta} - \Psi\boldsymbol{u}\|^2 + \frac{\lambda^2}{\sigma_\varepsilon^2}\|\boldsymbol{u}\|^2. \tag{1.6}$$

This criteria for fitting a nonparamteric regression function is a form of ridge regression, which also arises in fitting the linear mixed effects model (Henderson 1950). Consider the linear mixed effects model

$$\mathbf{y} = X\boldsymbol{\beta} + \Psi\boldsymbol{u} + \boldsymbol{\varepsilon}. \tag{1.7}$$

In (1.7), $\boldsymbol{\varepsilon} \sim N(0, R)$, $\boldsymbol{u} \sim N(0, G)$ and G and R are covariance matrices. Conditioning on $\boldsymbol{u}$,

$$\boldsymbol{y}|\boldsymbol{u} \sim N(X\boldsymbol{\beta} + \Psi\boldsymbol{u}, R) \text{ and } \boldsymbol{u} \sim N(0, G).$$

The augmented likelihood of $(\boldsymbol{y}, \boldsymbol{u})$ is

$$\frac{1}{(2\pi)^{\frac{N}{2}}|R|^{\frac{1}{2}}} \exp\left\{-\frac{1}{2}(\boldsymbol{y} - X\boldsymbol{\beta} - \Psi\boldsymbol{u})'R^{-1}(\boldsymbol{y} - X\boldsymbol{\beta} - \Psi\boldsymbol{u})\right\} \frac{1}{(2\pi)^{\frac{K}{2}}|G|^{\frac{1}{2}}} \exp\left\{-\frac{1}{2}\boldsymbol{u}'G^{-1}\boldsymbol{u}\right\}.$$

Fixing R and G and taking the log of the augmented likelihood gives

$$-\frac{1}{2}(\boldsymbol{y} - X\boldsymbol{\beta} - \Psi\boldsymbol{u})'R^{-1}(\boldsymbol{y} - X\boldsymbol{\beta} - \Psi\boldsymbol{u}) - \frac{1}{2}\boldsymbol{u}'G^{-1}\boldsymbol{u}. \tag{1.8}$$

Maximizing equation (1.8) is equivalent to the minimization of

$$(\boldsymbol{y} - X\boldsymbol{\beta} - \Psi\boldsymbol{u})'R^{-1}(\boldsymbol{y} - X\boldsymbol{\beta} - \Psi\boldsymbol{u}) + \boldsymbol{u}'G^{-1}\boldsymbol{u}. \tag{1.9}$$

Taking $R = \sigma_\varepsilon^2 I_N$ and $G = \sigma_u^2 I_K$ and comparing (1.6) to (1.9) shows that the mixed effects model leads to criterion (1.6) where $\lambda^2 = \sigma_\varepsilon^2/\sigma_u^2$.

Instead of the truncated line basis other bases can be used. In what follows we use low-rank cubic radial basis functions,

$$f(x) = \beta_0 + \beta_1 x + \sum_{i=1}^{K} u_i|x - \kappa_i|^3.$$

Using a cubic radial basis will often lead to faster convergence in the MCMC framework, and a fitted curve that is aesthetically more appealing than the fit obtained with truncated line basis, see Figure 1.2. In equation (1.5), Ψ and D now become

$$\Psi_K = \begin{pmatrix} |x_1-\kappa_1|^3 & \cdots & |x_1-\kappa_K|^3 \\ \vdots & \ddots & \vdots \\ |x_n-\kappa_1|^3 & \cdots & |x_n-\kappa_K|^3 \end{pmatrix} \text{ and } D = \begin{pmatrix} 0_{2\times 2} & 0_{2\times K} \\ 0_{K\times 2} & \Omega_{K\times K} \end{pmatrix}.$$

The subscript K on Ψ will be explained below. The (k,l)th element of Ω_K is $|\kappa_k - \kappa_l|^3$. Using the connection between the roughness penalty approach and mixed effects models leads to the model,

$$\boldsymbol{y} = X\boldsymbol{\beta} + \Psi_K \boldsymbol{u} + \boldsymbol{\varepsilon},\ \mathrm{Cov}(\boldsymbol{u}) = \sigma_u^2 \Omega_K^-,\ \mathrm{Cov}(\boldsymbol{\varepsilon}) = \sigma_\varepsilon^2 I_N. \tag{1.10}$$

Note that since Ω_K is not a positive definite matrix, we have used the generalized inverse notation in (1.10). To make model (1.10) a valid mixed effects model, $\mathrm{Cov}(\boldsymbol{u})$ needs to be a positive definite matrix. This can be done by expressing $\mathrm{Cov}(\boldsymbol{u})$ as $\mathrm{Cov}(\boldsymbol{u}) = \sigma_u^2(\Omega_K^{-\frac{1}{2}})(\Omega_K^{-\frac{1}{2}})$ where $\Omega_K^{-\frac{1}{2}}$ is obtained from the singular value decomposition. Letting $\boldsymbol{b} = \Omega_K^{-\frac{1}{2}}\boldsymbol{u}$ and $\Psi = \Psi_K \Omega_K^{-\frac{1}{2}}$, the mixed model (1.10) is equivalent to

$$\boldsymbol{y} = X\boldsymbol{\beta} + \Psi\boldsymbol{b} + \boldsymbol{\varepsilon},\ \mathrm{Cov}\begin{pmatrix} \boldsymbol{b} \\ \boldsymbol{\varepsilon} \end{pmatrix} = \begin{pmatrix} \sigma_b^2 I_K & 0 \\ 0 & \sigma_\varepsilon^2 I_n \end{pmatrix}. \tag{1.11}$$

Another popular basis uses B-splines. The following is based on Hastie et al. (2001). We first augment the knot sequence. Let $\xi_0 < \xi_1$ and $\xi_K < \xi_{K+1}$ be two boundary knots which typically define the domain over which we will evaluate our spline. We define the augmented knot sequence τ such that

- $\tau_1 \le \tau_2 \le \cdots \le \tau_M \le \xi_0$
- $\tau_{j+M} = \xi_j,\ j = 1, \ldots, K$
- $\xi_{K+1} \le \tau_{K+M+1} \le \tau_{K+M+2} \le \cdots \le \tau_{K+2M}$

The values of these knots beyond the boundary are arbitrary, and it is customary to make them all the same and equal to ξ_0 and ξ_{K+1}, respectively. Let $B_{i,m}(x)$, be the ith B-spline basis function of order m for the knot sequence τ, $m \leq M$. They are defined recursively in terms of divided differences:

$$B_{i,1} = \begin{cases} 1 & \text{if } \tau_i \leq x < \tau_{i+1} \\ 0 & \text{otherwise} \end{cases},$$

for $i = 1, \ldots, K + 2M - 1$. These are also known as Haar basis functions.

$$B_{i,m}(x) = \frac{x - \tau_i}{\tau_{i+m-1} - \tau_i} B_{i,m-1} + \frac{\tau_{i+m} - x}{\tau_{i+m} - \tau_{i+1}} B_{i+1,m-1}$$

for $i = 1, \ldots, K + 2M - m$. When $M = 4$, $B_{i,4}$, $i = 1, \ldots, K + 4$ are the $K + 4$ cubic spline basis functions for the knot sequence ξ. This recursion can be continued and will generate the B-splines for any order spline.

Another approach to smoothing is knot selection. This is done by first selecting a large number of knots and placing them equidistantly throughout the covariate or at specific percentiles. Each knot is assigned an indicator variable, 1 if the knot is to be retained at that location and 0 if the knot is to be removed from that location. In a Bayesian framework, the indicators are sampled at each iteration. (Thompson and Rosen 2008)

1.2 Robust Regression

The goal of robust regression is to dampen the effect of departures from the model assumptions. There are different types of robust regression methods such as locally weighted scatterplot smoothing, M-estimation and least absolute deviation regression (LAD) to name a few. LOESS, developed by Clevland (1979), fits successive linear regression functions in local neighborhoods in order to obtain a smooth fit. This is done by fitting a low-degree polynomial at each data point, using a neighborhood or a subset of the data points around the current point, fitting the polynomial by weighted least squares, giving more weight to

points close to the current data point and less weight to those farther away. The process ends when this has been done to all points in the dataset. In Kutner et al. (2004) the process of fitting a local polynomial is explained using the case where there are two predictor variables, (X_{j1}, X_{j2}), $j = 1, \ldots, n$. First, some distance is needed, for example the Euclidean distance,

$$d_i = [(X_{i1} - X_{j1})^2 + (X_{i2} - X_{j2})^2]^{\frac{1}{2}}, \tag{1.12}$$

where d_i is the distance for the ith observation. A proportion of q observations is taken nearest to the point to define the neighborhood around that point. To perform the weighted least squares, a weight function is necessary. One possible choice is the tricube weight function, w_i.

$$w_i = \begin{cases} (1 - (d_i/d_q)^3)^3 & \text{if } d_i < d_q \\ 0 & \text{otherwise.} \end{cases} \tag{1.13}$$

As evident in equation (1.13), any observations outside the neighborhood are assigned a weight of zero, and the weight increases as the distance decreases. Thus the mean response at (X_{i1}, X_{j1}) is fitted locally. After all the weights have been found for all observations, weighted least squares is then used to fit a first or second degree polynomial. Only the value of the fitted surface at (X_{i1}, X_{j1}) is retained.

M-estimation is a maximum likelihood type estimation, developed by Huber (1964). It was dubbed "M-estimation" because in the simplest cases the estimators turned out to be means and medians. M-estimators for location parameters are defined as $T = T_n(x_1, \ldots, x_n)$, where T minimizes $\sum_t \rho(x_i - T)$ and ρ is a nonconstant function. It is easy to see that for the class of estimators for which $\rho(t) = t^2$, T is equal to the sample mean, and when $\rho(t) = |t|$, T is equal to the sample median.

LAD regression is another approach to robust regression. The following explanation is based on Faria and Melfi (2004) and Kutner et al. (2004). LAD regression, also commonly

called L_1-norm regression, is one of the widely used robust regression procedures. Consider the linear regression model,

$$y_i = \beta_0 + \sum_{j=1}^{p} \beta_j x_{ij} + \epsilon_i, \text{ for } i = 1, \ldots, n, \tag{1.14}$$

where p is the number of explanatory variables, and ϵ_i is the error term. The vector $\boldsymbol{\beta} = (\beta_0, \ldots, \beta_p)'$ is determined by minimizing

$$\sum_{i=1}^{n} |y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij}|. \tag{1.15}$$

Since absolute deviations are used rather than squared deviations, LAD is more resistant to outliers than the method of least squares. What also separates LAD from least squares is that the residuals will generally not sum to zero and the estimated coefficients may not be unique.

Another form of robust regression is iteratively reweighted least squares (IRLS) robust regression. The idea behind IRLS is to weight an observation based on how outlying it is, using the residual as the measure. This dampens the influence of the outlying point making the method more robust. The weights are continually revised leading to new residuals after each iteration. The process is repeated until it converges. The following is a summary of the steps that can be found in Kutner et al. (2004):

1. Choose a weight function.
2. Obtain starting weights for all observations.
3. Using the starting weights, obtain the residuals from the fitted regression function.
4. Repeat step 3 to obtain new weights.
5. Continue until the weights converge.

There are many choices for weight functions that may be used for reducing the influence of outliers, such as the tricube weight function (see 1.13). Two other widely used weight functions are the Huber function

$$w = \begin{cases} 1 & \text{if } |u| \leq 1.345 \\ \frac{1.345}{|u|} & \text{otherwise} \end{cases} \tag{1.16}$$

and the bisquare weight function,

$$w = \begin{cases} [1 - (\frac{u}{4.685})^2]^2 & \text{if } |u| \leq 4.685 \\ 0 & \text{otherwise.} \end{cases} \tag{1.17}$$

Here w is the weight, and u is the scaled residual $\frac{e_i}{MAD}$, where

$$MAD = \frac{1}{.6745}\text{median}\{|e_i - \text{median}\{e_i\}|\}. \tag{1.18}$$

The constants in the respective weight functions are called tuning constants, chosen to make ILRS 95 percent efficient for data under a normal error term. When choosing a starting value, the weight function must be taken into account. For instance, with the Huber weight function, starting values can be obtained using ordinary least squares. When using the bisquare weight function, ordinary least squares cannot be used. Instead, the Huber weight function is used to get an initial robust regression fit. Another way to get starting value for the bisquare weights is by least absolute residuals regression. Choosing the number of iterations depends on how long it takes to converge. This may be observed by looking at iteration plots of the weights, residuals, and coefficients to notice any changes, ensuring that they are slight.

1.3 Heavy-tailed Error terms

In general, most models are based on the assumption that the error term is normally distributed. The resulting inferences may be sensitive to outliers, which is why sometimes a heavy-tailed error distribution is used. Using a heavy-tailed distribution provides not

only the means to detect outliers but also to accommodate them in the model (West, 1984). When datasets exhibit heavy-tailed behavior, one choice is Student's t distribution for the distribution of the error term. Fonseca et al. (2008) note that the use of Student's t distribution for the error term reduces the influence of outliers, making statistical analysis more robust. One example of this is using a t distribution with a small value of degrees of freedom (Gelman et al., 2004). The degrees of freedom are directly related to the degree of robustness, that is, the smaller the degrees of freedom the higher the degree of robustness (Fonseca et al., 2008). The degrees of freedom must be at least 3 for the mean and variance to exist. Barros et al. (2008) explore the possibility of using a heavy-tailed distribution for the error term for the Birnbaum-Saunders (1969) models. This is a probability model originating from a physical problem related to material fatigue. It is now commonly used in many medical problems such as chronic heart disease and different types of cancer. The Birnbaum-Saunders model is a particular case of the generalized Birnbaum-Saunders (GBS). The GBS is a standard symmetrical distribution on the real line. The Birnbaum-Saunders model is closely related to the normal distribution and is sensitive to outliers in the same respect. Consequently, maximum likelihood estimates from the Birnbaum-Saunders models are sensitive to outliers (Barros et al. 2008). In this paper they compare the log-Birnbaum-Saunders model with a normal error term with that of the log-Birnbaum-Saunders model with a Student's t error term on respective datasets that are regularly used with the Birnbaum-Saunders model. Their findings agree with their hypothesis that using Student's t error term for the log-Birnbaum-Saunders regression model is appropriate for fitting the data compared to the log-Birnbaum-Saunders regression model with a normal error term. Fonseca et al. (2008) develop objective Bayesian analyses based on Jefferys' prior and on the independence Jefferys' prior for the linear regression model with independent Student's t errors with unknown ν. They apply their approach to a dataset on per capita income and per capita spending in public schools by state in the United States in 1979. When looking at the linear and quadratic models under both Gaussian and Student's t errors against the linear model with Gaussian errors, their

framework clearly points to Student's-t linear model as the best model. West (1984) notes that the Student t distribution belongs to the class of scale mixtures of normals with a gamma mixing distribution. Therefore, the Student t may be modeled as a mixture of Gaussian distributions with different variances. Stigler (1973) makes note of Simon Newcomb as the first to introduce a mixture of normal densities to model a heavy-tailed density. In 1886, Newcomb was critical of the overuse of outlier criteria and did discard outliers, but only when the deviations were large. Instead, he used a mixture of normals with differing variances in order to capture the distribution of the data. Fernandez and Steel (1999) consider errors that are distributed as a scale of mixtures of normals when the assumption of normality is not quite tenable, and thicker tales are needed in order to capture the data properly.

Chapter 2

The Model, Priors, and Sampling Scheme

2.1 The Model

Parametric regression techniques are attractive when the regression function has a known shape. In other cases, nonparametric smoothing techniques may be more suitable. Consider model (1.3). Usually, ε_i is normally distributed with a zero mean and variance σ^2. Our goal in this thesis is to make the estimated regression function more resistent to outliers. To this end, we model the error term as a mixture of a Student's t density with fixed small degrees of freedom, ν, and a normal density. Thus, the density function, h, of ε_i is

$$h(\varepsilon_i) = p\, s(\varepsilon_i) + (1-p)\, g(\varepsilon_i), \tag{2.1}$$

where s is the probability density function of a $t_\nu(0,\sigma)$ random variable, given by,

$$s(\varepsilon_i) = \frac{\Gamma((\nu+1)/2)}{\Gamma(\nu/2)\sqrt{\nu\pi\sigma}} \left(1 + \frac{1}{\nu}\left(\frac{\varepsilon_i}{\sigma}\right)^2\right)^{-(\nu+1)/2}.$$

Note that $\text{var}(\varepsilon_i) = \frac{\nu}{\nu-2}\sigma^2$. The function g is the probability density function of a $N(0,\sigma_\varepsilon^2)$ random variable, and p is a mixing probability such that $0 \leq p \leq 1$. To motivate modeling the error term as a mixture, first recall that t_ν distributions with small values of ν have thicker tails than that of the normal distribution, see Figure 2.1. Modeling the error term to accommodate outliers can be done in various ways. An alternative to our approach is to use the $t_\nu(0,\sigma)$ with both values of ν and σ unknown (Fonesca et al., 2008). The mixture (2.1) can flexibly model a heavy tailed distribution, without requiring the estimation of ν.

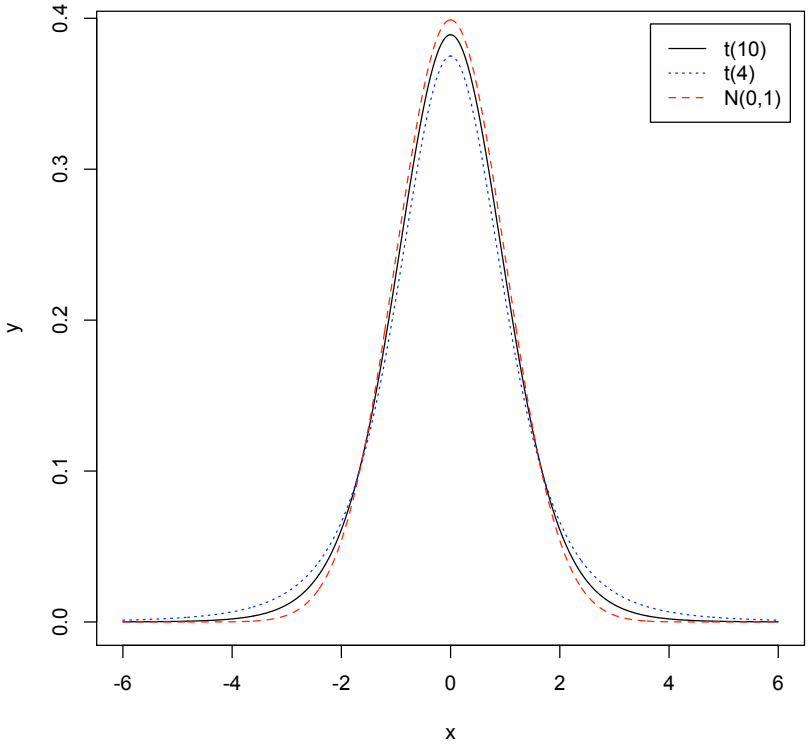

Figure 2.1: The solid line is a Student's t density with 10 degrees of freedom, the dotted line is a Student's t density with 4 degrees of freedom, and the dashed line is a standard normal density.

Rather, ν is fixed at a small enough value, say 4. To show that this mixture can give rise to varying degrees of tail thickness, we plot in Figure 2.2 four situations corresponding to $\nu = 6, 8, 10, 15$. In each plot, the $t_\nu(0, \sigma)$ density is shown in solid line overlaid with an appropriate mixture density. The parameter values corresponding to the mixture densities are given in Table 2.1.

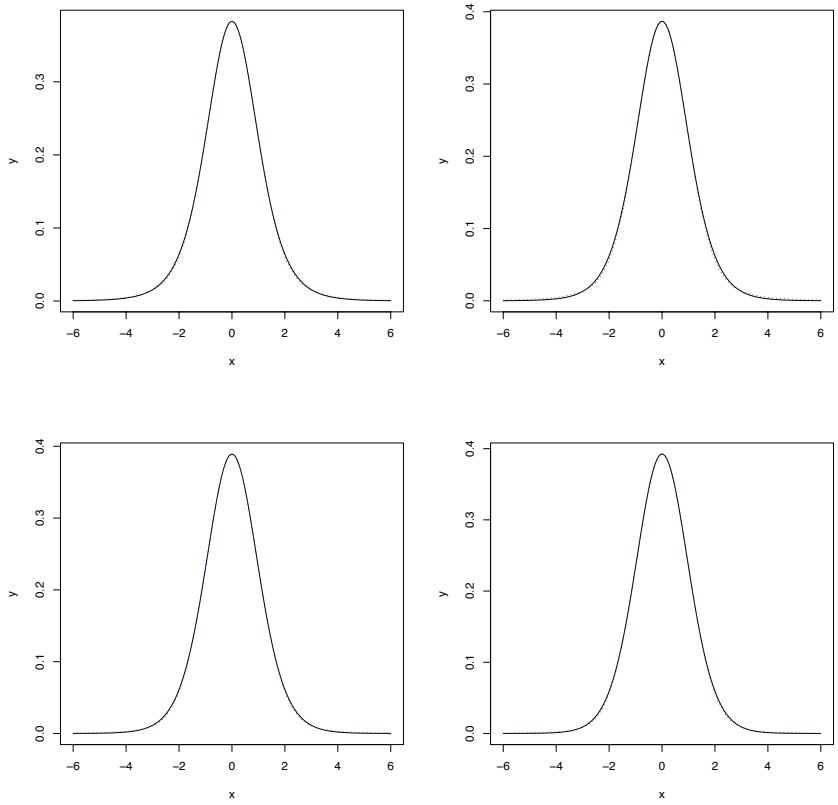

Figure 2.2: The solid line is a $t_\nu(0, \sigma)$ density with varying values of ν and σ. Clockwise starting from top left, the corresponding degrees of freedom are 6,8,15 and 10; the dashed line is the appropriate mixture density.

To model f, we use the mixed effects model (1.11) of Chapter 1,

$$\boldsymbol{y} = X\boldsymbol{\beta} + \Psi\boldsymbol{u} + \boldsymbol{\varepsilon},$$

Table 2.1: Parameter Values

ν	σ	σ_ε^2	p
6	1.09	.85	.569
8	1.23	.81	.4
10	1.05	.96	.38
15	1.09	.95	.25

where the error distribution is given by (2.1). The $t_\nu(0,\sigma)$ component in equation (2.1) is equivalent to the model $E_i \mid V_i \sim N(0, V_i)$ where $V_i \sim IG(\frac{\nu}{2}, \frac{\nu\sigma^2}{2})$ (see Gelman et al. 2004, page 303). To see this, compute the marginal density of E_i:

$$\begin{aligned}
s(\varepsilon_i) &= \int_0^\infty f(\varepsilon_i|v_i)g(v_i)dv_i \\
&\propto \int_0^\infty \frac{1}{\sqrt{v_i}} \exp\left\{-\frac{\varepsilon_i^2}{2v_i}\right\} \times v_i^{-(\frac{\nu}{2}+1)} \exp\left\{-\frac{\nu\sigma^2}{2v_i}\right\} dv_i \\
&= \int_0^\infty v_i^{-\left(\frac{\nu+1}{2}+1\right)} \exp\left\{-\frac{1}{v_i}\left(\frac{\varepsilon_i^2}{2} + \frac{\nu\sigma^2}{2}\right)\right\} dv_i \\
&\propto \left(\frac{\varepsilon_i^2}{2} + \frac{\nu\sigma^2}{2}\right)^{-\left(\frac{\nu+1}{2}\right)} \\
&\propto \left(1 + \frac{\varepsilon_i^2}{\nu\sigma^2}\right)^{-\left(\frac{\nu+1}{2}\right)},
\end{aligned}$$

which is the kernel of the $t_\nu(0,\sigma)$ density.

The augmented likelihood of $\boldsymbol{\theta}$ is

$$\begin{aligned} L(\boldsymbol{\theta} \mid \boldsymbol{y}, \boldsymbol{u}, \boldsymbol{v}, X) &= \prod_{i=1}^{n} \left[p \frac{1}{\sqrt{2\pi v_i}} \exp\left\{ -\frac{1}{2v_i}(y_i - \boldsymbol{x}_i'\boldsymbol{\beta} - \boldsymbol{\psi}_i'\boldsymbol{u})^2 \right\} \right. \\ &+ (1-p) \frac{1}{\sqrt{2\pi\sigma_\varepsilon^2}} \exp\left\{ -\frac{1}{2\sigma_\varepsilon^2}(y_i - \boldsymbol{x}_i'\boldsymbol{\beta} - \boldsymbol{\psi}_i'\boldsymbol{u})^2 \right\} \\ &\times \left. \frac{\left(\frac{\nu\sigma^2}{2}\right)^{\frac{\nu}{2}}}{\Gamma(\frac{\nu}{2})} v_i^{-(\frac{\nu}{2}+1)} \exp\left(-\frac{\nu\sigma^2}{2v_i} \right) \right] \\ &\times \frac{1}{(2\pi\sigma_u^2)^{\frac{K}{2}}} \exp\left\{ -\frac{1}{2\sigma_u^2}\boldsymbol{u}'\boldsymbol{u} \right\}, \end{aligned}$$

where $\boldsymbol{x}_i'$ and $\boldsymbol{\psi}_i'$, are the ith rows of X and Ψ, respectively, and $\boldsymbol{\theta} = (p, \boldsymbol{\sigma}_\varepsilon^2, \boldsymbol{\beta}')'$. In our computations we further augment the likelihood with unobservable mixture component indicators, z_i, $i = 1, \ldots, n$, where

$$z_i = \begin{cases} 1 & \text{if obs } i \text{ came from the } N(\mu_i, v_i) \text{ distribution} \\ 0 & \text{if obs } i \text{ came from the } N(\mu_i, \sigma_\varepsilon^2) \text{ distribution,} \end{cases}$$

where $\mu_i = \boldsymbol{x}_i'\boldsymbol{\beta} + \boldsymbol{\psi}_i'\boldsymbol{u}$. The augmented likelihood thus becomes,

$$\begin{aligned} L(\boldsymbol{\theta}|\boldsymbol{y}, \boldsymbol{u}, Z, X) &= \prod_{i=1}^{n} \left\{ \left(p \, \frac{1}{\sqrt{2\pi v_i}} \exp\left\{ -\frac{1}{2v_i}(y_i - \boldsymbol{x}_i'\boldsymbol{\beta} - \boldsymbol{\psi}_i'\boldsymbol{u})^2 \right\} \right)^{z_i} \right. \\ &\times \left. \left((1-p) \, \frac{1}{\sqrt{2\pi\sigma_\varepsilon^2}} \exp\left\{ -\frac{1}{2\sigma_\varepsilon^2}(y_i - \boldsymbol{x}_i'\boldsymbol{\beta} - \boldsymbol{\psi}_i'\boldsymbol{u})^2 \right\} \right)^{1-z_i} \right\} \\ &\times \prod_{i=1}^{n} \left(\frac{\left(\frac{\nu\sigma^2}{2}\right)^{\frac{\nu}{2}}}{\Gamma(\frac{\nu}{2})} v_i^{-(\frac{\nu}{2}+1)} \exp\left(-\frac{\nu\sigma^2}{2v_i} \right) \right) \\ &\times \frac{1}{(2\pi\sigma_u^2)^{\frac{K}{2}}} \exp\left(-\frac{1}{2\sigma_u^2}\boldsymbol{u}'\boldsymbol{u} \right). \end{aligned}$$

In a Bayesian framework, prior distributions need to be placed on all the parameters.

2.2 The Priors

We place the following priors on $\boldsymbol{\beta}$, σ_u^2, σ_ε^2, p and, σ^2.

1. $\boldsymbol{\beta} \sim N_2(\mathbf{0}, \sigma_\beta^2 I_2)$, where σ_β^2 is a large known value, I_2 is a 2×2 identity matrix, and N_2 is a bivariate normal distribution.

2. $\sigma_u^2 \sim U(0, c_u)$, where c_u is a large known value, and U is the continuous uniform distribution.

3. $\sigma_\varepsilon^2 \sim U(0, c_\varepsilon)$, where c_ε is a large known value.

4. $p \sim \text{beta}(\alpha_1, \alpha_2)$, where α_1 and α_2 are small known values.

5. $\sigma^2 \sim U(0, c_\sigma)$,where c_σ is a large known value.

2.3 The Sampling Scheme

This section uses Gibbs sampling to sample from the posterior distribution of all the parameters. This is done by sampling from the conditional distributions described below. We start by initializing the values of p, σ_u^2, σ_ε^2, $\boldsymbol{u}$, σ^2, v_i, $i = 1, \ldots, n$, and $\boldsymbol{\beta}$.

1. For each i, $i = 1, \ldots, n$, sample z_i from the Bernoulli distribution, with success probability

$$\begin{aligned} P(z_i = 1 \mid \boldsymbol{\theta}, \boldsymbol{u}, v_i, y_i, \boldsymbol{x}_i) &= p\frac{1}{\sqrt{2\pi v_i}} \exp\left\{-\frac{1}{2v_i}(y_i - \boldsymbol{x}_i{}'\boldsymbol{\beta} - \boldsymbol{\psi}_i{}'\boldsymbol{u})^2\right\} \\ &\Big/ \left[p\frac{1}{\sqrt{2\pi v_i}} \exp\left\{-\frac{1}{2v_i}(y_i - \boldsymbol{x}_i{}'\boldsymbol{\beta} - \boldsymbol{\psi}_i'\boldsymbol{u})^2\right\}\right. \\ &+ \left.(1-p)\frac{1}{\sqrt{2\pi\sigma_\varepsilon^2}} \exp\left\{-\frac{1}{2\sigma_\varepsilon^2}(y_i - \boldsymbol{x}_i{}'\boldsymbol{\beta} - \boldsymbol{\psi}_i'\boldsymbol{u})^2\right\}\right]. \end{aligned}$$

2. Sample σ_ε^2 from the truncated inverse gamma distribution, $IG(\frac{1}{2}\left[(n - \sum_{i=1}^n z_i) - 1\right], \frac{1}{2}(\boldsymbol{y} - C\boldsymbol{\gamma})'Z_d(\boldsymbol{y} - C\boldsymbol{\gamma}))I_{[0,c_\varepsilon)}(\sigma_\varepsilon^2)$, where $Z_d = \text{diag}\,(1 - z_1, \ldots, 1 - z_n)$ and

$$I_{[0,c_\varepsilon)}(\sigma_\varepsilon^2) = \begin{cases} 1 & \text{if } 0 \le \sigma_\varepsilon^2 \le c_\varepsilon \\ 0 & \text{otherwise.} \end{cases}$$

   Details on how to sample from the truncated inverse gamma distribution can be found in Appendix A.

3. Sample p from the beta distribution, beta$(\sum_{i=1}^{n} z_i + \alpha_1, n - \sum_{i=1}^{n} z_i + \alpha_2)$.

4. Sample σ_u^2 from the truncated inverse gamma distribution $IG(\frac{K}{2} - 1, \frac{1}{2}\boldsymbol{u}'\boldsymbol{u})I_{[0,c_u)}(\sigma_u^2)$.

5. Sample σ^2 from the truncated gamma distribution $\mathrm{G}(\frac{\nu}{2}\sum_{i=1}^{n} z_i + 1, \frac{\nu}{2}\sum_{i=1}^{n} \frac{z_i}{v_i})I_{[0,c_\sigma)}(\sigma^2)$. Details on how to sample from a truncated gamma distribution can be found in Appendix A.

6. For each i, $i = 1, \ldots, n$, sample v_i from the inverse gamma distribution $IG(\frac{z_i}{2} + \frac{\nu}{2}, \frac{1}{2}[z_i(y_i - \boldsymbol{C}_{i,}'\boldsymbol{\gamma})^2 + \nu\sigma^2])$, where $\boldsymbol{C}_i'$ is the ith row of the C matrix.

7. Sample $\boldsymbol{\gamma}$ from a multivariate Normal, $MVN(\boldsymbol{\mu}, \Sigma)$ where,

$$\Sigma = \left(C'Z_{1d}C + C'Z_{2d}C + \frac{1}{\sigma_u^2}D + \frac{1}{\sigma_\beta^2}P\right)^{-1},$$

$$\boldsymbol{\mu} = \Sigma[C'Z_{1d}\boldsymbol{y} + C'Z_{2d}\boldsymbol{y}],$$

and,

$$Z_{1d} = \mathrm{diag}\left(\frac{z_1}{v_1}, \ldots, \frac{z_n}{v_n}\right), \qquad Z_{2d} = \mathrm{diag}\left(\frac{1 - z_1}{\sigma_\varepsilon^2}, \ldots, \frac{1 - z_n}{\sigma_\varepsilon^2}\right),$$

$$D = \begin{pmatrix} 0_{2\times 2} & 0_{2\times K} \\ 0_{K\times 2} & I_{K\times K} \end{pmatrix}, \text{ and } P = \begin{pmatrix} I_{2\times 2} & 0_{2\times K} \\ 0_{K\times 2} & 0_{K\times K} \end{pmatrix}.$$

8. Repeat for M iterations.

Chapter 3

Simulation Study

To further test our model we perform a small simulation study. We consider the following four settings with two shapes for the regression function

1. $n = 500$, $p = .9$
2. $n = 100$, $p = .9$
3. $n = 500$, $p = .1$
4. $n = 100$, $p = .1$,

where p is the mixing weight of the t component. The advantage to using the same shape is to be able to compare the above four different settings which represent a large data set with a heavy-tailed distributed error term, a large data set whose error term is not heavy-tailed distributed, a small data set with a heavy-tailed distributed error term, and a small dataset whose error term is not heavy-tailed. In addition to looking at how the model performs in these different settings, we also compare it to a model that assumes a normal error term. One hundred datasets are generated in each setting. Each dataset is fitted using the Gibbs sampler with a total of 10,000 iterations, 2000 of which are used as burn-in. In order to do the comparison we use the mean square error (MSE) and pointwise credible intervals . The MSE is defined as

$$\text{MSE}(\hat{f}) = \frac{1}{n}\sum_{i=1}^{n}(f(x_i) - \hat{f}(x_i))^2,$$

where f is the true function and $\hat{f}$ is its estimate. A $100(1-\alpha)\%$ credible interval is constructed as follows. Let $\hat{f}_t(x)$ be the estimated function value at x on the tth iteration. A $(1-\alpha)$-level credible interval for $f(x)$ is $[f^L(x), f^U(x)]$, where

- $f^L(x)$ is the $\alpha/2$ empirical quantile of $\hat{f}_1(x), \ldots, \hat{f}_T(x)$
- $f^U(x)$ is the $(1-\alpha/2)$ empirical quantile of $\hat{f}_1(x), \ldots, \hat{f}_T(x)$
- $1, \ldots, T$ are the last T iterations (after burn-in).

In what follows, MSE($\hat{f}_{mix}$) denotes the MSE of the fit corresponding to the model with the mixture error term distribution. MSE($\hat{f}_{nor}$) is the MSE corresponding to the model with a normal error term.

3.1 Simulation Study 1

The true regression function we use in the first simulation study is

$$f(x) = 2\sin(4\pi x).$$

Setting 1: $n = 500$, $p = .9$

For the first setting, $h(\varepsilon_i) = .9s(\varepsilon_i) + .1g(\varepsilon_i)$, where $s(\varepsilon_i) \sim t_4(0, 1)$ and $g(\varepsilon_i) \sim N(0, 1)$. The x_i's, $i = 1, \ldots, 500$, are equidistant in the range $[0, 1]$. Figure 3.1 shows one realization with the true function (black solid line), $\hat{f}_{mix}(x)$ (red dashed line) and $\hat{f}_{nor}(x)$ (blue dotted line). Here we can see that when we assume a normal error term when the true error is from a heavy-tailed distribution, the fit appears to be influenced by outliers more so than the fit based on the mixture error term. Figure 3.2 is a histogram of $\hat{p}_k$, $k = 1, \ldots, 100$, where $\hat{p}_k$ is the estimate based on the kth dataset. Here it can be seen that approximately 45% of the estimates fell between .9 and 1. The overall mean is $\frac{1}{100}\sum_{i=1}^{100}\hat{p}_k = .831$.

Figure 3.3 shows side by side boxplots of the MSE for each approach, MSE($\hat{f}_{mix}$) is on the left. This further supports our approach, since MSE($\hat{f}_{mix}$) is smaller on average than MSE($\hat{f}_{nor}$). To further compare the two models, we look at 95% pointwise credible intervals based on each approach. In particular, at each of 10 covariate values, we compute the actual coverage based on the 100 simulated datasets. Examples of pointwise credible intervals based on a single realization are presented in Figure 3.4 and Figure 3.5 for the

mixture and normal cases, respectively. Table 3.1 contains the numbers of times the true function falls within each credible interval at 10 points chosen equidistantly throughout the range of x. The mixture approach appears to perform better.

Table 3.1: Simulation 1, setting 1: Pointwise Credible Intervals Comparison

Model	x_1	x_{56}	x_{112}	x_{167}	x_{223}	x_{278}	x_{334}	x_{389}	x_{445}	x_{500}
Mix	92	89	97	92	97	97	93	98	95	85
Nor	99	69	98	41	52	93	92	99	82	96

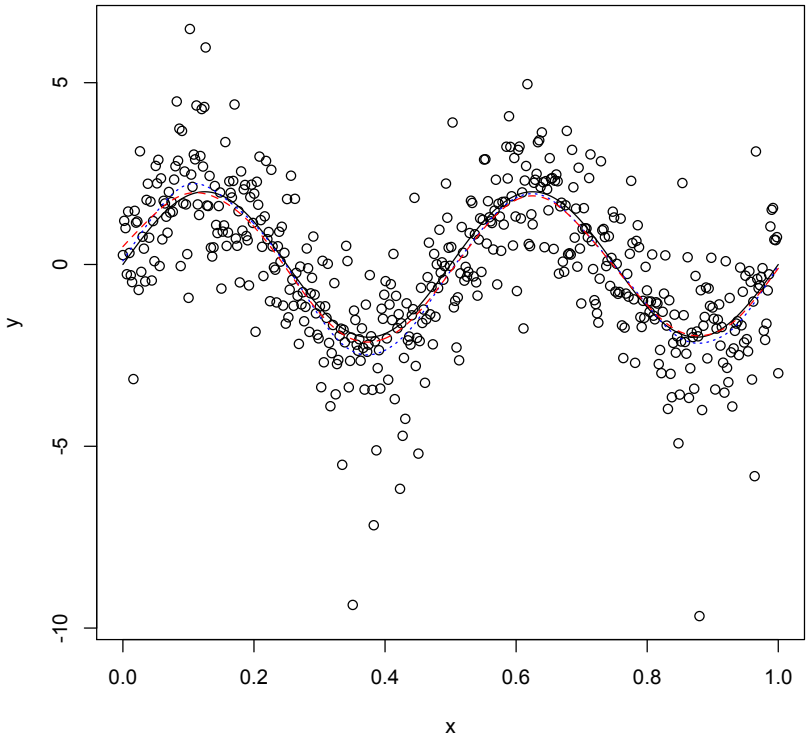

Figure 3.1: Simulation 1, setting 1: True function is the solid black line, $\hat{f}_{mix}$ is the red dashed line, and $\hat{f}_{nor}$ is blue dotted line.

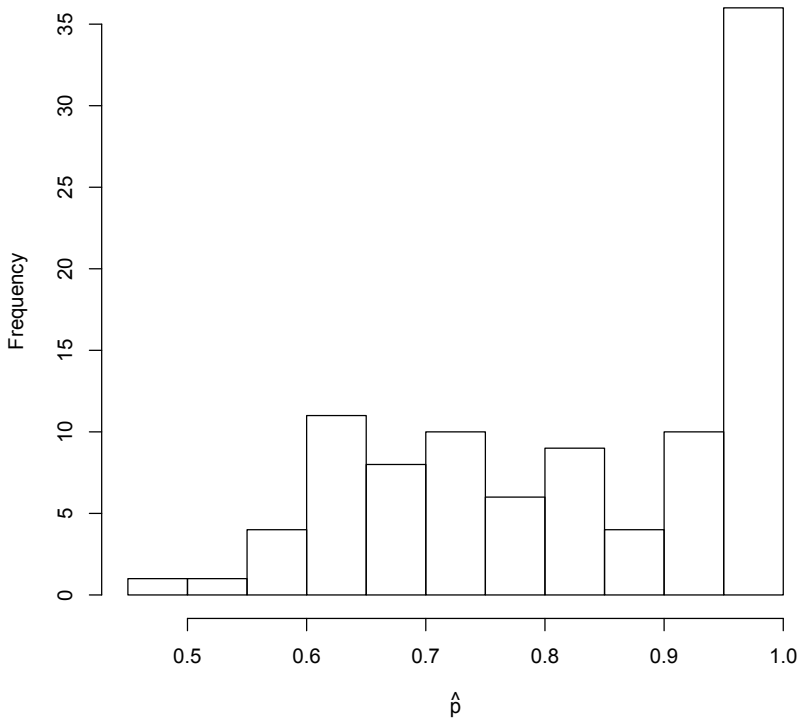

Figure 3.2: Simulation 1, setting 1: Distribution of $\hat{p}_k$, $k = 1, \ldots, 100$.

Setting 2: $n = 100$, $p = .9$

For the second setting, $h(\varepsilon_i) = .9s(\varepsilon_i) + .1g(\varepsilon_i)$, where $s(\varepsilon_i) \sim t_4(0, 1)$ and $g(\varepsilon_i) \sim N(0, 1)$. The x_i's, $i = 1, \ldots, 100$, are equidistant in the range $[0, 1]$. Figure 3.6, the histogram of the $\hat{p}_k$, $k = 1, \ldots, 100$, shows that over 80% of the estimates fall between .8 and 1, with a mean of .91.

Figure 3.7 shows side by side boxplots of the MSE for each approach, MSE($\hat{f}_{mix}$) is on the left. This may indicate that when working with a smaller dataset, the normal error term performs better, since MSE($\hat{f}_{mix}$) is larger on average than MSE($\hat{f}_{nor}$). Table 3.2 contains the numbers of times the true function falls within each credible interval at 10 points chosen equidistantly throughout the range of x. The model with the normal error

term distribution appears to perform better than the model with the mixture error term.

Table 3.2: Simulation 1, setting 2: Pointwise Credible Intervals Comparison

Model	x_1	x_{12}	x_{23}	x_{34}	x_{45}	x_{56}	x_{67}	x_{78}	x_{89}	x_{100}
Mix	73	72	92	73	81	88	78	93	76	72
Nor	92	98	95	100	99	89	98	99	100	98

Setting 3: $n = 500$, $p = .1$

For the third setting, $h(\varepsilon_i) = .1s(\varepsilon_i) + .9g(\varepsilon_i)$, where $s(\varepsilon_i) \sim t_4(0, 1)$ and $g(\varepsilon_i) \sim N(0, 1)$. The x_i's, $i = 1, \ldots, 500$, are equidistant in the range $[0, 1]$. Figure 4.8 is the histogram of

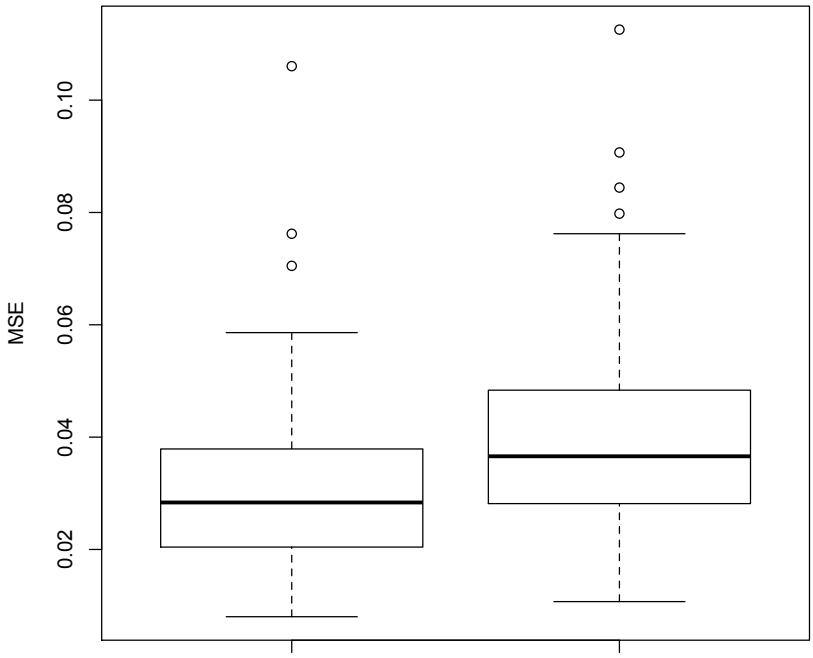

Figure 3.3: Simulation 1, setting 1: Left is MSE($\hat{f}_{mix}$). Right is MSE($\hat{f}_{nor}$).

$\hat{p}_k$, $k = 1, \ldots, 100$, with mean .22.

Figure 3.9 shows side by side boxplots of the MSE for each approach, MSE($\hat{f}_{mix}$) is on the left. MSE($\hat{f}_{mix}$) appears almost equal to MSE($\hat{f}_{nor}$). Table 3.3 contains the numbers of times the true function falls within each credible interval at 10 points chosen equidistantly throughout the range of x. The coverage probabilities are better for the mixture approach.

Setting 4: $n = 100$, $p = .1$

For the fourth setting, $h(\varepsilon_i) = .1s(\varepsilon_i) + .9g(\varepsilon_i)$, where $s(\varepsilon_i) \sim t_4(0, 1)$ and $g(\varepsilon_i) \sim N(0, 1)$. The x_i's, $i = 1, \ldots, 100$, are equidistant in the range $[0, 1]$. Figure 3.10 shows one realization

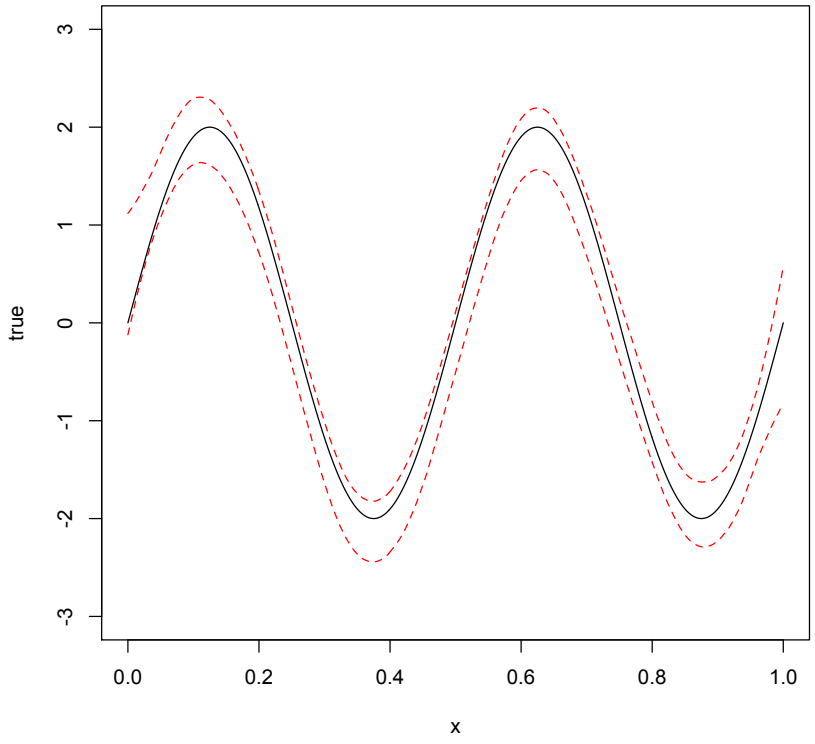

Figure 3.4: Simulation 1, setting 1: The red dashed lines represent 95% pointwise credible intervals for the approach assuming a mixture error term, and the solid black line is the true function.

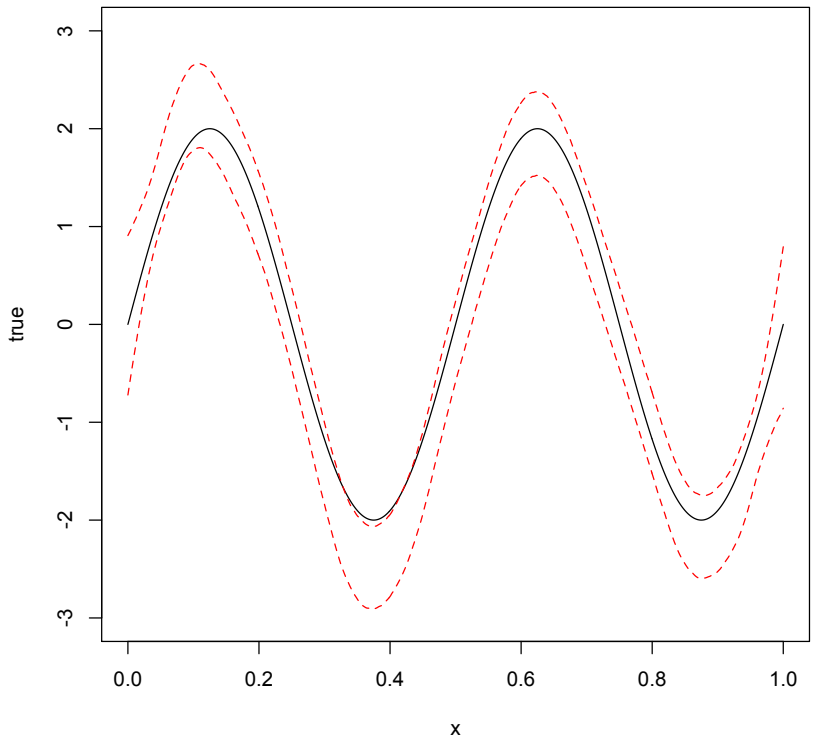

Figure 3.5: Simulation 1, setting 1: The red dashed lines represent 95% pointwise credible intervals for the approach assuming a normal error term, and the solid black line is the true function.

Table 3.3: Simulation 1, setting 3: Pointwise Credible Intervals Comparison

Model	x_1	x_{56}	x_{112}	x_{167}	x_{223}	x_{278}	x_{334}	x_{389}	x_{445}	x_{500}
Mix	91	89	94	93	96	93	94	97	93	93
Nor	92	85	93	96	92	92	91	94	90	60

with the true function (black solid line), $\hat{f}_{mix}(x)$ (red dashed line) and $\hat{f}_{nor}(x)$ (blue dotted line). Notice that in this plot $\hat{f}_{nor}$ is better than $\hat{f}_{mix}$. Figure 3.11 is a histogram of $\hat{p}_k$,

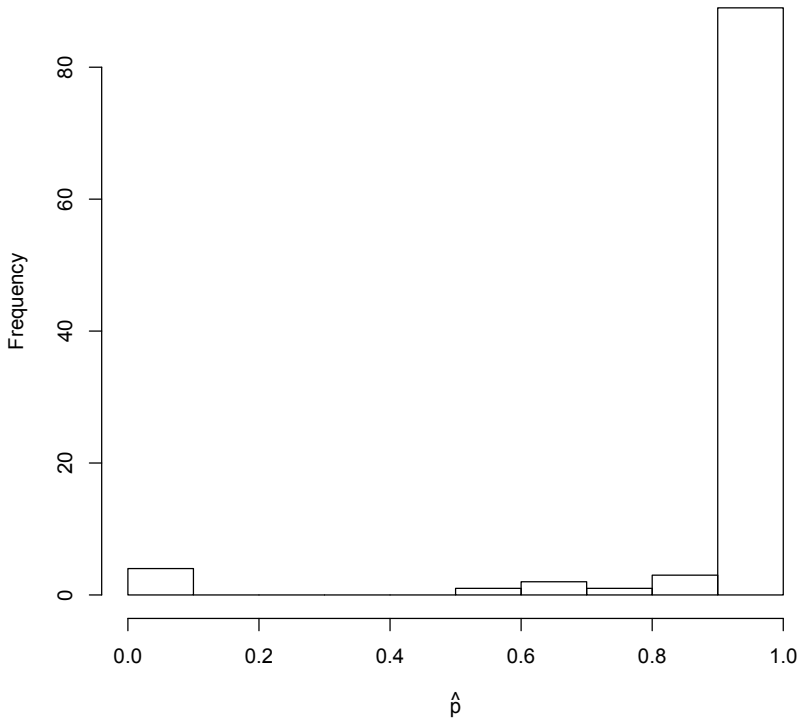

Figure 3.6: Simulation 1, setting 2: Distribution of $\hat{p}_k$, $k = 1, \ldots, 100$.

$k = 1, \ldots, 100$, with mean .694, which shows that $\hat{p}_k$ was estimated high.

Figure 3.12 shows side by side boxplots of the MSE for each approach, MSE($\hat{f}_{mix}$) is on the left. The normal error approach is performing well, MSE($\hat{f}_{mix}$) is larger than MSE($\hat{f}_{nor}$). Table 3.4 contains the numbers of times the true function falls within each credible interval at 10 points chosen equidistantly throughout the range of x. The normal error term approach appears to perform better.

Table 3.4: Simulation 1, setting 4: Pointwise Credible Intervals Comparison

Model	x_1	x_{12}	x_{23}	x_{34}	x_{45}	x_{56}	x_{67}	x_{78}	x_{89}	x_{100}
Mix	79	78	95	77	89	86	80	90	82	79
Nor	97	40	99	91	94	75	96	95	98	64

3.2 Simulation Study 2

The true regression function we use in the first simulation study is

$$f(x) = 2^x.$$

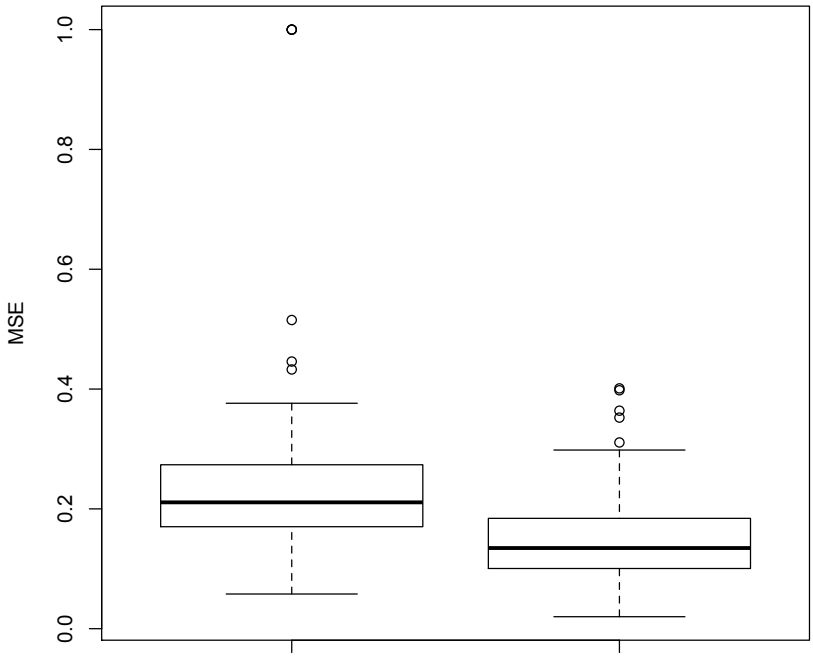

Figure 3.7: Simulation 1, setting 2: Left is MSE($\hat{f}_{mix}$). Right is MSE($\hat{f}_{nor}$).

Setting 1: $n = 500$, $p = .9$

For the first setting, $h(\varepsilon_i) = .9s(\varepsilon_i) + .1g(\varepsilon_i)$, where $s(\varepsilon_i) \sim t_4(0, 1)$ and $g(\varepsilon_i) \sim N(0, 1)$. The x_i's, $i = 1, \ldots, 500$, are equidistant in the range $[0, 5]$. Figure 3.13 shows one realization with the true function (black solid line), $\hat{f}_{mix}(x)$ (red dashed line) and $\hat{f}_{nor}(x)$ (blue dotted line). Figure 3.14 shows a histogram of $\hat{p}_k$, $k = 1, \ldots, 100$. Here it can be seen that all the estimates are above .5, the overall mean of the $\hat{p}_k$, $k = 1, \ldots, 100$, is .81.

Figure 3.15 shows side by side boxplots of the MSE for each approach, MSE($\hat{f}_{mix}$) is on the left. This further supports our model as the more appropriate choice, since MSE($\hat{f}_{mix}$) is smaller on average than MSE($\hat{f}_{nor}$). To further compare the two models, we look at 95% pointwise credible intervals based on each approach. Examples of pointwise credible

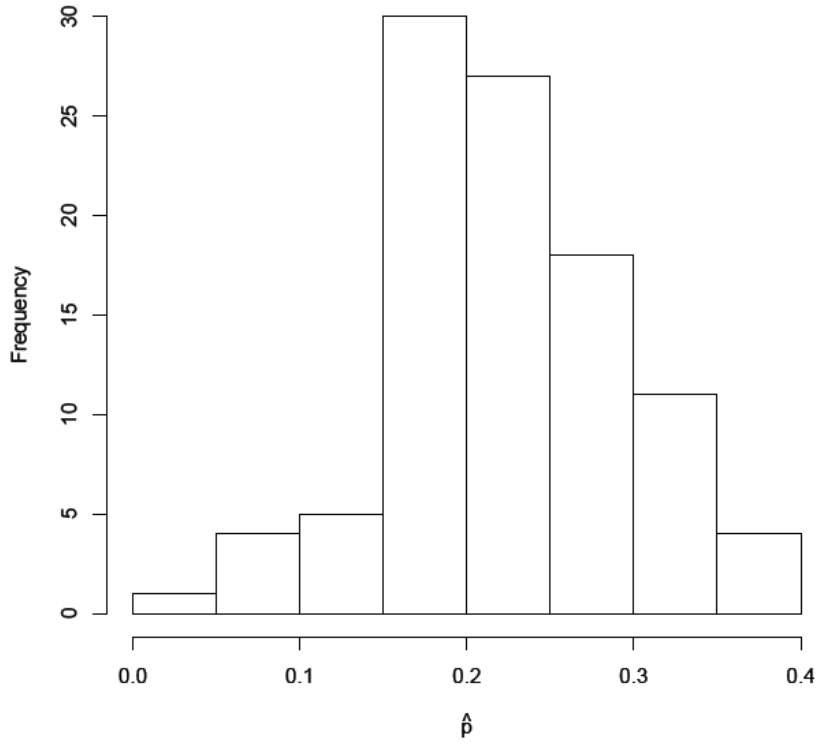

Figure 3.8: Simulation 1, setting 3: Distribution of $\hat{p}_k$, $k = 1, \ldots, 100$.

intervals based on a single realization are presented in Figure 3.16 and Figure 3.17 for the mixture and normal cases, respectively. Table 3.5 contains the numbers of times the true function falls within each credible interval at 10 points chosen equidistantly throughout the range of x. Our approach performs better.

Table 3.5: Simulation 2, setting 1: Pointwise Credible Intervals Comparison

Model	x_1	x_{56}	x_{112}	x_{167}	x_{223}	x_{278}	x_{334}	x_{389}	x_{445}	x_{500}
Mix	96	95	95	98	99	100	99	98	97	93
Nor	95	90	94	98	99	98	97	98	96	92

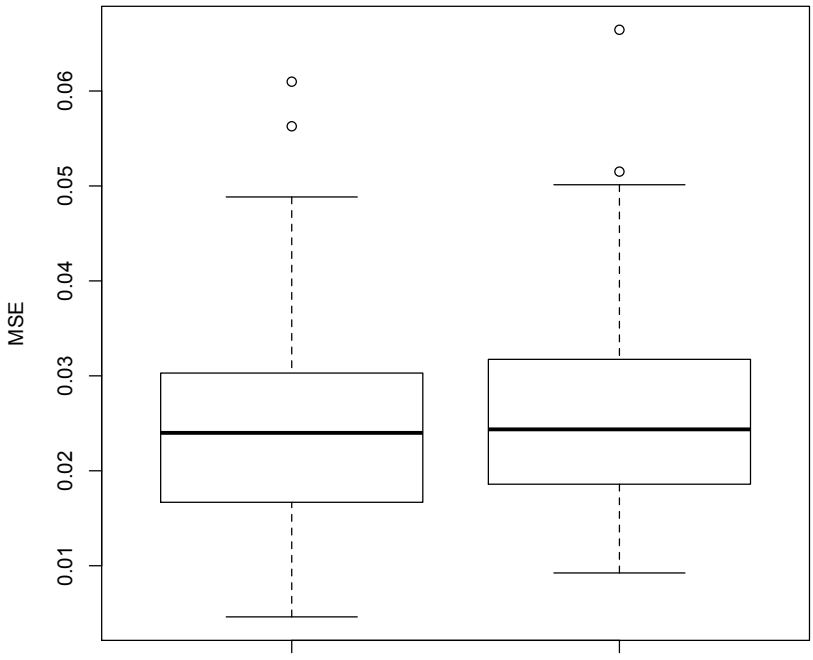

Figure 3.9: Simulation 1, setting 3: Left is MSE($\hat{f}_{mix}$). Right is MSE($\hat{f}_{nor}$).

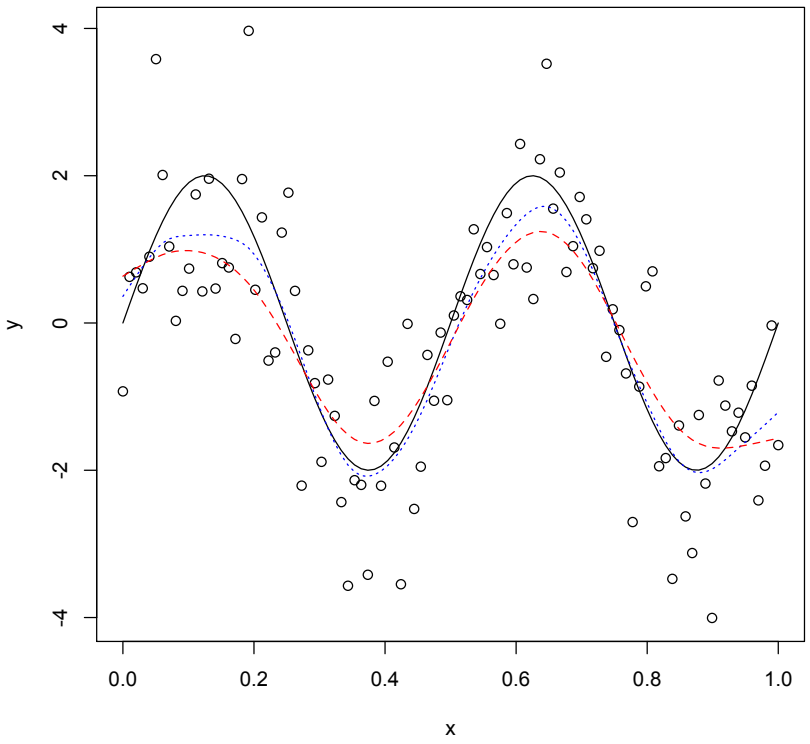

Figure 3.10: Simulation 1, setting 4: True function is the solid black line, $\hat{f}_{mix}$ is the red dashed line, and $\hat{f}_{nor}$ is blue dotted line.

Setting 2: $n = 100$, $p = .9$

For the second setting, $h(\varepsilon_i) = .9s(\varepsilon_i) + .1g(\varepsilon_i)$, where $s(\varepsilon_i) \sim t_4(0, 1)$ and $g(\varepsilon_i) \sim N(0, 1)$. The x_i's, $i = 1, \ldots, 100$, are equidistant in the range $[0, 5]$. Figure 3.18 shows a histogram of $\hat{p}_k$, $k = 1, \ldots, 100$, with a mean of .95.

Figure 3.19 shows side by side boxplots of the MSE for each approach, MSE($\hat{f}_{mix}$) is on the left. MSE($\hat{f}_{mix}$) is smaller on average than MSE($\hat{f}_{nor}$). Table 3.6 contains the numbers of times the true function falls within each credible interval at 10 points chosen equidistantly throughout the range of x. Our approach appears to perform better than the approach using the normal error term.

Table 3.6: Simulation 2, setting 2: Pointwise Credible Intervals Comparison

Model	x_1	x_{12}	x_{23}	x_{34}	x_{45}	x_{56}	x_{67}	x_{78}	x_{89}	x_{100}
Mix	96	95	99	99	98	98	96	97	94	94
Nor	91	98	99	100	100	100	96	96	98	92

Setting 3: $n = 500$, $p = .1$

For the third setting, $h(\varepsilon_i) = .1s(\varepsilon_i) + .9g(\varepsilon_i)$, where $s(\varepsilon_i) \sim t_4(0, 1)$ and $g(\varepsilon_i) \sim N(0, 1)$. The x_i's, $i = 1, \ldots, 500$, are equidistant in the range $[0, 5]$. Figure 3.20 is the histogram of $\hat{p}_k$, $k = 1, \ldots, 100$, with mean .22.

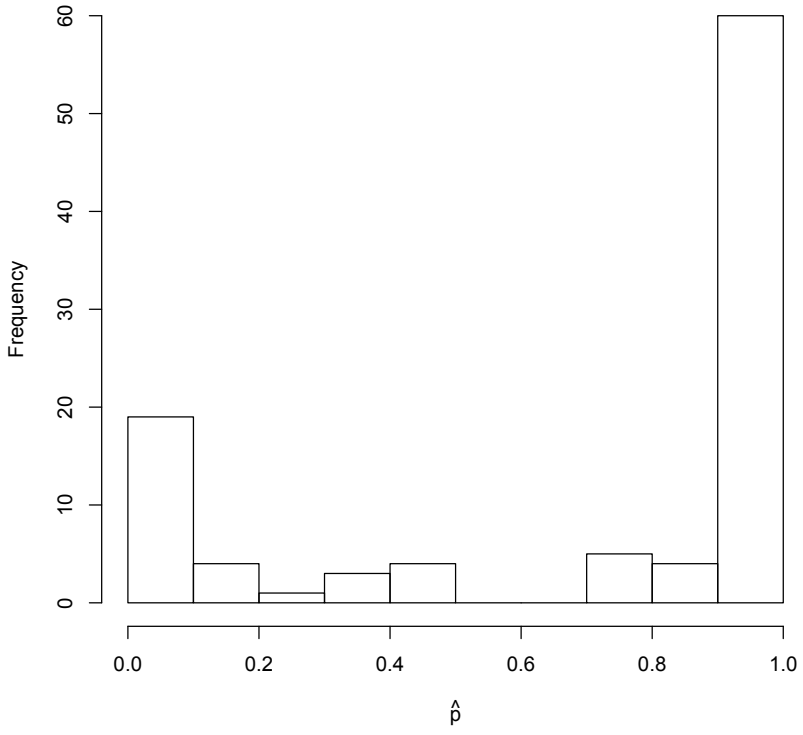

Figure 3.11: Simulation 1, setting 4: Distribution of $\hat{p}_k$, $k = 1, \ldots, 100$.

Figure 3.21 shows side by side boxplots of the MSE for each approach, MSE($\hat{f}_{mix}$) is on the left. MSE($\hat{f}_{mix}$) appears almost equal to MSE($\hat{f}_{nor}$). Table 3.7 contains the numbers of times the true function falls within each credible interval at 10 points chosen equidistantly throughout the range of x. The coverage probabilities are better for the mixture approach.

Table 3.7: Simulation 2, setting 3: Pointwise Credible Intervals Comparison

Model	x_1	x_{56}	x_{112}	x_{167}	x_{223}	x_{278}	x_{334}	x_{389}	x_{445}	x_{500}
Mix	95	93	98	98	96	97	96	98	97	93
Nor	96	79	83	86	97	97	94	87	96	92

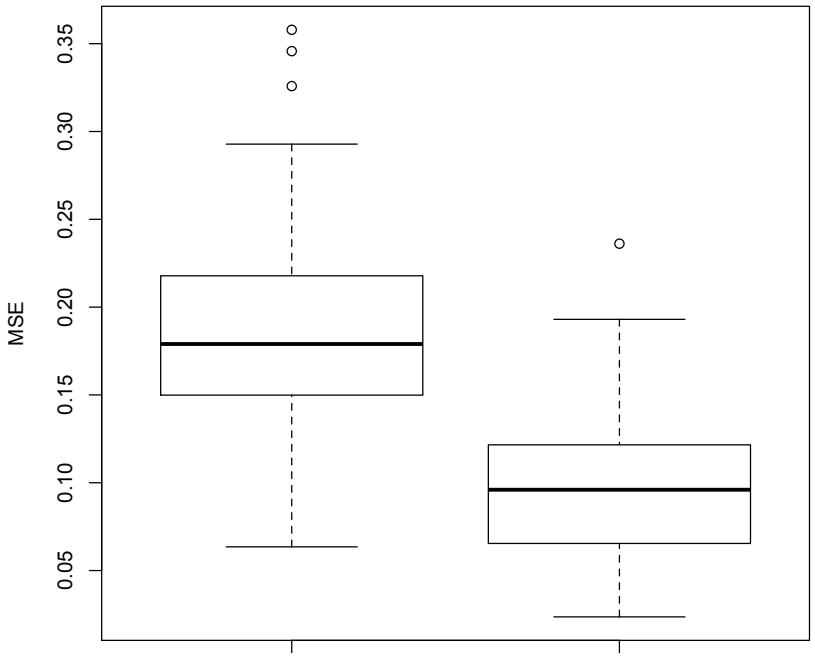

Figure 3.12: Simulation 1, setting 4: Left is MSE($\hat{f}_{mix}$). Right is MSE($\hat{f}_{nor}$).

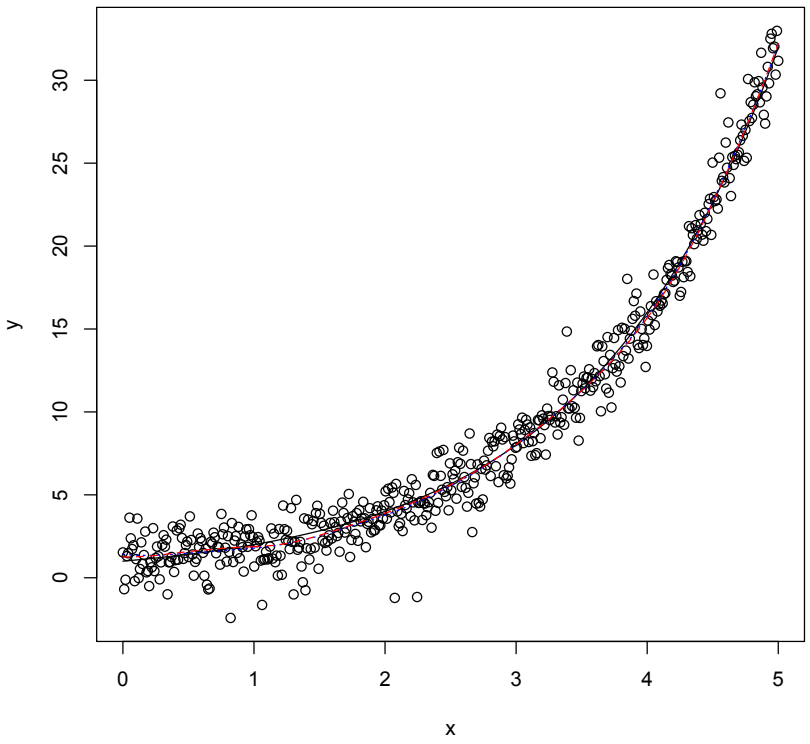

Figure 3.13: Simulation 2, setting 1: True function is the solid black line, $\hat{f}_{mix}$ is the red dashed line, and $\hat{f}_{nor}$ is blue dotted line.

Setting 4: $n = 100$, $p = .1$

For the fourth setting, $h(\varepsilon_i) = .1s(\varepsilon_i) + .9g(\varepsilon_i)$, where $s(\varepsilon_i) \sim t_4(0, 1)$ and $g(\varepsilon_i) \sim N(0, 1)$. The x_i's, $i = 1, \ldots, 100$, are equidistant in the range $[0, 5]$. Figure 3.22 is a histogram of $\hat{p}_k$, $k = 1, \ldots, 100$, with mean .552, much higher than .1.

Figure 3.23 shows side by side boxplots of the MSE for each approach, MSE($\hat{f}_{mix}$) is on the left. The MSEs for both approaches are almost the same. Table 3.8 contains the numbers of times the true function falls within each credible interval at 10 points chosen equidistantly throughout the range of x. Our approach gives better coverage probabilities than the ones corresponding to the model with the normal error term.

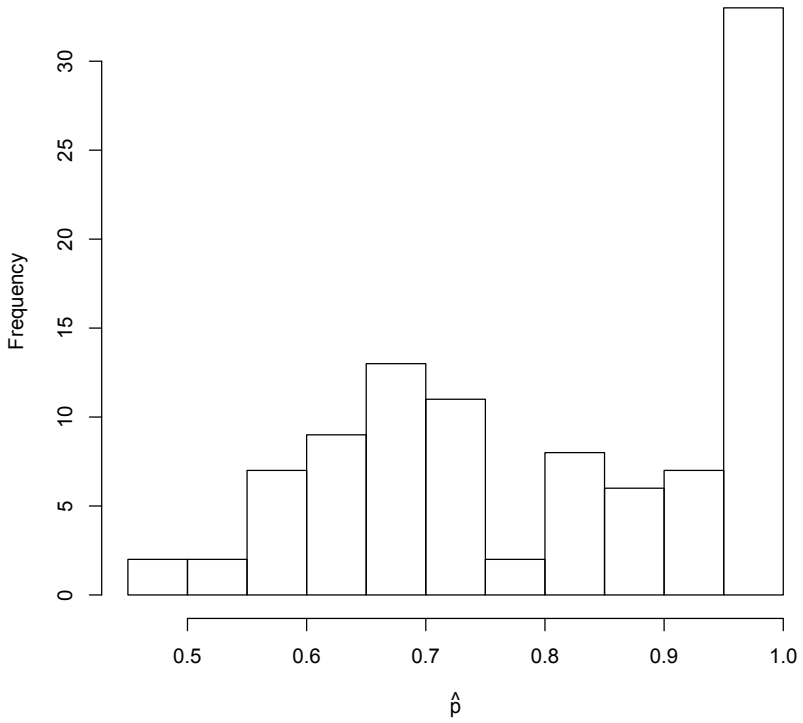

Figure 3.14: Simulation 2, setting 1: Distribution of $\hat{p}_k$, $k = 1, \ldots, 100$.

Table 3.8: Simulation 2, setting 4: Pointwise Credible Intervals Comparison

Model	x_1	x_{12}	x_{23}	x_{34}	x_{45}	x_{56}	x_{67}	x_{78}	x_{89}	x_{100}
Mix	96	97	98	100	98	97	98	100	97	93
Nor	88	94	96	98	96	82	98	89	61	87

3.3 Simulation Summary

In conclusion, the mixture approach is preferred in the majority of the settings. It performs better overall with the second simulation than the first simulation. When n is large and $p = .9$, in both simulations the average mixing probability estimate is around .81. When

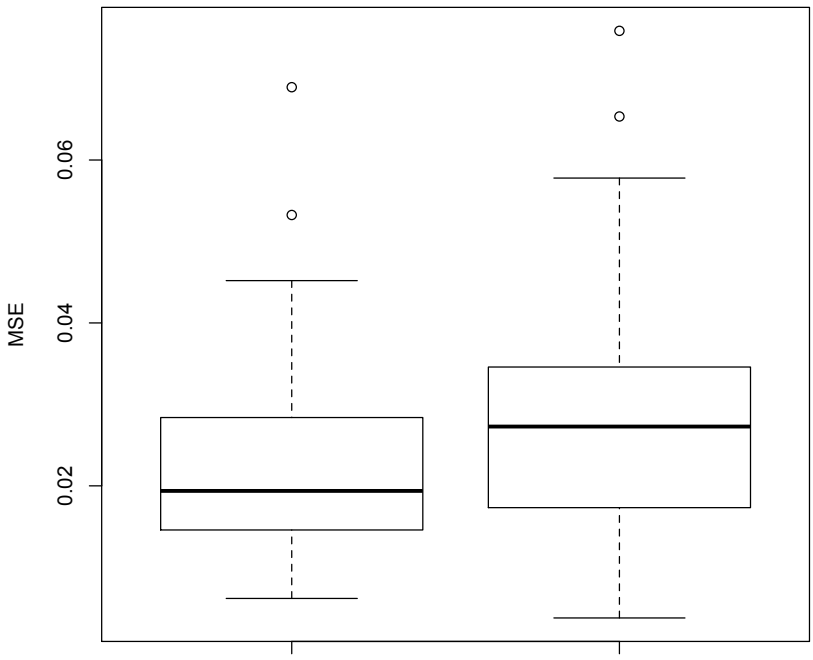

Figure 3.15: Simulation 2, setting 1: Left is MSE($\hat{f}_{mix}$). Right is MSE($\hat{f}_{nor}$).

n is smaller and $p = .9$, the estimates of p are around .9, although the two shapes led to different outcomes. In the first simulation, the normal error term approach is favored, and in the second simulation, the mixture error term is favored. When n is small and $p = .1$, the simulation results are similar to the ones from the previous setting, although in this case the estimates of p are not close to the true value for either simulation. When n is large and $p = .1$, the estimates of p are only slightly higher than the true value, and are identical for both simulations. Both approaches seem to perform equally well, although for the mixture approach, the coverage probabilities are better. When n is small, the choice of approach may rely heavily on the shape of the data. In the case of large n, the mixture approach is preferred.

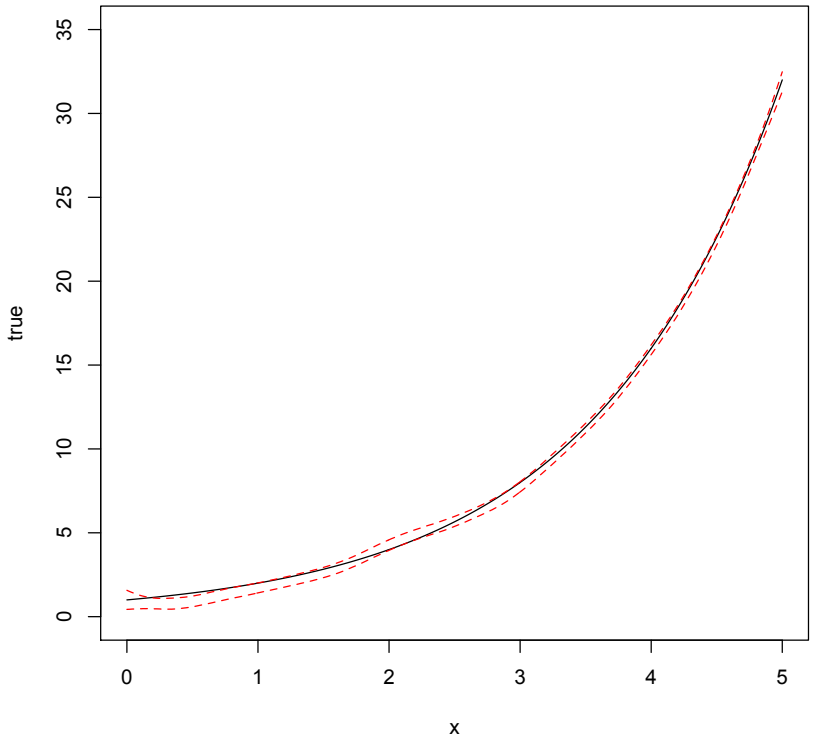

Figure 3.16: Simulation 2, setting 1: The red dashed lines represent 95% pointwise credible intervals for the approach assuming a mixture error term, and the solid black line is the true function.

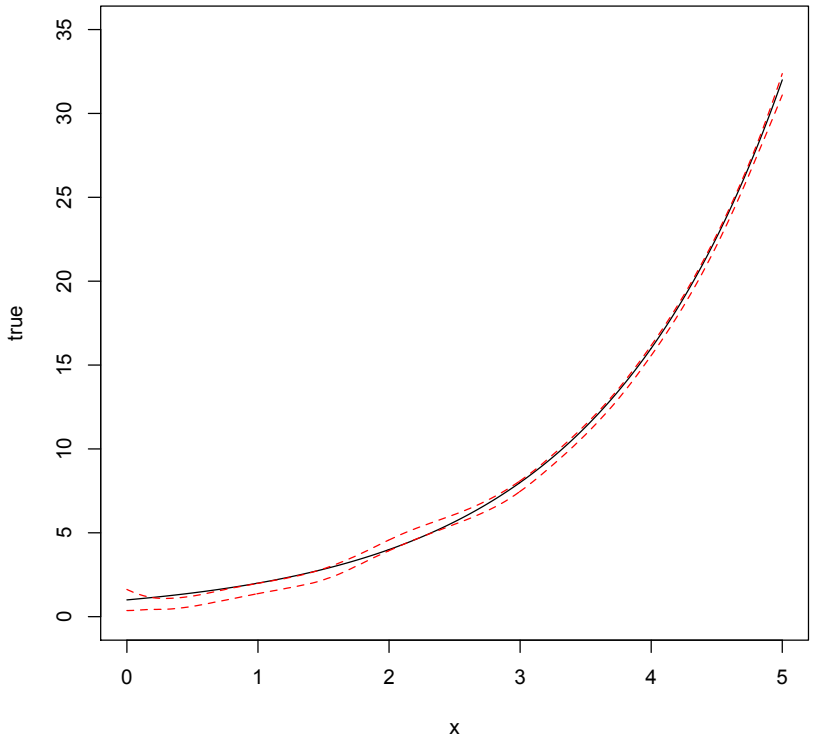

Figure 3.17: Simulation 2, setting 1: The red dashed lines represent 95% pointwise credible intervals for the approach assuming a normal error term, and the solid black line is the true function.

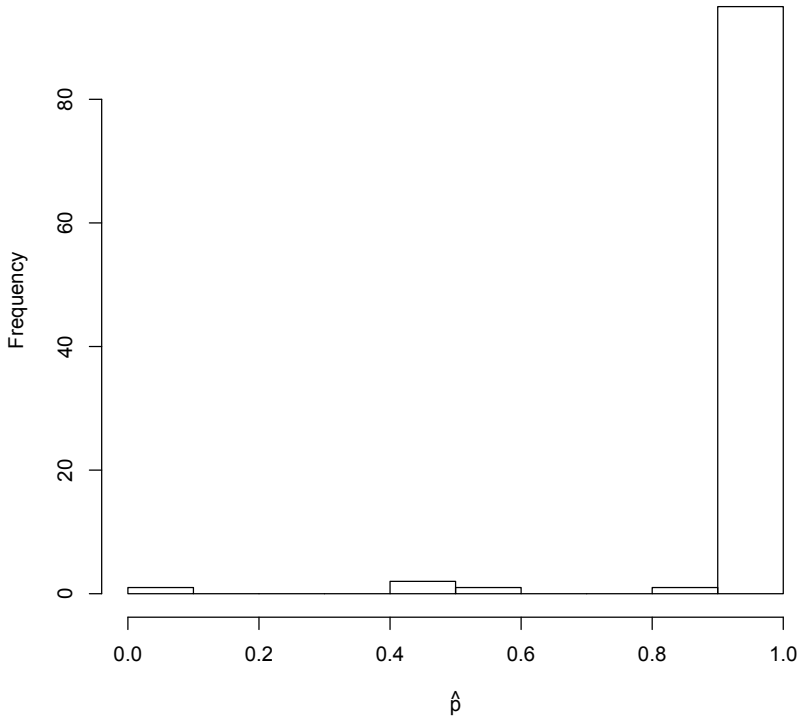

Figure 3.18: Simulation 2, setting 2: Distribution of $\hat{p}_k$, $k = 1, \ldots, 100$.

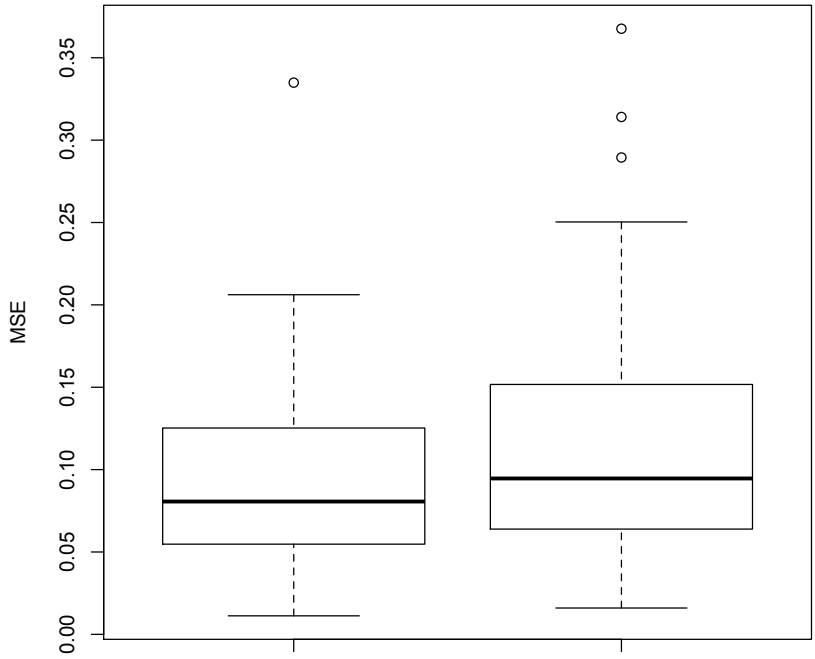

Figure 3.19: Simulation 2, setting 2: Left is MSE($\hat{f}_{mix}$). Right is MSE($\hat{f}_{nor}$).

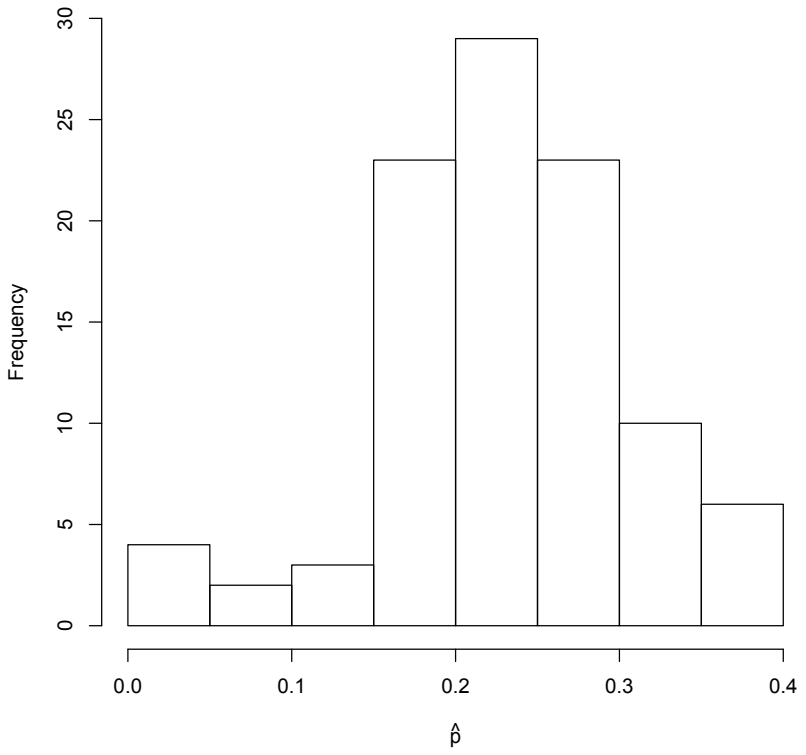

Figure 3.20: Simulation 2, setting 3: Distribution of $\hat{p}_k$, $k = 1, \ldots, 100$.

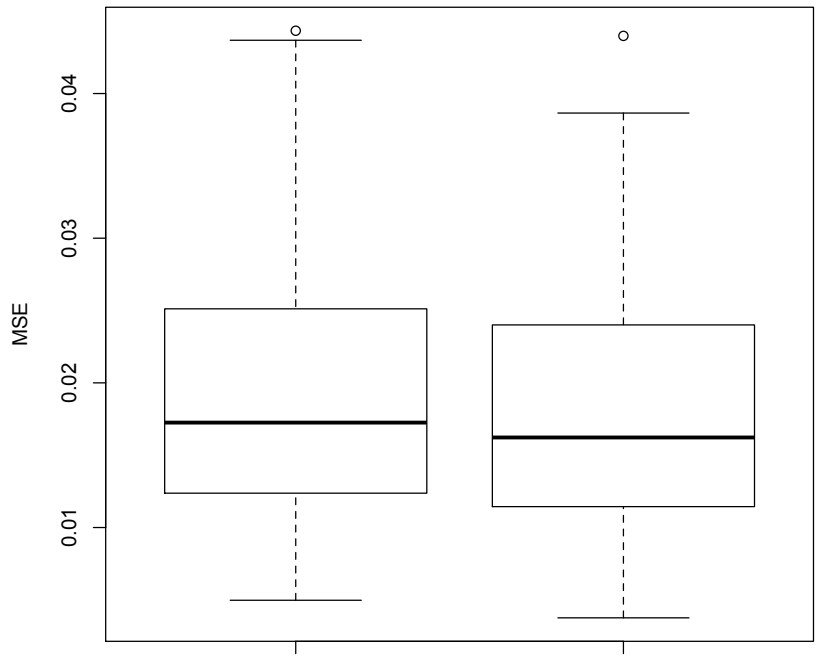

Figure 3.21: Simulation 2, setting 3: Left is MSE($\hat{f}_{mix}$). Right is MSE($\hat{f}_{nor}$).

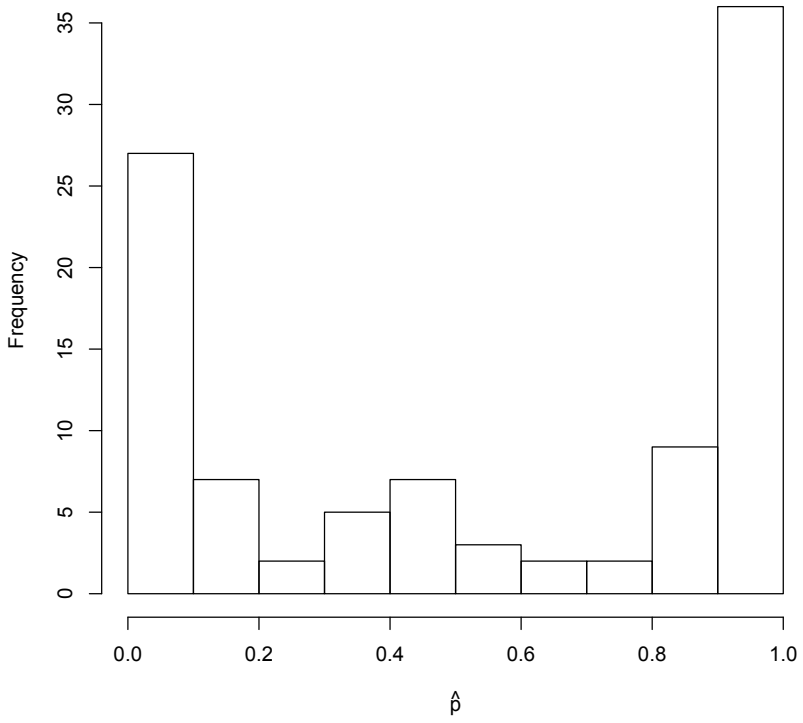

Figure 3.22: Simulation 2, setting 4: Distribution of $\hat{p}_k$, $k = 1, \ldots, 100$.

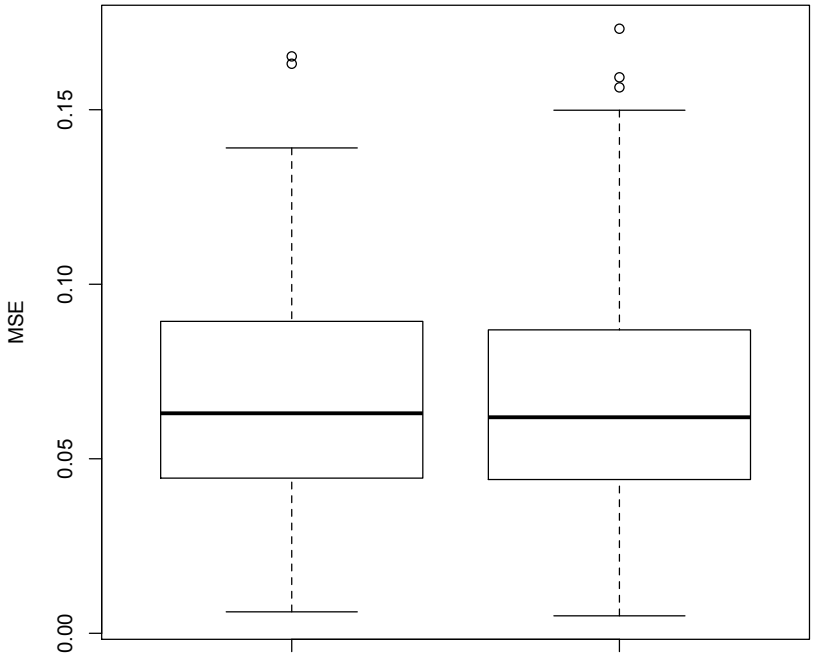

Figure 3.23: Simulation 2, setting 4: Left is MSE($\hat{f}_{mix}$). Right is MSE($\hat{f}_{nor}$).

Chapter 4

Data Analysis

In this chapter we apply our approach to two datasets. We fit our model to each dataset using a total of 10000 iterations.

4.1 1979 Education Spending Data

The first data set consists of per capita income and per capita spending in public schools by state in the United States in 1979. In order to analyze this dataset, the variables were standardized. This dataset was previously analyzed by Greene (1997), Cribari-Neto et al. (2000) and Fonseca et al. (2008). Similar to the previous analyses, we take per capita spending as the dependent variable, and per capita income as the explanatory variable.

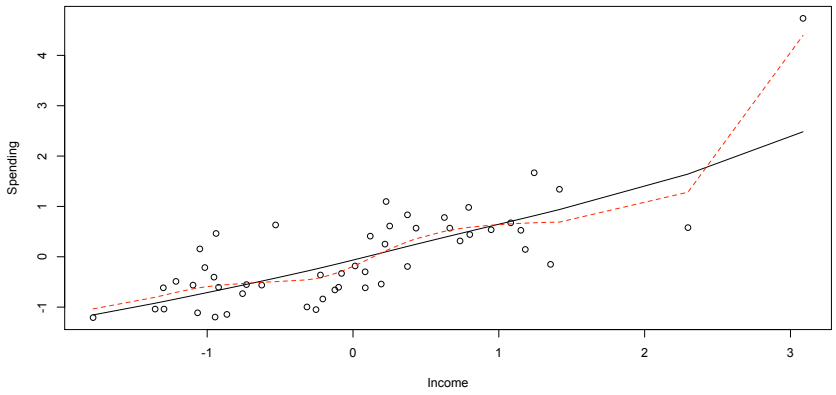

Figure 4.1: The solid line is the fit based on the mixture model. The dashed line is the fit when the error term is normally distributed.

Figure 4.1 shows the dependent variable against the explanatory variable, as well as the fit with our proposed model and the fit when the error term is normally distributed. Here it is evident that there is an outlying observation that corresponds to Alaska. Fonseca et al. (2008) stated that when the error term is normally distributed, and standard analyses are performed, a quadratic or a linear model is selected as the best model depending on the inclusion or exclusion of Alaska, respectively. They came to the conclusion that the linear model with a Student t error term is preferred when using their approach.

Our model fit, shown as the solid black line, agrees with the conclusion that was made by Fonseca et al. (2008) and is more robust than the model fit with the normal error term. The estimated mixing probability for this dataset is .95.

4.2 Math Proficiency

The second dataset is from The Educational Testing Service study *America's Smallest School: The Family*. In this study the relation between educational achievement of students to their home environment is investigated. The data were obtained from the 1990 national Assessment of Educational Progress for 37 states, the District of Columbia, the Virgin Islands and Guam. We particularly look at the percentage of eighth grade students with three or more types of reading materials at home, such as books, encyclopedias, magazines and newspapers as the explanatory variable, and the average mathematics proficiency as the dependent variable. This dataset is also analyzed in Kutner et al. (2003).

Figure 4.2 shows the dependent variable against the explanatory variable, as well as the two model fits. There appears to be three distinct outliers, the District of Columbia and the Virgin Islands are outliers with respect to the explanatory variable, where Guam appears to be an outlier with respect to both the explanatory and the dependent variables. Although our model does appear to be slightly influenced by the outliers, it appears to have a more

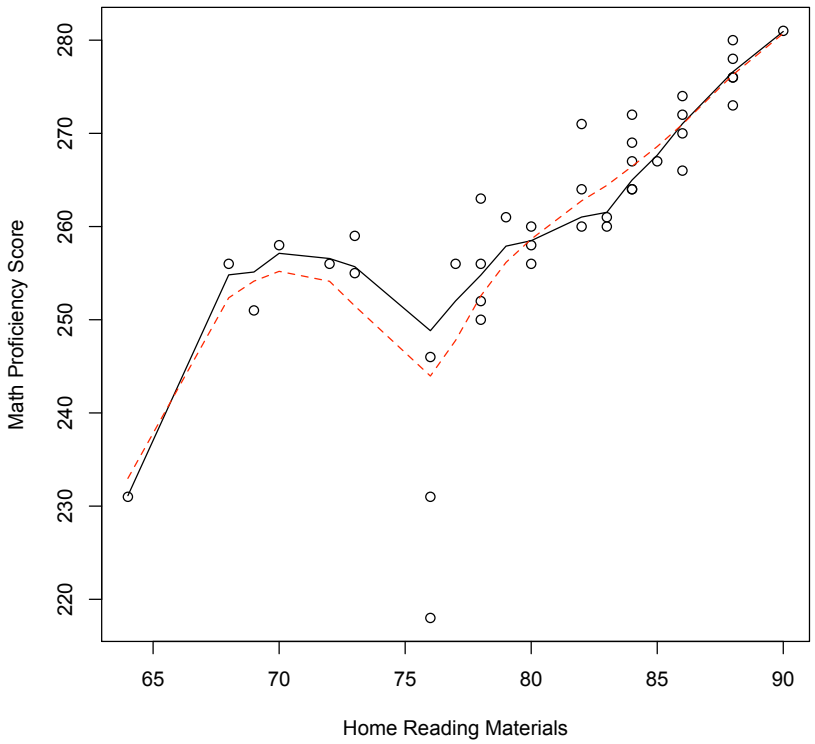

Figure 4.2: The solid line is the fit when the error term is a mixture. The dashed line is the fit when the error term is normally distributed.

robust fit than that of the normal error term model. The estimated mixing probability for this dataset was .93.

References

[1] Banks, D.L., Olszewski, R.T. and Maxion, R.A. (2003). Comparing Methods for Multivariate Nonparametric Regression, *Communications in Statistics - SImulation and Computation*, **32:2**, 541-571.

[2] Barros, M., Paula, G.A. and Leiva, V. (2008). A New Class of Survival Regression Models with Heavy-tailed Errors: Robustness and Diagnostics, *Lifetime Data Anal*, **14**, 316-332.

[3] Cleveland, W.S. (1979). Robust Locally Weighted Regression and Smoothing Scatterplots, *Journal of the American Statistical Association*, **74**, 829-836.

[4] Davison, A.C. (2003). *Statistical Models*, Cambridge University Press.

[5] Faria, S. and Melfi, G. (2004). LAD regression and Nonparametric Methods for Detecting Outliers and Leverage Points, *Student*, **5**, 256-272.

[6] Fonseca, T.C.O., Ferreira, M.A.R. and Migon, H.S. (2008). Objective Bayesian Analysis for the Student-t Regression Model, *Biometrika*, **95**, 2, 325-333.

[7] Greene, W.H. (1997). *Econometric Analysis*, Prentice-Hall, Upper Saddle, New Jersey

[8] Gelman, A., Carlin, A., Stern, H.S. and Rubin, D.B. (2004). *Bayesian Data Analysis*, Chapman & Hall/CRC, Boca Raton.

[9] Hastie, T., Tibshirani, R. and Friedman, J. (2001). *The Elements of Statistical Learning*, Springer.

[10] Henderson, C. (1950). Estimation of Genetic Parameters, *The Annals Of Mathematical Statistics*, **21**, 309-310.

[11] Huber, P. (1964). Robust Estimation of a Location Parameter, *The Annals Of Mathematical Statistics*, **35**, 73-101.

[12] Kutner, M., Nachtsheim, C.J. and Neter, J. (2004). *Applied Linear Regression Models*, McGraw-Hill Irwin, Boston.

[13] Rosen, O. and Thompson, W. (2009). A Bayesian Regression Model for Multivariate Functional Data , *Journal of Computational Statistics and Data Analysis*, **53**, 3773-3786.

[14] Rupert, D., Wand, M.P. and Carroll, R.J. (2003). *Semiparametric Regression*, Cambridge University Press, Cambridge.

[15] Stigler, S.M. (1984). Simon Newcomb, Percy Daniell, and the History of Robust Estimation 1885-1920, *Journal of the American Statistical Association*, **68**, 344, 872-879.

[16] Schwarz, G. (1978). Estimating the Dimension of a Model, *The Annals of Statistics*, **6**, 461-464.

[17] Thompson, W. and Rosen, O. (2008). A Bayesian Model for Sparse Functional Data. *Biometrics*, **64**, 54-63.

[18] West, M. (1984). Outlier Models and Prior Distributions in Bayesian Linear Regression, *Journal of the Royal Statistical Society, Series B*, **46**, 431-439.

Appendix A

Sampling Descriptions

A.1 Truncated Inverse Gamma

Let F be the cumulative distribution function (cdf) of the inverse gamma distribution, τ^2 is a variance component and c is an upper limit. We want to sample from $F_{\tau^2}(t \mid \tau^2 < c) = p(\tau^2 \leq t \mid \tau^2 < c)$.

$$p(\tau^2 \leq t \mid \tau^2 < c) = \frac{p(\tau^2 \leq t, \tau^2 < c)}{p(\tau^2 < c)} = \frac{p(\tau^2 \leq t)}{p(\tau^2 < c)} = \frac{F_{\tau^2}(t)}{1 - c_1} = u$$

Since the basic R package does not have the inverse gamma distribution, we have to write the above in terms of the gamma distribution.

$$p(\tau^2 < t) = p\left(\frac{1}{\tau^2} > \frac{1}{t}\right) = 1 - p\left(\frac{1}{\tau^2} \leq \frac{1}{t}\right) = 1 - G\left(\frac{1}{t}\right)$$

where G is the cdf of the gamma distribution. So $p(\tau^2 \leq t \mid \tau^2 \leq c) = \frac{1-G\left(\frac{1}{t}\right)}{1-c_1} = u$, from which we get $G\left(\frac{1}{t}\right) = 1 - u(1 - c_1)$ which leads to $t = \frac{1}{G^{-1}[1-u(1-c_1)]}$ as a variate from the truncated inverse gamma. Also,

$$p(\tau^2 < c) = F(c) = p\left(\frac{1}{\tau^2} > \frac{1}{c}\right) = 1 - p\left(\frac{1}{\tau^2} \leq \frac{1}{c}\right) = 1 - G\left(\frac{1}{c}\right) = 1 - c_1.$$

A.2 Truncated Gamma

Let G be the cdf of the gamma distribution and c be the upper limit. We want to sample $G_{\sigma^2}(t \mid \sigma^2 < c) = p(\sigma^2 \leq t \mid \sigma^2 < c)$.

$$p(\sigma^2 \leq t \mid \sigma^2 < c) = \frac{p(\sigma^2 \leq t, \sigma^2 < c)}{p(\sigma^2 < c)} = \frac{p(\sigma^2 \leq t)}{p(\sigma^2 < c)} = \frac{F_{\sigma^2}(t)}{c_1} = u,$$

then $t = F^{-1}(c_1 u)$ is a variate from the truncated gamma and

$$p(\sigma^2 < c) = G(c) = c_1.$$

Appendix B

Code

The programming language used was R,here you will find the R code of our approach. The code here is set up to support the simulation study, it can be easily modified to support a single dataset.

n< −length(x)

y_hat < − matrix(0,n,8001)

ci_max < − matrix(1,n,100)

ci_min < − matrix(1,n,100)

h_hat < − matrix(1,n,100)

sigmean < − rep(1,100)

pmean < − rep(1,100)

thetamean < − matrix(1,32,100)

MSE < − rep(1,100)

for (j in 1:100)

y < − h[,j] ♯ the data, for the simulation studies the data was stored in in a n by 100 matrix, called h.

T < − 10000 ♯ number of times update

N < − length(y) ♯ The length of the data

K < − 30 ♯ number of knots

q < − 2 ♯ number of parameters

ssqb < − 100 ♯large known value

nu < − 4

c_u < − 100 ♯large known value

```
c_ep <- 100 #large known constant
c_sig <- 100
alpha <- 3 # constant for the prior on p
ssq_ep <- rep(.01,T+1) # vector to hold the reps of the sigmasq epsilon
ssq_u <- rep(.1,T+1) # vector to hold all reps of sigmasq_u
ssq <- rep(.1,T+1)
v <- matrix(.01,N, T+1)
z <- matrix(0,N,2) # the z matrix
p <- rep(.5,T+1) # vector to hold the values of p
prob <- matrix(0, nrow = N, ncol = 2)
pf <- matrix(0, nrow = N, ncol = 2)
C <- matrix(0,n,K+q) #X and Z matrices bound together
X <- matrix(0,n,q) # covariate matrix
L <- matrix(0,n,K) # matrix of bases
D <- matrix(0,q+K,q+K)
P <- matrix(0,q+K,q+K)
theta <- matrix(0,K+q,T+1)
beta <- matrix(0,q,1)
u <- matrix(1,K,1)
X[,1] <- rep(1,n)
X[,2] <- x
theta[,1] <- rbind(beta,u)
knots <- seq(min(x),max(x),length=K+2)[-c(1,K+2)]
D <- diag(c(c(0,0),rep(1,K)))
P <- diag(c(c(1,1),rep(0,K)))
# radial basis functions
svd.Omega <- svd(abs(outer(knots, knots, "-"))^ 3)
matrix.sqrt.Omega <- t(svd.Omega$v % * % (t(svd.Omega$u)* sqrt(svd.Omega$d)))
```

```
L < − abs(outer(x, knots, ”-”)ˆ3)%∗%solve(matrix.sqrt.Omega)
C < − cbind(X,L)
for(t in 1:T){
pf[,2] < − (1-p[t])*dnorm(y- C%*%theta[,t], 0,sqrt(ssq_ep[t]))
for( i in 1:N){
pf[i,1] < − p[t]*dnorm(y[i]-C[i,]%*%theta[,t],0,sqrt(v[i,t]))
prob[i,2] < − pf[i,2]/(pf[i,2] + pf[i,1])
e< −runif(1)
ifelse(e ¡ prob[i,2], z[i,2] < − 1, z[i,2] < − 0)
ifelse(z[i,2]¿0, z[i,1] < − 0, z[i,1] < − 1)
}♯end for i
sum_one < − sum(z[,1])
sum_two < − sum(z[,2])
d_one < − diag(z[,1])
d_two < − diag(z[,2])
epshape < − (.5*(sum_two)-1)
w < − y-C%*%theta[,t]
w_1 < − t(w)%*%d_two%*%w
eprate < − .5*w_1
ifelse(epshape < 0, epshape < − .1, epshape < − epshape)
ifelse(eprate < 0, eprate < − .1, eprate < − eprate)
ut < − runif(1,0,1)
const1 < − pgamma(1/c_ep,shape=epshape, rate=eprate)
const2 < − 1-ut*(1-const1)
ssq_ep[t+1] < − 1/qgamma(const2, shape=epshape, rate=eprate)
p[t+1] < − rbeta(1, sum_one+alpha, sum_two + alpha) ushape < − (K/2)- 1
urate < − .5*t(u)%*%u
ut < − runif(1,0,1)
```

```
const1 < − pgamma(1/c_u,shape=ushape, rate=urate)
const2 < − 1-ut*(1-const1)
ssq_u[t+1] < − 1/qgamma(const2, shape=ushape, rate=urate)
for(i in 1:N){
v[i,t+1] < − 1/rgamma(1,shape= nu/2 + z[i,1]/2, rate = ((z[i,1]*(y[i]-C[i,]%∗%theta[,t])^(2))/2)
+ nu*ssq[t]/2)
}
z1d < − rep(n,0)
z2d < − rep(n,0)
vsum =0
for(i in 1 :N){
z1d[i] < − z[i,1]/v[i,t+1]
vsum < − vsum + 1/v[i,t+1]
}
zvsum < − sum(z1d)
ut < − runif(1,0,1)
const1 < − pgamma(c_sig,shape=((nu)/2)*sum_one + 1,rate= (nu/2) * zvsum)
ssq[t+1] < − 1/qgamma(const1*ut, shape=((nu)/2)*sum_one + 1,rate= (nu/2) * zvsum)
ifelse(ssq_ep[t+1] ¡ .001, z2d < − z[,2], z2d < − z[,2]/ssq_ep[t+1])
zsum < − diag(z2d)+ diag(z1d)
part1 < − (t(C)%*%zsum%*%C)
varcov < − solve(part1 + D/ssq_u[t+1] + P/ssqb)
mean < − varcov %*% (t(C)%*%zsum%*%y)
theta[,t+1] < − mvrnorm(1,mean,varcov)
if(t > 2000){
i= t-2000
y_hat[,i] < − C%*%theta[,t+1]
}
```

```
beta <- theta[1:q,t+1]
u <- theta[3:K+2,t+1]
}#end T
for(i in 1:n)
ci_min[i,j] <- quantile(y_hat[i,], .025)
ci_max[i,j] <- quantile(y_hat[i,], .975)
}
thetamean[,j] <- apply(theta[,2001:10001],1,mean)
sigmean[j] <- mean(ssq_ep[2001:10001])
pmean[j] <- mean(p[2001:10001])
h_hat[,j] <- C%*%thetamean[,j]
true=2*sin(4*pi*x)#2^x, depending on what the true function was this changed
MSE[j] = (1/length(x))*sum ((true-h_hat[,j])^2)
#lines(x,true,col="blue", lty = "dotted")
}#end j
```

Curriculum Vitae

Courtney Barnes was born in Fairfax, Virginia. The youngest child and only daughter of Brendan Nohilly and Patricia Nohilly, she graduated from Gar-Field Senior High School, Woodbridge, Virginia, in the spring of 2000. In the Spring of 2002 she entered El Paso Community College, El Paso, Texas with a full softball athletic scholarship. She entered The University of Texas at El Paso in the fall of 2004 to pursue a degree in mathematical sciences with a minor in computer science. She received her bachelor's in science degree in the spring of 2007, graduating cum laude. In the fall she entered the Graduate School at the University of Texas at El Paso.

Permanent address: 4616 Loma Del Rey Circle
El Paso, Texas 79934