Analyzing Health Utility Data with Generalized Additive Models

By

Hoi Suen Wong, BSc

A Thesis

Submitted to the School of Graduate Studies

in Partial Fulfilment of the Requirements

for the Degree

Master of Science

McMaster University

MASTER OF SCIENCE — McMaster University

(2011 Statistics) — Hamilton, Ontario

TITLE: Analyzing Health Utility Data with Generalized Additive Models

AUTHOR: Hoi Suen, Wong

B.Sc. (University of Toronto)

SUPERVISOR: Dr. Aaron Childs and Dr. Eleanor Pullenayegum

NUMBER OF PAGES: xi, 80

Contents

List of Tables

List of Figures

Acknowledgements

I would like to say thanks and express my gratitude to my supervisors Dr. Eleanor Pullenayegum and Dr. Aaron Childs for their time and patience throughout my thesis. It would not be possible for the completion of the thesis without their valuable guidance and explanation.

I would like to thank Dr. Roman Viveros and Dr. Amadou Sarr for being the examiners of this project.

I would also like to thank the Diabetes Hamilton group, in particular Dr Hertzel Gerstein, Janet Greb, and also Dr Daria O'Reilly for the study used in our analysis and the CANNeCTIN biostatistics methodology group, for partial funding of the research stipend.

Finally, I would like to thank my grand-parents, parents, and my sisters for their constant support, encouragement, and the time spent with me.

Abstract

The money put into health care has been increasing dramatically. Comparison of different programs is important in helping government to make decision as to what health care services to be provided. The cost-effectiveness of different health care programs can be compared based on the improvement of a person's health states and the cost it incurred. To measure the health states of a person, health utility scores can be used. But health utility data exhibit features such as non-normality, heteroscedasticity of variances, and the majority of observations attaining values close to or at the maximum of the measurement scale. This brings challenges to analyzing health utility data. For example, linear regression with the assumption of normality might not be valid since non-normality is present in health utility data. To address these problems, other methods are used. In this study, we investigate the performance of generalized additive models (GAMs) in handling health utility data. In GAMs, the relationship between the response and the predictor variables can be non-linearly defined. So GAM methods give more options in the assumption of the relationship between the response and the predictors. For comparison, we also use ordinary least squares. To evaluate the performance of generalized additive models, simulation is used. Data are generated from the simulation model, and GAMs will

be used to fit the simulated data. Bias and coverage probability will be used to assess the performance of the GAM method. A comparison between OLS and GAM will be also be done in the study. And, as an illustration, GAM will be applied to a real data set called Diabetes Hamilton which was collected from some diabetic patients who participated in a community program based in Hamilton, Ontario.

When GAM is applied to the real data set, similar results to the OLS method in terms of the estimates of the parameters is observed. Both methods give similar coefficient values for each parameter. From the simulation results, the estimate given by the GAM method is closer to the true value than the OLS method in general. The bias produced by the GAM is smaller than the OLS method. So overall, GAM method seems to be valid in analyzing the data set such as those used in this study and also the general health utility data.

Chapter 1

Introduction

1.1 Motivation

Health utilities are the quality weight used to calculate quality-adjusted life years, the preferred outcome in cost-effectiveness analysis. Different methods have been proposed previously for analyzing health utility data. But due to the characteristics of health utility data, those previously proposed methods might not be appropriate. In this study, generalized additive models are proposed as an alternative in analyzing health utility data. Valid results from the analysis of health utility data can give an indication of the effectiveness of the health care programs being provided.

Due to advancement of technology and science, once incurable diseases have become treatable. More treatments and diagnostic tools have become available for different diseases. But newer or better treatment and diagnostic tools also come with a greater cost. For example, there is an increasing use of MRI machines in both Canada and the U.S.. An MRI machine can cost between U.S. 1 million and 3

million. In 2009, Canada had 266 MRI scanners and the number of exams done was about 41 per 1000 population during 2008 and 2009. Each MRI examination is also very expensive. More drugs and interventions have also become available to people with diabetes. Diabetes is a disorder of the metabolism. There are three types of diabetes. The most common type of diabetes is type-2 diabetes. About 90 to 95 percent of diabetic patients suffer from type-2 diabetes [Shaw et al., 2005]. Type-2 diabetes is a condition when the pancreas of the body can produce enough insulin but for some reason the body cannot make use of the insulin effectively. Glucose will build up in the body and the body cannot make efficient use of the food intake. One of the drugs that is used to treat the type-2 diabetes condition is called Pioglitazone. The drug can cost three to five dollars per pill. It can be a burden to society. Also only some drugs are covered under government programs.

There is only a limited amount of resources available in society to allocate to different needs. In developed countries, people are living longer and longer. The population as a whole has been aging. People aged 65 and above constitute a higher and higher proportion of the population in Canada and the U.S.. Demand for better health care services and easier access have put pressure on the government to allocate resources spent on health care more effectively. For example, in 2009 spending on health care in Canada was about 180 billion with about 128.6 billion being public sector health care spending. Since only limited resources are available, some services or treatments needed to be forgone. To decide which treatments or interventions to forgo, the benefits of different treatments need to be weighed. Health economics provides a framework in the decision making as to what services or interventions to provide. Economic evaluations relate the costs and benefits of different treatments

in health care delivery. It relates the benefits produced by the different treatments against the resources consumed. Different types of economic evaluation methods exist, and depending on the benefits they measure, they take different names. Cost-utility analysis is an evaluation that assesses the costs and benefits of interventions [Kernick, 2003]. It is useful in comparing different programs across different areas. The outcome of alternative treatments is compared on the quality-adjusted-life-year (QALY). The QALY gives an indication of both quality and quantity of life. The overall effectiveness, QALY, is calculated by combining the utility of the patient's health state and the time spent in that health state. Different health state description and valuation systems exist. The more common ones are the Health Utility Index [Feeney et al., 1995], the Quality of Well-being State [Clarke et al., 2002], and the EuroQOL Group's EQ5D [Shaw et al., 2005]. Each of the systems has a classification for the respondent's health state and the preferences for that health state.

The EQ5D-US has two parts. The first part is a descriptive system that classifies respondents into 1 of 243 distinct health states. The descriptive system consists of 5 dimensions which are mobility, self-care, usual activities (e.g. work, study), pain and discomfort, and anxiety. Each dimension has 3 levels with level one meaning "no problem", level two meaning "some problem" and level three meaning "extreme problem". After the data is collected, a scoring function will be used to assign a value (EQ5D index score) to self-reported health state from a set of population-based preference weights. A scoring algorithm based on the U.S. preference weights will be used to calculate the EQ5D-US index score. Most studies using the EQ5D in the U.S. have used the weights derived from the general population of the U.K.. They were used with the assumption that there is minimal difference in preference between the

population in the U.S. and the U.K.. A set of U.S. population-based preference weights had not been established until 2005 [Shaw et al., 2005]. In the motivating data set that we used, the EQ5D is calculated based on the weights developed from the U.S. population-based preferences.

For the scoring algorithm used, the possible EQ5D-US scores range from -0.11 (i.e. 33333) to 1.0 (i.e. 11111) where 0.0 means death and 1.0 means perfect health. A negative score means a health state worse than death. The scoring algorithm is based on the work presented in the article by James W. Shaw et al. [Shaw et al., 2005]. The second part of the EQ5D-US questionnaire consists of using a 20-cm visual analog score (EQ-VAS) that has end points labeled "best imaginable health state" and "worst imaginable health state" anchored at 100 and 0. Respondents are asked to indicate their current health state by drawing a line from an anchor box to that part on the EQ-VAS.

Health utility data often exhibit features such as their distribution being non-normal and left-skewed. Usually a proportion of the population will attain a utility score of 1. So a proportion of people will achieve the upper-bound of 1. This could be due to the inability of the measurement tools to detect small deviations from the full-health score of 1. Due to the non-normality and the possibly bimodal features of health utility data, analysis methods such as simple linear regression may not be appropriate. For simple linear regression to work, one of the assumptions is that the residuals need to be normally distributed with a constant variance. But utility data often violates this assumption. Also in a linear model for health utility with a continuous covariate, a linear relationship between the mean and the covariate is unlikely to hold as the utility value approaches 1 since by definition utility value is

bounded at 1.

Alternative methods have been proposed for the analysis of health utility data. For example, the Tobit models assume that there is a latent variable $Y_i^\star$ satisfying $Y_i^\star = X_i\beta + \epsilon_i$ with $\epsilon_i \sim N(0, \sigma^2)$ but the observed variable is Y_i instead of $Y_i^\star$ where

$$Y_i = \begin{cases} Y_i^\star & \text{if } Y_i^\star \leq 1 \\ 1 & \text{otherwise} \end{cases}$$

But this model assumes that the health status can be greater than one and the observed data is truncated at one. But the definition of the utility is that it is bounded at one. So it is not appropriate to use the Tobit model for the analysis of health utility data. Another proposed model is the CLAD model, which models the medians instead of the means. In estimating the parameters, it minimizes the sum of the absolute deviations

$$\sum_{i=1}^{n} \mid Y_i - min(1, \boldsymbol{X}_i\boldsymbol{\beta}) \mid$$

But this model also assumes that the utility value can be greater than one and makes the assumption that it has been censored at one. Both of the models are inappropriate for the study at hand since we are interested in analyzing utility data which by definition cannot be greater than one. Models that don't assume censoring have also been proposed. Ordinary least squares has been proposed as a method for analyzing health utility data. The mean of the response variable, health utility in this case, is modeled as a linear combination of some covariates of interest for example age, duration of diabetes, etc. Generalized additive models are proposed in this study as an alternative method of analyzing health utility data. Both OLS and GAM methods don't assume any form of censoring for the response variable.

1.2 Ordinary Least Squares (OLS)

In the linear regression model, the expected value of a response variable Y of interest is modeled as a linear function of some other variables. The response variable Y is assumed to be continuous and depends on some variables $X_1, X_2, .., X_p$. The model has the form $Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + ... + \beta_p X_p + \epsilon_i$ where ϵ_i represents the deviation between the observed value of Y and its expected value $\epsilon_i = Y_i - E(Y_i)$. For example, if Y is a student score on an exam, the independent variables $X_1, X_2, ..., X_p$ of interest could be the number of hours spent studying for the exam, the student's mid-term score, and number of hours spent watching TV each day. Since our response variable of interest is health utility, the independent variables could be blood pressure, glucose level, age, and diabetes complication. Given a set of values of $X_1, X_2, ..., X_p$, the expected value of Y is assumed to be given by

$$\mu = E(Y_i \mid X) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + ... + \beta_p X_p \tag{1.2.1}$$

The error terms, ϵ_i, include the measurement error of the tools of measurement. For example, the measured value of Y might be off from the actual Y value. The error terms take all the discrepancy into account. The error terms are assumed to be on average equal to zero. If the model is correct for the data at hand, the difference between the observed response and the predicted response should be on average close to zero, so $E(\epsilon_i) = 0$. Variance of the ϵ_i is assumed to be constant, so $Var(\epsilon_i) = \sigma^2 \ \forall i$. The errors are assumed to be independent, so $Cov(\epsilon_i, \epsilon_j) = E[(\epsilon_i - E(\epsilon_i))(\epsilon_j - E(\epsilon_j))] = 0 \ \forall i \neq j$. To estimate the parameters $\beta_0, \beta_1, ..., \beta_p$, the least-squares method is employed. Least-squares estimation gives the values of $\beta_0, \beta_1, ..., \beta_p$ that minimize the sum of squares of the deviation between

the observed values and the predicted values. We will use $\boldsymbol{\epsilon}$ to denote the vector $(\epsilon_1, ..., \epsilon_n)$, $\boldsymbol{Y}$ to denote the vector $(y_1, ..., y_n)$, $\boldsymbol{\beta}$ to denote the vector $(\beta_1, ..., \beta_p)$ and X to denote the measurement values of $x_1, ..., x_p$. Given a data set, the estimated $\boldsymbol{\beta}$ minimizes $\boldsymbol{\epsilon}'\boldsymbol{\epsilon} = [\boldsymbol{Y} - E(\boldsymbol{Y})]'[\boldsymbol{Y} - E(\boldsymbol{Y})] = \boldsymbol{Y}'\boldsymbol{Y} - 2\boldsymbol{\beta}'X'\boldsymbol{Y} + \boldsymbol{\beta}'X'X\boldsymbol{\beta}$. The estimate of $\boldsymbol{\beta}$ is then given by $\hat{\boldsymbol{\beta}} = (X'X)^{-1}X'\boldsymbol{Y}$ where $X'X$ is assumed to be invertible. If the error terms are assumed to be normally distributed with same mean and constant variance $\epsilon_i \sim N(0, \sigma^2)$, then $Y \sim N(X\boldsymbol{\beta}, \sigma^2 I_n)$ and $\hat{\boldsymbol{\beta}} = (X'X)^{-1}X'\boldsymbol{Y} \sim N(\boldsymbol{\beta}, (X'X)^{-1}\sigma^2)$. So hypothesis testing regarding β with $H_0 : \hat{\beta}_i = \beta_i$ can be done with the statistic: $\frac{\hat{\beta}_i - \beta_i}{\sqrt{(x^{i+1,i+1})\hat{\sigma}^2}} \sim t_{n-p}$ where p is the number of parameters in the model, $x^{i+1,i+1}$ is the $(i+1)^{th}$ diagonal element of the $(X'X)^{-1}$ matrix and $\hat{\sigma^2}$ is the estimate of the variance of the error terms. But since health utility data exhibits features of non-normality, testing hypotheses about β whilst making the assumption of ϵ_i being normally distributed and having constant variance is not appropriate. Robust standard errors and bootstrapping p-values can be considered. But $\hat{\boldsymbol{\beta}}$ is still an unbiased estimate of $\boldsymbol{\beta}$ since $E(\hat{\boldsymbol{\beta}}) = E((X'X)^{-1}X'\boldsymbol{Y}) = (X'X)^{-1}E(X'\boldsymbol{Y}) = (X'X)^{-1}X'E(\boldsymbol{Y}) = (X'X)^{-1}X'X\boldsymbol{\beta} = \boldsymbol{\beta}$. The alternative method to the OLS are generalized additive models. In generalized additive models, the response variable can be modeled to be non-linearly related to the predictor variables.

1.3 Generalized Additive Models (GAM)

The generalized additive model is a generalization of the linear regression model [Hastie and Tibshirani, 1986]. For a standard multiple linear regression model with p independent variables, the model is given by $Y = \mathbf{X}'\boldsymbol{\beta} + \epsilon$ where $Y = (y_1, y_2, ..., y_n)^T$

and $\boldsymbol{X} = (1, X_1, X_2, ..., X_p)'$ with $E(\epsilon) = 0$ and $Var(\epsilon) = \sigma^2$, $\boldsymbol{X}$ could be composed of some measured values of the variables $X_1, ..., X_p$ or values that are chosen in advance. There are many ways to generalize the linear regression models [Hastie and Tibshirani, 1990]. One way is to model Y as $Y = s(x_1, ..., x_p) + \epsilon$ called surface smoothers. The model can be thought of as a non-parametric estimate of the regression model. But there are problems with such generalizations. To get a model value at a point, the observations used are values in the neighbourhood of the point. In defining the local neighbourhood of a target point, the length of the neighbourhood to capture the data points will increase dramatically when the number of variables in the model increases [Hastie and Tibshirani, 1990]. For example, given a p-dimensional unit cube with data points uniformly scattered around, if a neighbourhood at the origin is to capture a $100\times K\%$ of the data, the length of the sub-cube increases as p increases by a phenomenon called the curse of dimensionality. Given say p=1, so with one variable only, to capture $100\times0.1\%$=10% of the data, the length will be 0.1 since 0.1×1=0.1 where 1 is the total volume or length in this case. With p=2, to capture the same amount of data, the length of the side will have to be about 0.32. With p=10, the length of the sub-cube will have to be about 0.8. The concept of 'local' seems to degenerate as the number of variables increases. Also as more variables are added to the model, the interpretation of the model becomes more challenging. A model with two variables Y=$s(X_1, X_2)$ can be fitted and seen with the use of software. The changes of the response Y and the relationship between Y and X_1, and X_2 can be seen easily from a plot. But with p bigger than 2, the interpretation will become harder since it is not easy to visualize the relationship. Other generalizations of multiple linear regression models have been devised. For example, projection

pursuit regression, and the regression tree [Hastie and Tibshirani, 1990].

On the other hand, with a linear model, the interpretation becomes more straightforward since the effect of each predictor $X_1, .., X_p$ can be seen from the coefficients β. The additive model generalizes the linear model with the function $s_1(x_1), s_2(x_2), ..., s_p(x_p)$ replacing the $\beta_1 x_1, \beta_2 x_2, ..., \beta_p x_p$. $s_1(x_1), ..., s_p(x_p)$ could be some arbitrary univariate functions of the variables. The model is then given by

$$Y = \alpha + s_1(x_1) + s_2(x_2) + ... + s_p(x_p) + \epsilon \tag{1.3.2}$$

with $Cov(\epsilon_i, \epsilon_j) = 0$, $E(\epsilon_i) = 0$, and $Var(\epsilon_i) = \sigma^2 \; \forall i, j$. Also $Es_i(x_i)=0 \; \forall i$ is assumed so there will be no free constant for each $s_i(X_i)$. Each of the predictors contributes to the model through their own functions. The overall effect is additively combined from each individual function. Once the model has been fitted, the individual contribution to the model of each variable can be explained separately. Interpretation of the model is less complicated since there are no interaction terms in the model. By varying one variable and holding all the others fixed, the change in the response due to the specific variable can be observed. And the change of the response will not depend on other variables except the one that is varying. Then by varying each variable in turn and fixing all the other variables, and plotting the changes of response against the variable that is changing, the individual contribution of each variable can be seen. But additive models may have components which are functions of more than one variable. For example, the component functions can be of the form $s(x_1, x_2)$, $s(x_1, x_3)$. The functions can also be of categorical variables. Although models with functions of more than one variable are possible, interpretation can be more complicated. More details of these scenarios can be found in [Hastie and

Tibshirani, 1990].

Generalized linear models generalize the linear regression model by allowing the possibilities of different links in linking the parameter μ with the predictors $X_1, ..., X_p$. The parameter μ is related to $X_1, ..., X_p$ by $\eta(\mu) = \beta_0 + \beta_1 x_1 + ... + \beta_p x_p$ where $\eta(\mu)$ is called the link function. The generalized linear model assumes that the response variable is from the exponential density family $f(y_i \mid \theta, \phi) = exp(\frac{y_i\theta - b(\theta)}{a(\phi)} + c(y_i, \phi))$ where θ is called the natural parameter and ϕ called the dispersion parameter. For example, a binary variable response can be modeled as $log(\frac{\mu}{1-\mu}) = \beta_0 + \beta_1 x_1 + ... + \beta_p x_p$ by using the logit link. The more common link is the canonical link where $\eta(\mu) = \theta$. The GAM generalizes the additive model by also allowing the possibilities of different links. For example, with the logit link, the model of a binary response variable will become $log(\frac{\mu}{1-\mu}) = \alpha + s_1(x_1) + ... + s_p(x_p) + \epsilon$. The estimation of α and $s_1(x_1), ..., s_p(x_p)$ is done by using a procedure called local scoring procedure where the backfitting algorithm is employed.

1.4 Backfitting Algorithm

To estimate the $s_0, s_1(\bullet), \ldots, s_p(\bullet)$ in the additive model, a backfitting algorithm is used. The backfitting algorithm is an iterative procedure. The motivation of the algorithm is similar to the ordinary least squares method when the residuals are minimized with respect to the β's. The additive model is given as

$$Y = s_0 + \sum_{j=1}^{p} s_j(X_j) + \epsilon \tag{1.4.3}$$

The partial residual of the j^{th} observation is defined as

$$R_j = Y - s_0 - \sum_{k \neq j} s_k(X_k) \tag{1.4.4}$$

then $E(R_j \mid X_j) = s_j(X_j)$. So the function being minimized is

$$E(Y - s_0 - \sum_{k=1}^{p} s_k(X_k))^2 \tag{1.4.5}$$

Given some estimates of s_i where $i \neq j$, the estimate of s_j can be devised iteratively by the procedure of backfitting algorithm. The details of the backfitting algorithm can be found in the article by Friedman and Stuetzle [Friedman and Stuetzle, 1981].

Backfitting Algorithm

Initialization:

$s_0 = E(Y), s_1^1(\bullet) \equiv 0, s_2^1(\bullet) \equiv 0,$

$s_3^1(\bullet) \equiv 0, \ldots, s_p^1(\bullet) \equiv 0,$ m=0.

Iterate:

$m = m$+1 for j=1 to p do the following :

$R_j = Y - s_0 - \sum_{k=1}^{j-1} s_k^m(X_k) - \sum_{k=j+1}^{p} s_k^{m-1}(X_k)$

$s_j^m(X_j) = E(R_j \mid X_j).$

Until:

The quantity $E(Y - s_0 - \sum_{j=1}^{p} s_j^m(X_j))^2$ stop decreasing.

In the backfitting algorithm, the estimate of $s_j(\bullet)$ at the m th iteration is denoted by $s_j^m(\bullet)$. To get the current estimate of the term $s_j(\bullet)$, the terms from the previous iteration are used. For example, to estimate the $s_j(\bullet)$ at the current step, the estimate of $s_1(X_1), s_2(X_2), \ldots, s_{j-1}(X_{j-1})$ from the current step is used and the estimates of $s_{j+1}(X_{j+1}), s_{j+2}(X_{j+2}), \ldots, s_p(X_p)$ in the previous step are used.

1.5 Local Scoring Algorithm

The local scoring algorithm is used to fit the generalized additive model. It makes use of the backfitting algorithm in the process.

Initialization:

$$s_0 = g(E(Y)), s_1^0(\bullet) \equiv s_2^0(\bullet) \equiv ... \equiv s_p^0(\bullet) \equiv 0, m = 0$$

Iterate:

$$m = m + 1$$

1 Form the adjusted dependent variable $Z_i = \eta_i^{m-1} + (Y_i - \mu_i^{m-1})(\frac{\partial \eta_i}{\partial \mu_i^{m-1}})$, where $\eta_i^{m-1} = s_0 + \sum_{j=1}^{p} s_j^{m-1}(X_{ij})$ and $\eta_i^{m-1} = g(\mu_i^{m-1})$.

2 Form the weights $w_i = (\frac{\partial \mu_i}{\partial \eta_i^{m-1}})^2 V^{-1}$ where V is var(Y_i).

3 Fit an additive model to $Z = (Z_1, ..., Z_n)$ using the backfitting algorithm with weights w_i, so that the function $\sum_{j=1}^{n} w_i(y_i - s_0 - s_1(X_{i1}) - ... - s_p(X_{ip}))^2$ is minimized, then get estimated functions $s_j^m(\bullet)$ and model η^m.

Until: E dev(Y, μ^m) fails to decrease where dev is the deviance between Y, and the new estimate μ^m.

The details can be found in [Hastie and Tibshirani, 1986].

1.6 Smoothing

In the GAM model, the function $s_i(X_i)$ could be of parametric form or non-parametric form. For example, the function $s_i(X_i)$ could be some closed-form function like $\sin(X_i)$ or $X^2 + 1$. But it can also be some kind of non closed-form function. The

model gives an average change of the response variable with respect to the change in the explanatory variables. Different kinds of averaging can be done. Since the purpose of the averaging process is to get a relatively smooth overall representation of the data, the term *smoother* is used to label the different averaging techniques collectively. Depending on the amount of smoothness produced and the way it is done by the smoothers, different specific names are given to each. With one explanatory variable, if there is replication of the explanatory variable values, then the smoothed value at a particular point can be calculated by just averaging the observed response values at that point. But replications of explanatory variables are not always observed, so it is not always possible to just calculate the smooth at the point of interest, the target point, in that way. The nearby points around the target point can be used to calculate the smooth at the target. Those points constitute a neighbourhood of the target point. The average value of the observed response values at the points inside the neighbourhood will give the smooth at the target point of interest.

A decision needs to be made as to how big a neighbourhood is chosen and how the average is calculated. Some smoothers may provide a smoother plot, while others may provide a more wiggly representation. A bigger neighbourhood will include more neighbours in calculating the smooth and the variance of the estimate is generally smaller, but with a bigger bias. On the other hand, a smaller neighbourhood uses fewer points in the averaging, but the estimate will have bigger variance and less bias. There is a trade off between the variance and the bias when determining how big of a neighbourhood to take. How the averaging is done depends on the choice of the smoother used. For example, for a running-mean smoother, the average is being

done for a symmetric neighbourhood. An equal number of points is taken from the left and right of the target point and the smooth will just be equal to the average of the observed response at those points in the neighbourhood. So equal weight is given to each point inside the neighbourhood. For a running-line smoother, the averaging is done by using a least squares line. After a neighbourhood is chosen for the target point, a least squares line will be fit using those points that are in the neighbourhood. Then the smooth at the target point will be the least squares fit evaluated at that particular point. Similarly, a least squares fit is done for each neighbourhood of other target values. So in the running-line smoother, an asymmetric neighbourhood is used since it is not possible to have an equal number of points taken on each side of the target point.

Also different weights can be given to the points inside a neighbourhood. Points that are further away from the target can be given less weight in computing the smooth since their values might be considered less influential than those points that are closer to the target point. For example, a smoother called kernel smoother assigns different weights to different points according to how far they are from the target. The weight assigned to the different points is given by $S_{0j} = \frac{C_0}{\lambda} d(|\frac{x_0 - x_j}{\lambda}|)$ where $d(|t|)$ is a function. The function d can be the standard normal density function, so points that are further away will be given less and less weight gradually [Hastie and Tibshirani, 1990]. So the kernel smoother gives a smoother fit since the weight is only decreasing gradually whereas some smoothers give zero weight to all points outside of a certain range, and this can lead to a more wiggly representation.

The percentage of points to be included in a neighbourhood is called the *span* of the data points or the *smoothing parameter*. The smoothing parameter can be

denoted as λ. When the smoothing parameter varies, the bias and variance of the estimate also changes. This can be illustrated with the running-mean smoother. For the running-mean smoother, a symmetric neighbourhood is normally taken. So an equal number of points say k from the left and k from the right of the target point x_i is taken. An asymmetric neighbourhood can also be taken when there are not enough points available on the left or on the right of the target point. A total of $2k+1$ points are taken. For the running-mean smoother case, the smooth is given by $\hat{f}(x_i) = \sum_{j \in N_k^s(x_i)} \frac{y_j}{2k+1}$ which is just the average of the observed y-values of those $2k+1$ points inside the neighbourhood. The expectation of $\hat{f}_k(x_i)$ is then given by $E(\hat{f}_k(x_i)) = \sum_{j \in N_k^s(x_i)} \frac{f(x_j)}{2k+1}$. The variance of the smooth is given by $Var(\hat{f}_k(x_i)) = \frac{\sigma^2}{2k+1}$ where $V(\epsilon) = \sigma^2$ from the model $Y = f(x) + \epsilon$. When the number of neighbours used increases, i.e. the percentage of points used increases, the bias tends to increase and the variance tends to decrease. On the other hand, when the percentage of points used decreases, the bias tends to decrease, but the variance tends to increase.

It is possible to find a smoothing parameter that gives an optimal result of bias and variance [Hastie and Tibshirani, 1990]. A criterion that combines both the bias and variance is the mean squared error. For example, for the running-mean smoother, the mean squared error is $E(\hat{f}_k(x_i) - f(x_i))^2 = var(\hat{f}_k(x_i)) + [E\hat{f}_k(x_i) - f(x_i)]^2$. The bias of the linear smoother can be shown using a Taylor series expansion to be approximately equal to $\frac{k(k+1)}{6} f''(x_i) \triangle^2$ where $\triangle$ is defined as $x_{i+1} - x_i = \triangle$ and assuming that the data are equally spaced. From the approximation, it can be seen that the bias increases as k increases. An optimal value of k can be found by minimizing the mean squared error formula, and it can be shown to be equal to

approximately $k_{opt} = (\frac{9\sigma^2}{2\triangle^4 f''(x_i)^2})^{1/5}$. Both formula requires the $f''(x_i)$ to be known, but $f''(x_i)$ is rarely known.

In the data analyst work, a different criterion is employed for choosing the smoothing parameter. Instead of choosing a parameter that minimizes the mean squared error, a sum of squares called cross-validation is minimized instead. The cross-validation square error is defined as $CV(\lambda) = \frac{1}{n}\sum_{i=1}^{n} E(Y_i - \hat{s}_\lambda^{-i}(x_i))^2$ where $\hat{s}_\lambda^{-i}(x_i)$ is the fitted value at x_i computed after leaving out the point x_i. The cross-validation works by leaving out the point (x_i, y_i) one at a time and then estimating the smooth at the points x_i based on the remaining $n-1$ points. The cross-validation sum of squares is similar to another quantity called the average predictive squared error. The average predictive squared error is defined as $PSE(\lambda) = \frac{1}{n}\sum_{i=1}^{n} E(Y_i^\star - \hat{f}_\lambda(x_i))^2$, where the $Y_i^\star$ is a new observation at x_i, so $Y_i^\star = f(x_i) + \epsilon_i^\star$ where $\epsilon_i^\star$ is independent of the $\epsilon_i s$. The value of the cross-validation sum of squares can be shown to be on average equal to the average predictive squared error, i.e. $E\{CV(\lambda)\} \approx PSE(\lambda)$.

1.6.1 Splines

In our data analysis, the non-parametric part of the GAM will employ smoothing splines. A smoothing spline is one kind of smoothing technique. Smoothers such as the running-line smoother and running-mean smoother are constructed explicitly, and the smooth at the target point is then computed. For the smoothing spline, the spline function is chosen such that a certain criterion is met. A function $f(x)$ with two continuous derivatives is chosen such that the following is minimized: $\sum_{i=1}^{n}\{y_i - f(x_i)\}^2 + \lambda\int_a^b \{f''(t)\}^2 dt$ where λ is a fixed constant and

$a \leq x_1 \leq ... \leq x_n \leq b$. The criterion is composed of two parts. The first part gives the difference between the observed value and the predicted value. The second part gives a condition in terms of the second derivative of the function. The parameter λ acts as a span, so that different values of λ give rise to different amount of smoothness. It can be shown that the function that minimizes the criteria above is a cubic smoothing spline with knots at the value of x_i. A cubic smoothing spline is a series of cubic polynomials of the form $f(x) = ax^3 + bx^2 + cx + d$ joined together smoothly. The derivation of the criteria can be found in [Reinsch, 1967].

1.6.2 LOESS

The other smoothing method that is employed in our data analysis is the 'Loess' method. 'Loess' stands for locally-weighted running-line smoothers. It is a method developed by Cleveland [Cleveland, 1979]. The 'Loess' can be used to compute the estimate at a target point of interest. To compute the estimate at a target point say x_0, a neighbourhood is first chosen. The neighbourhood contains the k nearest points to the target point. A weight is then given for each point in the neighbourhood. Let x_i be a point in the neighbourhood. Then the weight assigned to the point x_i is given by the weight function $W(\frac{|x_0 - x_i|}{\triangle(x_0)})$ where $\triangle(x_0)$ is defined to be the distance between the target point and the point inside the neighbourhood that lies furthest away from the target point. The function w is defined as

$$W(u) = \begin{cases} (1-u^3)^3 & \text{if } 0 \leq u \leq 1 \\ 0 & \text{otherwise} \end{cases}$$

Then the estimated value at x_0 is computed by fitting a weighted least squares line using those weights assigned by the weight function. The steps can be summarized as below:

1 The k nearest neighbours of x_0 are chosen and denoted as $N(x_0)$.

2 $\triangle(x_0) = max_{N(x_0)}|x_0 - x_i|$ is calculated, so the maximum distance is calculated.

3 Weights are assigned to each point in $N(x_0)$ by the weight function.

4 $s(x_0)$ is computed as a fit at x_0 from weighted least squares using those observed y values corresponding to the points inside $N(x_0)$.

There is no definite way to compare among different smoothers as to which is more suitable [Hastie and Tibshirani, 1990].

The remainder of the thesis is orgainzed as follows. In chapter 2, the real data, Diabetes Hamilton, is analyzed with the OLS method. Robust standard errors and bootstrapping method will be used. In Chapter 3, simulation studies will be done to investigate the performance of GAM method. Also a comparsion study will be done between OLS method and GAM method. Then in chapter 4, GAM will be applied to analyze the real data set, Diabetes Hamilton.

Chapter 2

Motivating Data Set

2.1 Diabetes Hamilton Data

The Diabetes Hamilton data set is from some of the people who participated in the program called Diabetes Hamilton, in Hamilton Ontario [O'Reilly et al.]. Diabetes Hamilton is a program that provides diabetes information and resources to people with diabetes. Some of the participants in the program completed the EQ-5D questionnaires in addition to other questionnaires that they were asked to complete. There are 1147 participants who completed the EQ-5D questionnaires. The people who completed the EQ-5D questionnaires answered a list of questions that indicate their current health state. The EQ-5D utility scores are then calculated using the U.S. based EQ-5D scoring algorithm. In addition to the EQ-5D utility scores, other information about the participants was also collected. For example, information such as age, gender, duration of diabetes, BMI (body mass index), HbA1C (hemoglobin A1c), and diabetes complications are also recorded. The diabetes complications

Table 2.1: Summary of the demographic characteristics of the Diabetes Hamilton data set.

Variable	*Mean*	*Median*	*Minimum*	*Maximum*
EQ5D scores	0.747246	0.799794	-0.0383556	1
Age	63.7378	63.8877	39.8986	92.9041
Duration of Diabetes	11.16413	8.2219	0.1699	64.874

recorded include the following: foot or leg amputation, heart attack, stroke, and kidney failure. From the data set, we would like to find out how the different diabetes complications have an impact on the quality-of-life of people carrying the diabetes disease because the number of people who are affected by the disease has been increasing. And it can reduce the burden of society if timely interventions could be done so that extra costs of health care or treatment could be avoided if possible. The demography summary of the Diabetes Hamilton data is summarized in Table 2.1. The mean age of the people in the data set is 63.7 with the youngest being 39 years of age and oldest being 92 years of age. The mean of the EQ5D scores from Diabetes Hamilton data set is 0.747 with the minimum observed score being -0.038 and the highest score being 1. The frequency of the diabetic complications are reported in Table 2.2. A total of 157 people reported having a heart attack in the past, 82 people reported stroke, 28 people reported kidney failure, and 14 people reported amputation. So the most common diabetes complication is heart attack. The mean duration of diabetes is reported to be 11.1, so on average the patients in the study have carried the disease for about 11 years.

Table 2.2: Diabetes complications summary.

Variable	*Frequency*
Heart attack	157
Stroke	82
Kidney failure	28
Foot or Leg amputation	14

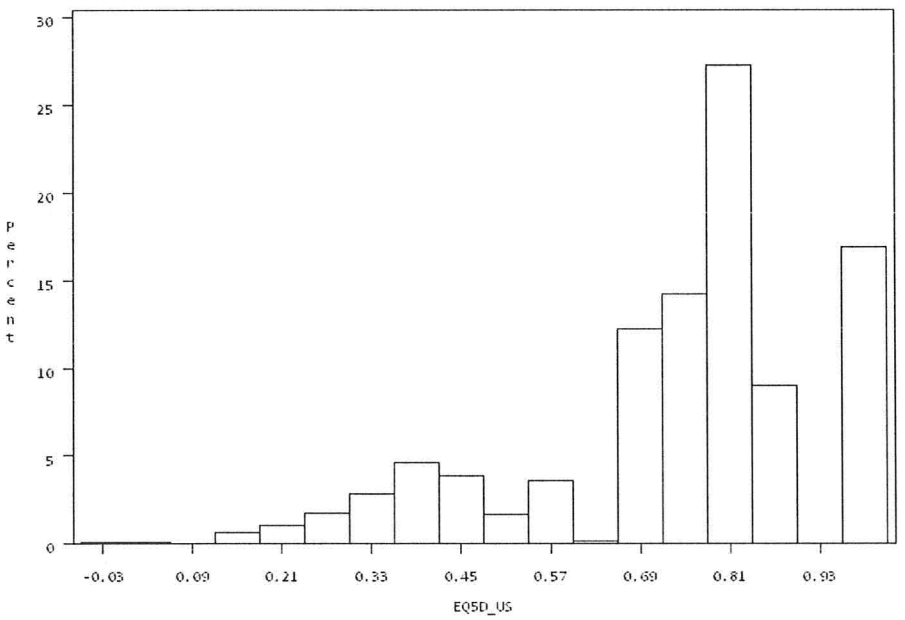

Figure 2.1: Histogram of the EQ5D values of Diabetes Hamilton data.

In the data set, about 18% of the observations have EQ-5D utility scores equal to one where one stands for perfect health. About 27% of the observations have utility scores equal to 0.8 as seen in Figure 2.1. The majority of people reported a value that indicates perfect health or a state that is close to perfect health. A large proportion of people achieving perfect or close to perfect health states can be due to the inability of the measurement tools to differentiate between perfect health and small deviations from perfect health.

In the Diabetes Hamilton data set, the variables of diabetes complications are: amputation, heart attack, stroke, and kidney failure. Each variable takes on values of 1, 2, and 3, where 1 stands for within the past year, 2 stands for more than 12 months ago, 3 stands for never. OLS is used to analyze the Diabetes Hamilton data set. We are interested in comparing health utility score between "ever had the complication" to "never". Therefore, we have recoded the values of the diabetes complications so that now each of the variables amputation, heart attack, stroke, and kidney failure only take on the values 0 and 1. 0 replaces the value 3, and 1 replaces the value of 1 and 2 in the original data set. So now 0 stands for never and 1 stands for complications that have ever happened. Each coefficient of the diabetes complication variables turn out to be negative. So people who had a diabetes complication are expected to have a negative impact on the health utility score compared with those who have no diabetes complications. The coefficients of the parameters of amputation, stroke, heart attack, and kidney failure are reported in Table 2.3. They are -0.06311, -0.04617, -0.05859, and -0.10182 respectively. The standard errors for each of the parameters are given to be 0.05257, 0.02275, 0.01727, and 0.03775 respectively. The t-value of the diabetes complication parameters are

Table 2.3: Estimates of parameters and their standard error and p-value with the use of regression analysis.

Variable	*DF*	*Parameter Estimate*	*Standard Error*	*t value*	*P-value*
Intercept	1	0.51736	0.03824	13.53	<0.0001
Foot Leg Amputation	1	-0.06311	0.05257	-1.2	0.2302
Stroke	1	-0.04617	0.02275	-2.03	0.0426
Heart Attack	1	-0.05859	0.01727	-3.39	0.0007
Kidney Failure	1	-0.10182	0.03775	-2.7	0.0071
Age	1	0.00291	0.00056062	5.2	<0.0001
Gender	1	0.05151	0.01159	4.44	<0.0001
Duration of Diabetes	1	-0.00152	0.000615	-2.47	0.0136

given as -1.2, -2.03, -3.39, and -2.7. And the p-value of the test of the significance of the parameters amputation, stroke, heart attack, and kidney failure are given to be 0.2302, 0.0426, 0.0007, and 0.0071. So all diabetes complications are considered to be important in contributing to the change of the EQ5D utility score at the 5% significance level except the parameter amputation.

In order for conclusions drawn from the OLS model to be valid, assumptions regarding the model need to be satisfied. These assumptions are homogeneity of error terms, independence between error terms. If these assumptions are satisfied then no patterns should be observed when residuals are plotted against the predicted values. The plot of residuals against the predicted values is shown in Figure 2.2. Some diagonal lines are observed from the plot, but other than that, the rest of the points seem to be randomly scattered around.

The error terms are also assumed to be normally distributed with the same

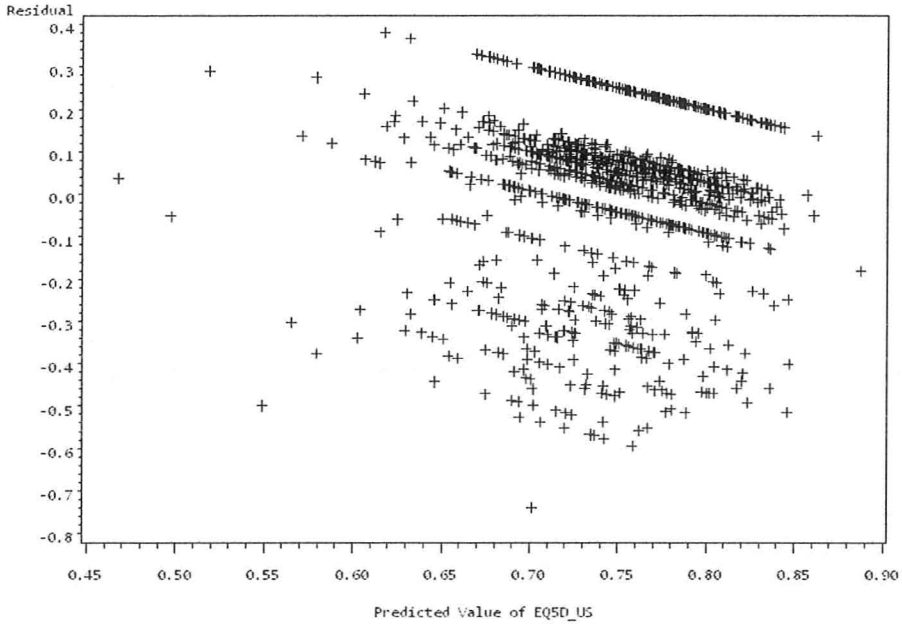

Figure 2.2: Plot of residuals values against the predicted values of EQ5D.

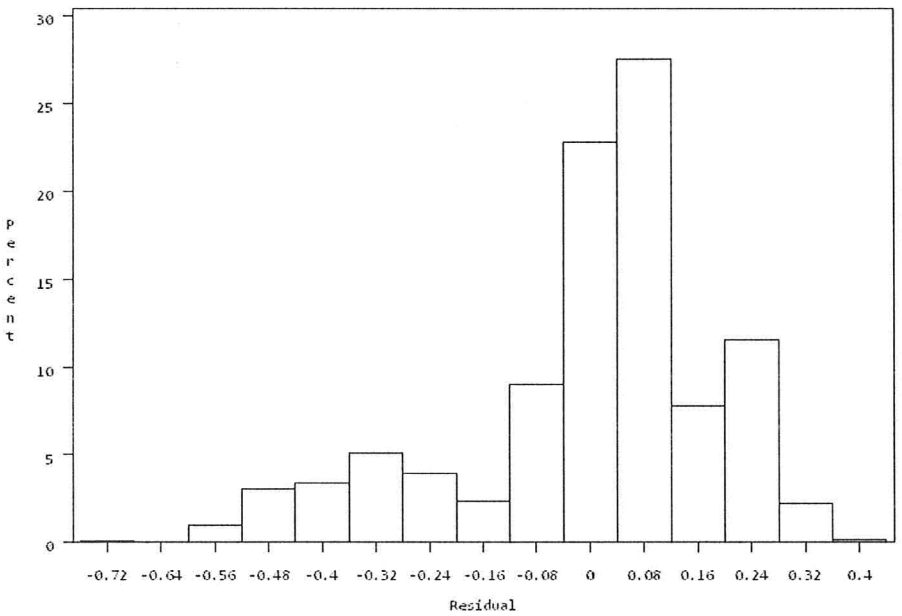

Figure 2.3: Histogram of the residuals of OLS model.

mean and having constant variance. A histogram of the residuals is shown in Figure 2.3. From the histogram, it seems that the residuals don't follow normal distribution.

A q-q plot of the residuals is shown in Figure 2.4. From the q-q plot, the points don't seem to fall on a straight line. It seems that the normality assumption of the error terms is violated.

Since health utility data exhibits features such as heteroscedasticity of error terms, alternative methods such as robust standard errors and bootstrapping for the errors terms are considered. Robust standard error can be used when heteroscedasticity is present in the error terms.

For the robust standard error, the variance of the estimated coefficients is

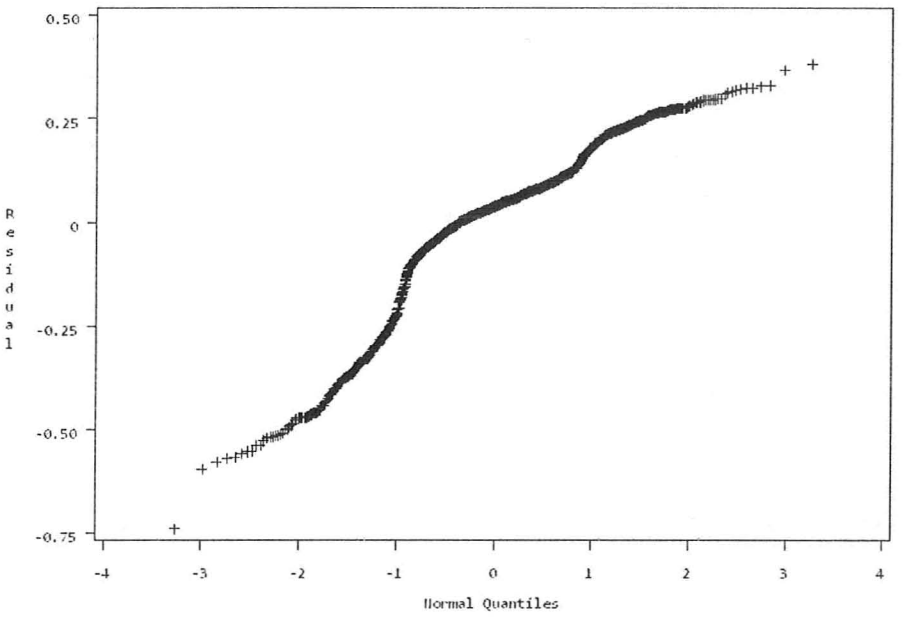

Figure 2.4: Q-Q plot of the residuals of OLS model.

given by $var(\hat{\boldsymbol{\beta}}) = (\sum_i \boldsymbol{X_i}\boldsymbol{X_i}')^{-1}(\sum_i \boldsymbol{X_i}Var(Y_i|\boldsymbol{X_i})\boldsymbol{X_i}')(\sum_i \boldsymbol{X_i}\boldsymbol{X_i}')^{-1}$, where $\boldsymbol{X_i}$ is a vector of the covariates for subject i and $Var(Y_i|\boldsymbol{X_i})$ is estimated by a function of the residual $Y_i - \boldsymbol{X_i}\beta$. The version of the covariance matrix for $\boldsymbol{\beta}$ that we use here is $Var(\hat{\boldsymbol{\beta}}) = (\sum_i \boldsymbol{X_i}\boldsymbol{X_i}')^{-1}(\sum_i \boldsymbol{X_i}(\frac{1-h_{ii}}{Y_i-\boldsymbol{X_i}\beta})^2\boldsymbol{X_i}')(\sum_i \boldsymbol{X_i}\boldsymbol{X_i}')^{-1}$ where $h_{ii} = \boldsymbol{X_i}'(\boldsymbol{X_i}\boldsymbol{X_i}')^{-1}\boldsymbol{X_i}$ [Ervin and Long, 2000]. The robust standard errors of the diabetes complications are reported in Table 2.4. The robust errors are 0.0566896, 0.0236896, 0.0170977, and 0.0467273 respectively for amputation, stroke, heart attack, and kidney failure. The robust p-value of each of the diabetes complications coefficients are given to be 0.2302, 0.051348, 0.000632, and 0.029536. So all parameters are considered to be significant at the 10% level except for amputation.

Another method that is employed in calculating the p-value for the parameters

Table 2.4: Robust standard errors, robust p-value, standard deviation, bootstrap p-values of each parameter estimate.

Variable	*Parameter Estimate*	*Std. Error*	*Robust Standard Error*	*Bootstrap Standard Error*	*p-value*	*Robust p-value*	*Bootstrap p-value*
Intercept	0.051736	0.03824	0.0399201	0.0005683	<0.0001	0	0
Foot Leg Amputation	-0.06311	0.05257	0.0566896	0.0513040	0.2302	0.2302	0.281953
Stroke	-0.04617	0.02275	0.0236896	0.0233411	0.0426	0.051548	0.055459
Heart Attack	-0.05859	0.01727	0.0170977	0.0179490	0.0007	0.000632	0.000406
Kidney Failure	-0.10182	0.03775	0.0467273	0.0479629	0.0071	0.029536	0.034695
Age	0.00291	0.000561	0.0005717	0.0005683	<0.0001	4.085×10^{-7}	4.971×10^{-7}
Gender	0.05151	0.01159	0.0116898	0.0112295	<0.0001	0.00012	0.000018
Duration of Diabetes	-0.00152	0.000615	0.0005927	0.0005986	0.0136	0.010452	0.012781

in the model is bootstrapping. Bootstrapping provides a sampling distribution for a statistic of interest. Bootstrapping involves sampling from the data set already collected at hand. There are parametric bootstrap and non-parametric bootstrap methods. In the parametric bootstrap case, some assumptions are made regarding the underlying distribution of the population. For the non-parametric bootstrap case, there is no assumption being made about the underlying population. So the statistics of interest are estimated empirically from the sampling distribution [Efron and Tibshirani, 1993].

In our data analysis work, the non-parametric bootstrap is employed to get standard errors of the parameters of interest. Sampling with replacement is done to the data set called S at hand. The sample S is the Diabetes Hamilton data. Then 1000 samples each of size 1147 is drawn with replacement from the data set S. So we have 1000 bootstrapped samples each denoted as $S_i^\star$ where $S_1^\star = \{X_{1,1}, X_{1,2}, ..., X_{1,1147}\}, S_2^\star = \{X_{2,1}, X_{2,2}, ..., X_{2,1147}\}, ..., S_i^\star = \{X_{i,1}, X_{i,2}, ..., X_{i,1147}\}, ...,$

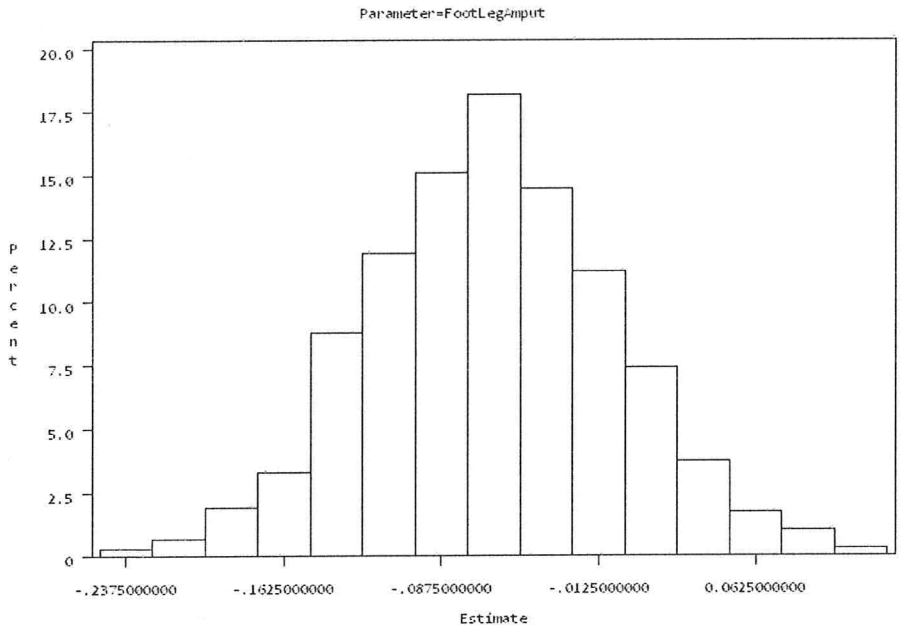

Figure 2.5: Bootstrap distribution for the coefficient of amputation.

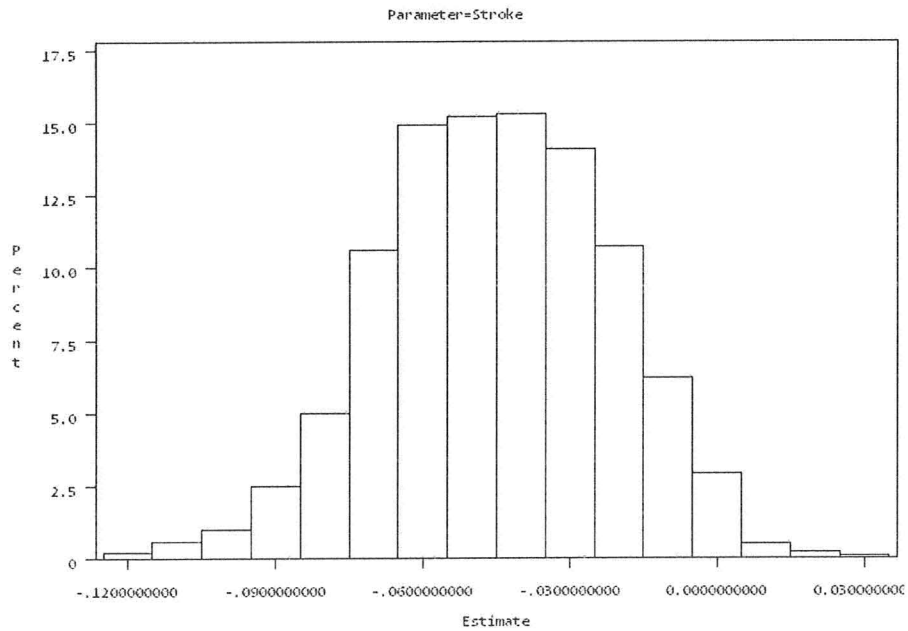

Figure 2.6: Bootstrap distribution for the coefficient of stroke.

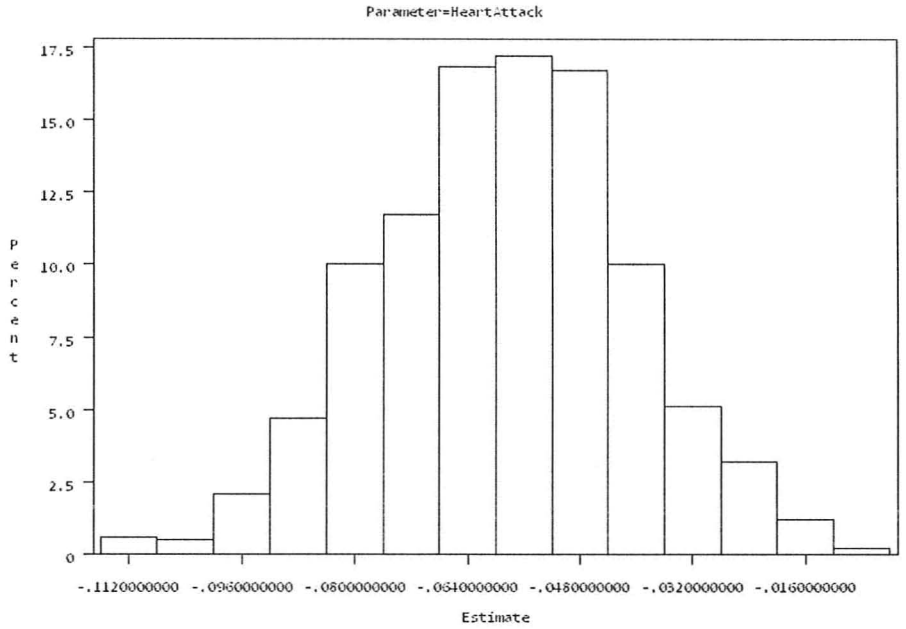

Figure 2.7: Bootstrap distribution for the coefficient of heart attack.

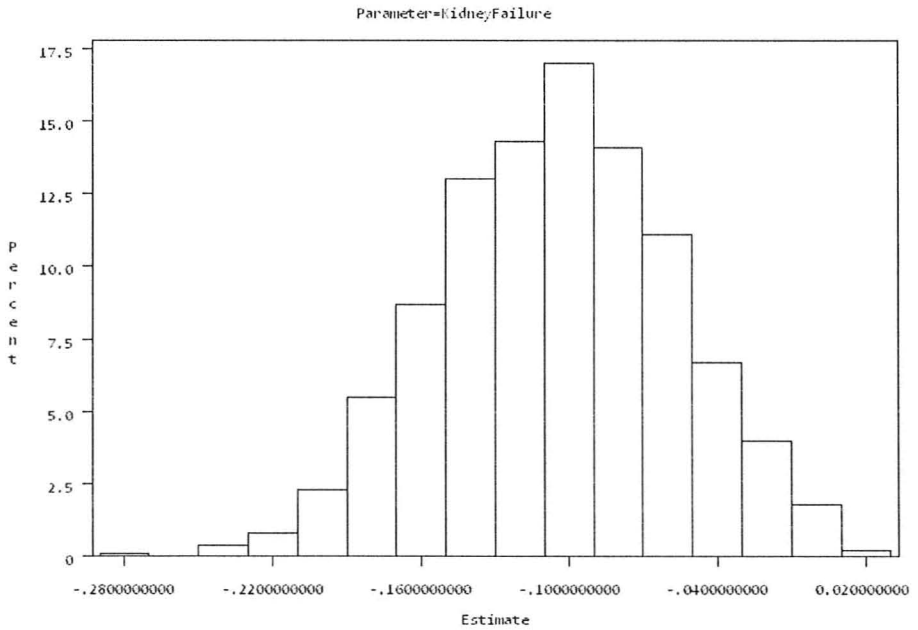

Figure 2.8: Bootstrap distribution for the coefficient of kidney failure.

$S^{\star}_{1000} = \{X_{1000,1}, X_{1000,2}, ..., X_{1000,1147}\}$. The statistics of interest say T in this case are the estimates of the β coefficients. So $T^{\star}_1 = t(S^{\star}_1), T^{\star}_2 = t(S^{\star}_2), ..., T^{\star}_{1000} = t(S^{\star}_{1000})$ are recorded. Then the average of the bootstrapped statistics $\bar{T}^{\star} = \hat{E}^{\star}(T^{\star}) = \frac{\sum_{i=1}^{n} T^{\star}_i}{n}$ estimates the expected value of the original statistics of interest i.e. T (and n=1000 in our case). So $\hat{B}^{\star} = \bar{T}^{\star} - T$ is an estimate of the bias of T, i.e. $E(T) - \theta$ where θ is the β coefficient of the model. And the variance of the bootstrapped statistics, i.e. $V(T^{\star}) = \frac{\sum_{i=1}^{n} (T^{\star}_i - \bar{T})^2}{n-1}$ can be an estimate of the variance of the original statistics of interest i.e. T. The bootstrap sample is to the sample as sample is to the population [Efron and Tibshirani, 1993]. $t(\theta^{\star}) - t(\hat{\theta}) \sim t(\hat{\theta}) - t(\theta) \Rightarrow Var(t(\theta^{\star}) - t(\hat{\theta})) \approx Var(t(\hat{\theta}) - t(\theta)) \rightarrow Var(\beta^{\star}_i) \approx Var(\hat{\beta}_i)$.

Bootstrap standard errors are reported in Table 2.4. The bootstrap p-value of the parameters amputation, stroke, heart attack, and kidney failure are given to be 0.281953, 0.055459, 0.000406, and 0.034695, respectively. The bootstrap distribution of each of the estimates for the parameters of amputation, stroke, heart attack, and kidney failure are also shown respectively in Figures 2.5–2.8. From the graphs, they seem to follow a normal distribution quite well. Therefore the bootstrap p-value can be relied upon. So all diabetes complications are shown to be significant at 10% in terms of explaining the change of the health utility score.

From the scatter plot of residuals against the predicted EQ-5D values, no obvious patterns are observed. But there could be a trend in the data that is not obvious. To aid in evaluating whether a trend exists in the plot of residuals against the variable age and duration of diabetes, a loess line that runs through the graph is used. The plot of the residuals against the predicted values with the loess curve is shown in Figure 2.9. From the plot, the curve seems to be running quite steadily from

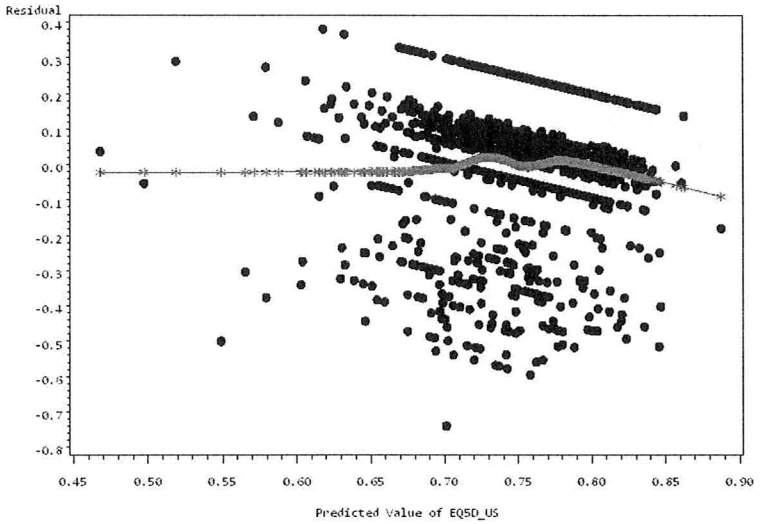

Figure 2.9: Residuals plot with the use of Locally-Weighted Smoother ('loess').

the left to the right. There seems to not be vertical variation in the curve. Therefore, it seems that no trend exists in the plot of the residuals against the predicted values.

The residuals are also plotted against the variables age and duration of diabetes in Figure 2.10 and 2.11, respectively. From the graphs, there seem to be some upward pattern coming out from the loess curve that runs through the graph for residuals against duration of diabetes and a downward trend that coming out from the graph of residuals against age. It seems that both the variables age and duration of diabetes are non-linearly related to the EQ5D utility response. So models that can encompass a non-linear relationship between the response variable and some of the predictors variables might be more appropriate. A GAM model could be used in this situation and will be illustrated later on in the study.

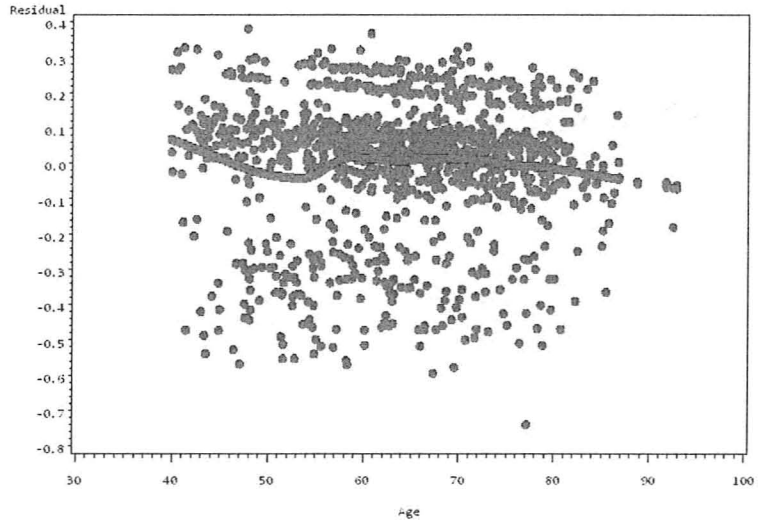

Figure 2.10: Residuals vs age with the use of Locally-Weighted Smoother ('loess').

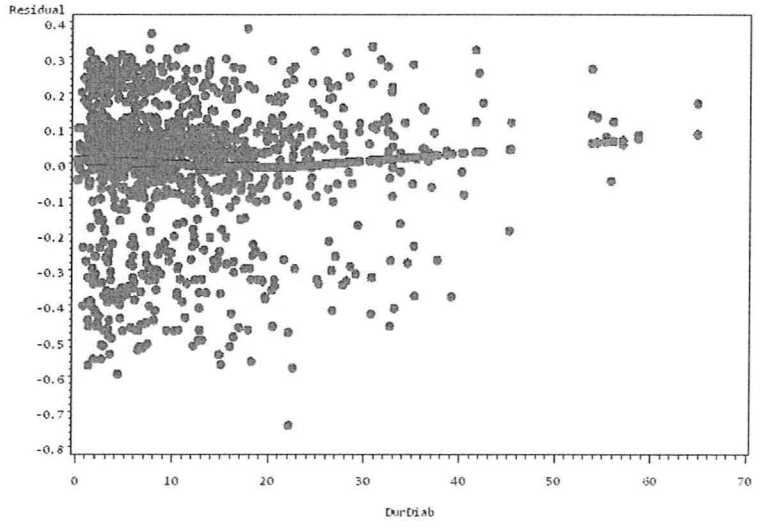

Figure 2.11: Residuals vs duration of diabetes with the use of Locally-Weighted Smoother ('loess').

Chapter 3

Simulation

To assess the performance of GAMs in analyzing health utility data and for a comparative study to be done between the OLS and GAM methods, simulation is employed in our study. Simulation is performed using a software package that can handle computationally intensive tasks. The package that is used in our study is the SAS 9.1 system. The use of simulation studies in the medical literature has been increasing [Burton et al., 2006]. A number of things need to be addressed when simulation is used. For example, the purpose of the simulation, the model that is used for simulating the data, the variables of interest, the method of comparing across different models, etc. The simulation model should possess characteristics that resemble the features of the population of interest [Burton et al., 2006]. However since it is not always possible to perform an experiment repeatedly or more than once, if a simulation model can mimic a real-life situation, costly error or risks might be avoided by doing a simulation study. After data are simulated from the simulation models, the data are fit using some model. From the simulation study, we would like to find

out GAM method's accuracy of estimating the change of the health utility scores for people who previously had a heart attack had they not had a heart attack after adjusting for each person's age, so we are interested in the bias of the GAM method. Also by simulating a large number of data sets and fitting OLS and GAM to each, we would like to find out which method can be more appropriately applied to health utility data.

The models are compared against each other in terms of certain criteria. There are a number of criteria that can be used to compare among the different models to evaluate which is a "better" one or more suitable for the particular situation. The criteria that we use for our study are bias, empirical standard error (i.e. the standard error of the estimator), average standard errors, and coverage probability. The empirical standard error (ESE) should be close to the average standard error (ASE) if the estimates are unbiased [Schafer and Grapham, 2002]. Also the Type-1 error rate for testing the significance of parameter is considered. Not every model gives a standard error of the estimates, so only certain models are compared using coverage probability criteria. But most models are compared based on the bias of the estimates that they produce. The details of the design of simulation studies can be found in [Burton et al., 2006]. Two simulation models are used in our study. The first model that is used in our study is a model based on Basu & Manca. The second simulation model is a two-part logarithmic model which represents a more realistic scenario for our study.

3.1 Simulation Model 1: Beta Models with a lump mass at 1

This model is adapted from the Basu & Manca model. It produces response values from zero to one. The response variable is Y. It is modeled as a mixture distribution. When Y is equal to one, it is modeled as a Bernoulli random variable and when Y is below one, it is modeled as a beta random variable denoted as Y_{beta}. The Bernoulli random variable X_2 is an added component to the Basu & Manca model and it is correlated with X_1 since we are interested in simulating a scenario where we are interested in the effect of the binary variable X_2 on health utility but with the presence of a confounding variable. The Bernoulli variable X_2 depends on the variable X_1 where X_1 is a uniform random variable on the domain zero to one. It is related to the mean of the Y_{beta} variable through a logit link. The relationship between the mean of the response variable Y_{beta} and the X_1, X_2 variables is linear in the logit link. The parameter *logor* in equation (3.1.1) can be varied to give different correlations between X_1 and X_2. A logor value equal to 1 is used so that a moderate correlation exists between X_1 and X_2. The simulation model 1 is given in 3.1.1.

$$\begin{cases} \mathrm{X}_1 \sim Uniform(0,1), X_2 \sim Bernoulli(p), logit(p) = logor * X_1 \\ Y = 1 * Ceiling + Y_{beta} * (1 - Ceiling), Ceiling \sim Bernoulli(q) \\ q = \frac{1}{1+exp(1+\gamma_1 X_1 + \gamma_2 X_2)} \\ Y_{beta} \sim Beta(a,b), a = exp(2), b = exp(\alpha_1 X_1 + \alpha_2 X_2) \end{cases} \tag{3.1.1}$$

Two sets of α_2 and γ_2 values will be used in the simulation model. First,

$\alpha_2 = 2$ and $\gamma_2 = 0.2$ are used, so that a correlation of about 0.5 will exist between the variable Y_{beta} and X_2. Then $\alpha_2 = 0$ and $\gamma_2 = 0$ will be used, so that type-I error rates of different models can be investigated.

From the simulation model, the expected value of the response variable Y is given by

$$E(Y|X) = \frac{1}{1 + e^{1+\gamma_1 X_1 + \gamma_2 X_2}} \left(1 + \frac{e^{1+\gamma_1 X_1 + \gamma_2 X_2}}{1 + e^{-2+\alpha_1 X_1 + \alpha_2 X_2}}\right) \tag{3.1.2}$$

So the marginal effect of the variable X_2 is given by

$$E_{X_1}(\mu(X_1, 1) - \mu(X_1, 0)|X_2 = 1) \tag{3.1.3}$$

where $\mu(X_1, 1)$ is the expected value of Y for a person who has previously had a heart attack and $\mu(X_1, 0)$ is the expected value of Y for a person who has never had a heart attack before.

A histogram of 100000 simulated values of Y is produced from the model and plotted in Figure 3.1. Compared to the Diabetes Hamilton data set, the simulation model produces approximately equal proportions of observations having a value close to perfect health.

A plot of the expected value of the response Y given $X_2 = 1$ and $X_2 = 0$ respectively is shown in Figure 3.2. The expected value of Y is non-linearly related to the variable X_1 and X_2 as seen from equation 3.1.2. Interaction can be seen from the plot as the change of the response given a fixed X_1 value is different for X_2=1 and X_2=0.

We model the mean value of the response using the OLS model and the GAM model. First, both models are without an interaction term. Then an interaction between X_1 and X_2 is added to both models. For the GAM model, the variable

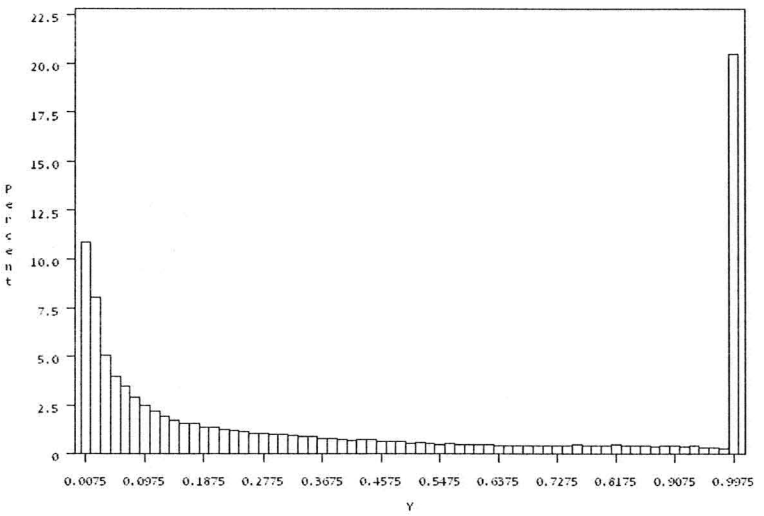

Figure 3.1: Histogram of the simulated value of Y from simulation model 1.

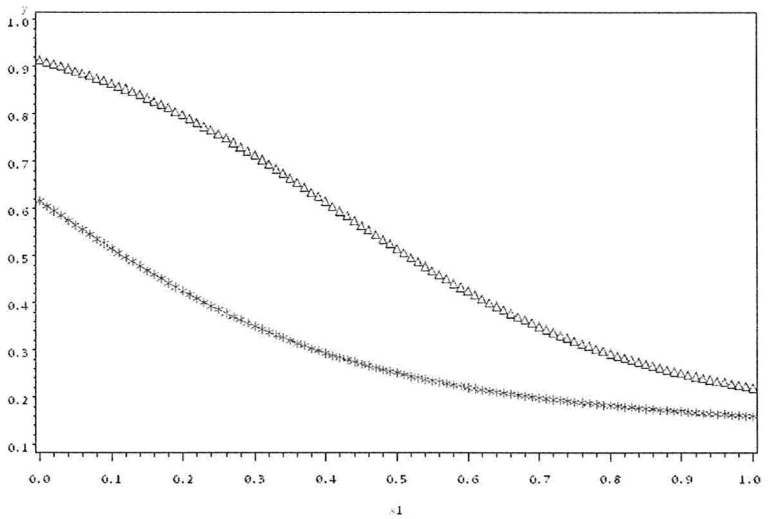

Figure 3.2: Simulation 1: Expected value of Y given $X_2 = 1$ and $X_2 = 0$ respectively where $X_2 = 1$ is represented by star (i.e., the lower curve).

X_2 is modeled to be linearly related to the response variable. The variable X_1 is modeled to be non-parametrically related to the response variable. So the GAM model contains both a parametric and non-parametric component. The OLS only contains a parametric component.

Since the expected value of Y is non-linearly related to X_1, if OLS is fitted to the generated data, it should produce a biased estimate of the marginal effect of X_2. We would like to see how the GAM model will perform for the same simulated data set. A GAM model with no interaction term is first fitted to the simulated data, then a GAM model with an interaction term is also fitted to the same simulated data.

One thousand data sets each of size 100 are simulated from the simulation model 1. An OLS model without interaction is fitted to each of the 1000 simulated data sets. For each of the 1000 data sets, the estimate of the coefficient of X_2 is computed and stored. So 1000 estimates of the coefficients of X_2 are recorded. Then the average of those 1000 estimates is calculated and compared to the true value of the marginal effect of X_2 from the simulation model to get an estimate of the bias of the model.

To calculate the true value of the marginal effect of X_2, the equation of the expected value of Y from the simulation model is used (Equation 3.1.2). Given $X_2 = 1$, the equation of the expected value of Y is integrated over the whole range of X_1, i.e. from 0 to 1. Then let X_2 to be zero in the same equation and integrate the equation over the whole range of X_1 again, from 0 to 1. The difference between the two calculated integrals will give the true value of the marginal effect of X_2. The exact value of each of the integrals is not calculated. Instead, we calculate an approximate value for each integral and then subtract the value of the integral when

$X_2 = 0$ from the value of the integral when $X_2 = 1$. To approximate the integral, we separate the integral into pieces. Since the integral can be considered as the area under the curve produced by the function 3.1.2, so we calculate the area by first separating into 5000 pieces, each of equal width. And then recalculate the integral but with 10000 pieces since by separating the area into more pieces, the calculated approximation will be closer to the true value.

3.1.1 $\alpha_2 = 2$ and $\gamma_2 = 0.2$

When 5000 pieces is used, the marginal effect of X_2 is found to be -0.2403657 and when 10000 pieces is used, the marginal effect of X_2 is found to be -0.2403775. So we take the value of -0.240378 to be the real value of the marginal effect of X_2.

The result of bias, ESE, ASE, and coverage probability (CP) for 1000 simulations are reported in Table 3.1. The bias of the OLS method without interaction is reported to be 0.0092908 and the bias of the GAM method without interaction is reported to be 0.0089532. The coverage probability of both OLS and the GAM model seems to be a little bit off from the expected value of 0.950 where the coverage probability of the GAM is 0.952 and the coverage probability of the OLS method is 0.955. The reason the coverage probability is off from the expected value could be due to random variation. When an interaction term is added to each of the models, the bias is reduced significantly for the GAM model. The bias of the GAM now becomes -0.0033721 whereas the bias of the OLS method is -0.0105578. So the GAM seems to provide a better estimate since it has a less bias than the OLS method.

To reduce the fluctuation of the estimate due to random variation, we increase the number of simulations from 1000 to 5000. The result of bias, ESE, ASE, and

Table 3.1: Comparison between OLS and GAM method for Simulation Model 1 with 1000 simulations when $\alpha_2 = 2, \gamma_2 = 0.2$.

Model($\alpha_2 = 2, \gamma_2 = 0.2$)	*Bias with 1000 simulations*	*ESE with 1000 simulations*	*ASE with 1000 simulations*	*CP with 1000 simulations*
OLS no interaction	0.0092908	0.0629193	0.0677317	0.955
GAM no interaction	0.0089532	0.0635494	0.0669408	0.952
OLS with interaction	-0.0105578	0.0678680	N/A	N/A
GAM with interaction	-0.0033721	0.0681806	N/A	N/A

Table 3.2: Comparison between OLS and GAM method for Simulation Model 1 with 5000 simulations when $\alpha_2 = 2, \gamma_2 = 0.2$.

Model($\alpha_2 = 2, \gamma_2 = 0.2$)	*Bias with 5000 simulations*	*ESE with 5000 simulations*	*ASE with 5000 simulations*	*CP with 5000 simulations*
OLS no interaction	0.0097171	0.0643309	0.0675503	0.948
GAM no interaction	0.0096444	0.0649810	0.0667545	0.945
OLS with interaction	-0.0108189	0.0701467	N/A	N/A
GAM with interaction	-0.0022346	0.0700146	N/A	N/A

coverage probability with 5000 simulations are reported in Table 3.2. When 5000 simulations are used, the bias of the OLS without interaction and GAM methods without interaction are shown to be 0.0097171 and 0.0096444 respectively. The bias of both method are close to each other. The coverage probability of the OLS method now becomes 0.948 and the coverage probability for GAM method is now 0.945. Both coverage probability are still close to the value of 0.950 when compared to the 1000 simulations result. So both models seem to be providing a good estimate. When an interaction term is added to both model, the bias of the OLS and the GAM methods are reported to be -0.0108189 and -0.0022346 respectively. So the GAM method still provides a closer estimate to the true value when 5000 simulations are used. Hence overall the GAM method seems to provide a better estimate since the bias resulting from the GAM method is less than the bias resulting from the OLS method.

3.1.2 $\alpha_2 = 0$ and $\gamma_2 = 0$

When $\alpha_2 = 0$ and $\gamma_2 = 0$ are used, the result of bias, ESE, ASE, and the coverage probability of both OLS and GAM method without interaction are shown in Table 3.3. When 1000 simulations are used, the bias of the OLS method and the GAM method are reported to be 0.0002774 and 0.0002263 respectively. Both bias values are close to each other and are close to zero which indicate both methods provide a good estimate to the true value. The coverage probability for the OLS and the GAM are reported to be 0.945 and 0.950 respectively. So both methods have about 5% chance of making Type-I error which also indicates both methods seem to be appropriate for analyzing the data generated by the simulation model.

To reduce the fluctuations of the estimate due to random variation, we in-

Table 3.3: Comparison between OLS and GAM method for Simulation Model 1 with 1000 simulations when $\alpha_2 = 0, \gamma_2 = 0$.

Model($\alpha_2 = 0, \gamma_2 = 0$)	*Bias with 1000 simulations*	*ESE with 1000 simulations*	*ASE with 1000 simulations*	*CP with 1000 simulations*
OLS no interaction	0.000277494	0.0559283	0.0577013	0.945
GAM no interaction	0.000226373	0.0567553	0.0571113	0.950

Table 3.4: Comparison between OLS and GAM method for Simulation Model 1 with 5000 simulations when $\alpha_2 = 0, \gamma_2 = 0$.

Model($\alpha_2 = 0, \gamma_2 = 0$)	*Bias with 5000 simulations*	*ESE with 5000 simulations*	*ASE with 5000 simulations*	*CP* with 5000 simulations
OLS no interaction	0.000236465	0.0572811	0.0577717	0.950
GAM no interaction	0.000114621	0.0577234	0.0572267	0.945

crease the number of simulations to 5000. The result from the simulations is shown in Table 3.4. When 5000 simulations are used, the bias of the OLS and the GAM methods are reported to be 0.0002364 and 0.0001146 respectively. There is a significant drop of bias for the GAM method, the bias is reduced by about 50% when 5000 simulations are used. The bias for the OLS method stays about the same. The bias of the GAM method is also significantly smaller than the bias produced by the OLS method. The bias given by the GAM is about half of the the bias of the OLS method.

The coverage probability of the OLS and the GAM methods are now 0.950

and 0.945 respectively. So both methods seem to be providing good estimates of the true value since both method's confidence intervals cover the true value close to 95% of the time. But overall the bias resulting from the GAM method is less than the bias resulting from the OLS method. So it seems that GAM is a better method in analyzing the data simulated from the given simulation model.

3.2 Simulation Model 2: Two-part logarithmic model

The two-part log-linear model is given in equation 3.2.4. This model is used because it reflects a closer representation of what is being observed from the Diabetes Hamilton data set. For the two-part log-linear model, the response Y is modeled to be related to the variables X_1, and X_2. The variable X_1 is taken to be the variable age. And the variable X_2 is taken to be one of the most frequent complications of the data set. So X_2 is taken to be heart attack in this case since it is the most frequent complication in the Diabetes Hamilton data set. For the response variable Y, the probability of attaining a value of 1 (i.e., full-health) is dependent on a Bernoulli variable. The parameter of the Bernoulli variable is dependent upon the variable X_1, and X_2 through a logit link. So the response variable Y is modeled as a mixture distribution with Y equal to one being modeled as a Bernoulli variable and when Y attaining value below 1 being modeled as a variable $Y_{lognormal}$ where $Y_{lognormal} = 1 - e^Z$ where $Z = log(1 - Y_{lognormal})$ is distributed normally with mean $\beta_0 + \beta_1 X_1 + \beta_2 X_2$ and variance σ^2.

$$\begin{cases} Y = 1 * Ceiling + Y_{lognormal} * (1 - Ceiling), Ceiling \sim Bernoulli(q) \\ logit(q) = \alpha_0 + \alpha_1 X_1 + \alpha_2 X_2 \\ log(1 - Y_{lognormal}) \sim N(\beta_0 + \beta_1 X_1 + \beta_2 X_2, \sigma^2) \end{cases} \tag{3.2.4}$$

From (3.2.4), to get the $\beta_0, \beta_1, \beta_2$, and σ^2, we fit a model with $log(1 - Y)$ as response, and age and heart attack as the predictors, to those who have a health utility score less than one in the Diabetes Hamilton data. Since for those who have a health utility score equal to one, their value of $log(1-Y)$ is undefined and is modeled as the variable Ceiling in 3.2.4 as shown. Ceiling is modeled as a Bernoulli variable with parameter q, where q is the probability of attaining perfect health (i.e., a value of 1). To get the value of α_0, α_1, and α_2, we let Ceiling to be 1 in the Diabetes Hamilton for those who have health utility score equal to one, and Ceiling to be zero for those who have health utility score below one. Then we model the logit of q to the variable age and heart attack and then get the α_0, α_1, and α_2 from the model.

The estimates of $\beta_0, \beta_1, \beta_2$, and σ^2 are found to be -0.95334, -0.00609, -0.09721, and $\frac{225.58497}{947}$. And the estimate of α_0, α_1, and α_2 are found to be -2.3193, 0.0126, and -0.7951.

A histogram of some 100000 generated Y values from the model is plotted in Figure 3.3. From the plot, it can be seen that there are two modes in the distribution. About 17.5% of the observations attain a value close to or equal to 1. Also a big proportion of values of Y attain values in the upper-end of the distribution. This reflects more closely to the Diabetes Hamilton data set when compared to the values simulated from simulation model 1.

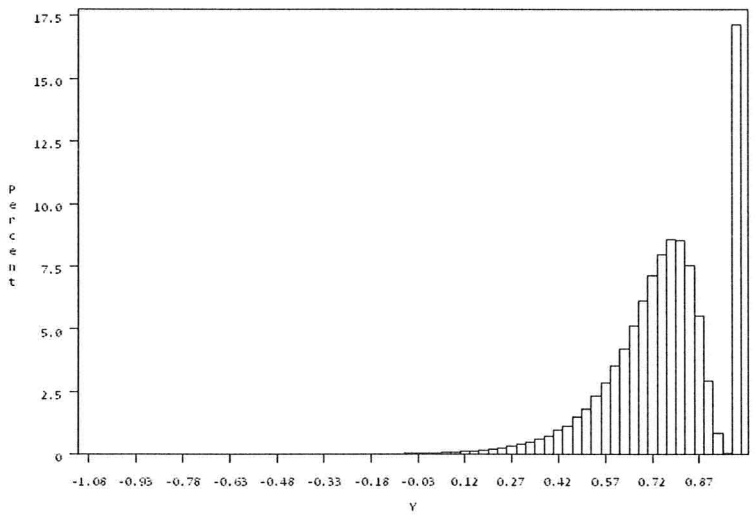

Figure 3.3: A histogram of 100000 generated values of Y from simulation model 2.

After data are simulated from the model, we fit OLS and GAM to the data. By using the $\beta_0, \beta_1, \beta_2, \sigma^2, \sigma^2, \alpha_0, \alpha_1$, and α_2 values given by fitting respective models to the Diabetes Hamilton data, we can simulate data from the model. To simulate from the model, we first pick 100 people by simple random sample from the Diabetes Hamilton data set. And then based on the value of each person's q, $\beta_0, \beta_1, \beta_2$, and σ^2, we will get the values of $\beta_0 + \beta_1 X_1 + \beta_2 X_2$, and σ^2, and also the values of $\alpha_0 + \alpha_1 X_1 + \alpha_2 X_2$. We simulate 1000 observations from each person. And by rearranging the observations, we get 1000 data sets each with 100 observations.

From both OLS and GAM methods, we would like to estimate the change of the health utility score for people who previously have had a heart attack had they not had a heart attack. We would like to find out the bias of each method. Also we would like to compare OLS and GAM methods based on certain criteria such as bias and coverage probability. In order to get the bias of each method, we need to find

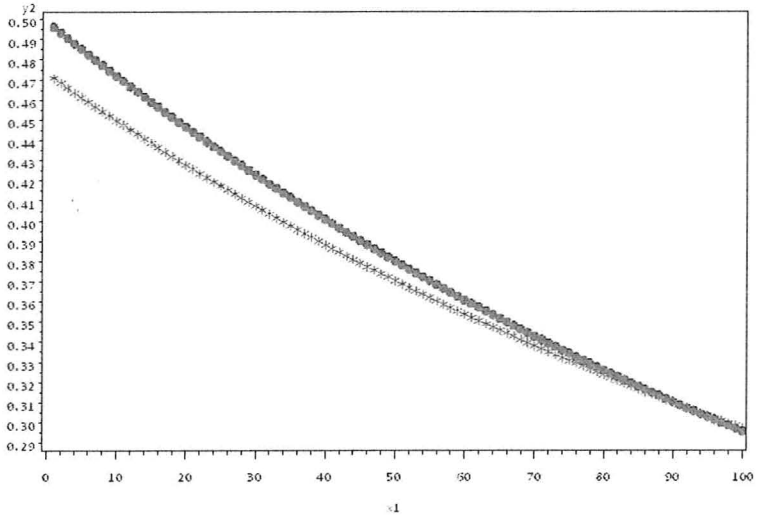

Figure 3.4: Simulation 2: Expected value of Y given $X_2 = 1$ and $X_2 = 0$ respectively where $X_2 = 1$ is represented by circle (i.e., the upper line).

out the true value of the marginal effect of heart attack.

For this simulation model, the expected value of the response Y is given by

$$E(Y|X) = q + (1-q)(1 - e^{\beta_0+\beta_1 X_1+\beta_2 X_2+\sigma^2/2}) \tag{3.2.5}$$

where $q=\frac{e^{\alpha_0+\alpha_1 X_1+\alpha_2 X_2}}{1+e^{\alpha_0+\alpha_1 X_1+\alpha_2 X_2}}$.

So the marginal effect of the variable X_2 is given by

$$E_{X_1}(\mu(X_1, 1) - \mu(X_1, 0)|X_2 = 1) \tag{3.2.6}$$

where $\mu(X_1, 1)$ is the expected value of Y for a person who has previously had a heart attack and $\mu(X_1, 0)$ is the expected value of Y for a person who has never had a heart attack.

To calculate the true value of the marginal effect of X_2, the equation of the expected value of Y from the simulation model is used (Equation 3.2.5). From the

random sample of 100 people from the Diabetes Hamilton data set, we pick those who have a heart attack value equal to one i.e. those who have had a heart attack. There are 13 people in total out of the 100. Then we take X_2 to be one in the equation 3.2.5 and substitute the value of age into the same equation for each person, and then calculate the equation's value. Then we take X_2 to be zero in the same equation, and substitute the value of age of each person, and calculate the equation's value. Then we calculate the difference of those two values for each of the 13 people. The average of those 13 difference will give us the estimate of the true marginal effect of X_2 i.e. heart attack. And it is calculated to be -0.0531128. So we take this value to be our true value.

A plot of the expected value of Y given $X_2 = 1$ and $X_2 = 0$ is shown in Figure 3.4. From the plot, it seems that the response variable is more linearly related to X_1, and X_2 than in simulation model 1.

The results of ESE, ASE, and CP for 1000 simulations are also reported in Table 3.5.

From the table, the bias produced from each model whether there is interaction or not seem to be close to each other. The bias of the OLS model and GAM methods without interaction are given to be 0.0002752, and 0.0002743, respectively. And the bias of OLS model and GAM methods with interaction are given to be 0.0002495 and 0.0002895, respectively. It seems that adding an interaction term to the model doesn't really improve the bias of the estimate. This can also be seen from Figure 3.4 where the curve that corresponds to $X_2 = 1$ and the curve that corresponds to $X_2 = 0$ seems to be more parallel to each other when compared to simulation model 1. The coverage probability of the OLS method without interaction

Table 3.5: Comparison between OLS and GAM method for Simulation Model 2 with 1000 simulations.

Model	*Bias with 1000 simulations*	*ESE with 1000 simulations*	*ASE with 1000 simulations*	*CP with 1000 simulations*
OLS no interaction	0.0002752	0.0559021	0.0537033	0.943
GAM no interaction	0.0002743	0.0561183	0.0533899	0.941
OLS with interaction	0.0002495	0.0560295	N/A	N/A
GAM with interaction	0.0002895	0.0561718	N/A	N/A

Table 3.6: Comparison between OLS and GAM method for Simulation Model 2 with 5000 simulations.

Model	*Bias with 5000 simulations*	*ESE with 5000 simulations*	*ASE with 5000 simulations*	*CP with 5000 simulations*
OLS no interaction	0.0002753	0.0539125	0.0537516	0.948
GAM no interaction	0.0003435	0.0540007	0.0534436	0.945
OLS with interaction	0.0002660	0.0537937	N/A	N/A
GAM with interaction	0.0003470	0.0538759	N/A	N/A

is 0.943. And for the GAM method without interaction, the coverage probability is 0.941. Both coverage probabilities are off from the expected value of 0.950. It could be due to random variation. So instead of 1000 simulations, 5000 simulations could be used.

From the 5000 simulations, the coverage probability of both OLS and GAM improve as shown in Table 3.6. The GAM coverage probability becomes 0.945 which is much closer to the expected value of 0.950. For the OLS method, the coverage probability is now 0.948 and is also closer to 0.950. The bias of the OLS without interaction and the GAM without interaction stay about the same when more simulations are run. The bias of the OLS and GAM are now found to be 0.0002753 and 0.0003435, respectively. And when an interaction term is added to both models, for the 5000 simulations, the bias becomes 0.000266 and 0.000347, respectively for the OLS and the GAM. There is not much of an improvement in the bias.

As a verification to the true value of the marginal effect of X_2 (i.e., heart attack) computed earlier, we also fit a two-part log-linear model to the generated data. The estimate resulting from the two-part log-linear model should be close to the true value and should give a zero bias estimate since the data are generated originally from a two-part log-linear model. We fit the two-part log-linear model to the same 1000 simulations data set generated in simulation model 2. The estimate of the bias and ESE are reported in Table 3.7. As seen from the table, the estimate is close to the true value with a bias of 0.0024654. But since the true value is also an estimate, so the bias could have been magnified. The true value is correct to 7 decimals point. The empirical standard error of the estimate is 0.0544059.

Overall, both the OLS and the GAM methods seem to give similar results

Table 3.7: Fitting the two-part log-linear model to Simulation Model 2.

Model	*True value*	*Estimate*	*Bias*	*ESE*
Two-Part Log-linear	-0.0531228	-0.0555882	0.0024654	0.0544059

when applied to the simulation 2 data. The bias of both methods are close to each other, and coverage probabilities are also close to each other.

Chapter 4

Application to Real Data Set

For the Diabetes Hamilton data, from the plot of residuals against the predictor variables in Figures 2.10 and 2.11, some of the variables seem to be having a non-linear relationship with the response variable. Curvatures are observed when residuals are plotted against some of the predictors. From the q-q plot in Figure 2.4, non-normality also seems to be observed from the data. We would like to fit the data using the generalized additive models since non-linearity can be handled by specifying non-parametric terms in the models.

From the residuals plot against the predictors, curvature is observed for the predictors age, and duration of diabetes. For our generalized additive models, non-linear relationship is going to be assumed between the response and the predictors of age, and duration of diabetes. And for other predictors such as amputation, stroke, heart attack, kidney failure, and gender, the relationship is assumed to be linear. Overall, the response variable is modeled to be semi-parametrically related to the predictors with spline applied to the variable age, and duration of diabetes.

Table 4.1: Estimates of parameters and their standard error and p-value with the use of Generalized additive models with splines for the parameters age, and duration of diabetes.

Variable	*Parameter Estimate (OLS)*	*Parameter Estimate (GAM)*	*Standard Error (GAM)*	*t value (GAM)*	*P-value (GAM)*
Intercept	0.51736	0.50991	0.03799	13.42	<0.0001
Foot Leg Amputation	-0.06311	-0.06029	0.05224	-1.15	0.2487
Stroke	-0.04617	-0.04655	0.02260	-2.06	0.0397
Heart Attack	-0.05859	-0.05636	0.01716	-3.28	0.0011
Kidney Failure	-0.10182	-0.10429	0.03751	-2.78	0.0055
Linear (Age)	0.00291	0.00303	0.00055703	5.44	<0.0001
Gender	0.05151	0.05204	0.01152	4.52	<0.0001
Linear (Duration of Diabetes)	-0.00152	-0.00161	0.00061107	-2.63	0.0087

The analysis result given by the GAM method is given in Table 4.1. From the output, each of the parameters are shown to be significant at the 10% level except the parameter amputation. The estimates of the parameters of amputation, stroke, heart attack, kidney failure, and gender are given to be -0.06029, -0.04655, -0.05635, -0.1429, and 0.05204, respectively.

For age and duration of diabetes, the linear part of the estimates are given to be 0.00303 and -0.00161 respectively. The p-values of each of the parameters of amputation, stroke, heart attack, and kidney failure are given to be 0.2487, 0.0397, 0.0011, and 0.0055, respectively. Compared to the p-value given by the OLS model, the respective p-values are about the same. The estimates of the coefficients of the parameters are also about the same. The parameters that are significant in OLS analysis are also shown to be significant in the GAM analysis. Overall the OLS and GAM seem to give similar results in terms of the estimates of the parameters and the significance of each parameter.

Chapter 5

Conclusion & Future Research

Generalized additive models give more options in representing the relationship between the response and the predictors variables. The relationship can be modeled to be non-linearly related. More flexibility is given using the GAM models instead of the OLS where a linear relationship is assumed. Health utility data exhibit features such as heteroscedasticity and non-normality. The assumption of normality in the use of inference of parameters for the OLS method is not appropriate.

On the other hand, GAM can uncover non-linearity in the relationship between the response and the predictors variable. From the simulation studies in Chapter 3, the estimate of the parameters of interest produced by the GAM method is in general closer to the true value than the estimate produced by the OLS method. So GAM method give a smaller bias. Both coverage probabilities are close to the expected value of 0.950. The confidence interval of the GAM method covers the true parameter value 95% of the time, which gives us an indication that the GAM method is a valid method for analyzing data like those from this study.

In the fitting of the GAM models with parametric terms, when smoothing is done by a spline function, homoscedasticity of errors terms is assumed [Durban et al., 1999]. But from the residual plot of the Diabetes Hamilton data shown previously in Figure 2.2, the error terms don't seem to attain constant variance. So the assumption of constant variance doesn't hold and inference of the significance of parameters might not be appropriate. So in this case, bootstrapping of the standard errors might be more appropriate since the assumption is violated. But overall, the bias of the estimate given by the GAM method is generally smaller and hence can be a good alternative model for estimation of the parameters of interest.

Other models that could be investigated in the future include the TPM (two-part models) and the LCM (latent class models). The two-part models can be used to model the ceiling effect whereas the latent class models can be used to model a bi-modal distribution which is also what being observed in our real data set. Also in analyzing health utility data, since homoscedasticity of error term doesn't generally hold, further work on developing models that can handle such situation might be valuable instead of relying on bootstrapping.

Appendix

```
/* Simulation Model 1 */
/*Part 1: Get the real-value of the marginal effect of X2*/
data sum;
do i=1 to 10000;
y=((1/(1+exp(1+0.5*(i/10000)+0.2)))*
(1+((exp(1+0.5*(i/10000)+0.2))/(1+exp(-2+5*(i/10000)+2))))-
((1/(1+exp(1+0.5*(i/10000))))*(1+((exp(1+0.5*i/10000))/(1+exp(-2+5*i/10000))))))/10000;
output;
end;
run;
data sum2;
set sum;
if first.i then cum_y=y;
else cum_y+y;
run;
/*Compare between the integrals when we separate the integral into 10000 pieces and
when we sepaparte it into 5000 pieces*/
data sum3;
do i=1 to 5000;
y=((1/(1+exp(1+0.5*(i/5000)+0.2)))*
(1+((exp(1+0.5*(i/5000)+0.2))/(1+exp(-2+5*(i/5000)+2))))-
((1/(1+exp(1+0.5*(i/5000))))*(1+((exp(1+0.5*i/5000))/(1+exp(-2+5*i/5000))))))/5000;
output;
end;
run;
data sum5;
```

```
set sum3;
if first.i then cum_y=y;
else cum_y+y;
run;
/*Part 1a: For Simulation model 1, GAM without interaction onto x1 and x2*/
/* alpha_2 = 2 and gamma_2 = 0.2 */
data simulate;
call streaminit(1234);
do i=1 to 5000;
do j=1 to 100;   /* each iteration has 100 observations, the 100 reflect the
real life situation and also if testing for model, if it works
for 100 obs, it will also work for 1000 obs*/
x1=rand('uniform');
p=exp(x1)/(1+exp(x1));
x2=rand('bernoulli',p);
q=1/(1+exp(1+0.5*x1+0.2*x2));
ceiling=rand('bernoulli',q);
a=exp(2);
b=exp(5*x1+2*x2);
Ybet=rand('beta',a,b);
Y=ceiling+Ybet*(1-ceiling);
output;
end;
end;
run;
ods output "Parameter Estimates"=gam_para_est;
proc gam data=simulate;
model Y=param(x2) spline(x1);
by i;
run;
proc sql;
create table gam_est_table as
select Parameter, Estimate, StdErr, Estimate-1.96*StdErr as low_95, Estimate+1.96*StdErr as up_95
from gam_para_est
where Parameter='x2';
run;
```

```
data gam_est;
set gam_est_table;
if low_95 LT -0.240377 AND up_95 GT -0.240377 Then indicator=1;
else indicator=0;
run;
/* with 1000 simulations the coverage probability is 0.952 */
/* with 5000 simulations the coverage probability is 0.9444 */
proc means data=gam_est;
var indicator;
run;
/* with 1000 sim, the ESE=0.0635494 ASE=0.0669408 */
/* with 5000 sim, the ESE=0.0649810 ASE=0.0667545 */
proc means data=gam_est_table;
var estimate stderr;
run;
/* with 1000 sim, the model's estimate is -0.2493302, so bias of gam= 0.0089532 */
/* with 5000 sim, the model's estimate is -0.2500214, so bias of gan= 0.0096444 */
data bias_gam;
real=-0.240377;
gam_model=-0.2500214;
bias=real-gam_model;
run;
/* 1b:  OLS without interaction*/
ods output "Parameter Estimates"=ols_para_estimates;
proc reg data=simulate;
model Y=x1 x2;
by i;
run;
ods output close;
proc sql;
create table ols_est_table as
select Variable, Estimate, StdErr, Estimate-1.96*StdErr as low_95, Estimate+1.96*StdErr as up_95
from ols_para_estimates
where Variable='x2';
run;
data ols_est;
```

```
set ols_est_table;
if low_95 LT -0.240377 and up_95 GT -0.240377 then indicator=1;
  else indicator=0;
run;
/*with 1000 sim, the coverage probability is 0.955 */
/*with 5000 sim, the coverage probability is 0.9504 */
proc means data=ols_est;
var indicator;
run;
/*with 1000 sim, the ESE=0.0629193 ASE=0.0677317 */
/*with 5000 sim, the ESE=0.0643309 ASE=0.0675503 */
proc means data=ols_est_table;
var estimate stderr;
run;
/* with 1000 sim, the model's estimate is -0.2496678, so bias of ols= 0.0092908 */
/* with 5000 sim, the model's estimate is -0.2500941, so bias of ols= 0.0097171 */
data bias_ols;
real=-0.240377;
ols_model=-0.2500941;
bias=real-ols_model;
run;
/* Part 2a: GAM method with interaction*/
data gam_with_interaction;
set simulate;
x11=x1*x2;
x10=x1*(1-x2);
run;
proc sql;
create table gam_with_int as
select i,j,x1,p,x2,y,x11,x10
from gam_with_interaction;
run;
data gam_x21_int;
set gam_with_int;
where x2=1;
run;
```

```
proc gam data=gam_with_interaction;
model y=param(x2) spline(x11) spline(x10);
score data=gam_x21_int out=gam_x21_int_pred;
by i;
run;
data gam_x20_int;
set gam_x21_int;
x2=0;
x11=x1*x2;
x10=x1*(1-x2);
run;
proc gam data=gam_with_interaction;
model y=param(x2) spline(x11) spline(x10);
score data=gam_x20_int out=gam_x20_int_pred;
by i;
run;
data mergegam_int;
merge gam_x21_int_pred (drop=x11 x10 x2 P_x11 P_x10) gam_x20_int_pred (drop=x11 x10 x2
P_x11 P_x10 IN=gam_x20_int_pred rename=(P_Y=P_Yx20));
run;
data gam_wint;
set mergegam_int;
diff=P_y-P_Yx20;
run;
ods output Summary=differences_gam_int;
proc means data=gam_wint;
var diff;
by i;
run;
ods trace off;
ods output close;
/* with 1000 sims, the 1000 average differences = -0.2370049 and the ESE=0.06818064*/
/* with 5000 sims, the 5000 average differences = -0.2381424 and the ESE=0.0700146 */
ods output Summary=gam_with_int_average;
proc means data=differences_gam_int;
var diff_mean;
```

```
run;
ods output close;
/* with 1000 sims,  the bias of gam =-0.0033721 */
/* with 5000 sims,  the bias of gam =-0.0022346 */
data bias_gam_int;
real=-0.240377;
gam_model=-0.2381424;
bias=real-gam_model;
run;

/* Part 2b: Ordinary linear regression model with interaction*/
data ols_int;
set simulate;
x3=x1*x2;
run;
proc sql;
create table ols_with_int as
select i,j,x1,p,x2,x3,y
from ols_int;
run;
data olsx21_int;
set ols_with_int;
where x2=1;
run;
proc gam data=ols_with_int;
model y=param(x1 x2 x3);
score data=olsx21_int out=olsx21_int_pred;
by i;
run;
data olsx20_int;
set olsx21_int;
x2=0;
x3=x1*x2;
run;
proc gam data=ols_with_int;
model y=param(x1 x2 x3);
```

```
score data=olsx20_int out=olsx20_int_pred;
by i;
run;
data mergeols_int;
merge olsx21_int_pred (drop=x1 x3 x2) olsx20_int_pred (drop=x1 x3 x2 IN=olsx20_int_pred
rename=(P_Y=P_Yx20));
run;
data ols_interaction;
set mergeols_int;
diff=P_y-P_Yx20;
run;

ods output Summary=differences_ols_int;
proc means data=ols_interaction;
var diff;
by i;
run;
ods trace off;
ods output close;

     /* with 1000 sims, the average turns out to be -0.2297792 and the ESE=0.0678680413 */
    /* with 5000 sims, the average turns out to be -0.2295171 and the ESE=0.0701467 */
ods output Summary=ols_with_int_average;
proc means data=differences_ols_int;
var diff_mean;
run;
ods output close;

/* with 1000 sims, bias of ols = -0.0105578 */
/* with 5000 sims, bias of ols = -0.0108189 */
data bias_ols_int;
real=-0.240337;
model_ols=-0.2295171;
bias_ols=real-model_ols;
run;
```

```
/*Part 3: get type 1 error  alpha_2 =0   gamma_2 =0 */
 /* ols without interaction*/
data type_1;
call streaminit(1234);
do i=1 to 5000;
do j=1 to 100; /*each interaction has 100 observations*/
x1=rand('uniform');
p=exp(x1)/(1+exp(x1));
x2=rand('bernoulli',p);
q=1/(1+exp(1+0.5*x1));
ceiling=rand('bernoulli',q);
a=exp(2);
b=exp(5*x1);
Ybet=rand('beta',a,b);
Y=ceiling+Ybet*(1-ceiling);
output;
end;
end;
run;

ods output "Parameter Estimates"=ols_para_est;
proc reg data=type_1;
model y=x1 x2;
by i;
run;
ods output close;

proc sql;
create table ols_table as
select i, Variable, Estimate, StdErr, Estimate-1.96*StdErr as lower95,
Estimate+1.96*StdErr as upper95
from ols_para_est
where variable='x2';
run;
data ols_type1_error;
set ols_table;
```

```
if lower95 LT 0 AND upper95 GT 0 then grade=1;
else grade=0;
run;

/*with 1000 sim, the CP is 0.9450 , so type one error is 5.5% */
/*with 5000 sim, the CP is 0.9504*/
proc means data=ols_type1_error;
var grade;
run;
/* since the true value is 0,
for 1000 sim, the bias is the average of the estimate which is
0.000277494, ESE=0.0559283, ASE=0.0577013
for 5000 sim, the bias is the average of the estimate which is
0.000236465, ESE=0.0572811, ASE=0.0577717*/

proc means data=ols_table;
var estimate StdErr;
run;

/* for gam without interaction*/
ods output "Parameter Estimates"=gam_para_est;
proc gam data=type_1;
model y=param(x2) spline(x1);
by i;
run;
ods output close;

proc sql;
create table gam_table as
select i, Parameter, Estimate, StdErr, Estimate-1.96*StdErr as lower95,
Estimate+1.96*StdErr as upper95
from gam_para_est
where Parameter='x2';
run;
data gam_type1_error;
set gam_table;
```

```
if lower95 LT 0 AND upper95 GT 0 then grade=1;
else grade=0;
run;

/* with 1000 sims, the coverage turns out to be 0.95, so type-I error is only 5%  */
/* with 5000 sims, the CP=0.9464*/
proc means data=gam_type1_error;
var grade;
run;

/* the true value is 0,
    for 1000 sims, so the bias is the average of the estimate which is
    0.000226373, ESE=0.0567553, ASE=0.0571113
for 5000 sims, so the bias is the average of the estimate which is
0.000114621, ESE=0.0577234, ASE=0.0572267*/

proc means data=gam_table;
var Estimate StdErr;
run;

/*  For simulation model 2:  refit the OLS no interaction and GAM
no interaction with the use of 1000 or 5000 data sets this time,
and see if the coverage probability of each will get improved*/
data diat;
        set diabetefile;
        keep EQ5D_US footlegamput stroke heartattack kidneyfailure age gender durdiab;
        if Footlegamput in (1 2) then Footlegamput=1;
        if Footlegamput=3 then Footlegamput=0;
        if Stroke in (1 2) then Stroke=1;
        if Stroke=3 then Stroke=0;
        if HeartAttack in (1 2) then HeartAttack=1;
        if HeartAttack=3 then heartattack=0;
        if kidneyfailure in (1 2) then kidneyfailure=1;
        if kidneyfailure=3 then kidneyfailure=0;
        run;
proc univariate data=diat;
```

```
var EQ5D_US footlegamput stroke heartattack kidneyfailure age gender durdiab;
run;
data diat2;
      set diat;
      keep EQ5D_US heartattack age;
      run;

/* Part 2: simulate and rearrange so that there are 1000 data sets each with 100
observations */
data diabetes;
      set diat2;
      if EQ5D_US=1 then prob=1;  else prob=0;   zrep=log(1-EQ5D_US);
run;
proc reg data=diabetes;
model zrep=age heartattack;
run;
data diabeteshamilton;
      set diabetes;
      beta0=-0.95334; beta1=-0.00609; beta2=0.09721;
      alpha0=-2.3153; alpha1=0.0126; alpha2=-0.7951;
      q=exp(alpha0+age*alpha1+heartattack*alpha2)/(1+exp(alpha0+age*alpha1+heartattack*alpha2));
      me=beta0+beta1*age+beta2*heartattack;
      sig2=225.58457/947;    sig=sqrt(sig2);
      run;

      /* Get 100 values from a random sample of 100*/
proc surveyselect data=diabeteshamilton out=sample method=SRS SAMPSIZE=100 seed=12321;
      run;

data sim2simulate;
call streaminit(1234);
      set sample;
      do j=1 to 5000;
                  ceiling=rand('bernoulli',q);
                  z=rand('normal',me,sig);
                  Y_log=1-exp(z);
```

```
                        Y=ceiling+(1-ceiling)*Y_log;
output;
        end;
run;

/* Part 1: first OLS with no interaction is used*/
proc sort data=sim2simulate;
        by j;
        run;

ods output "Parameter Estimates"=para_estimate;
proc reg data=sim2simulate;
        model Y=age heartattack;
        by j;
        run;
ods output close;

proc sql;
        create table olsim2 as
        select j, Variable, Estimate, StdErr, Estimate-1.96*StdErr as lower95,
        Estimate+1.96*StdErr as upper95
        from para_estimate
        where Variable="HeartAttack";
        run;

data ols2final;
        set olsim2;
        if lower95 LT -0.0531228 AND upper95 GT -0.0531228 then grade=1;
        else grade=0;
        run;

/* the Coverage probability for ols with 1000 simulations is 0.943 */
/* the Coverage probability for ols with 5000 simulations is 0.948*/

proc means data=ols2final;
        var grade;
```

```
        run;
/* with 1000 simulations ESE=0.0559021, ASE=0.0537033*/
/* with 5000 simulations ESE=0.0539125, ASE=0.0537516*/

proc means data=ols2final;
        var estimate stderr;
        run;
/* with 1000 simulations, ols_no_interaction=-0.053398*/
/* with 5000 simulations, ols_no_interaction=-0.0533981*/
/* with 1000 simulations, bias=0.0002752*/
/* with 5000 simulations, bias=0.0002753*/

data biasols;
        real_marginal=-0.0531228;
        olsmodel_marginal=-0.0533981;
        bias=real_marginal-olsmodel_marginal;
        run;

/* part 2: GAM with no interaction*/
ods output "Parameter Estimates"=gampara;
proc gam data=sim2simulate;
        model y=param(heartattack) spline(age);
        by j;
        run;
ods output close;

proc sql;
        create table gamsim2 as
        select j, Parameter, Estimate, StdErr, Estimate-1.96*StdErr as lower95,
        Estimate+1.96*StdErr as upper95
        from gampara
        where Parameter="HeartAttack";
        run;

/* for coverage probability*/
data gamsim2final;
```

```
        set gamsim2;
        if lower95 LT -0.0531128 AND upper95 GT -0.0531128 then grade=1;
        else grade=0;
        run;

/* with 1000 sim, GAM's CP=0.941*/
/* with 5000 sim, GAM'S CP=0.945*/
proc means data=gamsim2final;
        var grade;
        run;

/* with 1000 sim, GAM's ESE=0.0561183, ASE=0.0533899  */
/* with 5000 sim, GAMS' ESE=0.0540007, ASE=0.0534436 */
proc means data=gamsim2final;
        var estimate stderr;
        run;

/* with 1000 simulations, gam_no_interaction=-0.0533871*/
/* with 5000 simluations, gam_no_interaction=-0.0534563*/
/* with 1000 simulations, GAM's Bias=0.0002743 */
/* with 5000 simulations, GAM's Bias=0.0003435 */
data biasgam;
        real_marginal=-0.0531128;
        gam_marginal=-0.0534563;
        bias=real_marginal-gam_marginal;
        run;
/* Part 3: OLS with interaction*/
data ols_x3;
set sim2simulate;
x3=age*heartattack;
run;
proc sql;
create table olsdata as
select j,age,heartattack,x3,y
from ols_x3;
run;
```

```
data ols_heartone;
set olsdata;
where heartattack=1;
run;

proc gam data=olsdata;
model y=param(age heartattack x3);
score data=ols_heartone out=ols_heartone_pred;
by j;
run;
data ols_heartzero;
set ols_heartone;
heartattack=0;
x3=age*heartattack;
run;
proc gam data=olsdata;
model y=param(age heartattack x3);
score data=ols_heartzero out=ols_heartzero_pred;
by j;
run;
data ols_merge_heart;
merge ols_heartone_pred (drop=age x3 heartattack)
ols_heartzero_pred (drop=age x3 heartattack IN=ols_heartzero_pred rename=(P_Y=P_Yx20));
run;
data ols_difference;
set ols_merge_heart;
diff=P_Y-P_Yx20;
run;

ods output Summary=ols_sim2;
proc means data=ols_difference;
var diff;
by j;
run;
ods trace off;
ods output close;
```

```
/* Across 1000 simulations, ESE=0.0560295 ASE=0.0410332*/
/* Across 5000 simulations, ESE=0.0537937 ASE=0.041357 */
proc means data=ols_sim2;
var diff_Mean diff_StdDev;
run;
ods output Summary=ols_with_interaction;
proc means data=ols_sim2;
var diff_mean;
run;
ods output close;

/* with 1000 simulations, ols_with_interaction=-0.0533723*/
/* with 5000 simulations, ols_with_interaction=-0.0533888*/
/* for 1000, the bias=0.0002495*/
/* for 5000, the bias=0.000266*/
data ols_bias_int;
ols_marg=-0.0533723;
real_marg=-0.0531228;
bias=real_marg-ols_marg;
run;
/* Part 4: GAM with interaction*/
data gamx11x10;
set sim2simulate;
x11=age*heartattack;
x10=age*(1-heartattack);
run;
proc sql;
create table gamdata as
select j,age,heartattack,y,x11,x10
from gamx11x10;
run;
data gam_heartone;
set gamdata;
where heartattack=1;
run;
```

```
proc gam data=gamx11x10;
model y=param(heartattack) spline(x11) spline(x10);
score data=gam_heartone out=gam_heartone_pred;
by j;
run;
data gam_heartzero;
set gam_heartone;
heartattack=0;
x11=age*heartattack;
x10=age*(1-heartattack);
run;
proc gam data=gamx11x10;
model y=param(heartattack) spline(x11) spline(x10);
score data=gam_heartzero out=gam_heartzero_pred;
by j;
run;
data gam_merge_heart;
merge gam_heartone_pred(drop=x11 x10 heartattack P_x11 P_x10)
gam_heartzero_pred (drop=x11 x10
heartattack P_x11 P_x10 IN=gam_heartzero_pred rename=(P_Y=P_Yx20));
run;
data gam_difference;
set gam_merge_heart;
diff=P_Y-P_Yx20;
run;

ods output Summary=gam_sim2;
proc means data=gam_difference;
var diff;
by j;
run;
ods trace off;
ods output close;

/* For 1000 simulations: ESE=0.0561718 ASE=0.0741193*/
/* For 5000 simulations: ESE=0.0538759 ASE=0.074346*/
```

```
proc means data=gam_sim2;
var diff_Mean diff_StdDev;
run;
ods output Summary=gam_with_interaction;
proc means data=gam_sim2;
var diff_mean;
run;
ods output close;

/* with 1000 simulations, gam_with_interaction=-0.0534123*/
/* with 5000 simulations, gam_with_interaction=-0.0534698*/
/* with 1000 simulations, the bias=0.0002895*/
/* with 5000 simulations, the bias=0.000347*/
data gam_bias_int;
gam_marg=-0.0534123;
real_marg=-0.0531228;
bias=real_marg-gam_marg;
run;

/* For the simulation model 2: Fit the two-part log-linear model and check
that the marginal effect from this model should be close to the true
value and so the bias should be close to zero*/
proc import datafile='D:\SASsoftware\SAS9.1.3Portable\DiabetesUtilityFinal2nodob.csv'
        out=work.diabetefile; delimiter=',';
        run;
data diat;
        set diabetefile;
        keep EQ5D_US footlegamput stroke heartattack kidneyfailure age gender
durdiab;
        if Footlegamput in (1 2) then Footlegamput=1;
        if Footlegamput=3 then Footlegamput=0;
        if Stroke in (1 2) then Stroke=1;
        if Stroke=3 then Stroke=0;
        if HeartAttack in (1 2) then HeartAttack=1;
        if HeartAttack=3 then heartattack=0;
        if kidneyfailure in (1 2) then kidneyfailure=1;
```

```
        if kidneyfailure=3 then kidneyfailure=0;
        run;
data diat2;
        set diat;
        keep EQ5D_US heartattack age;
        run;

/* Part 2: simulate and rearrange so that there are 1000 data sets each
with 100 observations */
data hamilton;
        set diat2;
        zrep=log(1-EQ5D_US);
run;

proc reg data=hamilton;
model zrep=age heartattack;
run;
data hamilton_diabetes;
        set hamilton;
        beta0=-0.95334; beta1=-0.00609; beta2=0.09721;
        alpha0=-2.3153; alpha1=0.0126; alpha2=-0.7951;
        q=exp(alpha0+age*alpha1+heartattack*alpha2)/
        (1+exp(alpha0+age*alpha1+heartattack*alpha2));
        me=beta0+beta1*age+beta2*heartattack;
        sig2=225.58457/947;
        sig=sqrt(sig2);
        run;

        /* Get 100 values from a random sample of 100*/
proc surveyselect data=hamilton_diabetes out=samp method=SRS SAMPSIZE=100 seed=12321;
        run;
/*data samp1;
        set samp;
        if first.EQ5D_US then id=1;
                 else id+1;
                run;*/
```

```
data simulate;
        call streaminit(1234);
        set samp;
        do j=1 to 1000;
                        ceiling=rand('bernoulli',q);
                        z=rand('normal',me,sig);
                        Y_log=1-exp(z);
                        Y=ceiling+(1-ceiling)*Y_log;
                        output;
        end;
run;
proc univariate data=simulate;
histogram y;
run;
data ceilingzero;
        set simulate;
        where ceiling=0;
        run;
proc sort data=ceilingzero;
        by j;
        run;
proc sort data=simulate;
        by j;
        run;
proc gam data=ceilingzero;
        model z=param(age heartattack);
        score data=simulate out=heartone;
        by j;
        run;
/* get the estimate of the sigma^2*/
ods output "Analysis of Variance"=anala_var;
proc reg data=ceilingzero;
model z=age heartattack;
by j;
run;
ods output close;
```

```
proc sql;
create table error_term as
select j, Source, MS
from anala_var
where Source="Error";
run;
proc sort data=simulate;
by j;
run;
proc logistic data=simulate descending;
  model ceiling = age heartattack;
score data=simulate out=prob_heart1;
  by j;
  run;

/* Part 3: now put everything together*/
data one;
merge prob_heart1 heartone;
run;
data two;
set one;
keep age heartattack P_1 P_0 P_z j;
run;
data three;
merge two error_term;
by j;
run;
data four;
set three;
where heartattack=1;
run;

/* now try to get marginal effect when x2=0*/
data heartzero;
set four;
```

```
heartattack=0;
run;
proc gam data=ceilingzero;
model z=param(age heartattack);
score data=heartzero out=z_heartzero;
by j;
run;
proc logistic data=simulate descending;
  model ceiling = age heartattack;
score data=heartzero out=prob_heart0;
  by j;
  run;
data z_heartzero2;
set z_heartzero;
predzero=P_z;
run;
data z_heartzero3;
set z_heartzero2;
drop P_z;
run;
data last;
merge z_heartzero3 prob_heart0;
run;

/* Part 4: now get the difference of the expected value of Y when X2=1 and X2=0,
i.e. when Heartattack is present and when hearattack is absent*/
data last2;
set last;
Y_X2one=P_1+(P_0)*(1-exp(P_z+MS/2));
Y_X2zero=P_12+(P_02)*(1-exp(predzero+MS/2));
diff=Y_X2one-Y_X2zero;
run;

/* get the average of the difference within each of the 1000 data sets*/
ods output Summary=difference1000;
proc means data=last2;
```

```
var diff;
by j;
run;
ods output close;

/* now get the average across the 1000 values of the "averages" to be the marginal
effect of hearattack value produced by fitting the two-part log-linear model*/
/* the average is calculated to be -0.0555882*/
/* and the ESE is equal to 0.0544059       */
ods output Summary=two_part_log;
proc means data=difference1000;
var diff_mean;
run;
ods output close;
/* Part 5: real-value of the marginal effect of heartattack*/
/*the answer turns out to be -0.0531228*/
data real_marg;
set samp;
where heartattack=1;
diff= exp(beta0+beta1*age+sig2/2)-(exp(alpha0+alpha1*age)/(1+exp(alpha0+alpha1*age)))*exp(beta0
+beta1*age+sig2/2)
-exp(beta0+beta1*age+beta2+sig2/2)+
(exp(alpha0+alpha1*age+alpha2)/(1+exp(alpha0+alpha1*age+alpha2)))
*exp(beta0+beta1*age+beta2+sig2/2);
run;
proc means data=real_marg;
var diff;
run;
/* Part 6: Bias of the two-part log-linear model*/
/* the bias is calculated to be 0.0024654*/
data bias_two_part;
real=-0.0531228;
model=-0.0555882;
bias=real-model;
run;
```

Bibliography

A. Burton, D.G. Altman, P. Royston, and R.L. Holder. The design of simulation studies in medical statistics. *Statistics in Medicine*, 25:4279–4292, 2006.

P. Clarke, A. Gray, and R. Holman. Estimating utility values for health states of type 2 diabetic patients using the EQ5D. *Medical Decision Making*, 22:340–348, 2002.

W.S. Cleveland. Robust locally-weighted regression and smoothing scatterplots. *Journal of the American Statistical Association*, 74:829–836, 1979.

M. Durban, C. A. Hackett, and I.D. Currie. Approximate standard errors in semi-parametric models. *Biometrics*, 55:699–703, 1999.

B. Efron and R.J. Tibshirani. *An Introduction to the Bootstrap*. Chapman and Hall, New York, 1993.

L.H. Ervin and J.S. Long. Using heteroscedasticity consistent standard errors in the linear regression model. *Journal of the American Statistical Association*, 54: 217–24, 2000.

D. Feeney, W. Furlong, M. Boyle, and G. W. Torrance. Multi-attribute health status

classification systems: Health utilities index. *PharmacoEconomics*, 7:490–502, 1995.

J.H. Friedman and W. Stuetzle. Projection pursuit regression. *Journal of the American Statistical Association*, 76:817–823, 1981.

T.J. Hastie and R.J. Tibshirani. Generalized additive models. *Statistical Sciences*, 1 Number 3:297–318, 1986.

T.J. Hastie and R.J. Tibshirani. *Generalized Additive Models*. Chapman and Hall, Bury St Edmunds, Suffolk, Great Britain, 1990.

D.P. Kernick. Introduction to health economics for the medical practitioner. *Postgraduate Medical Journal*, 79:147–150, 2003.

D.J. O'Reilly, F. Xie, E.M. Pullenayegum, H.C. Gerstein, J. Greb, G.K. Blackhouse, J.E. Tarride, J. Bowen, and R.A. Goeree. Estimation of the impact of diabetes-related complications on health utilities for patients with type 2 diabetes in ontario, canada. *Quality of Life Research (in press)*.

C. Reinsch. Smoothing by spline functions. *Numerical Mathematics*, 10:177–83, 1967.

J.L. Schafer and J.W. Grapham. Missing data: our view of the state of the art. *Psychological Methods*, 7:147–177, 2002.

J.W. Shaw, J.A. Johnson, and S.J. Coons. US valuation of the EQ5D health states development and testing of the D1 valuation model. *Medical Care*, 43 Number 3: 203–220, 2005.