# Simulation Study for Single-Index Models

by

Thomas J. Fisher

Submitted to the
Department of Mathematical Sciences
of Clemson University

in partial fulfillment of
the requirements for the degree of
Master of Science in
Mathematical Sciences
May, 2006

Advisor: Dr. K.B. Kulasekera
Committee: Dr. C.M. Gallagher
Committee: Dr. W.J. Padgett

Approved: _____
Committee Chair

**Acknowledgements**

The research conducted, and consequentially this paper would not be possible without the help of the following; I'd like to thank each of them for their contribution in the completion of this project:

Dr. K.B. Kulasekera, for his guidance and expertise on the subject matter. His enthusiasm towards the topic provided motivation and most importantly, his believing and encouragement in me.

Dr. C.M. Gallagher, for his friendship and help in the programming aspects required. Without his help, much of the code would be in an inefficient or incorrect state. His friendship during the final few weeks provided a much needed stress reducer.

Dr. W.J. Padgett, for his inspiration to join the field of statistics. His editorial review greatly improved the original drafts of this paper.

My family, sister Barbara and father Charles Ronald, for their support in all of my endeavors.

**Abstract**

Single-Index Models (SIMs) generalize regression. In this project we perform a simulation study comparing SIMs to the linear regression technique. Results demonstrate the SIM as a viable alternative to linear modeling techniques. We provide examples and discuss the importance of estimating the unique projection vector in the SIM. Potential methods to estimate the unique projection vector are discussed.

# Contents

# 1 Introduction

In many scientific investigations such as, econometric studies, dose response models in biometrics, reliability studies, analysis of electrical signals, etc. the ability to capture and model a signal (typically the mean function of a response based on predictor variables) is essential. In many cases, a semi-parametric method can be used as a compromise between too restrictive parametric models and extremely flexible nonparametric models (Lin 2002). In this project, we compare the typical linear modeling method to a semi-parametric modeling method commonly called the Single-Index Model.

## 1.1 General Linear Modeling

The most common form of modeling a set of responses is that of linear regression. Suppose a set of responses $Y = (Y_1, Y_2, \ldots, Y_n)^T$ are observed, along with accompanying predictor variables $X_i = (x_{i,1},\ x_{i,2},\ \ldots,\ x_{i,k})$ for each $i = 1,\ldots,\ n$. The responses can be modeled in the form

$$Y_i = \beta_0 + \beta_1 x_{i1} + \ldots + \beta_k x_{ik} + \epsilon_i;\ i = 1, \ldots, n$$

which is typically written in the general form

$$Y = X\beta + \epsilon$$

where $X$ is the matrix of $x_{ij}$'s, $i = 1,\ldots,n$; $j = 1,\ldots,k$ and $\epsilon = (\epsilon_1,\epsilon_2,\ldots,\epsilon_n)^T$, with $E[\epsilon] = 0$. Upon making the observations $Y_1,Y_2,\ldots,Y_n$, if the distribution of $\epsilon$ is known, likelihood methods can be used to estimate the vector of coefficients $\beta$. If the distribution of $\epsilon$ is not known, but suitable assumptions on the distribution can be made (i.e. mean zero, finite variance), the method of least squares (LS) can be used efficiently for estimating the vector of coefficients $\beta$ (Horowitz 1998). Much literature exists on the details of estimating $\beta$. The details for estimation and testing for the $\beta$'s for linear models can be found in Graybill (1976), and many other texts on linear models. If we were to use ordinary LS method for the vector $\beta$, the estimator is given by

$$\hat{\beta} = X^{-}Y = (X^T X)^{-1} X^T Y.$$

One can then find a vector of predicted $Y$'s given by

$$\hat{Y} = X\hat{\beta}$$

and typically the error term's variance, $\sigma^2$, is estimated by

$$\hat{\sigma}^2 = Y^T(I - XX^{-})Y/(n - k - 1).$$

Inference about the model and the error structure can be performed based on the assumptions made on $\epsilon_i$'s.

## 1.2 Single-Index Modeling

The familiar parametric method described above, linear modeling, can be quite restrictive since certain assumptions must be made (some knowledge about the distribution of $\epsilon$, linearity of the mean function in terms of the coefficients $\beta_1, \ldots, \beta_k$) for inference. When linearity of the model is not readily known, a single-index model is a viable alternative. Single-index models relax some of the model restrictions, such as linearity of the mean function, are relatively easy as far as computing is concerned and maintain many of the desirable features of linear model and least-squares methods (Horowitz 1998). Given random variables $(Y, X)$, we can define a function $g(x) = E[Y|X = x]$. This is known as the mean function. Then, a general model can be written as

$$Y = g(X) + \epsilon \tag{1}$$

and as before, $\epsilon$ is a mean zero random variable independent of $X$. We can consider $E[Y|X = x]$ as a projection of $(X, Y)$ onto a lower dimensional space, typically $\mathbb{R}$. Generally $g : \mathbb{R}^p \to \mathbb{R}$, but a specific case that will be studied in this project is $g(x) = h(\theta^T x)$ where $\theta \in \mathbb{R}^p$ and $h(\cdot)$ is a smooth univariate function. Therefore $g(x)$ is constant on the contours $\theta^T x$ and as shown in Lin (2002), the contour line $\theta^T x$ provides as much information as $X$ about $g$ under this particular model. This is referred to as the *single-index model* (SIM). Thus, hereafter we assume that each of our observations $(x_i, Y_i)$, where $x_i$ is $p$-dimensional, are generated from the model

$$Y_i = h(\theta^T x_i) + \epsilon_i;\ i = 1, \ldots, n$$

where $h$ is a smooth univariate function, $\theta$ is a $p$-vector with $\left\|\theta\right\| = 1$ and $\theta_1 > 0$. These restrictions are needed for the function $h$ to be identifiable (Lin and Kulasekera 2006). Also $\epsilon_1, \ldots, \epsilon_n$ are iid random variables with mean zero and a finite variance. The parameter, $\theta$, is called the *index*and can be thought of as a projection vector (from $\mathbb{R}^p$ to $\mathbb{R}$). Upon making the observations, the goal is to then estimate $h(\cdot)$ and $\theta$ based on our observations $(x_i, Y_i); i = 1, \ldots, n$. As defined, the single-index model is a particular case of the general model (1) described above. When $h$ is the identity, this becomes a linear model.

## 1.3 Model Fitting

Upon making observations $(x_i, Y_i)$, $i = 1, \ldots, n$, a model fitting technique will be applied in an attempt to capture the original signal, $h(\theta^T x)$, and find an estimator for the variance of $\epsilon$, our error term. If our function $h$ is linear,

a general linear model captures the mean function to a high degree and we can estimate for the variance of $\epsilon$ using an appropriate sum of squares of residuals.

### 1.3.1 Linear Modeling

Issues arise when our link function, $g(x)$ is nonlinear (e.g. polynomial, sine, cosine, exponential, logistic). We can approximate these non-linear functions with a linear model by expanding $g(\cdot)$ in a Taylor series to include squares, cubes, etc... of our individual covariates. As an example, suppose the link function $h$ is given by $h(\theta^T x) = (\theta^T x)^2$ for $p = 2$ for our observations; $x_1$, $x_2$ with responses $y_i$. We can fit a model that includes the square terms, along with the crossproduct terms; $x_1$,$x_2$,$x_1^2$,$x_2^2$,$x_1x_2$ and get a perfect linear model in all these terms, that will be an exact fit for $h(\theta^T x)$. Our new model then has $q = 5$ covariates. Performing a regression would effectively capture the quadratic signal. However, with the expansion of the number of covariates $q$, in this simple example, the degrees of freedom for the unbiased estimator of the variance becomes smaller, although we get a near perfect model.

Using a method identical to our first example, a general linear model can be used for a trigonometric or an exponential mean function using a Taylor series expansion. If our link function were $h(\theta^T x) = \exp(\theta^T x)$ for instance, we could use the Taylor Series expansion idea to include squares, cubes, etc. Following the first example, we could write an approximation of this function into a new model, including the terms: $x_1$,$x_2$,$x_1^2$, $x_2^2$, $x_1x_2$, etc. This would be a better approximation for the actual model function than a linear model $\beta_0 + \beta_1 x_1 + \beta_2 x_2$. However, in this purely nonlinear case, we will never be able to completely represent the highly nonlinear model function, but the approximation becomes better with more and more terms. The same degrees of freedom issue as the first example arises; as the number of parameters increase, our variance estimator becomes unstable. Furthermore, increasing the number of covariates can become computationally tedious and a sense of precise *modeling* is lost.

### 1.3.2 Semiparametric Modeling

We can take a semiparametric approach to approximate purely nonlinear models, such as trigonomic, exponential and logistic models by using a single-index model to represent our observations. Due to its familarity and simplicity to code on a computer, we use the projection pursuit regression (PPR) method with one-step to estimate the corresponding vector $\theta$. Let $\hat{\theta}$ be a suitable estimate of $\theta$. A kernel density estimator can be used to estimate

our $h$ using an appropriate kernel function and a bandwidth. A detailed description for estimators of $h$ is provided in the next chapter, giving their properties. It should be noted that several methods are available for estimating $\theta$ and $h$. In performing the simulations as part of this project, the PPR method for estimating $\theta$ and kernel smoothing for estimating $h$ were chosen for convenience in computation. Although harder to estimate than a linear model, the SIM eliminates some of the inference issues in estimating the variance of our error term, $\epsilon$.

## 1.4 Simulation Study

We provide details on approximating the SIM. First by exploring ways to estimate the projection vector, $\theta$, and then estimating the mean function, $h$. We then conduct a simulation study comparing the performance of the single-index model against the performance of linear modeling techniques. In each case, the single-index model is compared to a sequence of linear regression models by comparing the estimation of the variance of $\epsilon$ to the known variance. The estimation of the signal for each method is compared to the exact known signal, $h(\theta^T x)$. Akaike Information Criteria (AIC) (Akaike 1974), as well as the Scharz-Bayesian (hereafter referred to as Bayesian) Information (BIC) (Schwarz 1978), are used for model selection, penalizing linear modeling techniques for including too many covariate terms. Several $\theta$ values are explored, as well as a multitude of mean functions, $h$, both polynomial based and nonlinear. In each case, $h$ is selected to be univariate and smooth. Results are supplied, along with a discussion of those results. Exploration into the estimation of $\theta$ is also discussed.

# 2 Single-Index Model

## 2.1 Estimation of Projection Vector $\theta$

As described in the previous chapter, the single-index model (SIM) has both an unknown projection parameter, $\theta$, and an unknown univariate function $h$. It is more complicated than the general linear model with coefficients vector $\beta$ since both, a projection vector $\theta$ and the mean function $h$, must be estimated. Upon making observations $(x_i, Y_i)$, $i = 1, \ldots, n$ where $x_i$ is a $p$-dimensional covariate vector, $Y_i$ are modeled as

$$Y_i = h(\theta^T x_i) + \epsilon_i;\ i = 1, \ldots, n.$$

Here $h$ is smooth and univariate, $\theta$ is a $p$-vector with $\left\|\theta\right\| = 1$ and $\theta_1 > 0$, $\epsilon_1, \ldots, \epsilon_n$ are iid random variables with mean zero and a finite variance. Our goal is to estimate both $\theta$ and $h$ based on the observations.

### 2.1.1 Projection Pursuit Regression

In this project, for the simulation portion, we use the projection-pursuit regression technique to estimate our $\theta$; we also explore a weighted least-squares approach to estimating $\theta$. The projection pursuit regression technique is attributed to Friedman and Stuetzle (1981). It approximates a regression surface by a sum of empirically determined univariate functions, i.e. $E[Y|X = x]$ is approximated by $\sum_{j=1}^{r} g_j(\beta_j^T x)$. The complete details of the estimation of $g$'s and $\beta$'s can be found in Friedman and Stuetzle. The algorithm finds a set of vectors $\beta_1$,...,$\beta_r$ (for the SIM, $r = 1$) that minimizes the sum of squares of the residuals. The approximation is constructed in an iterative manner: (1) First the residuals and the term counter are initialized,

$$r_i \leftarrow Y_i;\ i = 1, \ldots, n$$

$$M \leftarrow 0$$

(2) For a given projection, $\alpha^T x$, a smooth represention $S_\alpha(\alpha^T x)$ is constructed. Then a "Figure of Merit" (criterion of fit), $I(\alpha)$ for that particular linear combination is computed by

$$I(\alpha) = 1 - \sum_{i=1}^{n}(r_i - S_\alpha(\alpha^T x_i))/\sum_{i=1}^{n} r_i^2$$

The coefficient $\alpha_{M+1}$ that maximizes $I(\alpha)$ (the "projection pursuit") is then found, along with the corresponding smoother $S_{\alpha_{M+1}}$.

(3) If the "figure of merit" is smaller than a user specified threshold, the algorithm terminates and outputs the computed $\alpha_{M+1}$, else the residuals and term counter are updated as follows:

$$r_i \leftarrow r_i - S_{\alpha_{M+1}}(\alpha_{M+1}^T x_i),\ i = 1, \ldots, n$$

$$M \leftarrow M + 1,$$

and step two is then repeated. The complete details of the algorithm are described in Section 2 of Friedman and Stuetzle (1981).

Thus the PPR algorithm finds a $\widetilde{\theta} = \alpha_{M+1}$ that *works* in estimating our $h(\theta^T X)$ by setting $r = 1$, the number of terms to include in a final PPR model. However the PPR algorithm does not necessarily estimate the unique $\theta$ used in the model.

### 2.1.2 Weighted Least-Squares

Another technique that can be used to estimate $\theta$ is the least-squares method. Define

$$h(u|\beta) = E[Y|\beta^T X = u].$$

When we have $\beta = \theta$,

$$h(u|\theta) = E[Y|\theta^T X = u] = E[h(\theta^T X) + \epsilon|\theta^T X = u] = h(u).$$

Whence,

$$h(u|\beta) \to h(u), \text{ as } \beta \to \theta.$$

Thus, given the uniqueness of $\theta$, if $h(u|\hat{\theta})$ is approximately equal to $h(u)$ for some estimator $\hat{\theta}$, we can assume our estimator, $\hat{\theta}$ is close to the actual projection vector, $\theta$. Therefore, given $h$, the $\hat{\theta}$ value corresponding to minimizing the sum of distances, $\sum_{i=1}^n d(h(u_i|\beta), h(u_i))$ with respect to $\beta$, where $d$ is a distance measure (typically absolute value or squared distance) is taken as our estimator $\hat{\theta}$. Unfortunately, neither $h(u|\beta)$ or $h(u)$ are known, and therefore must be estimated. We estimate $h(u_i)$ by $Y_i$, and $h(u|\beta)$ by a kernel estimator or local linear regression or some other suitable method. We give a brief description of kernel estimation of $h(u|\beta)$ in the next subsection. Then if the estimator of $h(\cdot|\beta)$ is $\hat{h}(\cdot|\beta)$, one can estimate the above distance function by

$$\hat{S}(\beta) = \sum_{i=1}^{n} (Y_i - \widehat{h}_i(\beta^T x_i|\beta))^2.$$

Now, we can estimate $\theta$ by minimizing $\hat{S}(\beta)$ with respect to $\beta$ (Lin 2002).

For the simulation study in this project, the projection pursuit regression technique for estimating $\theta$ is used exclusively. The weighted least-squares and methods to estimate the *unique* $\theta$ are discussed in the conclusion section, although no simulations using that approach are currently conducted.

## 2.2 Estimation of Link function $h$

Upon estimating the $\theta$ with some $\hat{\theta}$, our next step is to estimate the univariate function $h$. In this project, a kernel smoothing estimator is used for estimating the smooth function $h$. However local linear regression or spline regression could also be used in this estimation.

To estimate our mean function, a Nadaraya-Watson kernel smoothing is used,

$$\hat{Y}_i = \hat{h}_i(u_i|\hat{\theta}) = \frac{\sum_{j=1}^{n} K_a(u_i - u_j)Y_j}{\sum_{j=1}^{n} K_a(u_i - u_j)} \tag{2}$$

where $u_i = \hat{\theta}^T x_i$, $K_a(u) = K(u/a)$ for some symmetric kernel function $K$ (e.g. standard normal). The value $a$ is the bandwidth parameter for the kernel estimation and must be specified. This is usually done by some data based selection criterion. A popular method is the cross-validation (CV) criterion. The simple CV criterion selects a bandwidth $a$ through minimization of

$$CV(a) = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i^*)^2$$

where $\hat{Y}_i^*$ is the predicted $Y$ at the $i^{\text{th}}$ data point from a reduced kernel density estimate that does not use the $i^{\text{th}}$ observation to estimate the values of $g$ at that point. That is, $\hat{Y}_i^*$ is given by the formula

$$\hat{Y}_i^* = \frac{\sum_{j\neq i}^{n} Y_j K_a(u_i - u_j)}{\sum_{j\neq i}^{n} K_a(u_i - u_j)}.$$

By minimizing $CV(a)$ with respect to $a$, we find the best bandwidth for the given observations. Using the bandwidth in (2) we can then find our estimators for the function $h$ evaluated at $\hat{\theta}^T x_i$.

## 2.3 Inference about the Single-Index Model

Once we have an estimate for the model's link function $h(\theta^T x_i)$, we can make inference. In this project we calculate a mean squared error (MSE) of our estimate against the actual signal. This MSE is an indication as to how close our estimated link function $\hat{h}(\hat{\theta}^T x_i)$ is to the actual mean function.

We also find an estimate for the variance of our error term, $\epsilon$. To find an unbiased estimator for the variance of $\epsilon$, we follow the method described in Hastie and Tibshirani (1990). The kernel smoothing method for capturing the signal finds a linear smoother operator, $S_a$, such that $\hat{Y} = S_a Y$, where $a$ is the bandwidth. In estimating the variance of our error term, $\epsilon$, we first find the Sum of Squared Errors (SSE). Hastie and Tibshirani provide several forms for the degrees of freedom for our smoothing model, and hence allow us to find an unbiased estimator for $\sigma^2$. In a typical regression example (such as the general linear models described in Chapter 1), we have $n-k$ as the degrees of freedom, where $n$ is the sample size, and $k$ is the number of covariates. An unbiased estimator for $\sigma^2$ can made by

$$\hat{\sigma}^2 = \frac{\sum_{i=1}^{n}(\hat{Y}_i - Y_i)^2}{n-k}.$$

An analogous result is $n - \text{tr}(2S_a - S_a S_a^T)$, as the approximate degrees of freedom for error, for the linear smoothing operator, $S_a$ (Hastie and Tibshirani 1990). Whence we can find a reasonable estimator for $\sigma^2$ by

$$\hat{\sigma}^2 = \frac{\sum_{i=1}^{n}(\hat{Y}_i - Y_i)^2}{n - \text{tr}(2S_a - S_a S_a)}.$$

The smoothing matrix, $S_a$, is symmetric and each element has the form

$$s_{ij} = \frac{K_a(u_i - u_j)}{\sum_{l=1}^{n} K_a(u_i - u_l)}.$$

With some matrix algebra, we can simplify the denominator for our error variance estimator. First we have:

$$\text{tr}(S_a) = \sum_{i=1}^{n} \frac{K_a(0)}{\sum_{l=1}^{n} K_a(u_i - u_l)}.$$

Since $S_a$ is symmetric, simplification shows that

$$\text{tr}(S_a^2) = \sum_{i=1}^{n}\sum_{j=1}^{n} s_{ij}^2,$$

which allows us to calculate the approximate degrees of freedom used for the estimator of $\sigma^2$. The variance of the error term $\epsilon$, is then estimated from

$$\hat{\sigma}^2 = \frac{\sum_{i=1}^{n}(\hat{Y}_i - Y_i)^2}{n - 2\text{tr}(S_a) + \text{tr}(S_a^2)}. \tag{3}$$

# 3 Simulation Details

In this section, a detailed description of the simulation performed is provided. The goal of the simulation is to compare the performance of linear modeling techniques with that of the semiparametric technique, single-index model. This is done by generating a random sample from some known mean function, $h$, and a projection vector, $\theta$. The sample is then fitted to an assortment of linear models and a single-index model. The predicted means are then compared to the actual mean function used in generating the data and the variance of the error term in the model is then estimated based on the observations and the predicted values using linear models and the SIM, respectively.

## 3.1 Algorithm for Simulation

- The first step in the algorithm is generate a random sample. This is done using a suitable random number generator (RNG). The predictor variables, $x_i = (x_{i1}, x_{i2}, \ldots, x_{ip})$, for the $i^{\text{th}}$ observation in a sample of size $n$ is obtained such that $x_{ij} \sim \text{Unif}(0,1)$, $j = 1, \ldots, p$. Error terms are generated $\text{N}(0,\sigma^2)$, with $\sigma^2$ being the known variance of the error terms. The $i^{\text{th}}$ response variable is then calculated by

$$Y_i = h(\theta^T x_i) + \epsilon_i;\ i = 1, \ldots, n.$$

- Now, the observations $(Y_i, x_i)$, $i = 1,\ldots,n$, where $x_i$ is a $p$-dimensional vector are fitted to a sequence of linear models as follows:

  - Using the method described in section 1.3, create new predictor variables using the components of $X$ to include degree 2, degree 3, ..., degree 10 terms, including crossproduct terms. We will then fit a linear model using this new set of predictor variables and the response variables. This can be seen with an example. Suppose $p = 2$. Our original data matrix $X$ is of the form:

$$\mathbf{X} = \begin{pmatrix} x_{11} & x_{12} \\ \vdots & \vdots \\ x_{i1} & x_{i2} \\ \vdots & \vdots \end{pmatrix} \text{ for } i = 1, \ldots, n.$$

    We will then create a new matrix of predictor variables by includ-

ing squares and crossproduct terms:

$$\mathbf{X^{2*}} = \begin{pmatrix} x_{11} & x_{12} & x_{11}^2 & x_{12}^2 & x_{11}x_{12} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{i1} & x_{i2} & x_{i1}^2 & x_{i2}^2 & x_{i1}x_{i2} \\ \vdots & \vdots & \vdots & \vdots & \vdots \end{pmatrix} \text{ for } i = 1, \ldots, n$$

where the $2*$ represents the model of degree 2 ($3*$ would include cubes, and so on). Our new set of predictor variables has $p^{2*} = 5$. We can continue this process for cubes, quartics and so on, this simulation expands to degree 10.

– We can then use a linear modeling routine (*glm* in **R**) with $Y$ as the set of response variables, and $X^{2*}$ as the set of predictor variables, to get a set of estimated response variables $\hat{Y}^{2*}$ where

$$\hat{Y}_i^{2*} = \hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \hat{\beta}_2 x_{i2} + \hat{\beta}_3 x_{i1}^2 + \hat{\beta}_4 x_{i2}^2 + \hat{\beta}_5 x_{i1} x_{i2};\ i = 1, \ldots, n.$$

Using similar methods, we'll get a set of 10 different estimated response vectors, $\hat{Y}^{1*}$, $\hat{Y}^{2*}$, ..., $\hat{Y}^{10*}$ for each of our model types.

- Since both the exact signal, $h(\theta^T x)$, and variance of error, $\sigma^2$ are known, the performance of the linear modeling technique is compared to the known values. We first estimate an integrated mean squared error for the $j^{\text{th}}$ model, $j = 1,\ldots,10$, against the known signal, $h(\theta^T x)$. That is, calculate at the $k^{\text{th}}$ simulation

$$\gamma_k^{j*} = \frac{1}{n} \sum_{i=1}^{n} \left( \hat{Y}_i^{j*(k)} - h(\theta^T x_i) \right)^2;\ \text{for } j = 1, \ldots, 10$$

where $k$ is the simulation number ($1 \leq k \leq M$) and $\hat{Y}_i^{j*(k)}$ is the predicted $i^{\text{th}}$ response for the $j^{\text{th}}$ model at the $k^{\text{th}}$ simulation. A running sum, one for each execution of the algorithm, of each $\gamma_k^{j*}$ for $j = 1,\ldots,10$ is stored to compute the estimated average integrated mean squared error for the $j^{\text{th}}$ model. That is, at the completion of the all simulations, set

$$\bar{\gamma}^{j*} = \frac{1}{M} \sum_{k=1}^{M} \gamma_k^{j*};\ \text{for } j = 1, \ldots, 10,$$

where $M$ is the number of simulations. Thus, $\bar{\gamma}^{j*}$ for $j = 1,\ldots,10$ is a reasonable estimator of the integrated mean squared error for each of the linear model regressions performed. These averages will later be compared to the same indicator for the single-index model.

- The next step is to estimate the variance of the error term. This is done in the typical way. At the $k^{\text{th}}$ simulation, calculate

$$\hat{\sigma}_{j*}^{2(k)} = \frac{1}{n - p^{j*} - 1} \sum_{i=1}^{n} \left( \hat{Y}_i^{j*(k)} - Y_i \right)^2; \text{ for } j = 1, \ldots, 10.$$

  Each estimator for $\sigma^2$ is then compared to the known value using

$$d_k(\hat{\sigma}^2_{j*(k)}, \sigma^2) = \left( \hat{\sigma}_j^{2(k)} - \sigma^2 \right)^2; \text{ for } j = 1, \ldots, 10$$

  where $k$ represents the simulation number. As with the integrated mean squared error, at the conclusion of the simulations, we will calculate the average distance to see, on average, how accurate our estimated variance is compared to the known value, i.e.

$$\bar{d}(\hat{\sigma}^2_{j*}, \sigma^2) = \frac{1}{M} \sum_{k=1}^{M} d_k(\hat{\sigma}_{j*}^{2(k)}, \sigma^2); \text{ for } j = 1, \ldots, 10$$

  with $M$ being the number of simulations. The average distances will later be compared to a similar quantity calculated for a single-index model.

- As another measure to choose the best linear model, Akaike Information Criteria (AIC) and Bayesian Information Criteria (BIC) are used. Both AIC and BIC are used to penalize linear models for including too many parameters. This is particularly valuable since in our examples Taylor expansion can be used almost indefinitely for nonlinear models. The AIC and BIC are calculated for each model, then the model with smallest AIC and BIC, respectively, are chosen as the *best* linear model. The integrated MSE and variance estimator for the best AIC and best BIC model at each simulation iteration are then summed as above, and the average is taken at the conclusion of $M$ simulations. The AIC and BIC are the minimization with respect to $p$ of

$$AIC = -2(\text{maximum log-likelihood}) + 2 * p$$

$$BIC = -2(\text{maximum log-likelihood}) + \ln(n) * p$$

  where $p$ parameters in the model and $n$ is the number of observations. Under the normal error model, (Burnham and Anderson 2002) the AIC and BIC can be computed by

$$AIC = n \ln(\frac{RSS}{n}) + 2 * p$$

$$BIC = n\ln(\frac{RSS}{n}) + \ln(n) * p$$

where $RSS$ is the standard Residual Sum of Squares.

- Once the linear models and the comparative measures have been calculated, the single-index model will be fitted. The details of calculating the SIM are provided in Chapter 2. Using a Projection Pursuit Regression routine (*ppr* in **R**) with the number of terms set to 1, a $\hat{\theta}$ is calculated. Upon having an estimate for the projection vector, a kernel smoothing routine can be used to estimate the univariate function $h$. First we find a suitable bandwidth for the kernel smoothing using the Cross-Validation Criterion (*h.select* in **R**) discussed in Chapter 2. Once projection vector and bandwidth are found, kernel smoothing will result in a set of predictor variables:

  $$\hat{Y} = S_a Y$$

  where $S_a$ is the smoothing matrix resulting from input values $\hat{\theta}^T x_i$ and our observed $Y_i$'s, with bandwidth $a$. In this project, we follow the routine *ksmooth* in **R** to perform our kernel smoothing. This routine takes the specified bandwidth and scales the kernels so their quartiles are at +/- '0.25' times the specified bandwidth. That is,

  $$a^* = \frac{0.25 * a}{z_{0.750}}$$

  where $P\{Z < z_{0.750}\} = 0.75$ for the r.v. $Z \sim$N(0,1). This adjustment is made in the *ksmooth* routine for boundary correctness. To compensate, when computing our smoothing matrix, we actually find $S_{a^*}$ and use it to approximate the degrees of freedom for the kernel smoothing. We use the Gaussian kernel function as the $K$ function.

- Once the projection vector and univariate function are approximated, an integrated MSE is then calculated

  $$\gamma_k^{SIM} = \frac{1}{n}\sum_{i=1}^{n}\left(\hat{h}(\hat{\theta}^T x_i) - h(\theta^T x_i)\right)^2$$

  where $k$ is for the $k^{\text{th}}$ simulation, $1 \leq k \leq M$. As with the linear model, at the conclusion of the simulation runs, we calculate the average mean squared error as

  $$\bar{\gamma}^{SIM} = \frac{1}{M}\sum_{k=1}^{M}\gamma_k^{SIM}.$$

This value will then be compared to the calculated average MSEs for the linear models. Likewise, an estimator for the variance of error will be calculated using (3). The estimator for $\sigma^2$ is then compared to the known value using the squared difference

$$d_k(\hat{\sigma}^2_{SIM}, \sigma^2) = \left(\hat{\sigma}^2_{SIM} - \sigma^2\right)^2$$

where $k$ represents the $k^{\text{th}}$ simulation iteration. Following completion of $M$ simulations, the average distance is calculated by

$$\bar{d}(\hat{\sigma}^2_{SIM}, \sigma^2) = \frac{1}{M}\sum_{k=1}^{M} d_k(\hat{\sigma}^2_{SIM}, \sigma^2).$$

This average distance will then be compared to that of the linear models.

- At the conclusion of $M$ simulations, we will compare the performance of the linear modeling techniques, with the performance of the single-index model. This is done by a simple ratio calculation:

$$\text{ratio}(\gamma^{j*}) = \bar{\gamma}^{j*}/\bar{\gamma}^{SIM};\ \text{for } j = 1, \ldots, 10.$$

A value greater than 1, would indicate the single-index model performed better than the linear model in estimating the mean function, on average. Likewise, a similar ratio for variance is computed:

$$\text{ratio}(d(\hat{\sigma}^2_{j*}, \sigma^2)) = \bar{d}(\hat{\sigma}^2_{j*}, \sigma^2)/\bar{d}(\hat{\sigma}^2_{SIM}, \sigma^2);\ \text{for } j = 1, \ldots, 10.$$

If the ratio is greater than 1, this would indicate the single-index model performed better than the linear model in estimating the variance for the error term.

- During the $M$ simulations, we also want to count the number of times a glm model performs better than the single-index model. That is, during any given iteration $k$, if $\gamma_k^{j*} < \gamma_k^{SIM}$ for some $j$, then model $j$ performed better than the single-index model in predicting the mean function. Likewise, if $d_k(\sigma^2_{j*}, \sigma^2) < d_k(\sigma^2_{SIM}, \sigma^2)$ for some $j$, then model $j$ performed better than the single-index model in predicting the variance of our error term.

# 4 Simulation Results

A series of simulations are performed to compare the performance of the single-index model to the linear modeling techniques.

## 4.1 Simulation for mean function $h(u) = u(\sin(4\pi u)+1.5)$

In this subsection, we provide simulation results on the mean function, $h(u) = u(\sin(4\pi u) + 1.5)$. Over the interval (-1.5, 1.5), this mean function has the form shown in Figure 1.

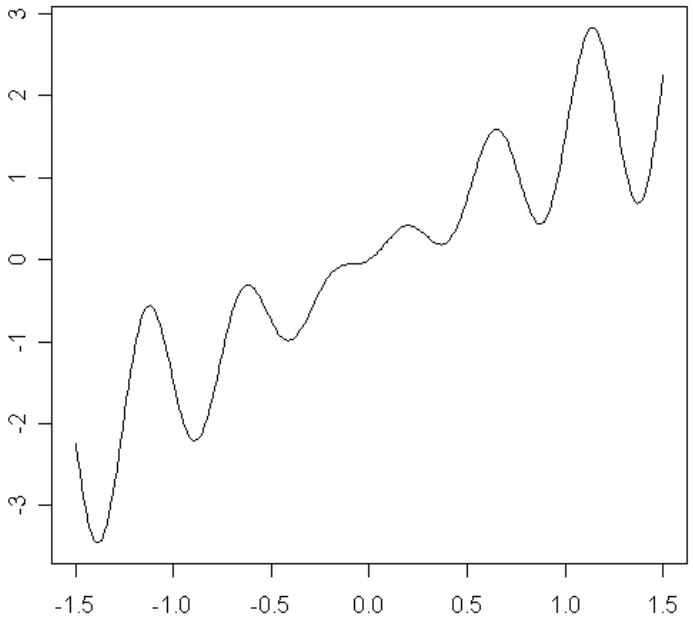


Figure 1: Mean Function $h(u) = u(\sin(4\pi u) + 1.5)$

### 4.1.1 Graphical Analysis

We first perform some graphical analysis for this particular mean function.

- $X = (x_1, x_2)^T$ are fixed such that $\theta^T x$ results in a sequence of fixed $u$ values, and hence $h(u)$ is fixed.
- The true index vector $\theta = (\frac{\sqrt{3}}{2}, \frac{1}{2})^T$

- The error term $\epsilon$ is normally distributed with $\mu = 0$ and $\sigma^2 = 0.64$
- $M = 50$ simulations will be run, with a sample size $n = 101$.
- The SIM is computed for each simulation iteration based on the observed $Y$'s and the fixed $X$. The average for each $\hat{Y} = SY$ will be calculated to visually inspect the accuracy of the SIM and the graph of the average approximated SIM and the fixed x values are seen in the figure 2 below. Approximations for the MSE and variance are calculated.
- This graphical analysis can be thought of as analogous to the numerical comparisons below. However it should be noted that since the $x$'s are fixed in this graphically study, numerical estimations are not consistent with the numerical comparison performed in the next subsection.

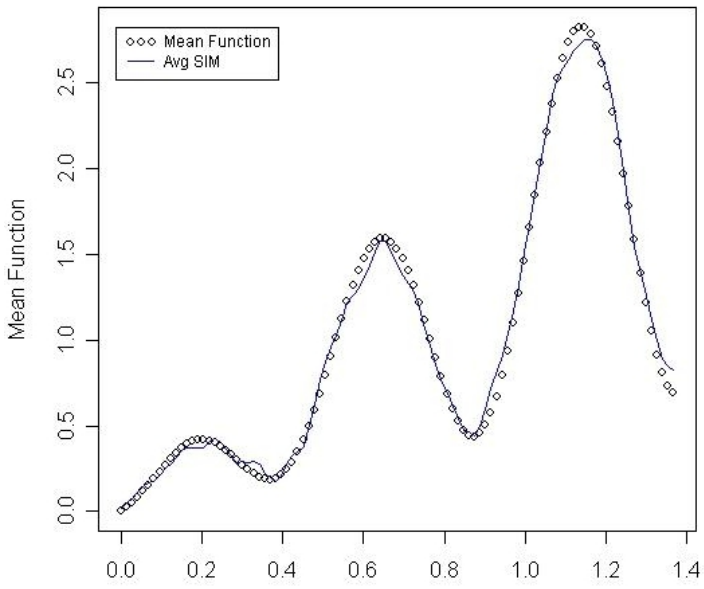


Figure 2: Average SIM estimation

As seen in Figure 2, the SIM (plotted with a solid line) appears to capture the signal (plotted by points). In fact, the averaged (over 50 simulations) mean squared error is 0.1204063. The SIM also does a decent job of estimating the variance of the error term. The average distance between the estimated variance, and the known variance is 0.01195378 over 50 simulations.

In the next graphical analysis, we see how the SIM performs over the interval $(-\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})$.

- As before, $X = (x_1, x_2)^T$ are fixed such that $\theta^T x$ results in a sequence of fixed $u$ values, and hence $h(u)$ is fixed.
- The true index vector $\theta = (\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}})^T$
- The error term $\epsilon$ is normally distributed with $\mu = 0$ and $\sigma^2 = 0.64$
- $M = 50$ simulations are run, with a sample size $n = 101$.
- The SIM is computed for each simulation iteration based on the observed $Y$'s and the fixed $X$. The average for each $\hat{Y} = SY$ will be graphed to visually inspect the accuracy of the SIM. Approximations for the MSE and variance are calculated.
- The graph of the average approximated SIM and the fixed $x$ values are seen in Figure 3 below.

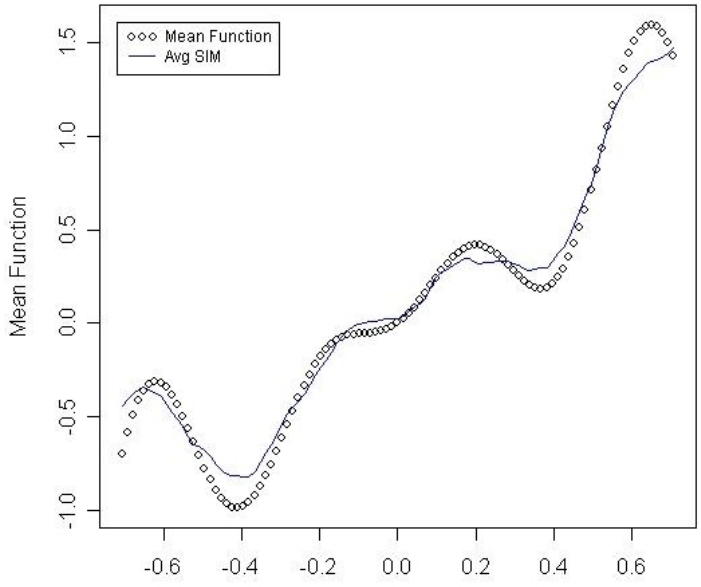


Figure 3: Average SIM estimation

As the image shows, the SIM does a reasonable job of capturing the oringal signal. Visually, it does not seem to perform as well as the previous

graph; however the averaged mean squared error decreases to 0.07803931. This decrease is attributed to the range of the function over this interval. In the previous image, although visually the SIM appears to do as well of a job, the difference in the MSE is attributed to the increase in the function as the x-values increase. That is, as $\theta^T x$ grows, $h$ grows, and the accuracy of the SIM decreases. The SIM also does a decent job of estimating the variance of the error term. The average distance between the estimated variance and the known variance is 0.01257754 over 50 simulations, indicating no difference between the two graphical simulations in the accuracy between the approximated variance.

### 4.1.2 Numerical Comparisons against GLM

In this section we compare the SIM to the GLM numerically. Tables 1 and 2 give the results of this particular simulation. Table 1 gives the ratio of the performance of the linear modeling technique to the single-index model. That is, the first column comprises

$$\bar{\gamma}^{j*}/\bar{\gamma}^{SIM}, \text{ for } j = 1, \ldots, 10$$

and the second column gives

$$\bar{d}(\hat{\sigma}^2_{j*}, \sigma^2)/\bar{d}(\hat{\sigma}^2_{SIM}, \sigma^2), \text{ for } j = 1, \ldots, 10.$$

A value greater than one suggests the single-index model performed better. The model numbers correspond to the degree of the Taylor expansion, that is, model 3 includes cubed terms. The AIC and BIC models represent the models that are selected based on the respective criteria. Table 2 provides counting statistics that show how many simulations had a linear model perform better.

- $X = (X_1, X_2)^T$ with $X_1$, $X_2$ ~Uniform[0,1]
- The true index vector $\theta = (\frac{\sqrt{3}}{2}, \frac{1}{2})^T$
- The error term $\epsilon$ is normally distributed with $\mu = 0$ and $\sigma^2 = 0.16$
- $M = 500$ simulations, sample size $n = 67$.

| Model | MSE Ratio (GLM/SIM) | Variance Ratio (GLM/SIM) |
|---|---|---|
| 1 | 4.5621 (0.2615/0.0573) | 22.4073 (0.0793/0.0035) |
| 2 | 4.3388 (0.2487/0.0573) | 21.1126 (0.0748/0.0035) |
| 3 | 3.8691 (0.2218/0.0573) | 17.5334 (0.0621/0.0035) |
| 4 | 3.3747 (0.1934/0.0573) | 13.7897 (0.0488/0.0035) |
| 5 | 2.2641 (0.1298/0.0573) | 5.0659 (0.0179/0.0035) |
| 6 | 1.5425 (0.0884/0.0573) | 1.0740 (0.0038/0.0035) |
| 7 | 1.6470 (0.0944/0.0573) | 0.6087 (0.0022/0.0035) |
| 8 | 1.9021 (0.1090/0.0573) | 0.6929 (0.0025/0.0035) |
| 9 | 2.3236 (0.1332/0.0573) | 1.2170 (0.0043/0.0035) |
| 10 | 2.7730 (0.1589/0.0573) | 17.2299 (0.0610/0.0035) |
| AIC | 2.7730 (0.1589/0.0573) | 17.2299 (0.0610/0.0035) |
| BIC | 2.8790 (0.1650/0.0573) | 6.4809 (0.0227/0.0035) |

Table 1: GLM vs. SIM

| | Number of Simulation |
|---|---|
| Better GLM MSE | 57 |
| Better GLM VAR | 348 |
| Better AIC MSE | 13 |
| Better AIC VAR | 50 |
| Better BIC MSE | 13 |
| Better BIC VAR | 50 |

Table 2: # Times GLM Performed Better

The following observations were made:

- On average, the SIM performed better than the GLM in estimating the mean function.
- Linear Models 7 and 8 performed better than the SIM in estimating the variance of the error term.
- Using the AIC and BIC for model selection tends to select models 10 and 9, respectively; the SIM performs better than the AIC and BIC models, on average.
- In 348 simulations, a GLM model estimated the variance more accurately. When the AIC or BIC selection method is used, this number

dropped to 50. Similarly, the number of times a GLM performed better in approximating the mean function decreasing by nearly 80% when AIC or BIC model selection is used.

Now we modify the true index vector to $\theta = (\frac{\sqrt{3}}{2}, -\frac{1}{2})^T$ The following results were observed.

| Model | MSE Ratio (GLM/SIM) | Variance Ratio (GLM/SIM) |
|---|---|---|
| 1 | 1.4246 (0.0601/0.0422) | 3.3263 (0.0049/0.0015) |
| 2 | 1.4878 (0.0627/0.0422) | 3.1651 (0.0047/0.0015) |
| 3 | 1.3652 (0.0576/0.0422) | 2.3336 (0.0034/0.0015) |
| 4 | 1.4069 (0.0593/0.0422) | 1.8199 (0.0027/0.0015) |
| 5 | 1.4883 (0.0628/0.0422) | 1.2596 (0.0019/0.0015) |
| 6 | 1.7000 (0.0717/0.0422) | 1.1005 (0.0016/0.0015) |
| 7 | 2.0994 (0.0885/0.0422) | 1.3123 (0.0019/0.0015) |
| 8 | 2.5716 (0.1084/0.0422) | 1.7609 (0.0026/0.0015) |
| 9 | 3.1439 (0.1326/0.0422) | 3.2377 (0.0048/0.0015) |
| 10 | 3.7725 (0.1591/0.0422) | 39.7704 (0.0587/0.0015) |
| AIC | 3.7688 (0.1590/0.0422) | 32.7886 (0.0492/0.0015) |
| BIC | 3.3586 (0.1417/0.0422) | 7.7563 (0.0116/0.0015) |

Table 3: GLM vs. SIM

| | Number of Simulation |
|---|---|
| Better GLM MSE | 150 |
| Better GLM VAR | 416 |
| Better AIC MSE | 0 |
| Better AIC VAR | 51 |
| Better BIC MSE | 16 |
| Better BIC VAR | 72 |

Table 4: # Times GLM Performed Better

The following observations were made:

- On average, the SIM performed better than the GLM in estimating the mean function and the variance of the error term.
- Using the AIC and BIC for model selection tends to select models 10 and 8, respectively; the SIM performs better than the AIC and BIC models, on average.

- In 416 simulations, a GLM model estimated the variance more accurately. When the AIC or BIC selection method is used, this number dropped to 72 or lower, a drop of 80%. The number of GLM models in estimating the link function, drops by nearly 90%.

Now, we modify the projection vector to $\theta = (\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})^T$ and the results were as follows.

| Model | MSE Ratio (GLM/SIM) | Variance Ratio (GLM/SIM) |
|---|---|---|
| 1 | 5.6013 (0.2839/0.0507) | 44.7234 (0.0869/0.0019) |
| 2 | 5.4328 (0.2754/0.0507) | 44.4596 (0.0864/0.0019) |
| 3 | 4.8652 (0.2466/0.0507) | 38.0543 (0.0739/0.0019) |
| 4 | 4.3633 (0.2212/0.0507) | 32.5107 (0.0632/0.0019) |
| 5 | 2.6070 (0.1321/0.0507) | 8.9024 (0.0173/0.0019) |
| 6 | 2.0149 (0.1021/0.0507) | 3.5829 (0.0070/0.0019) |
| 7 | 1.8450 (0.0935/0.0507) | 1.2041 (0.0023/0.0019) |
| 8 | 2.1123 (0.1071/0.0507) | 1.2685 (0.0025/0.0019) |
| 9 | 2.5693 (0.1302/0.0507) | 2.3070 (0.0045/0.0019) |
| 10 | 3.0906 (0.1566/0.0507) | 27.4264 (0.0533/0.0019) |
| AIC | 3.0906 (0.1566/0.0507) | 27.4264 (0.0533/0.0019) |
| BIC | 3.2601 (0.1653/0.0507) | 11.2887 (0.0214/0.0019) |

Table 5: GLM vs. SIM

| | Number of Simulation |
|---|---|
| Better GLM MSE | 26 |
| Better GLM VAR | 337 |
| Better AIC MSE | 5 |
| Better AIC VAR | 45 |
| Better BIC MSE | 5 |
| Better BIC VAR | 46 |

Table 6: # Times GLM Performed Better

The following observations were made:

- On average, the SIM performed better than the GLM in estimating the mean function and the variance of the error term.

- Using the AIC and BIC for model selection tends to select models 10 and 9 respectively, the SIM performs better than the AIC and BIC models, on average.
- In 337 simulations, a GLM model estimateded the variance more accurately. When the AIC or BIC selection method is used, this number dropped to 46 or lower, a drop of 86%. The number of GLM models in estimating the link function, drops by nearly 80%.

Lastly, we compare the SIM to the GLM using $\theta = (\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}})^T$, where our $\theta^T x_i$ will be distributed on $(-\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})$, symmetric about the y-axis.

| Model | MSE Ratio (GLM/SIM) | Variance Ratio (GLM/SIM) |
|---|---|---|
| 1 | 1.3251 (0.0482/0.0364) | 2.8343 (0.0030/0.0010) |
| 2 | 1.4155 (0.0515/0.0364) | 2.7116 (0.0028/0.0010) |
| 3 | 1.5493 (0.0564/0.0364) | 2.6038 (0.0027/0.0010) |
| 4 | 1.2527 (0.0456/0.0364) | 1.1398 (0.0012/0.0010) |
| 5 | 1.5862 (0.0577/0.0364) | 1.3081 (0.0014/0.0010) |
| 6 | 1.9636 (0.0714/0.0364) | 1.4527 (0.0015/0.0010) |
| 7 | 2.4429 (0.0889/0.0364) | 1.6499 (0.0017/0.0010) |
| 8 | 2.9957 (0.1090/0.0364) | 2.2593 (0.0024/0.0010) |
| 9 | 3.6429 (0.1325/0.0364) | 4.1994 (0.0044/0.0010) |
| 10 | 4.3727 (0.1590/0.0364) | 53.7057 (0.0560/0.0010) |
| AIC | 4.3727 (0.1590/0.0364) | 53.7057 (0.0560/0.0010) |
| BIC | 3.6546 (0.1330/0.0364) | 10.1696 (0.0102/0.0010) |

Table 7: GLM vs. SIM

| | Number of Simulation |
|---|---|
| Better GLM MSE | 161 |
| Better GLM VAR | 386 |
| Better AIC MSE | 0 |
| Better AIC VAR | 32 |
| Better BIC MSE | 27 |
| Better BIC VAR | 52 |

Table 8: # Times GLM Performed Better

The following observations were made:

- On average, the SIM performed better than the GLM in estimating the mean function and the variance of the error term.
- Using the AIC and BIC for model selection tends to select models 10 and 8, respectively; the SIM performs better than the AIC and BIC models, on average.
- In 386 simulations, a GLM model estimated the variance more accurately. When the AIC or BIC selection method is used, this number dropped to 52 or lower, a drop of 86%. The number of GLM models in estimating the link function, drops by nearly 83%.

### 4.1.3 Conclusions

Based on the two sets of analysis, we conclude that the SIM is at least as good, if not better, as an estimator for the mean function $u(\sin(4\pi u) + 1.5)$. The linear approach typically performed its best around models 4-7. As the number of parameters increases, these model estimates will be infeasible to compute and the SIM will be a viable alternative. The value of the unique $\theta$ did not seem to effect the performance in the SIM, i.e. the SIM did not struggle with any specific $\theta$.

## 4.2 Simulation for a Logistic mean function

Here we perform a simulation study when the mean function is logistic, i.e. $h(u) = \frac{2}{1+e^3 e^{-5u}} + 1$. The mean function has the familar *S-shaped* form seen in Figure 4.

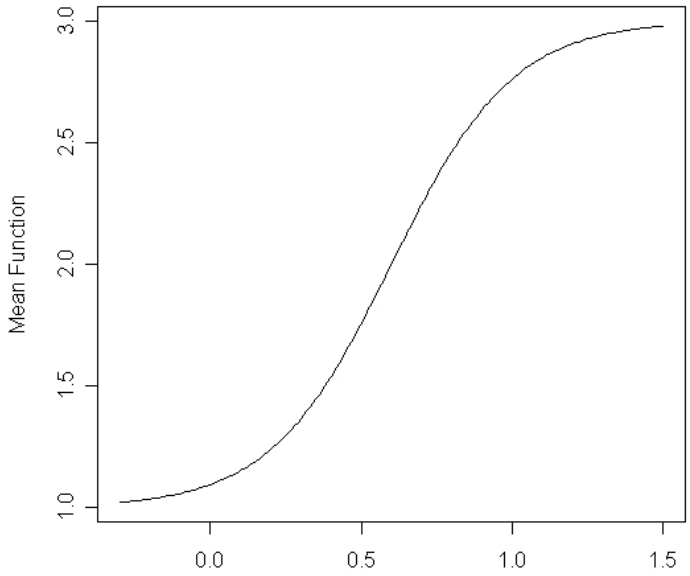


Figure 4: Logistic Mean Function $h$

### 4.2.1 Graphical Analysis

We first perform a visual inspection of the performance of the SIM.

- $X = (x_1, x_2)^T$ are fixed such that $\theta^T x$ results in a sequence of fixed $u$ values, and hence $h(u)$ is fixed.
- The true index vector $\theta = (\frac{\sqrt{3}}{2}, \frac{1}{2})^T$.
- The error term $\epsilon$ is normally distributed with $\mu = 0$ and $\sigma^2 = 0.64$.
- $M = 50$ simulations are run, with a sample size $n = 67$.
- The SIM is computed for each simulation iteration based on the observed $Y$'s and the fixed $X$. The average for each $\hat{Y} = SY$ will be graphed to visually inspect the accuracy of the SIM. Approximations for the MSE and variance are calculated.
- The graph of the average approximated SIM, the *best* linear model and the fixed x values are shown in Figure 5 below.

We observe the *best* linear model is the simple linear regression, that is, a first order Taylor expansion on our predictor variables. Although visually,

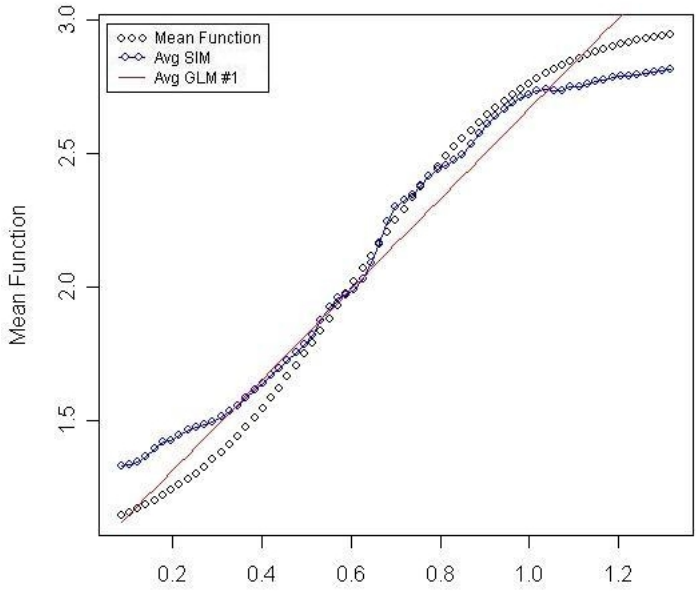


Figure 5: Average SIM and GLM estimation

the line does not capture the signal, it performs better numerically than the SIM. Visually, the SIM appears to capture the trend of the mean function. The SIM appears to be very accurate near the center of the mean function, but struggles near the tails, the location of the horizontal asymptotes. This seemingly allows the linear model to beat out the SIM in the numerical simulation performed below.

### 4.2.2 Numerical Analysis

- $X = (X_1, X_2)^T$ with $X_1$, $X_2$ ~Uniform[0,1].
- The true index vector $\theta = (\frac{\sqrt{3}}{2}, \frac{1}{2})^T$.
- The error term $\epsilon$ is normally distributed with $\mu = 0$ and $\sigma^2 = 0.16$.
- $M = 500$ simulations are run, each time our sample size is $n = 67$.

We observe that the *best* linear model is the degree 1 expansion, or a typical linear regression, this backs up the graphical claim. On average, the linear model performs better than the SIM, but not signifcantly better. Particularly when AIC or BIC, models 10 and 7.5 respectively, are used in the model selection, the SIM beats the linear models.

| Model | MSE Ratio (GLM/SIM) | Variance Ratio (GLM/SIM) |
|---|---|---|
| 1 | 0.5032 (0.0143/0.0284) | 0.7831 (0.0008/0.0010) |
| 2 | 0.6359 (0.0180/0.0284) | 0.8074 (0.0008/0.0010) |
| 3 | 0.8261 (0.0234/0.0284) | 0.7853 (0.0008/0.0010) |
| 4 | 1.2468 (0.0354/0.0284) | 0.8520 (0.0009/0.0010) |
| 5 | 1.7434 (0.0494/0.0284) | 0.9980 (0.0010/0.0010) |
| 6 | 2.3157 (0.0657/0.0284) | 1.1994 (0.0013/0.0010) |
| 7 | 2.9700 (0.0842/0.0284) | 1.5288 (0.0016/0.0010) |
| 8 | 3.7619 (0.1067/0.0284) | 2.1434 (0.0022/0.0010) |
| 9 | 4.6321 (0.1314/0.0284) | 3.7209 (0.0039/0.0010) |
| 10 | 5.5055 (0.1562/0.0284) | 45.5301 (0.0476/0.0010) |
| AIC | 5.5055 (0.1562/0.0284) | 45.5301 (0.0476/0.0010) |
| BIC | 4.2470 (0.1206/0.0284) | 10.1006 (0.0101/0.0010) |

Table 9: GLM vs. SIM

| | Number of Simulation |
|---|---|
| Better GLM MSE | 383 |
| Better GLM VAR | 435 |
| Better AIC MSE | 0 |
| Better AIC VAR | 51 |
| Better BIC MSE | 96 |
| Better BIC VAR | 125 |

Table 10: # Times GLM Performed Better

### 4.2.3 Conclusion

We conclude that overall, the SIM does not perform as well as the GLM, however it is not significantly worse. As will be discussed in the following sections, the SIM's performance improves when a better estimate for the projection vector $\theta$ is used. The SIM should be considered as an alternative to the GLM for a logistic function. This should particularly be true when the domain is wide.

## 4.3 Simulation to provide motivation for predicting more accurate $\theta$

In this subsection, we perform an abbreviated simulation similar to the simulations above. Analyzing the following mean functions and simulations will demonstrate, by example, the importance of predicting the projection vector $\theta$, and therefore provide motivation for estimating a more accurate projection vector.

### 4.3.1 Mean function $h(u) = \sin^2(2\pi u) + 1$, unknown $\theta$

In this subsection, we supply simulation results on the mean function, $h(u) = \sin^2(2\pi u) + 1$. Over the interval (-1.5, 1.5), this mean function has the form in Figure 6.

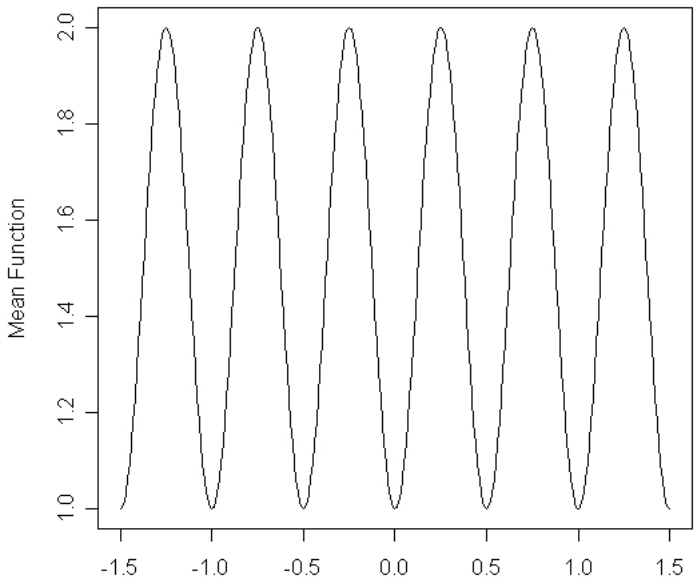


Figure 6: Mean Function $h(u) = \sin^2(2\pi u) + 1$

We note the high number of oscillations and suspect the kernel smoothing routine will struggle in capturing the mean function. Furthermore, we suspect a linear model bisecting ($\hat{Y} = \bar{Y}$) the range of the function will do a reasonable job numerically, in predicting the mean function and variance.

We first perform a graphical analysis. Here we fix our $x$-values and plot the original mean function, comparing it to the average predicted mean function by linear model #6 (typically the best numerically) and the SIM. As

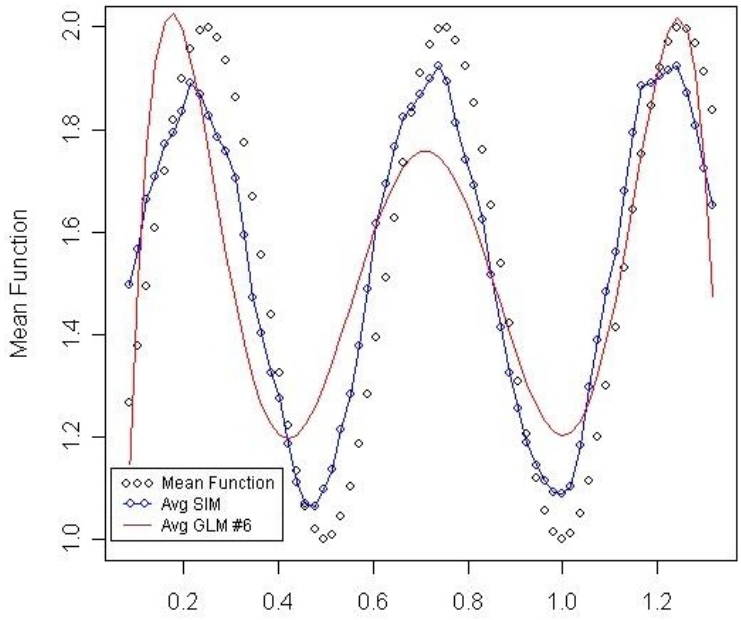


Figure 7: Average Sim and GLM estimates

can be seen in figure 7 graph, the GLM captures the first and last relative maximums fairly well and bisects the center oscillation resulting in a *good* numeric estimate for the signal and variance. The kernel smoother used in estimating the mean function in the SIM captures the overall trend of the mean functionbut there is a large bias. The estimated function is under-smoothed, and appears shifted to the left of the actual mean function. These two properties result in the inferior performance in the numeric comparisons.

- $X = (X_1, X_2)^T$ with $X_1$, $X_2$ ~Uniform[0,1].
- The true index vector $\theta = (\frac{\sqrt{3}}{2}, \frac{1}{2})^T$.
- The error term $\epsilon$ is normally distributed with $\mu = 0$ and $\sigma^2 = 0.16$.
- $M = 500$ simulations are run, each time our sample size is $n = 67$.

Results are provided in Table 11 and 12.

We see that the GLM predicts both the MSE and Variance better than the SIM. This can be attributed to the ocscillations in the mean function. That is, since the mean function has three complete periods, or oscillations in the interval $(0, \frac{\sqrt{3}}{2})$, a linear model can obtain a reasonable MSE or $\hat{\sigma}^2$ by

| Model | MSE Ratio (GLM/SIM) | Variance Ratio (GLM/SIM) |
|---|---|---|
| 1 | 1.2879 (0.1266/0.0983) | 1.7345 (0.0182/0.0105) |
| 2 | 1.2799 (0.1259/0.0983) | 1.6880 (0.0177/0.0105) |
| 3 | 1.2563 (0.1235/0.0983) | 1.5292 (0.0160/0.0105) |
| 4 | 1.1450 (0.1126/0.0983) | 1.1609 (0.0122/0.0105) |
| 5 | 0.8864 (0.0872/0.0983) | 0.4563 (0.0048/0.0105) |
| 6 | 0.8140 (0.0800/0.0983) | 0.1980 (0.0021/0.0105) |
| 7 | 0.9028 (0.0888/0.0983) | 0.1631 (0.0017/0.0105) |
| 8 | 1.1035 (0.1085/0.0983) | 0.2265 (0.0024/0.0105) |
| 9 | 1.3530 (0.1330/0.0983) | 0.3976 (0.0042/0.0105) |
| 10 | 1.6237 (0.1597/0.0983) | 5.6377 (0.0592/0.0105) |
| AIC | 1.6235 (0.1596/0.0983) | 5.0148 (0.0527/0.0105) |
| BIC | 1.5851 (0.1558/0.0983) | 1.3408 (0.0141/0.0105) |

Table 11: GLM vs. SIM

| | Number of Simulation |
|---|---|
| Better GLM MSE | 363 |
| Better GLM VAR | 463 |
| Better AIC MSE | 29 |
| Better AIC VAR | 151 |
| Better BIC MSE | 33 |
| Better BIC VAR | 155 |

Table 12: # Times GLM Performed Better

simply bisecting the function with respect to the range of the mean function. This can be seen in the graphical analysis above.

#### 4.3.2 Mean function $h(u) = \sin^2(2\pi u) + 1$, known $\theta$

In performing these simulations, the projection vector $\theta$ is *known*. That is, we do not use the PPR routine, or any other method of estimating a $\theta$. In essence, we are comparing the kernel smoothing technique to that of linear modeling for this particular function.

We first perform a visual comparison, shown in Figure 8. Visually, the SIM appears to predict the mean function almost exactly, while the linear model #6 deviates from the mean function. It should be noted that this particular GLM is not identical to the previous GLM graph in Figure 7, as

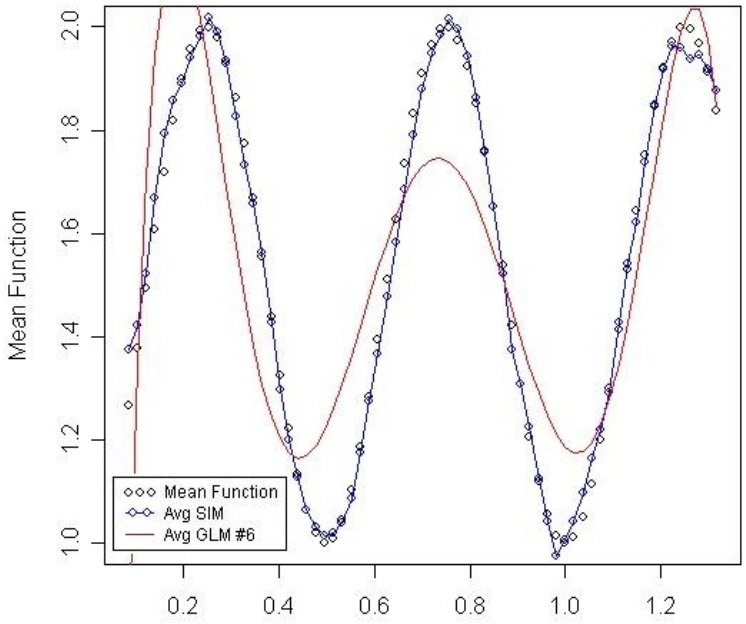


Figure 8: Average Sim and GLM estimates

variability in the observed $Y$'s alters the average GLM, and hence the graph.

We now do a numerical comparison.

- $X = (X_1, X_2)^T$ with $X_1$, $X_2$ ~Uniform[0,1]
- The true index vector $\theta = (\frac{\sqrt{3}}{2}, \frac{1}{2})^T$, however it is *known* and no longer estimated with the PPR routine.
- The error term $\epsilon$ is normally distributed with $\mu = 0$ and $\sigma^2 = 0.16$.
- $M = 500$ simulations are run, each time our sample size is $n = 67$.

| Model | MSE Ratio (GLM/SIM) | Variance Ratio (GLM/SIM) |
|---|---|---|
| 1 | 3.2939 (0.1259/0.0382) | 15.8286 (0.0181/0.0011) |
| 2 | 3.2625 (0.1247/0.0382) | 15.1976 (0.0174/0.0011) |
| 3 | 3.2161 (0.1230/0.0382) | 13.9723 (0.0160/0.0011) |
| 4 | 2.9210 (0.1117/0.0382) | 10.7494 (0.0123/0.0011) |
| 5 | 2.2705 (0.0868/0.0382) | 4.4383 (0.0051/0.0011) |
| 6 | 2.0859 (0.0798/0.0382) | 1.9777 (0.0023/0.0011) |
| 7 | 2.3250 (0.0889/0.0382) | 1.4493 (0.0017/0.0011) |
| 8 | 2.8696 (0.1097/0.0382) | 1.9942 (0.0023/0.0011) |
| 9 | 3.4905 (0.1335/0.0382) | 3.5504 (0.0041/0.0011) |
| 10 | 4.1636 (0.1592/0.0382) | 44.7675 (0.0513/0.0011) |
| AIC | 4.1636 (0.1592/0.0382) | 44.7675 (0.0513/0.0011) |
| BIC | 4.0802 (0.1559/0.0382) | 12.1516 (0.0134/0.0011) |

Table 13: GLM vs. SIM

The SIM performance increases by a factor of three with respect to the mean function estimator, and a factor of 10 with respect to the variance estimator. The SIM performs better than all the linear models when $\theta$ is known.

### 4.3.3 Mean function $h(u) = \sin(2\pi u) + 1.5$, unknown $\theta$

Following the previous simulation, we provide motivation to estimating a more accurate $\theta$. The mean function, $h(u) = \sin(2\pi u) + 1.5$ has the familar form in Figure 9. We will estimate the projection vector, using the PPR routine discussed in chapter 2.

We first perform a visual inspection of the SIM's performance, figure 10.

- $X = (x_1, x_2)^T$ are fixed such that $\theta^T x$ results in a sequence of fixed $u$ values, and hence $h(u)$ is fixed.
- The true index vector $\theta = (\frac{\sqrt{3}}{2}, \frac{1}{2})^T$.
- The error term $\epsilon$ is normally distributed with $\mu = 0$ and $\sigma^2 = 0.64$.
- $M = 50$ simulations are run, with a sample size $n = 101$.

We observe that the SIM does a good job capturing the *trend* of the mean function. That is, our SIM graph looks like a sine function, however, the SIM is shifted to the left and missing each respective fixed point, $h(\theta^T x_i)$.

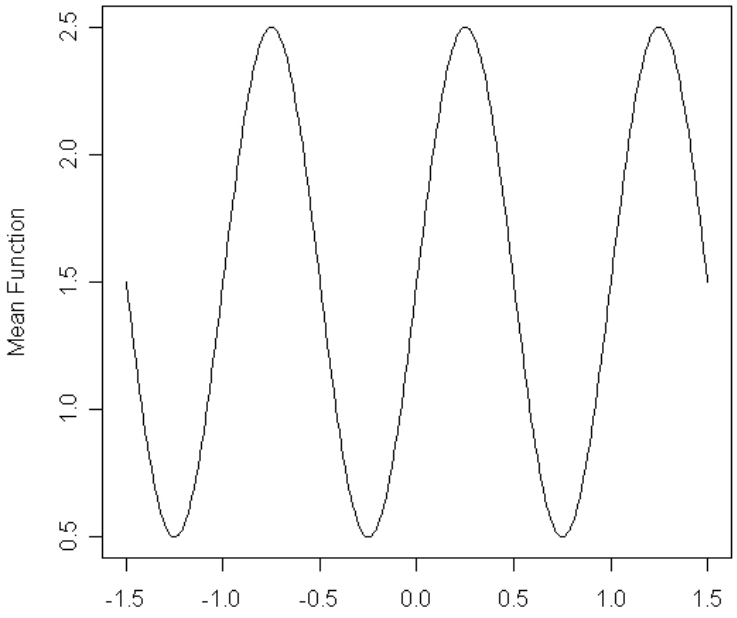


Figure 9: Mean Function $h(u) = \sin(2\pi u) + 1.5$

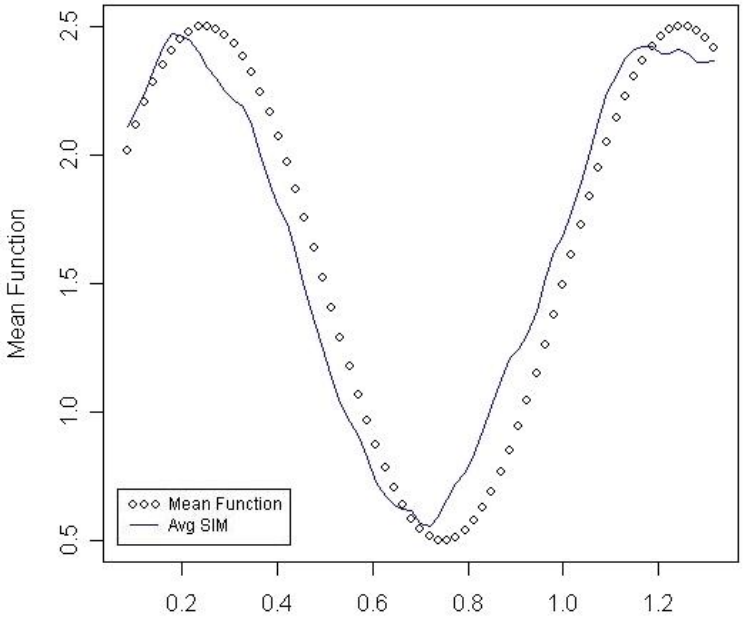


Figure 10: Average SIM estimation

Now we perform a numerical comparison between the GLM and the SIM.

- $X = (X_1, X_2)^T$ with $X_1$, $X_2$ ~Uniform[0,1].
- The true index vector $\theta = (\frac{\sqrt{3}}{2}, \frac{1}{2})^T$.
- The error term $\epsilon$ is normally distributed with $\mu = 0$ and $\sigma^2 = 0.16$.
- $M = 500$ simulations are run, each time our sample size is $n = 67$.

| Model | MSE Ratio (GLM/SIM) | Variance Ratio (GLM/SIM) |
|---|---|---|
| 1 | 3.8491 (0.4466/0.1160) | 5.1481 (0.2227/0.0433) |
| 2 | 1.1342 (0.1316/0.1160) | 0.4734 (0.0205/0.0433) |
| 3 | 0.7758 (0.0900/0.1160) | 0.1959 (0.0085/0.0433) |
| 4 | 0.3443 (0.0399/0.1160) | 0.0232 (0.0010/0.0433) |
| 5 | 0.4455 (0.0517/0.1160) | 0.0248 (0.0011/0.0433) |
| 6 | 0.5776 (0.0670/0.1160) | 0.0291 (0.0013/0.0433) |
| 7 | 0.7410 (0.0860/0.1160) | 0.0346 (0.0015/0.0433) |
| 8 | 0.9289 (0.1078/0.1160) | 0.0517 (0.0022/0.0433) |
| 9 | 1.1423 (0.1325/0.1160) | 0.0924 (0.0040/0.0433) |
| 10 | 1.3695 (0.1589/0.1160) | 0.9238 (0.0400/0.0433) |
| AIC | 1.3695 (0.1589/0.1160) | 0.9238 (0.0400/0.0433) |
| BIC | 1.2442 (0.1443/0.1160) | 0.2582 (0.0112/0.0433) |

Table 14: GLM vs. SIM

We see that, on average, the GLM dominates the SIM both with respect to the estimation of the MSE and the error variance. If the AIC and BIC criteria are used, the SIM's performs better in estimating the mean function, but still loses out with respect to the variance estimator.

#### 4.3.4 Mean function $h(u) = \sin(2\pi u) + 1.5$, known $\theta$

As in above analysis of the sine-squared mean function, the projection vector $\theta$ is known in this analysis. That is, we no longer estimate it using the PPR algorithm. Graphically, the SIM's performance increases dramatically as seen in figure 11. When $\theta$ was unknown, our SIM misses the mean function consistently. However when the known $\theta$ is used, the SIM no longer just captures the trend of the mean function, but captures the mean function itself to a very high degree. The numerical analysis below supports this claim.

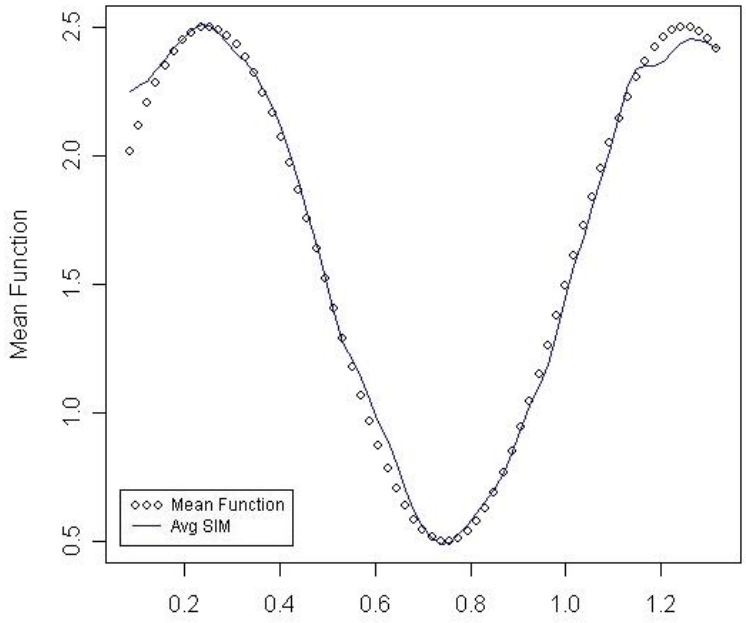


Figure 11: Average SIM estimation

Performing the same numeric simulation as above, except $\theta$ is known, results are in Tables 15 and 16: We observe from the numeric data in

| Model | MSE Ratio (GLM/SIM) | Variance Ratio (GLM/SIM) |
|---|---|---|
| 1 | 13.3515 (0.4416/0.0331) | 225.7493 (0.2144/0.0009) |
| 2 | 3.9869 (0.1319/0.0331) | 21.1651 (0.0201/0.0009) |
| 3 | 2.6593 (0.0880/0.0331) | 8.2259 (0.0078/0.0009) |
| 4 | 1.1483 (0.0380/0.0331) | 1.0317 (0.0010/0.0009) |
| 5 | 1.4912 (0.0493/0.0331) | 1.0905 (0.0010/0.0009) |
| 6 | 1.9761 (0.0654/0.0331) | 1.3279 (0.0013/0.0009) |
| 7 | 2.5617 (0.0847/0.0331) | 1.7336 (0.0016/0.0009) |
| 8 | 3.2054 (0.1060/0.0331) | 2.3145 (0.0022/0.0009) |
| 9 | 3.9184 (0.1296/0.0331) | 4.0672 (0.0039/0.0009) |
| 10 | 4.6848 (0.1550/0.0331) | 50.1703 (0.0476/0.0009) |
| AIC | 4.6848 (0.1550/0.0331) | 50.1703 (0.0476/0.0009) |
| BIC | 4.3415 (0.1437/0.0331) | 13.0491 (0.0117/0.0009) |

Table 15: GLM vs. SIM

| | Number of Simulation |
|---|---|
| Better GLM MSE | 153 |
| Better GLM VAR | 397 |
| Better AIC MSE | 0 |
| Better AIC VAR | 37 |
| Better BIC MSE | 7 |
| Better BIC VAR | 63 |

Table 16: # Times GLM Performed Better

Tables 15 and 16 that the SIM performs better, on average, than the GLM in all 10 models. Using the AIC and BIC criteria furthers this claim. In 500 simulations, only 7 BIC models did a better job estimating the mean function, and only 63 did better in estimating the variance of the error term.

### 4.3.5 Conclusions

These particular examples demonstrate that the estimation of the projection vector, $\theta$, is very important in getting an accurate SIM. That is, the kernel smoothing routine does a suitable job in estimating the univariate mean function $h$, when a unique $\theta$ has been accurately estimated. These examples provide motivation to estimate the unique $\theta$ as precise as possible, rather than just a compatible projection vector as the PPR routine provides.

# 5 Conclusion and Future Research

In this project, through a series of simulations, we demonstrated the single-index model as a viable alternative to linear modeling techniques. This is particularly the case for highly nonlinear models. In the future, the simulations will be expanded to include three and four-dimensional predictor vectors. With such expansion, it is hypothesized the SIM will be a viable option for many more functions, particularly those that require many terms in the Taylor expansion. With three or four predictor variables, the number of covariate terms in a Taylor expansion will grow exponentially, and such linear estimations will become infeasible. The SIM will then be an adequate estimate.

This project also provides motivation for estimating the projection, or index vector, more accurately. Simulation results demonstrate the importance of estimating an accurate index vector. In the future, the goal is to address the estimation of the projection vector. A weighted-least squares approach will be used. Currently we are working on a scheme that uses an initial grid search to approximate an initial *guess* projection vector. A nonlinear minimization technique is then used to get a more accurate approximation for the projection vector.

In the present cases, the grid search is limited to previous knowledge about the index vector, that is its dimensionality. A generalization of the grid search is one stated goal. The nonlinear minimization routine does not necessarily guarantee the necessary conditions for our projection vector to be unique and identifiable. That is, the nonlinear minimization tool does not include the constraints that $\left\|\theta\right\| = 1$ and $\theta_1 > 0$. Future work will attempt to use these methods to construct an algorithm that will consistently and accurately approximate the unique index vector.

# References


[1] Akaike, H. (1974), "A new look at the statistical model identification," IEEE Transactions on Automatic Control 19, 716-722.

[2] Burnham, K. P. and Anderson, D. R. (2002), "Model Selection and Multimodel Inference: A Practical Information-Theoretic Approach," second edition, Springer, New York, 60-64, 286.

[3] Friedman, Jerome H. and Stuetzle, Werner (1981), "Projection Pursuit Regression, " Journal of The American Statistical Association 76, 817-823.

[4] Graybill, F.A. (1976), "Theory and Application of the Linear Model." Duxbury Publishing, Pacific Grove.

[5] Hastie, T.J. and Tibshirani, R.J. (1990), "Generalized Additive Models." Chapman & Hall, London 52-54.

[6] Horowitz, Joel L. (1998), "Semiparametric Methods in Econometrics," Springer-Verlag, New York 5-22.

[7] Lin, Wei (2002), "Analysis of Single-Index Models," MS Report, Department of Mathematical Science, Clemson University.

[8] Lin, Wei and Kulasekera, K.B. (2006), "Identifiability of Single-index Models and Additive-Indices Models," To Appear in Biometrika.

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