TBASS: A Robust Adaptation of Bayesian Adaptive Spline Surfaces

Andy A. Shen, Kellin N. Rumsey, Devin C. Francom

Statistical Sciences Group (CCS-6): Los Alamos National Laboratory

Abstract

The R package TBASS is an extension of the BASS package created by Francom and Sansó (2019). The package is used to fit a Bayesian multivariate adaptive spline to a dataset that either follows a Student's t-distribution or has outliers. Much of the framework for TBASS is adapted from the concepts of Bayesian Multivariate Adaptive Regression Splines (BMARS), specifically the work done by Denison, Mallick, and Smith (1998). The spline function is fit using a Reversible-Jump Markov Chain Monte Carlo algorithm, followed by Gibbs Sampling. By including this more robust generalization, a dataset with outliers can accurately fit using the BMARS model, without the possibility of overfitting or variance inflation.

Keywords: splines, robust regression, Bayesian inference, nonparametric regression, sensitivity analysis

1 Introduction

Splines are a commonly used regression tool for fitting nonlinear data, both univariate and multivariate. Splines can act as basis functions, where each basis function combines to form the $\mathbf{X}$ matrix. The simplest way to create the ith basis functions can be represented as $\prod_{i=1}^{n}$

$$\prod \mathbf{B}_{ij} = [s_i(x_j - t_i)]_+ \tag{1}$$

Equation (1) is used to calculate the ijth element of the $\mathbf{B}$ matrix of basis functions, where $s_i \in \{-1, 1\}$, t_i is called a **knot** and $[a]_+ = \max(0, a)$.

For example, given the nonlinear data shown in Figure 1 below, we can use (1) to fit a spline model shown in Figure 2.

Figure 1 : Univariate Nonlinear Data

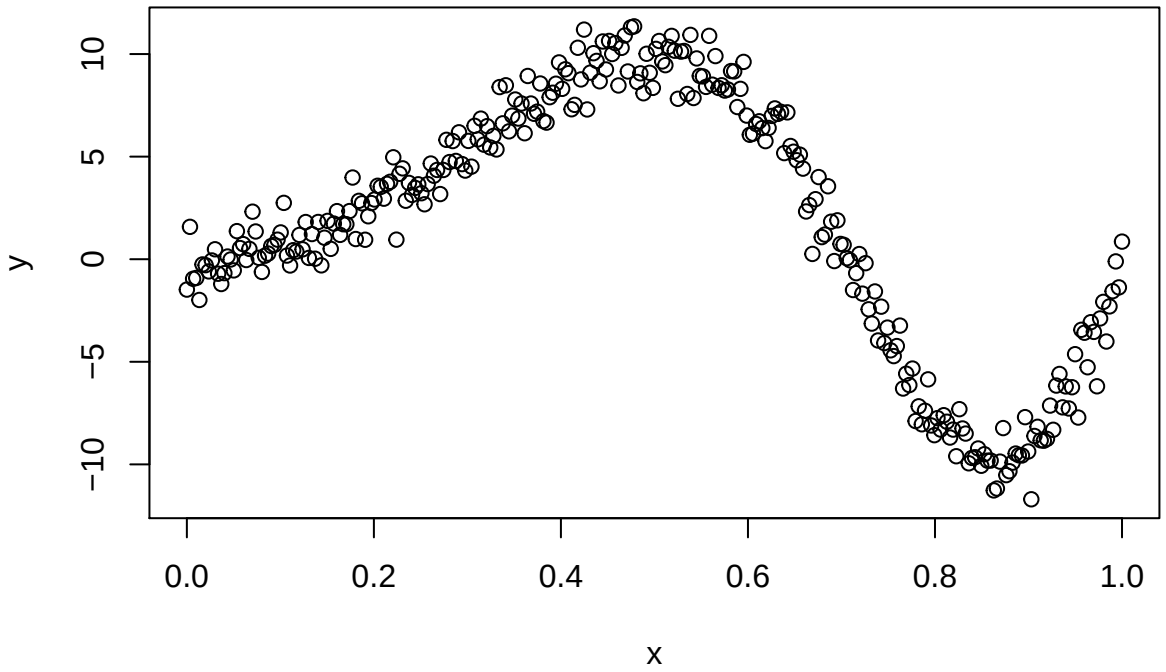


Figure 2 : Univariate Spline Function

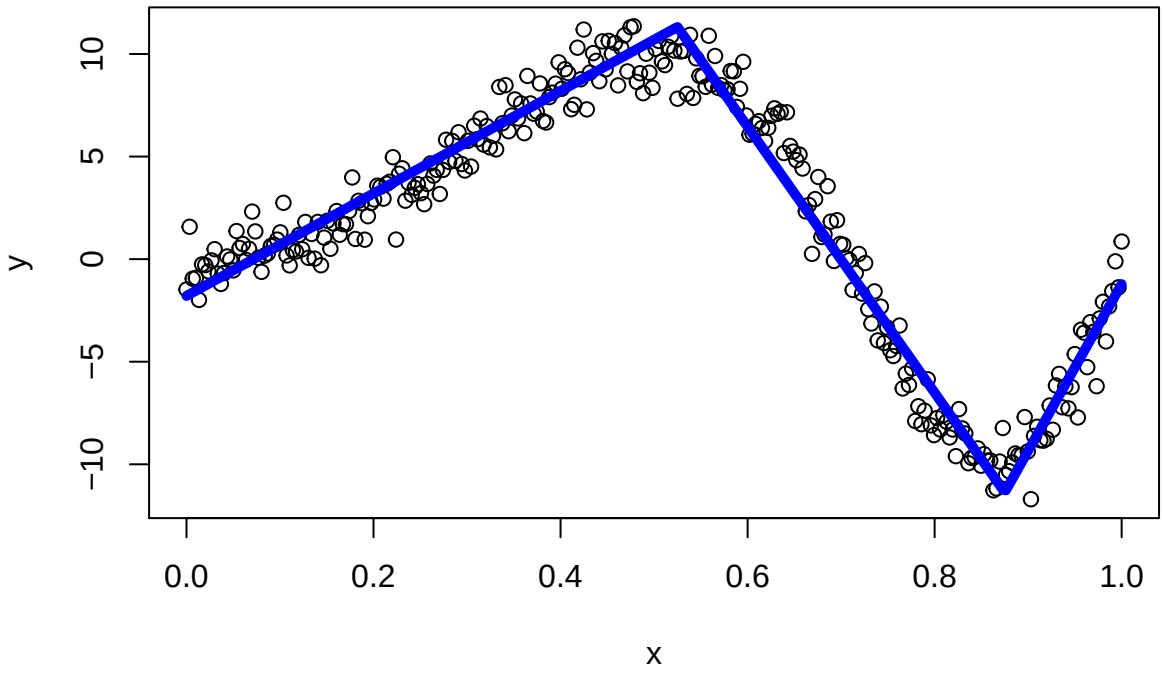


Figure 1: The blue line represents the spline function fit to the data using three knots

2 Robust BMARS

2.1 Overview

We want to extend the theory behind frequentist univariate spline regression to a multivariate Bayesian framework. Moreover, we want to accurately fit nonlinear data that has outliers. We adopt the Robust BMARS model based on the standard Gaussian framework used by Denison, Mallick, and Smith (1998) and Francom et al. (2019) when deriving our likelihood and full conditional distributions for the parameters.

In the presence of outliers, Gaussian BMARS will attempt to capture the excess noise by either adding basis functions (overfitting) or inflating the variance term (σ^2). The Robust BMARS model accounts for this sensitivity to outliers by avoiding overfitting or variance inflation when the degrees of freedom (ν) are low. When ν is high, the t-distribution closely mimics a normal distribution, so the Robust BMARS model behaves in a similar way.

The function used to fit the Robust BMARS model in the `TBASS` package is the `tbass()` command (see section 3).

2.2 Likelihood

In creating the Robust BMARS model, we introduce new auxiliary parameters not used in the Gaussian model (Gelman et al. (2013)).

Similar to Gaussian BMARS, we let y_i be our dependent variable and $\mathbf{x}_i$ be our independent variable. Without loss of generality, all independent variables $\mathbf{x}_i$ are scaled from zero to one (Francom and Sansó (2019)).

In the robust case, the dependent variable $\mathbf{y}$ is modeled as

$$\mathbf{y} = \mathbf{B}\boldsymbol{\beta} + \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim t_\nu \left(\mathbf{0},\ \sigma^2\mathbf{I}\right) \tag{2}$$

or equivalently,

$$y_i | V_i \sim \mathcal{N}\left(\mathbf{B_i}'\boldsymbol{\beta},\ \frac{\sigma^2}{V_i}\right). \tag{3}$$

We also assume a Gamma prior with shape and rate $\frac{\nu}{2}$ for V_i, such that

$$V_i \sim \Gamma\left(\frac{\nu}{2},\ \frac{\nu}{2}\right) \tag{4}$$

where $\mathbf{B}$ represents the matrix of basis functions by column, $\boldsymbol{\beta}$ is the vector of regression coefficients, $\boldsymbol{\epsilon}$ is the error term, ν represents the degrees of freedom in a Student's t-distribution, and $\sigma^2\frac{\nu}{\nu-2}$ is the variance term for y_i.

The basis functions themselves are produced the same way as in the `BASS` package by Francom and Sansó (2019).

2.3 Priors

Our regression coefficients $\boldsymbol{\beta}$ follow a Gaussian prior such that

$$\boldsymbol{\beta} \sim \mathcal{N}\left(0,\ \tau^2\mathbf{I}\right). \tag{5}$$

We assume an Inverse-Gamma prior for σ^2 with default shape $\gamma_1 = 0$ and default rate $\gamma_2 = 0$ such that

$$\sigma^2 \sim IG\ (\gamma_1,\ \gamma_2). \tag{6}$$

For λ, we assume a gamma prior such that $h_1 = h_2 = 10$ by default:

$$\lambda \sim \Gamma\left(h_1,\ h_2\right). \tag{7}$$

We use a $Uniform(0, 1)$ prior for t_i and a discrete uniform prior for s_i and J_i.

2.4 Posterior

The full posterior up to a constant is

$$\mathcal{N}\left(\mathbf{B_i}'\boldsymbol{\beta},\ \frac{\sigma^2}{V_i}\right)\ \Gamma\left(\frac{\nu}{2},\ \frac{\nu}{2}\right)\mathcal{N}\left(0,\ \tau^2\mathbf{I}\right)\ IG\ (\gamma_1,\ \gamma_2)\ \Gamma\left(h_1,\ h_2\right) \tag{8}$$

2.4.1 Regression Coefficients

We can marginalize $\boldsymbol{\beta}$ out of the posterior to obtain a marginal posterior that relies on the regression estimate

$$\hat{\boldsymbol{\beta}} = \frac{1}{\sigma^2}\left(\frac{1}{\sigma^2}\mathbf{B}'\mathbf{V}\mathbf{B} + \tau^{-2}\mathbf{I}\right)^{-1}\mathbf{B}'\mathbf{V}\ \mathbf{y} \tag{9}$$

where τ^2 is the prior variance for β_i.

Estimate (9) allows us to achieve our t-distributed likelihood function for the birth step. After marginalizing out $\boldsymbol{\beta}$ and σ^2, we can simplify our Likelihood L to

$$L(\mathbf{y} \mid \cdot) \propto \left(\tau^2\right)^{-\frac{M+1}{2}}\ |\mathbf{V}|^{-1/2}\ \left|\mathbf{H}^{-1}\right|^{-1/2}\exp\left(-\frac{1}{2}\mathbf{y}'\mathbf{V}^{-\mathbf{1}}\mathbf{y}\right)\exp\left(-\frac{1}{2}\hat{\boldsymbol{\beta}}'\mathbf{H}^{-\mathbf{1}}\hat{\boldsymbol{\beta}} - 2\hat{\boldsymbol{\beta}}'\mathbf{B}'\mathbf{V}^{-\mathbf{1}}\mathbf{y}\right)$$

where $\mathbf{H} = (\mathbf{B}'\mathbf{V}^{-\mathbf{1}}\mathbf{B} + \tau^{-2}\mathbf{I})$.

We then complete the square on the first exponential term using $\mathbf{y}'\ \mathbf{V}^{-\mathbf{1}}\mathbf{B}\ \mathbf{H}^{-\mathbf{1}}\ \mathbf{B}'\mathbf{V}^{-\mathbf{1}}\mathbf{y}$ which simplifies to $\hat{\boldsymbol{\beta}}'\mathbf{H}\hat{\boldsymbol{\beta}}$ since $\mathbf{B}'\mathbf{V}^{-\mathbf{1}}\mathbf{y} = \mathbf{H}\hat{\boldsymbol{\beta}}$ from (9).

From there, we have that

$$\begin{aligned}
L(\mathbf{y} \mid \cdot) &= \left(2\pi\tau^2\right)^{-\frac{M+1}{2}}\ |\mathbf{V}|^{-1/2}\ \exp\left(-\frac{1}{2}\left(\mathbf{y}'\mathbf{V}^{-\mathbf{1}}\mathbf{y} - \hat{\boldsymbol{\beta}}'\mathbf{H}\hat{\boldsymbol{\beta}}\right)\right)\exp\left\{-\frac{1}{2}\left(\boldsymbol{\beta}'\mathbf{H}\boldsymbol{\beta} - \boldsymbol{\beta}'\mathbf{H}\hat{\boldsymbol{\beta}} - \hat{\boldsymbol{\beta}}'\mathbf{H}\boldsymbol{\beta} + \hat{\boldsymbol{\beta}}'\mathbf{H}\hat{\boldsymbol{\beta}}\right)\right\} \\
&= \left(2\pi\tau^2\right)^{-\frac{M+1}{2}}\ |\mathbf{V}|^{-1/2}\ \exp\left(-\frac{1}{2}\left(\mathbf{y}'\mathbf{V}^{-\mathbf{1}}\mathbf{y} - \hat{\boldsymbol{\beta}}'\mathbf{H}\hat{\boldsymbol{\beta}}\right)\right)\exp\left\{-\frac{1}{2}\left[\left(\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}\right)'\mathbf{H}\left(\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}\right)\right]\right\} \\
&= \left(2\pi\tau^2\right)^{-\frac{M+1}{2}}\ |\mathbf{V}|^{-1/2}\ \exp\left(-\frac{1}{2}\left(\mathbf{y}'\mathbf{V}^{-\mathbf{1}}\mathbf{y} - \hat{\boldsymbol{\beta}}'\mathbf{H}\hat{\boldsymbol{\beta}}\right)\right)\ \mathcal{N}\left(\boldsymbol{\beta} \mid \hat{\boldsymbol{\beta}},\ \mathbf{H}^{-1}\right)\left|\mathbf{H}^{-1}\right|^{-1/2}(2\pi)^{\frac{M+1}{2}} \\
&= \left(\tau^2\right)^{-\frac{M+1}{2}}|\mathbf{V}|^{-1/2}\ \left|\mathbf{H}^{-1}\right|^{-1/2}\exp\left\{-\frac{1}{2}\left(\mathbf{y}'\mathbf{V}^{-\mathbf{1}}\mathbf{y} - \hat{\boldsymbol{\beta}}'\mathbf{H}\hat{\boldsymbol{\beta}}\right)\right\} \\
&= \left(\tau^2\right)^{-\frac{M+1}{2}}|\mathbf{V}|^{-1/2}\ \left|\mathbf{H}^{-1}\right|^{-1/2}\exp\left\{-\frac{1}{2}\left(\mathbf{y}'\mathbf{V}^{-\mathbf{1}}\mathbf{y} - \hat{\boldsymbol{\beta}}'\left(\mathbf{B}'\mathbf{V}^{-\mathbf{1}}\mathbf{B} + \tau^{-2}\mathbf{I}\right)\hat{\boldsymbol{\beta}}\right)\right\}
\end{aligned}$$

$$\Rightarrow L(\mathbf{y} \mid \cdot) = (\tau^2)^{\frac{M+1}{2}} |\mathbf{V}|^{-1/2} \left|(\mathbf{B'V^{-1}B} + \tau^{-2}\mathbf{I})^{-1}\right|^{-1/2} \exp\left\{ -\frac{1}{2}(\mathbf{y'V^{-1}y} - \hat{\boldsymbol{\beta}}'(\mathbf{B'V^{-1}B} + \tau^{-2}\mathbf{I})^{-1}\hat{\boldsymbol{\beta}})\right\} \tag{10}$$

Equation (10) is the likelihood for the current model with M basis functions.

2.5 Reversible-Jump Markov Chain Monte Carlo (RJ-MCMC)

Like Gaussian BMARS, the robust algorithm adaptively builds, deletes, and modifies basis functions, sampling candidate knot locations, signs, interaction degrees, and accepting or rejecting the candidate values using a RJ-MCMC algorithm.

RJ-MCMC is a generalization of the traditional Metropolis-Hastings algorithm in the sense that RJ-MCMC allows for parameter dimension change, allowing for simulation when the number of parameters is unknown. This is important for BMARS because we want to learn where knots should be placed and how many basis functions to have in our model, along with the degree of interaction for our basis functions. We also want to know if certain basis functions should be added, deleted, or changed.

Our Robust generalization for the RJ-MCMC algorithm is largely based off the work by Denison, Mallick, and Smith (1998).

The BMARS model has three possible move types, which is sampled using a discrete uniform:

- **Birth**: adding a basis function
- **Death**: deleting a basis function
- **Change**: changing a knot, sign, and values of a basis function

Once the move type is sampled, the RJ-MCMC algorithm is used to determine acceptance of that move type.

Our acceptance ratio α is denoted by

$$\alpha = min\left\{1,\ \frac{\pi(\theta')\ S(\theta' \to \theta)}{\pi(\theta)\ S(\theta \to \theta')}\right\} \tag{11}$$

where D represents the data, θ' represents the candidate model parameters and θ represents the current model parameters, L is the likelihood, p is the prior, and S is the proposal to jump from one model to another.

Section 2.6 details the RJ-MCMC algorithm for the birth step in detail. The death and change steps are very similar in nature.

2.6 Birth Step

2.6.1 Likelihood Ratio

Since a basis function is added for the birth step, we have that

$$L(\mathbf{y} \mid \theta' = M') \propto \left(\tau^2\right)^{-\frac{M+2}{2}} |\mathbf{V}|^{-1/2} \left|\mathbf{H}^{-1}\right|^{-1/2} \exp\left\{-\frac{1}{2}\left(\mathbf{y'V^{-1}y} - \hat{\boldsymbol{\beta}}'\left(\mathbf{B'V^{-1}B} + \tau^{-\mathbf{2}}\mathbf{I}\right)\hat{\boldsymbol{\beta}}\right)\right\} \tag{12}$$

Using (10), the likelihood ratio $\frac{L(\theta')}{L(\theta)}$ is then represented as

$$\frac{L(\mathbf{y} \mid \theta' = M' = M+1)}{L(\mathbf{y} \mid \theta = M)} = \frac{(\tau^2)^{-\frac{M+2}{2}} |\mathbf{V}|^{-1/2} \left|\mathbf{H}_c^{-1}\right|^{-1/2} \exp\left\{-\frac{1}{2}\left(\mathbf{y'V^{-1}y} - \hat{\boldsymbol{\beta}}_c' \left(\mathbf{B_c'V^{-1}B_c} + \tau^{-2}\mathbf{I_c}\right)\hat{\boldsymbol{\beta}}_c\right)\right\}}{(\tau^2)^{-\frac{M+1}{2}} |\mathbf{V}|^{-1/2} |\mathbf{H}^{-1}|^{-1/2} \exp\left\{-\frac{1}{2}\left(\mathbf{y'V^{-1}y} - \hat{\boldsymbol{\beta}}' \left(\mathbf{B'V^{-1}B} + \tau^{-2}\mathbf{I}\right)\hat{\boldsymbol{\beta}}\right)\right\}} \tag{13}$$

where the subscript c denotes the candidate move type of the algorithm (a birth step in this case).

The death is very similar to the birth step, with a candidate likelihood of $L(\mathbf{y} \mid \theta' = M' = M-1)$. There is no dimension change in the change step.

2.6.2 Prior Ratio

For the birth step, our prior ratio $\frac{p(\theta')}{p(\theta)}$ is of the following form, based on Francom et al. (2019):

$$\frac{\lambda}{M+1}\left(\frac{1}{2}\right)^J \binom{p}{J}^{-1} \left(\frac{1}{J_{max}}\right)(M+1) \tag{14}$$

where $\frac{\lambda}{M+1}$ is the prior for the number of basis functions with M current knots, $\left(\frac{1}{2}\right)^J$ is the prior for the signs, $\binom{p}{J}^{-1}$ is the prior for the number of possible variable combinations to create basis functions with, $\frac{1}{J_{max}}$ is the prior for the number of interactions, and $M+1$ is accounting for ordering the basis functions.

2.6.3 Proposal Ratio

Our proposal ratio for a birth step $\frac{S(\theta' \to \theta)}{S(\theta \to \theta')}$ can be expressed as:

$$\frac{\frac{1}{3}\frac{1}{M+1}}{\frac{1}{3}\frac{1}{J_{max}}\binom{p}{J}^{-1}(\frac{1}{2})^J} \tag{15}$$

which is effectively the probability of selecting a death step multiplied by the probability of selecting a specific basis function to kill, over the probability of proposing the already-proposed basis function.

The RJ-MCMC algorithm in `TBASS` calculates the likelihoods, priors, proposals, and acceptance ratios all on the log scale.

2.7 Gibbs Sampling

Once the basis function move type is complete, the model parameter values λ, V_i, σ^2, and $\boldsymbol{\beta}$ can then be sampled using Gibbs Sampling, since the full conditionals are all closed-form (Denison, Mallick, and Smith (1998)). The Gibbs Sampling steps shown below are not unique to the birth step as they are performed after every RJ-MCMC iteration.

Derived from section 2.3, the full conditionals are of the following form:

$$\lambda|\cdot \sim \Gamma\big(h_1 + M,\ h_2 + 1\big) \tag{16}$$

$$V_i|\cdot \sim \Gamma\left\{\frac{\nu+1}{2},\ \frac{1}{2\sigma^2}\sum_{i=1}^{n}(\mathbf{y}-\mathbf{X}\boldsymbol{\beta})^2\right\} \tag{17}$$

$$\sigma^2|\cdot \sim IG\left(\gamma_1+\frac{n}{2},\ \gamma_2+\frac{1}{2}(\mathbf{y}-\mathbf{B}\boldsymbol{\beta})'(\mathbf{y}-\mathbf{B}\boldsymbol{\beta})\right) \tag{18}$$

$$\boldsymbol{\beta}|\cdot \sim \mathcal{N}\left(\hat{\boldsymbol{\beta}},\left(\mathbf{B}'\mathbf{V}^{-\mathbf{1}}\mathbf{B}+\tau^{-2}\mathbf{I}\right)\right) \tag{19}$$

where $\hat{\boldsymbol{\beta}}$ is denoted in (9).

3 Simulation with `tbass()`

We now demonstrate the capabilities of the `TBASS` package using the main command, `tbass()`. The command uses an `Rcpp` interface (C++ functions) to optimize computation time. For all parameter values of this function, please refer to the help documentation by running `?tbass` after loading the package.

Software Requirements and Dependencies

At this time, you must have R Version 4.0.2 (Taking Off Again) or higher to install and use `TBASS`.

The package `mnormt` is REQUIRED in order to use `TBASS`. Please run `install.packages("mnormt")` to install the package. This dependency will be removed when the package is updated further. No other dependencies are required.

We begin by loading in the package and setting the seed for reproducibility. The package can be installed using the following command: `devtools::install_github("aashen12/TBASS")` which requires installing the package `devtools`.

When installing `TBASS`, you may be asked to update packages to a more recent version. Please update all packages (option 1).

When installing `RcppArmadillo`, you may be asked to install the package from sources which need compilation. Please type "no".

If you are not asked for these updates, please allow the installation to proceed normally.

```
set.seed(12)
library(TBASS)
```

3.1 Friedman Function Simulation

We fit the Robust BMARS model to the infamous Friedman Function (Friedman (1991)):

$$10\ sin(\pi x_1 x_2)\ +\ 20(x_3 - 0.5)^2\ +\ 10x_4\ +\ 5x_5 \tag{20}$$

In R:

```
f <- function(x) {
  10*sin(pi*x[,1]*x[,2]) + 20*(x[,3]-.5)^2 + 10*x[,4] + 5*x[,5]
}
```

We set our true value of $\sigma = 1$ and attempt to capture the same variability in using `tbass()`.

```
sigma <- 1 # TRUE noise sd
n <- 1000 # number of observations
x <- matrix(runif(n*5), nrow = n, ncol = 5)
y <- rnorm(n, mean = f(x), sd = sigma)
```

We then add extra noise to simulate a dataset with outliers, and categorize them for plotting later.

```
ind <- sample(n, size = 10) # convert 10 points to outliers
y[ind] <- rnorm(5, f(x[ind,]), 15)
```

```
col <- rep(1,n) # for coloring the outlier points in a later plot
col[ind] <- 2
```

From there, we can run the `tbass()` command with 30000 MCMC iterations (default 10000). Our first simulation is with $\nu = 10$ degrees of freedom to simulate a t-distribution with thicker tails.

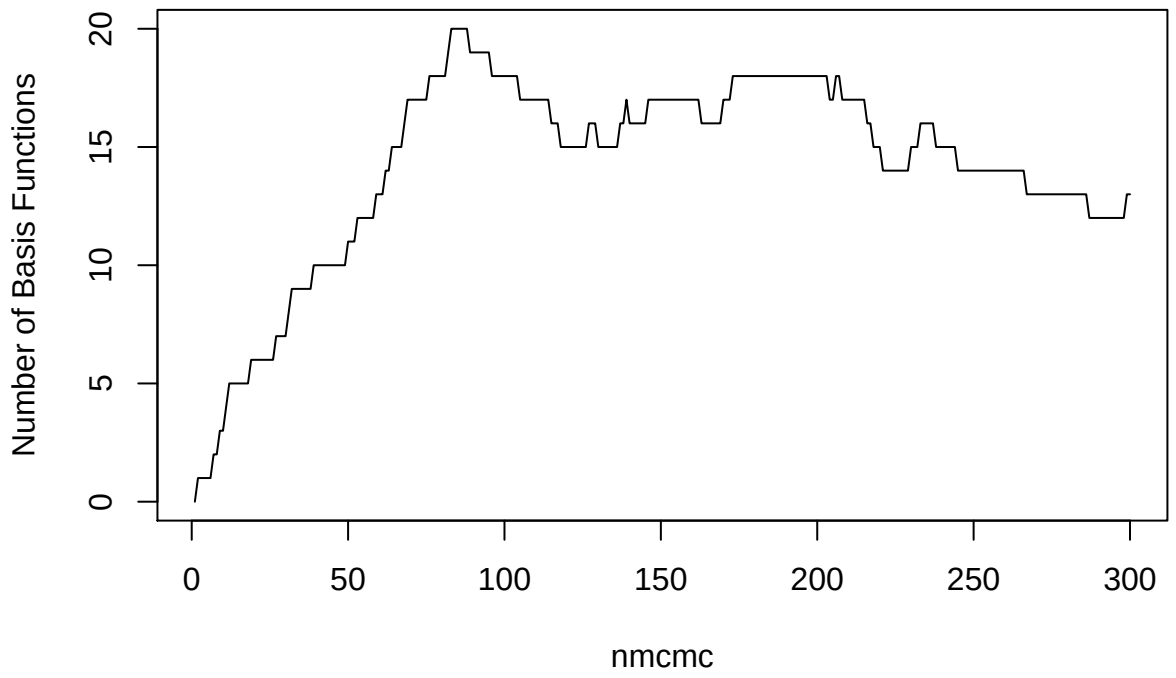
```
nmcmc <- 300 #number of iterations
tb <- tbass(x, y, nu = 10, nmcmc = nmcmc, verbose = FALSE)
```

3.2 Results

3.2.1 Overfitting and Prediction

We begin by plotting the number of basis functions throughout the entire simulation in Figure 3.

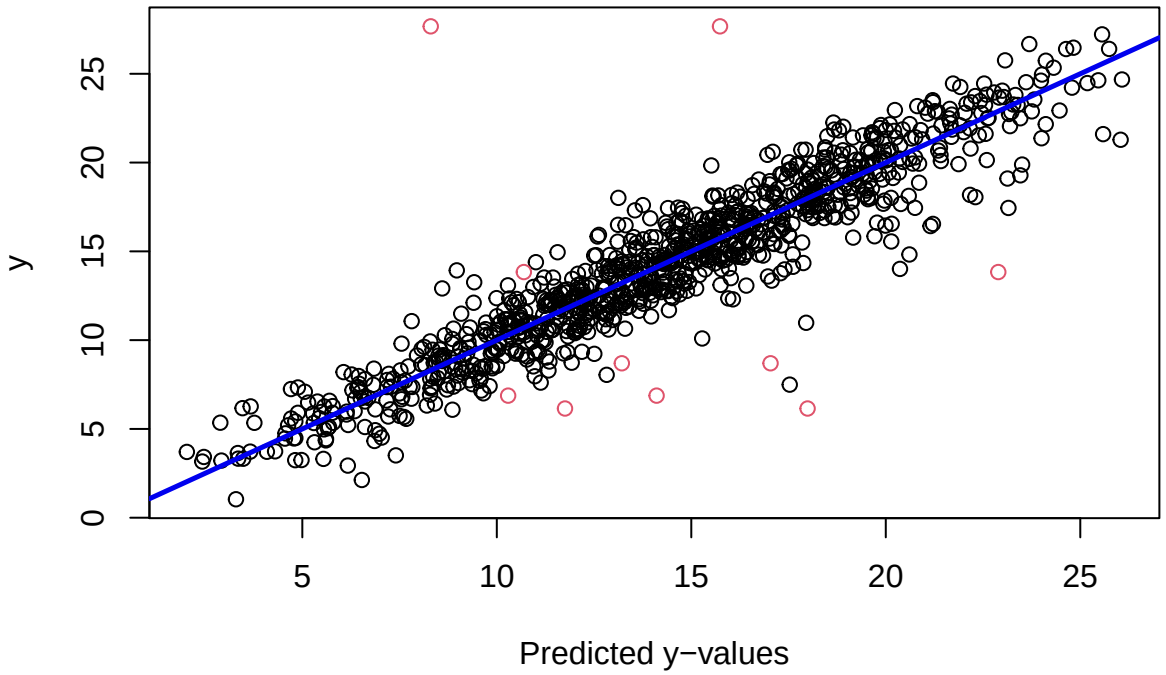
Figure 3: Trace plot of number of basis functions



We see that there are 9 basis functions from `BASS`, indicating that there was no overfitting in the `TBASS` model with 10 basis functions.

We then assess the accuracy of our predicted $\mathbf{y}$ values ($\mathbf{X}\boldsymbol{\beta}$) vs the actual y-values in Figure 4.

Figure 4: Predicted vs Actual Values



We see that the prediction is quite accurate despite the presence of outliers.

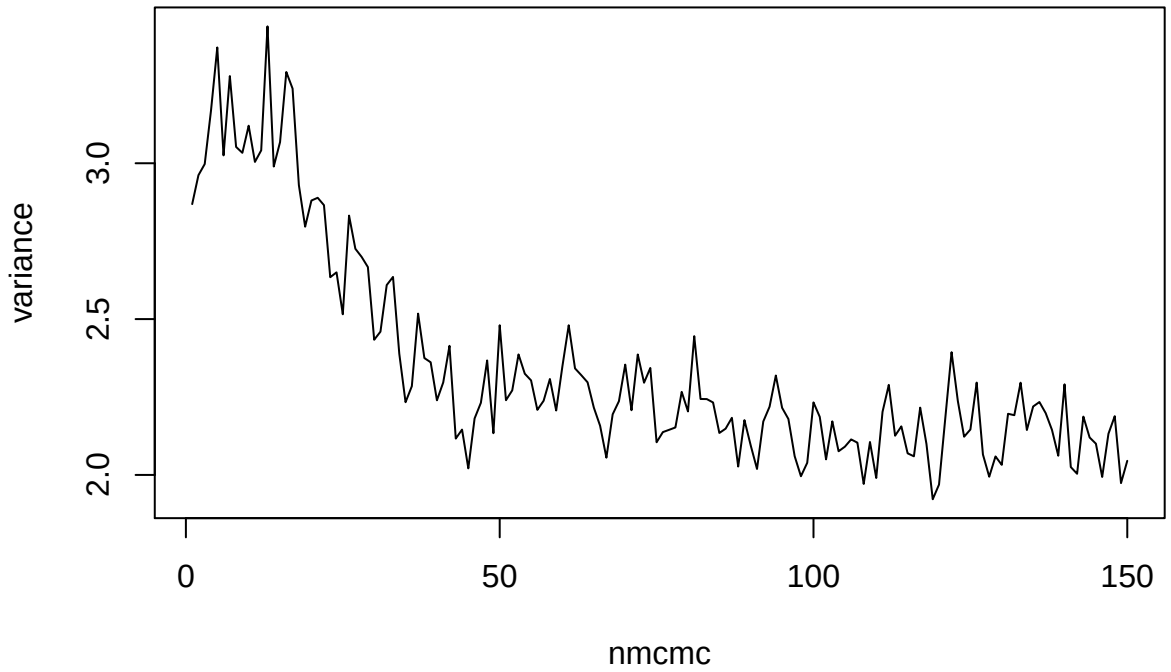
3.2.2 Gibbs Sampling Results

We set a burn-in value to the first 50% of samples.

```
burn_final <- nmcmc/2
burn <- 1:burn_final
```

From there, we plot our σ values in Figure 5 to ensure there was no variance inflation.

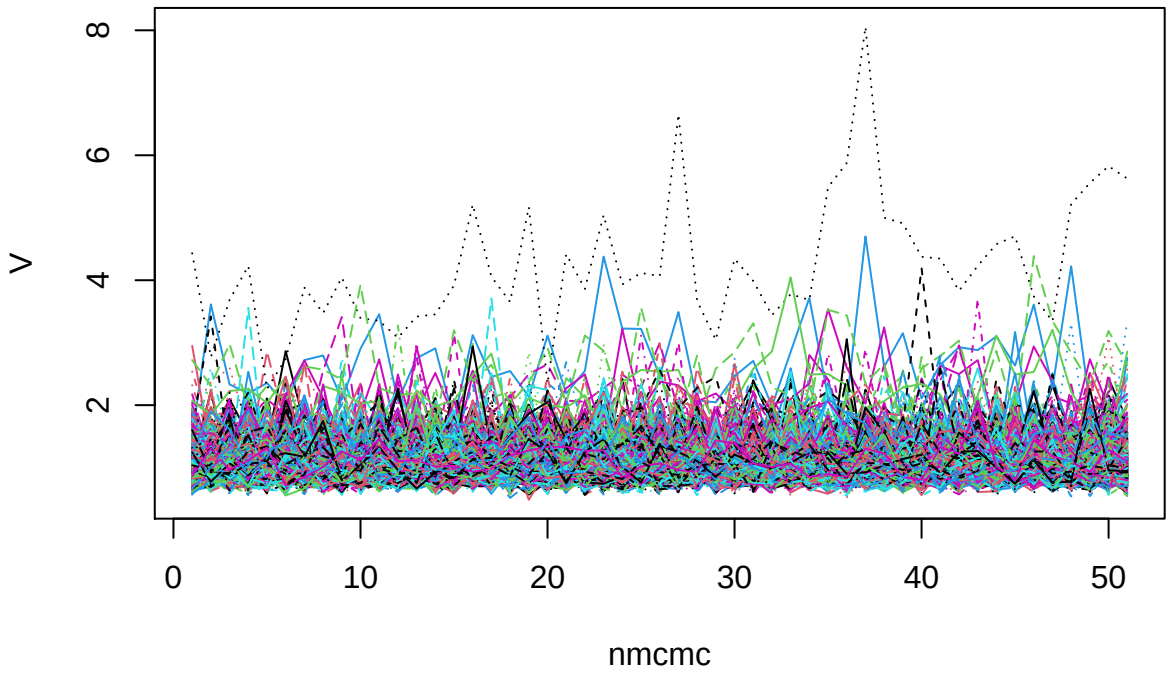
Figure 5: TBASS Variance Trace Plot



From this plot, we see that our simulated σ^2 values are close to 1, which matches our true value of σ^2 when we generated our data.

We then plot our $\frac{1}{V_i}$ values to see if the outliers are accounted for. We thin every `nmcmc/100` iterations after the burn-in. See Figure 6.

Figure 6: TBASS Trace Plot of V_i



We see that the peaks in V_i are accounting for the outliers, while the majority of V_i values are in the bottom portion of the plot.

3.3 Comparison with BASS

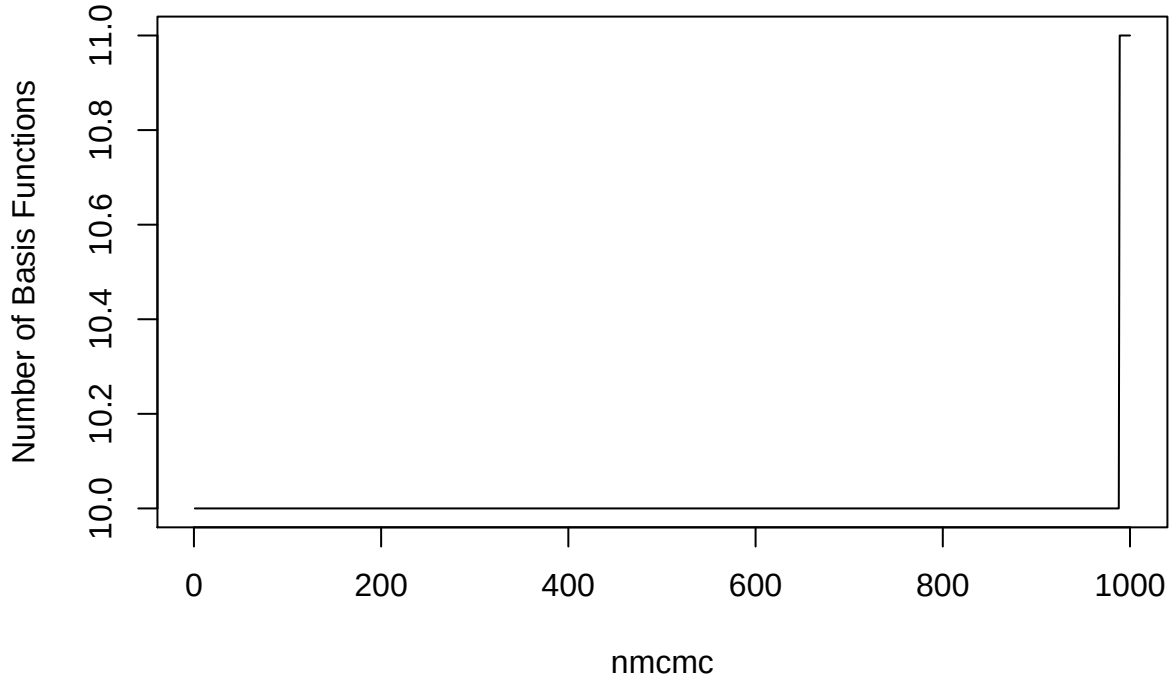
To account for the differences in behavior for Robust BMARS (TBASS) and Gaussian BMARS (BASS), we compare our results from TBASS to the output from BASS, which assumes a Gaussian likelihood.

```
library(BASS)
b <- bass(x,y,nmcmc = 10000) # automatically runs 10000 nmcmc iterations
# > MCMC Start #-- Sep 10 14:58:30 --# nbasis: 0
# > MCMC iteration 1000 #-- Sep 10 14:58:33 --# nbasis: 11
# > MCMC iteration 2000 #-- Sep 10 14:58:35 --# nbasis: 11
# > MCMC iteration 3000 #-- Sep 10 14:58:37 --# nbasis: 11
# > MCMC iteration 4000 #-- Sep 10 14:58:39 --# nbasis: 11
# > MCMC iteration 5000 #-- Sep 10 14:58:41 --# nbasis: 11
# > MCMC iteration 6000 #-- Sep 10 14:58:43 --# nbasis: 11
# > MCMC iteration 7000 #-- Sep 10 14:58:45 --# nbasis: 11
# > MCMC iteration 8000 #-- Sep 10 14:58:47 --# nbasis: 10
# > MCMC iteration 9000 #-- Sep 10 14:58:49 --# nbasis: 10
# > MCMC iteration 10000 #-- Sep 10 14:58:51 --# nbasis: 11
```

3.3.1 BASS Basis Functions

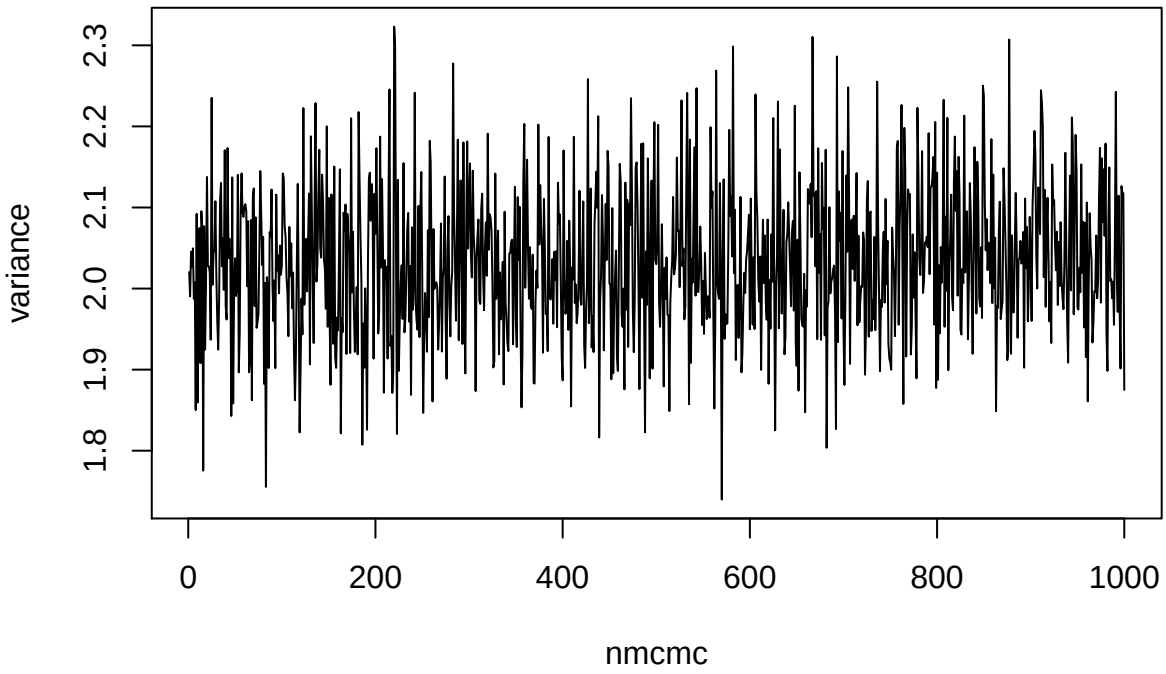
Figure 7 shows that there are also 10 basis functions once the BASS algorithm is complete, similar to TBASS.

Figure 7: BASS Basis Function Count



3.3.2 BASS Variance

Figure 8: BASS Variance Trace Plot



We see that the σ^2 from Gaussian BMARS is over two times as large as the σ^2 from `TBASS`, indicating that Gaussian BMARS is more sensitive to outliers and will inflate σ^2 when outliers are present.

4 Conclusion

We constructed a generalized version of Gaussian Bayesian Multivariate Adaptive Regression to accommodate data with outliers or data that is non-Gaussian. This framework provides reliable and accurate parameter estimation. The R package `TBASS` adopts this framework from the original `BASS` package. These two models have been tested and compared to show their differences and demonstrate how a low value of ν can emulate the results from a Student's t-distribution.

References

Denison, David GT, Bani K Mallick, and Adrian FM Smith. 1998. "Bayesian Mars." *Statistics and Computing* 8 (4): 337–46.

Francom, Devin, and Bruno Sansó. 2019. "Bass: An R Package for Fitting and Performing Sensitivity Analysis of Bayesian Adaptive Spline Surfaces." *Journal of Statistical Software* 2.

Francom, Devin, Bruno Sansó, Vera Bulaevskaya, Donald Lucas, and Matthew Simpson. 2019. "Inferring Atmospheric Release Characteristics in a Large Computer Experiment Using Bayesian Adaptive Splines." *Journal of the American Statistical Association* 114 (528): 1450–65.

Friedman, Jerome H. 1991. "Multivariate Adaptive Regression Splines." *The Annals of Statistics*, 1–67.

Gelman, Andrew, John B Carlin, Hal S Stern, David B Dunson, Aki Vehtari, and Donald B Rubin. 2013. *Bayesian Data Analysis*. CRC press.