Ş.KILIÇOĞLU

ATILIM UNIVERSITY 2022

COMPARISON AND ASSESSMENT OF SHRINKAGE METHODS IN CASE OF MULTICOLLINEARITY PROBLEM

THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
ATILIM UNIVERSITY

ŞEVVAL KILIÇOĞLU

A MASTER OF SCIENCE THESIS
IN
THE DEPARTMENT OF INDUSTRIAL ENGINEERING

JUNE 2022

COMPARISON AND ASSESSMENT OF SHRINKAGE METHODS IN CASE OF MULTICOLLINEARITY PROBLEM

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
ATILIM UNIVERSITY

BY

ŞEVVAL KILIÇOĞLU

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
THE DEPARTMENT OF INDUSTRIAL ENGINEERING

JUNE 2022

Approval of the Graduate School of Natural and Applied Sciences, Atılım University.

____

Prof. Dr. Ender KESKİNKILIÇ
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of **Master of Science in Industrial Engineering Department, Atılım University**.

____

Prof. Dr. Turan Erman ERKAN
Head of Department

This is to certify that we have read the thesis "COMPARISON AND ASSESSMENT OF SHRINKAGE METHODS IN CASE OF MULTICOLLINEARITY PROBLEM" submitted by ŞEVVAL KILIÇOĞLU and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science.

____

Asst. Prof. Dr. Fatma YERLİKAYA-ÖZKURT
Supervisor

Examining Committee Members:

Asst. Prof. Dr. Gözdem DURAL SELÇUK
Department of Business Administration,
Social Sciences University of Ankara
____

Asst. Prof. Dr. Fatma YERLİKAYA-ÖZKURT
Department of Industrial Engineering,
Atılım University
____

Asst. Prof. Dr. Kamil Demirberk ÜNLÜ
Department of Mathematics,
Atılım University
____

Date: 10.06.2022

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

ŞEVVAL KILIÇOĞLU

ABSTRACT

COMPARISON AND ASSESSMENT OF SHRINKAGE METHODS IN CASE OF MULTICOLLINEARITY PROBLEM

Kılıçoğlu, Şevval
M.S., Department of Industrial Engineering
Supervisor : Asst. Prof. Dr. Fatma Yerlikaya-Özkurt

June 2022, 100 pages

The use of data analysis and data interpretation are increasing in importance in many fields of applied science such as engineering, medicine, natural and social sciences. For this purposes, statistical methods are used to collect, analyze and interpret data. Among the statistical analysis methods, one of the most preferred one is multiple linear regression due to its simplicity and interpretation. It describes the relationship between more than one independent variables and a dependent variable.

However, sometimes, it can be observed that there is a multicollinearity (linear relationship) between the independent variables in data sets to which multiple linear regression models will be applied. This causes the variance of the estimated coefficients in the model to be large and their biases to be low, and in such cases, model predictions may not give accurate results and the reliability of the model may decrease. If there is a multicollinearity between the variables in the data set, it is of great importance to determine this in advance. For this purpose, there are many multicollinearity determination method and there are several methods developed to solve this problem. The most popular and powerful methods to handle this problem are shrinkage methods. Shrinkage methods aim to minimize the multicollinearity problem by reducing the variance of the estimated parameters in the model. Ridge Regression, Lasso, and Elastic Net are the most preferred shrinkage methods that set the coefficients of the variables in the model to zero or very close to zero.

In this thesis, Ridge Regression, Lasso, and Elastic Net were applied to nine different simulated data sets with different characteristics. The Copula function was used to create multicollinearity between independent variables for simulated data sets. Following that, all of the aforementioned shrinkage methods were also applied on three real-world data sets. These data sets were matched with the simulated data sets based on their sizes, which were classified as small, medium, and large. For the simulated data sets, a 10-fold Cross-Validation (CV) approach is applied to validate the shrinkage methods. On the other hand, the hold-out method, which relies on only one training and test split, is preferred for real-world data sets.

After all models were created, well-known performance measures were calculated for each method to determine which method gives better results in the data set in which characteristics. Mean squared error (MSE), mean squared error based on number of independent variables (PMSE), R-squared, mean absolute error (MAE) and explained variance are the performance measures used in decision making. Based on performance results, the methods were compared with TOPSIS, which is one of the multi-criteria decision making methods, and the order of preference was determined for each data set.

When all the performance and TOPSIS results are examined, it is seen that generally ridge regression gives the best results in small data sets, as the data set grows, that is, as the complexity increases, shrinkage methods tend to make variable selection to reduce the variance of the estimated coefficients, and therefore lasso or elastic net models give better results. If a general ranking is made among the models, they can be listed as lasso, elastic net and ridge regression.

Keywords: Multicollinearity, Shrinkage Methods, Ridge Regression, Lasso, Elastic Net, Multi Criteria Decision Making, TOPSIS.

ÖZ

ÇOKLU BAĞLANTI SORUNU DURUMUNDA KÜÇÜLTME YÖNTEMLERİNİN KARŞILAŞTIRILMASI VE DEĞERLENDİRİLMESİ

Kılıçoğlu, Şevval
Yüksek Lisans, Endüstri Mühendisliği Bölümü
Tez Yöneticisi : Dr. Öğr. Üyesi Fatma Yerlikaya-Özkurt

Haziran 2022, 100 sayfa

Mühendislik, tıp, doğa ve sosyal bilimler gibi uygulamalı bilimlerin birçok alanında veri analizi ve veri yorumlamanın kullanımı artarak önem kazanmaktadır. Bu amaç doğrultusunda veri toplamak, analiz etmek ve yorumlamak için istatistiksel yöntemler kullanılmaktadır. Basitliği ve kolay yorumlanması nedeniyle, en çok tercih edilen istatistiksel analiz yöntemlerinden biri, çoklu doğrusal regresyondur. Bu regresyon modelleri, birden fazla bağımsız değişken ile bir bağımlı değişken arasındaki ilişkiyi tanımlar.

Ancak bazen çoklu doğrusal regresyon modelinin uygulanacağı veri setlerinde bağımsız değişkenler arasında çoklu doğrusal bağlantı (iç ilişki) olduğu gözlemlenebilir. Bu da, modelde tahmin edilen katsayıların varyansının büyük ve yanlılıklarının düşük olmasına neden olmaktadır. Bu gibi durumlarda model tahminleri doğru sonuç vermeyebilir ve modelin güvenilirliği düşebilir. Veri setindeki değişkenler arasında çoklu doğrusal bağlantı varsa bunun önceden belirlenmesi büyük önem taşımaktadır. Bu amaç doğrultusunda kullanılan çok sayıda çoklu doğrusal bağlantı tespit etme yöntemi ve bu sorunu çözmek için geliştirilmiş çeşitli yöntemler bulunmaktadır. Çoklu doğrusal bağlantı problemini çözmek için en popüler ve güçlü yöntemler küçültme yöntemleridir. Küçültme yöntemleri, modelde tahmin edilen parametrelerin varyansını azaltarak çoklu doğrusal bağlantı sorununu en aza indirmeyi amaçlar. En çok tercih edilen küçültme yöntemlerinden olan Ridge Regresyon, Lasso ve Elastik Net modeldeki değişkenlerin katsayılarını direkt sıfır yapar veya sıfıra çok yaklaştırır.

Bu tez çalışmasında, Ridge Regresyon, Lasso ve Elastik Net, farklı özelliklere sahip dokuz farklı simüle edilmiş veri setine uygulanmıştır. Simüle edilmiş veri setlerindeki bazı bağımsız değişkenler arasında çoklu doğrusal bağlantıyı oluşturmak için Copula fonksiyonu kullanılmıştır. Daha sonra, yukarıda bahsedilen küçültme yöntemlerinin tümü, üç tane gerçek hayat veri setine de uygulanmıştır. Bu veri setleri, boyutlarına göre küçük, orta ve büyük olarak sınıflandırılan simüle edilmiş veri setleri ile eşleştirilmiştir. Simüle edilmiş veri setlerine uygulanan küçültme yöntemlerinin doğruluğunu ölçmek için 10 Katlamalı Çapraz Doğrulama yaklaşımı uygulanmıştır. Bunun yanında, gerçek hayat veri setleri için, veri setini yalnızca bir eğitim ve bir test verisine ayırmaya dayanan hold-out yöntemi tercih edilmiştir.

Tüm modeller oluşturulduktan sonra, hangi özelliklere sahip veri setlerinde hangi yöntemin daha iyi sonuç verdiğini belirlemek için her bir yöntem özelinde bazı performans ölçütleri hesaplanmıştır. Ortalama kare hatası (MSE), bağımsız değişken sayısına bağlı ortalama kare hatası (PMSE), R-kare, ortalama mutlak hata (MAE) ve açıklanan varyans, karar verme aşamasında kullanılan performans ölçütleridir. Performans sonuçlarından yola çıkarak, küçültme yöntemleri çok kriterli karar verme yöntemlerinden biri olan TOPSIS ile karşılaştırılmış ve her bir veri seti için yöntemlerin tercih sırası belirlenmiştir.

Tüm performans ve TOPSIS sonuçları incelendiğinde, genellikle küçük veri setlerinde en iyi sonuçları ridge regresyonunun verdiği, veri seti büyüdükçe yani karmaşıklık arttıkça küçültme yöntemlerinin tahmin edilen katsayıların varyansını azaltmak için değişken seçimi yapma eğiliminde olduğu ve bu yüzden de lasso ve elastik net modellerinin daha iyi sonuçlar verdiği görülmektedir. Modeller arasında genel bir sıralama yapılacak olursa lasso, elastik net ve ridge regresyonu olarak sıralanabilir.

Anahtar Kelimeler: Çoklu doğrusal bağlantı, Küçültme Yöntemleri, Ridge Regresyon, Lasso, Elastik Net, Çok Kriterli Karar Verme, TOPSIS.

To my family

ACKNOWLEDGMENTS

First and foremost, I would like to express my sincere gratitude to my thesis advisor, Asst. Prof. Dr. Fatma Yerlikaya-Özkurt, who guided and inspired me to choose this thesis topic. She provided me invaluable advice and patiently helped me in my difficult periods.

I am also deeply grateful to Prof. Dr. Turan Erman Erkan for his insightful comments and suggestions. He has always shed light on my academic path and guided me.

I would also like to thank my committee members, Asst. Prof. Dr. Gözdem Dural Selçuk and Asst. Prof. Dr. Kamil Demirberk Ünlü, for their efforts and contributions to this work.

Words cannot express my gratitude to Yalçın Zaim and Zerlin Zaim, who have supported and encouraged me in every part of my life and academic life and have been the cornerstone of my success.

At last but not in least, I'm extremely grateful to my family and friends, especially my mother, father and sister who supported me throughout my life, for their constant love, unwavering support and belief in me. I owe them big time.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF SYMBOLS/ABBREVIATIONS

p	: Number of independent variables
N	: Number of observations
$\boldsymbol{Y}$	: $Nx1$ vector of response variables
$\boldsymbol{X}$	: $Nx(p+1)$ matrix of observations on p independent variables
$\boldsymbol{\beta}$	: $(p+1)x1$ vector of unknown parameters which are called as regression coefficients
$\boldsymbol{\varepsilon}$	: $Nx1$ vector of random error
C	: Condition number
CI	: Condition Index
VIF	: Variance Inflation Factor
LM	: Linear Model
RR	: Ridge Regression
LASSO	: Least Absolute Shrinkage and Selection Operator
EN	: Elastic Net
$\widehat{\boldsymbol{\beta}}$	: Least squares estimator
$\widehat{\boldsymbol{\beta}}^{ridge}$	: Ridge estimator
$\widehat{\boldsymbol{\beta}}^{lasso}$	: LASSO estimator
$\widehat{\boldsymbol{\beta}}^{elastic\ net}$	: Elastic Net estimator

MCDM	: Multi Criteria Decision Making
MSE	: Mean Squared Error
PMSE	: Mean Squared Error Based on Number of Independent Variables
MAE	: Mean Absolute Error
TOPSIS	: Technique for Order Preference by Similarity to Ideal Solution
MOORA	: Multi-Objective Optimization by Ratio Analysis Method
CV	: Cross-Validation

CHAPTER 1

INTRODUCTION

Data is the organized form of information collected from any environment through measurement, experiment, observation or research, in order to be processed and interpreted by programs or people. There are many types of data and many areas where this data can be collected (e.g.: data in engineering, data in computer science, medical data, data in business management etc.). For example, in computer science, data can be encoded in the form of letters or numbers like represented in binary. Every data has a type and the type of this data is called data structure. A large number of different types of data are collected in databases. On the other hand, in business management, data can be used to record and track physical and monetary movements. Data quality plays an important role in this area. Data quality is generally related to the reliability and usability of the recorded data.

Statistics is used to collect, analyze and summarize data for the purpose of interpretation and presentation. In these days, many public institutions and private organizations share their data in order to obtain statistical analysis results and to get helpful information as the importance and needs of statistical analysis is increasing [1].

One of the most used tool among statistical analysis is linear regression model. Linear regression models are popular and well-known models for ease of application and interpretation in statistical research and analysis. They are used to explain and interpret the relationship between a dependent (or response or output) variable and one or more independent (or explanatory or input or predictor) variables [2], [3]. In order to explain this relationship, the ordinary least squares method is commonly used for the estimation of the regression parameters of a model [4]. The logic behind ordinary least squares is minimizing the sum of square of the residuals.

One of the assumptions accepted while creating linear regression models is that there is no correlation between the independent variables in the model [5]. Many results in the literature have shown that ordinary least squares estimation is no longer a good estimator in the presence of multicollinearity [6], [7]. Multicollinearity occurs when there is a linear relationship between one or more variables [8]. As a result of this, the variance of the parameter estimates will be high. This reduces the accuracy and reliability of the model. High correlation between independent variables poses a risk when making statistical conclusions about the importance of the regression coefficients [9]. That is to say, in case where there is multicollinearity in the data, linear models may not provide meaningful results.

There are multiple methods used to detect [10], [11] and eliminate the multicollinearity problem. Some of the methods used to detect multicollinearity are correlation coefficient (r), coefficient of determination (R^2), variance inflation factor (VIF), examining the eigenvalues of the correlation matrix X^TX, determinant of correlation matrix X^TX, tolerance value and determination of multicollinearity by F and t test based on regressing each independent variable on the remaining independent variables [9]. There are also many methods for solving the multicollinearity problem. To handle this problem, additional data collection, elimination of correlated variable or variables from the model, redefinition of independent variables causing correlation problem and regularization instead of least squares method can be considered [8], [12].

The most preferred methods for eliminating the multicollinearity problem are shrinkage methods. Shrinkage methods aim to handle the multicollinearity problem by minimizing the variance of the estimators in the model. Ridge Regression [13], Lasso [14], and Elastic Net [15] are well-known and popular shrinkage methods. These methods converge the values of the coefficients of the variables in the model to zero or very close to zero values.

Data sets may have different characteristics such as sample size and number of independent variables. Thus, it is a difficult problem to decide which shrinkage method is suitable for which data set. For that purposes, in this study, the performance of shrinkage methods obtained from different data sets with different characteristics will be examined by considering the well-known performance measures. In order to guide the practitioners, multi criteria decision making (MCDM) methods [16] will be used and the final decision will be made accordingly. The performance measures to be used in decision making are mean squared error (MSE), mean squared error based on number of independent variables (PMSE), coefficient of determination (R^2), mean absolute error (MAE) and explained variance. When comparing the models obtained by shrinkage methods, it is necessary to consider the effects of different criteria in MCDM approach. Therefore, Technique for Order Preference by Similarity to Ideal Solution (TOPSIS) and Multi-Objective Optimization by Ratio Analysis Method (MOORA) are considered among MCDM methods [17].

In this thesis study, the shrinkage methods mentioned are firstly applied to 9 simulated data sets with different characteristics. These data sets were produced by using Copula function in order to create multicollinearity between independent variables in the data sets. After that all the mentioned methods have also been applied to 3 real world data sets. These data sets were selected matched with the simulated data sets according to their sizes as small, medium and large. In the applications, to compare and validate the shrinkage methods, 10-fold Cross-Validation (CV) approach is used for the simulated data sets. In this approach, the original data set is randomly divided into ten sub-folds. While one sub-fold is hold for testing the model, the remaining nine sub-folds are used for training data, in other words, for building shrinkage models. For real world data sets, hold-out method is preferred which depends on just one training and one test split [18].

When the applications of shrinkage methods in the literature are examined, it is seen that they are generally applied to real world data sets with different characters [5], [8], [12], [15], [19]–[24]. This study aims to make suggestions for the use of the most appropriate shrinkage methods in data sets of different characters with multicollinearity among their independent variables. For this purpose, since it is not always possible to make a meaningful classification according to data characteristics and to find representative data in real world data sets, simulated data sets with both different characteristics and multicollinearity between variables were produced and comparisons were made on these data.

Thesis is organized as follows: in the second chapter, the background on linear regression and multicollinearity problem are mentioned. Then, in the third chapter, the well-known shrinkage methods which are Ridge Regression, Lasso and Elastic Net are explained. In the fourth chapter, simulated and real world data sets and the structures of these data sets are mentioned. In the fifth chapter, the criteria used to measure the performance of the methods and the comparison methods are explained. In the sixth chapter, applications and comparison results are presented. In the last part of the thesis, conclusion and future works are stated.

CHAPTER 2

BACKGROUND ON LINEAR REGRESSION MODELS AND MULTICOLLINEARITY PROBLEM

2.1. Linear Regression

One of the methods used to describe the relationship between one or more independent variables and a dependent variable in a data set is the linear regression model. Linear regression models are popular and well-known models in statistical research and data analysis because of their ease of application and interpretation [25].

Regression analysis with a single independent variable is called Simple Linear Regression, and regression analysis with more than one independent variable is called Multiple Linear Regression Model [3].

2.1.1. Simple Linear Regression

Simple linear regression models investigate the linear relationship between a dependent (Y) and an independent variable (x). The equation of the simple linear regression model is:

$$\boldsymbol{Y} = \boldsymbol{\beta x} + \boldsymbol{\varepsilon} \tag{1}$$

where $\boldsymbol{Y}$ is a $N \times 1$ vector of response variables, $\boldsymbol{x}$ is a $N \times (p+1)$ matrix of observations on p independent variables (p=1), $\boldsymbol{\beta}$ is a $(p+1) \times 1$ vector of unknown parameters which are called as regression coefficients and $\boldsymbol{\varepsilon}$ is a $N \times 1$ vector of random errors which are assumed as $\varepsilon \sim N(0, \sigma^2)$.

2.1.2. Multiple Linear Regression

A multiple linear regression equation is used to find the linear relationship between one dependent variable (Y) and more than one independent variable $(x_1, x_2, \dots, x_j)$. The equation of the multiple linear regression model is:

$$Y = \beta_0 + \beta_1 x_1 + \cdots + \beta_j x_j + \cdots + +\beta_p x_p + \varepsilon\,, \qquad (j = 1, 2, \dots, p) \tag{2}$$

The matrix vector notation of this model is expressed as:

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} = \begin{bmatrix} 1 & x_{11} & x_{12} & \cdots & x_{1p} \\ 1 & x_{21} & x_{22} & & x_{2p} \\ & \vdots & & \ddots & \vdots \\ 1 & x_{N1} & x_{N1} & \cdots & x_{Np} \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_p \end{bmatrix} + \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_N \end{bmatrix} \tag{3}$$

where,

$$\boldsymbol{Y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}, \boldsymbol{X} = \begin{bmatrix} 1 & x_{11} & x_{12} & \cdots & x_{1p} \\ 1 & x_{21} & x_{22} & & x_{2p} \\ & \vdots & & \ddots & \vdots \\ 1 & x_{N1} & x_{N1} & \cdots & x_{Np} \end{bmatrix}, \boldsymbol{\beta} = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_p \end{bmatrix}, \boldsymbol{\varepsilon} = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_N \end{bmatrix} \tag{4}$$

or in short:

$$\boldsymbol{y} = \boldsymbol{X\beta} + \boldsymbol{\varepsilon} \tag{5}$$

here, $\boldsymbol{Y}$ is a $N \times 1$ vector of response variables, $\boldsymbol{X}$ is a $N \times (p + 1)$ matrix of observations on p independent variables, $\boldsymbol{\beta}$ is a $(p + 1) \times 1$ vector of unknown parameters which are called as regression coefficients and $\boldsymbol{\varepsilon}$ is a $N \times 1$ vector of random errors which are assumed as $\varepsilon \sim N(0, \sigma^2)$.

2.1.3. Parameter Estimation Methods in Multiple Linear Regression

The purpose of regression analysis is to create the best model that can predict the dependent variable from the independent variables or to determine which independent variables are more affected by the dependent variable. One of the most used methods for the estimation of the parameters in the model to be created for this purpose is the least squares estimation method, which allows the estimation of the parameters in the model so that the sum of squares of the error is minimal [26].

2.1.3.1. Least Squares Estimation Method

The least squares estimation method is used to minimize the sum of the squares of the difference between the actual Y values and the predicted Y values. In other words, the coefficients of the variables that can satisfy the condition that the sum of the squares of the errors are minimum are estimated.

$$RSS(\boldsymbol{\beta}) = \sum_{i=1}^{N} (y_i - f(x_i))^2$$
$$= \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\, \beta_j)^2 \qquad (6)$$

We can write the residual sum-of-squares as:

$$RSS(\boldsymbol{\beta}) = (\boldsymbol{y} - \boldsymbol{X\beta})^{\boldsymbol{T}} (\boldsymbol{y} - \boldsymbol{X\beta}) \qquad (7)$$

Since $RSS(\boldsymbol{\beta})$ is a convex function, rest of the formulations hold to find the minimum.

Differentiating with respect to $\boldsymbol{\beta}$ we obtain:

$$\frac{\partial\, RSS(\boldsymbol{\beta})}{\partial\, \boldsymbol{\beta}} = -2\boldsymbol{X}^T (\boldsymbol{y} - \boldsymbol{X\beta}) \qquad (8)$$

We set the first derivation to zero:

$$\boldsymbol{X}^T (\boldsymbol{y} - \boldsymbol{X\beta}) = \boldsymbol{0} \qquad (9)$$

We obtain the unique solution:

$$\widehat{\boldsymbol{\beta}} = (\boldsymbol{X}^T \boldsymbol{X})^{-1} \boldsymbol{X}^T \boldsymbol{y} \qquad (10)$$

2.2. Multicollinearity

In linear regression, some assumptions must be valid so that parameters can be estimated using the least squares estimation method and multiple regression analysis can be applied. Some of these assumptions are; there is no multicollinearity between the independent variables, the distribution of the error term (ε_i) is normal, and the variance of the error term is constant [25], [27].

When predictor variables in a regression model are more closely related with other predictor variables than with the response variable, it means that multicollinearity exists [28]. In case of multicollinearity, the least squares estimates are unbiased, but the estimates may diverge from their true values as the variances get larger. Therefore, it is of great importance to check and determine whether there is multicollinearity among the independent variables before performing the regression analysis. If multicollinearity is detected among the variables, solutions should be applied to eliminate the negative effects of this.

2.2.1. Detect Multicollinearity

As known from the literature, there are many methods used to detect multicollinearity, and some of these methods will be discussed in this section.

2.2.1.1. Correlation Coefficient $(\boldsymbol{r})$

The correlation coefficient is the most useful and simple method used in the determination of multicollinearity [29]. It is the measure of the linear relationship between two variables and varies between -1 and 1.

$$Cor(X,Y) = r_{XY} = \frac{\sum_{i=1}^{N}(x_i-\bar{x})(y_i-\bar{y})}{\sqrt{\sum_{i=1}^{N}(x_i-\bar{x})^2}\sqrt{\sum_{i=1}^{N}(y_i-\bar{y})^2}} \tag{11}$$

If there is a close linear relationship between the X and Y independent variables, with r_{XY} being the correlation coefficient for the X and Y independent variables, the absolute value of the simple correlation coefficient $|r_{XY}|$ will take a value close to 1.

With this approach, only the relationship between two independent variables can be explained. It is not a highly preferred method because it is insufficient to explain the relationship between more than two variables.

2.2.1.2. Coefficient of Determination (R^2)

R-squared is a statistical measure of how close the data are to the fitted regression line.

$$Coefficient\ of\ Determination\ (R^2) = 1 - \frac{SS_{Regression}}{SS_{Total}} \tag{12}$$

Here,

$SS_{Regression}$: The sum of squares due to regression (explained sum of squares)

SS_{Total} : The total sum of squares

0% indicates that the model explains none of the variability of the response data around its mean. 100% indicates that the model explains all the variability of the response data around its mean.

With the help of this coefficient, the changes in R^2 are examined when new independent variables are added to the model. If there is no significant improvement in R^2, the multicollinearity problem may have arisen [30].

2.2.1.3.Tolerance Value

The percentage of variance in an independent variable in a model that cannot be explained by other independent variables is known as tolerance [31]. Tolerance is associated with each independent variable and ranges from 0 to 1.

$$Tolerance\ Value = (1 - R_j^2) \tag{13}$$

Here, R_j^2 is the multiple coefficient of determination in a regression of the j^{th} independent variable on all other independent variables.

A tolerance value of 0 indicates that there is a multicollinearity problem and a tolerance value equal to 1 indicates that there is no multicollinearity problem [32].

2.2.1.4. Variance Inflation Factor ($\boldsymbol{VIF}$)

In this method proposed by Marquardt [33], VIF is calculated for each independent variable in the regression model considered.

$$VIF_j = \frac{1}{\left(1 - R_j^2\right)} \qquad for\ j = 1, \dots, p \tag{14}$$

VIF's are the diagonal elements of $(\boldsymbol{X}^T\boldsymbol{X})^{-1}$ (the inverse of the covariance matrix) when the variables are standardized.

R_j^2 is the multiple coefficient of determination in a regression of the j^{th} independent variable on all other independent variables.

The coefficient of determination, denoted by R_j^2, will be zero if there is no multicollinearity problem for the j^{th} variable, and the VIF value will be equal to 1. In case of a multicollinearity, R_j^2 takes a value close to 1 and causes VIF_j to take a large value [20].

If any of VIF_j is higher than 10, multicollinearity exists in data and if any of VIF_j is higher than 30, it is assumed that a strong multicollinearity exists in data [34].

2.2.1.5. Eigenvalues and Condition Number

The eigenvalues $(\gamma_1, \gamma_2, \dots, \gamma_p)$ of the $\boldsymbol{X}^T\boldsymbol{X}$ correlation matrix, used in determining the multicollinearity, give information about the extent of the multicollinearity. If one or more of the $(\gamma_1, \gamma_2, \dots, \gamma_p)$ eigenvalues take a value close to zero, it indicates that there is a multicollinearity problem.

However, instead of examining each eigenvalue, the condition number (C) defined as the ratio of the largest eigenvalue to the smallest eigenvalue is used:

$$C = \frac{\gamma_{max}}{\gamma_{min}} \tag{15}$$

If $C < 100$, there is no problem with multicollinearity, if $100 < C < 1000$, there is a moderate multicollinearity and if $C > 1000$, there is a strong multicollinearity [29].

The square root of is known as condition index (CI). $\sqrt{C}$ around 10-30 states to moderate multicollinearity and higher values states to strong multicollinearity [10], [29].

2.2.2. Eliminate Multicollinearity

In linear regression analysis, if the multicollinearity problem has been identified using the methods mentioned above, different suggested methods can be used to eliminate the harmful effects of this problem. The most commonly used methods to solve this problem are; adding new observations, removing or combining some independent variables from the model, or applying biased estimation methods [27], [29]. However, each of these methods has advantages as well as disadvantages.

In cases where it is possible to add new observations to the data set, this solution method can be used to solve the problem. The purpose of this method is to increase the sample size and reduce the variance of the estimators. However, most of the time this option cannot be selected. Because there may be many independent variables that cannot be controlled in some fields. Therefore, new observations may contain the same patterns of correlation as previous observations.

Another solution method is to remove the variable or variables that cause the multicollinearity problem in the model. Although this method reduces multicollinearity, it also has the risk of reducing the accuracy of the model if the independent variable or variables removed from the model have a strong effect on the dependent variable. Besides, combining the independent variables that have a real relationship between them and making them a single variable also solves the multicollinearity problem.

In cases where methods such as adding new data or adding and removing independent variables cannot be applied, methods called Shrinkage Methods such as Ridge Regression, Lasso and Elastic Net can be used. These methods provide more precise estimations with a bias coefficient added to the model instead of the least squares method which is insufficient in the face of high variances when multicollinearity problems are encountered.

CHAPTER 3

SHRINKAGE METHODS

As mentioned in previous section, the method of least squares is used to estimate all coefficients of the linear regression model. However, due to the presence of multicollinearity among the independent variables, these estimators often have a small amount of bias, but have large variances. If there is a linear relationship between the coefficients in the regression model or the number of coefficients is large, the predicted model becomes difficult to interpret. At the same time, large variance of the estimators also negatively affects the accuracy of the estimation. Therefore, approximating some coefficients in the regression model to zero or evaluating them in the model as zero eliminates the negative effects on the accuracy of the model's estimation and provides a more accurate interpretation of the model. In order to actualize this, the bias of the predicted regression coefficients should be slightly increased and the variances should be reduced. For this purposes, in this section, Ridge Regression, Lasso and Elastic Net will be discussed as methods used to increase the accuracy of predictions in case of multicollinearity problem. The reasons for choosing these three shrinkage methods among available shrinkage methods in literature are that these biased regression methods are the main ones and other shrinkage methods were developed based on these methods. Also, they are easy to implement due to available packages [8]. Ridge regression is a method that only shrinks some of the predicted coefficients in the model, but does not eliminate them [35]. Conversely, lasso sets some of the coefficients that contribute less to the model or that have multicollinearity between them directly to zero. Elastic net, on the other hand, combines the good features of these two methods to estimate coefficients.

3.1. Ridge Regression

Ridge regression is a regression method first proposed by Hoerl and Kennard [13], [36] and is recommended as an alternative to the least squares estimation method when faced with the multicollinearity problem. This biased regression method is one of the techniques used to reduce the negative effects of multicollinearity on parameter estimations in linear regression [24], [37]. In other words, ridge regression is used to minimize the effects of independent variables on each other and to obtain more accurate coefficient estimates.

Ridge regression essentially uses a penalty to reduce the size of the regression coefficients. The operation of the ridge regression and the least squares method is almost the same. The only difference is that the ridge coefficients are aimed to be minimized by constraining them with a penalty term. Ridge regression solves the penalized least squares optimization problem, which aims to estimate the $\boldsymbol{\beta_j}$ parameters by reducing them to zero through a penalty term in addition to the residual sum of squares used in the least squares estimation method. In other words, a single constrained optimization problem is tried to be solved in ridge regression.

$\widehat{\boldsymbol{\beta}}^{ridge}$, which is the ridge estimator of $\boldsymbol{\beta}$ parameter, is obtained from the solution of equation (16). Here in equation (16), the objective function is the residual sum of squares and the constraint typically looks for the summation of the β_j^2 that limited the coefficients with a t value.

$$\widehat{\boldsymbol{\beta}}^{ridge} = \underset{\beta}{argmin} \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2$$

$$subject\ to \sum_{j=1}^{p} \beta_j^2 \leq t \tag{16}$$

This single constrained optimization problem can be reformulated as an unconstrained optimization problem as in equation (17). This unconstrained optimization problem contains two parts: the first part is the residual sum of squares and the second part is called the shrinkage penalty (or complexity term) that is tried to be minimized.

$$\hat{\boldsymbol{\beta}}^{ridge} = \underset{\beta}{argmin}\left\{\sum_{i=1}^{N}\left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j\right)^2 + \lambda\sum_{j=1}^{p}\beta_j^2\right\} \quad (17)$$

The λ is also known as tuning parameter (or ridge coefficient or the complexity parameter or regularization parameter) and the shrinkage is limited by this complexity parameter $\lambda \geq 0$. The amount of shrinkage is proportional to the value of lambda. If the value of λ increases, the amount of shrinkage also increases. In case the lambda is equal to 0, it means there is no effect from the penalty term and so, the ridge estimator of parameter $\boldsymbol{\beta}$ will be equal to the least squares estimator, i.e. $\hat{\boldsymbol{\beta}}^{ridge} = \hat{\boldsymbol{\beta}}$. The shrinkage penalty becomes more significant as $\lambda \to \infty$, and ridge regression coefficient estimates take values very close to zero [8].

While only a single set of coefficient estimates is produced in the least squares estimation method, a unique set of coefficient estimates are generated for each λ value in the ridge regression. Choosing the right λ value is of great importance for the accuracy of the estimations to be high. Although there are many methods for this, the cross-validation method is frequently used.

3.2. Lasso

Lasso is another shrinkage method, like ridge regression, but there are some important differences between these two methods. In ridge regression, some of the estimated coefficients are calculated very close to zero, but they are never directly equal to zero. However, in lasso, some coefficients that contribute less to the model are not included in the model when the coefficients are estimated and are set to zero. In other words, lasso performs variable selection.

Lasso has recently been seen as an alternative to ridge regression, as these parameter estimates can be problematic in some cases [8]. For example, if we consider a data set with eight independent variables and a dependent variable, in models that are created using the least squares estimation method or ridge regression, all eight variables will be included in the model and eight prediction parameters will be calculated. But let's assume that only five of these variables actually contribute to the model. Meanly, including all variables in the model will both reduce the accuracy of the estimations and make it difficult to interpret the model. This situation can also lead to bigger problems when the number of independent variables is quite large. In order to reduce the effect of these disadvantages, lasso has had a place in the literature.

Lasso estimator of $\boldsymbol{\beta}$ parameter, $\widehat{\boldsymbol{\beta}}^{lasso}$, is obtained from the solution of the minimization problem in equation (18).

$$\widehat{\boldsymbol{\beta}}^{lasso} = \underset{\beta}{argmin} \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2$$
$$subject\ to\ \sum\nolimits_{j=1}^{p} |\beta_j| \leq t \tag{18}$$

If this single constrained minimization problem is also transformed into an unconstrained minimization problem, equation (19) is obtained. In this equation, similar to the ridge regression, the first part is defined as the residual sum of squares, while the second part is the shrinkage penalty, that is, the part where the parameters are penalized.

$$\widehat{\boldsymbol{\beta}}^{lasso} = \underset{\beta}{argmin} \left\{ \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \right\} \tag{19}$$

When the lasso and ridge equations are compared, it is seen that they are very similar to each other. The only and most important difference between them is that the penalty term in ridge regression β_j^2 becomes $|\beta_j|$ in lasso. In fact, lasso also brings some coefficient estimates closer to zero, such as ridge regression. But the difference in the constraint used in lasso ensures that some coefficient estimates equal zero if the λ is large enough. Again, as in ridge regression, it is critical to determine the λ value while creating these models.

Graphical representation of ridge regression and lasso estimations in case of two parameters ($p = 2$) is given in Figure 1. Here, the constraints of the ridge regression and lasso form the gray shaded areas. The elliptical lines represent the residual sum of squares, and the first point of these lines touching the gray areas is the solution of both methods [19]. In lasso, one of the parameters equals zero if the solution is in the corner and if $(p > 2)$, more than one coefficient estimate can be calculated as equal to zero. Since the area for the ridge regression forms a circle, it has no corners and therefore the parameters are never equal to zero. Besides these, if the value of t is large enough, the ridge regression and lasso estimates will equal the least squares estimation, since the gray-painted areas will also include $\hat{\boldsymbol{\beta}}$.

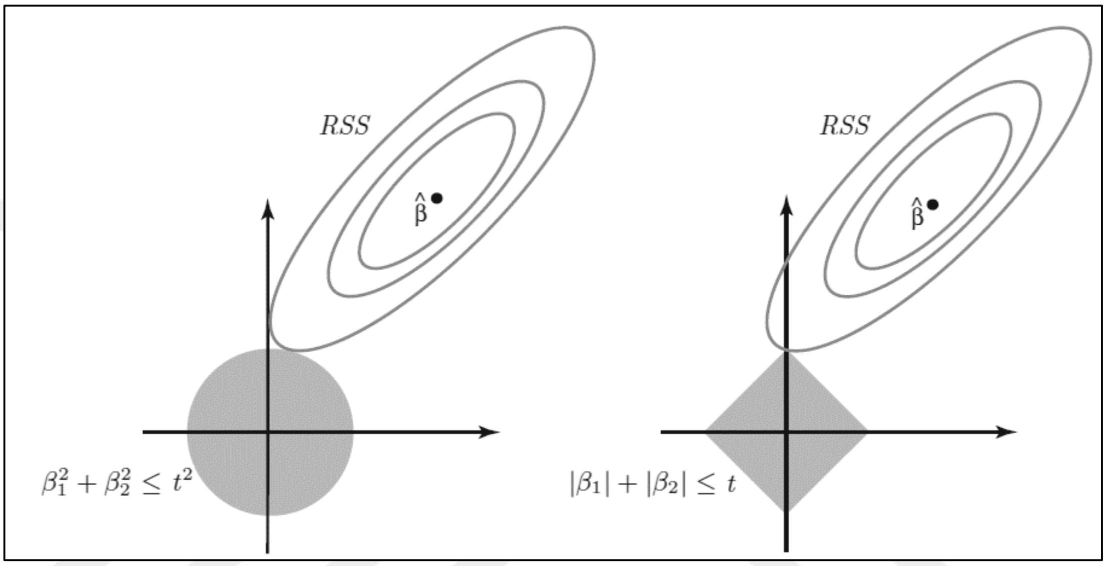

Figure 3.1 Graphical representation of Ridge Regression and Lasso estimations, respectively [8]

3.3. Elastic Net

Elastic net is basically a combination of ridge regression and lasso. This shrinkage method not only brings the correlated independent variables closer to zero, as ridge regression does, but also sets some correlated independent variables, as in lasso, to zero and removes them from the model.

In the previous chapters, some problems that arise in the application results of ridge regression were mentioned. Some coefficient estimates that are approximated to zero but never completely set to zero can complicate interpretation and reduce accuracy in models with a large number of independent variables. As a solution to this, it has been observed in the literature that the lasso method, which makes some variables close to zero and some variables directly zero, gives better results. However, Zou and Hastie mentioned that the lasso method also has some disadvantages. For example, when the lasso method is applied to a data set with a group of variables with very high pairwise correlations, lasso selects only one variable from this group and does not include it in the model, but does not care which one it chooses [15].

In the elastic net equation, given in equation (20), there are both ridge and lasso penalties. Similar to ridge regression and lasso's equations, elastic net equation also consists of two parts. The first part is called the residual sum of squares, while the rest is defined as the elastic net penalty.

$$\widehat{\boldsymbol{\beta}}^{elastic\ net} = \underset{\beta}{argmin} \left\{ \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 + \lambda_1 \sum_{j=1}^{p} |\beta_j| + \lambda_2 \sum_{j=1}^{p} \beta_j^2 \right\} \tag{20}$$

There are two special cases for the elastic net regression: when $\lambda_1 = 1$ and $\lambda_2 = 0$, elastic net regression turns into lasso, which aims to reduce the value of non-zero coefficients to zero, resulting in a sparse model that contains only a subset of the variables [38]. When $\lambda_1 = 0$ and $\lambda_2 = 1$, elastic net regression turns into ridge regression, which allows the model to include some correlated independent variables, thus removing the limitation on the number of independent variables included in the model.

The graphical representation of the regularization parts for ridge regression, lasso and elastic net is given in Figure 2. The outermost contour represents the area restricted by ridge regression, while the innermost contour denoted by dashed lines contains the area restricted by lasso. The contour, which is shown with a straight line in the middle of these two, is the area formed by the constraints of the elastic net regression and λ_1 and λ_1 are accepted as 0.5.

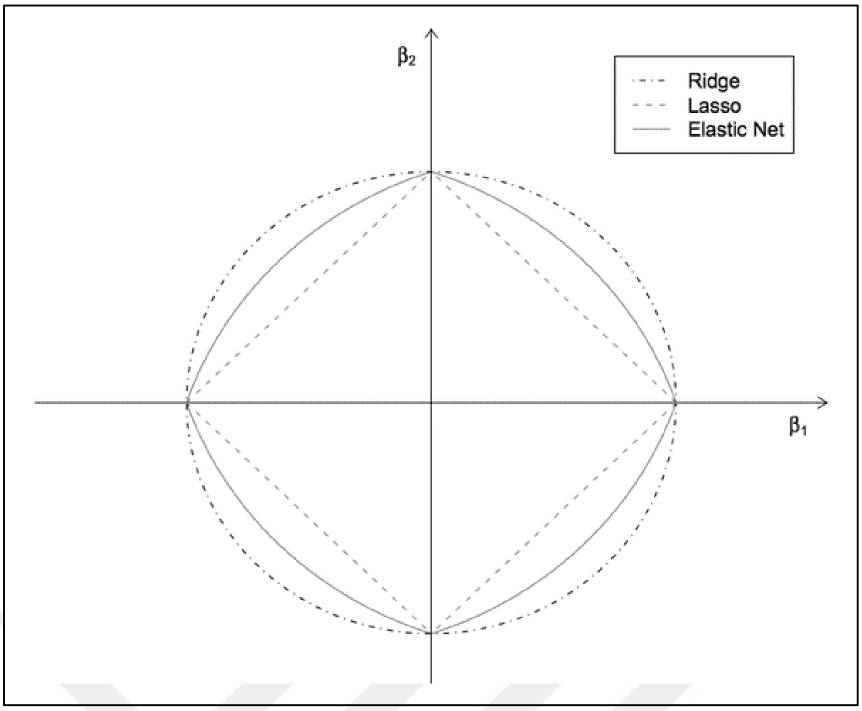

Figure 3.2 Visualization of the regularization parts for each shrinkage methods [15]

CHAPTER 4

DATA SETS

4.1. Simulated Data Sets

In order to apply shrinkage methods and compare their performances, nine simulated data sets were created with different characteristics in terms of number of independent variables and observations. These data sets also have multicollinearity between their independent variables. Copula function in MATLAB software was used to create multicollinearity between variables [39]. The copularnd('Gaussian',cormat,n) function in MATLAB creates a random matrix of n rows generated from a Gaussian copula using the cormat correlation matrix. The number of columns of the resulting random matrix is the same as the number of columns of the correlation matrix. In order to obtain multicollinearity between variables, firstly, the variables that are correlated to each other in the data set were determined randomly and a correlation matrix was created for these variables randomly. While determining the correlated variables, for example, five random numbers were generated for the data set with ten variables, and the next steps were taken so that there was a correlation between the independent variables corresponding to these five numbers. Similarly, while constructing the correlation matrix, a symmetric correlation matrix with randomly varying values between 0.5 and 1.0 was created for these five independent variables. If the correlation coefficient is in the range of 0.5 - 0.69, it is accepted that there is moderate correlation between the variables, if it is between 0.7 - 0.89, there is high correlation between the variables, and if it is between 0.9 - 1.0, there is a very high correlation between the variables [40]. In each data set, the number of variables with multicollinearity was determined to be almost half of the total number of independent variables. Then, the corresponding correlated variables were created based on the random correlation matrices. Other independent variables in the data set that do not have multicollinearity were created based on standard uniform distribution. And then, by combining all the mentioned independent variables, the design matrix ($\boldsymbol{X}$) was formed. In order to create the response variable vector ($\boldsymbol{y}$), regression equation $\boldsymbol{y} = \boldsymbol{X\beta} + \boldsymbol{\varepsilon}$ mentioned in Section 2.1.2., is used. Here, $\boldsymbol{\beta}$ and $\boldsymbol{\varepsilon}$ vectors are also produced based on the standard uniform distribution. After merging design matrix and response vector, the final form of the simulated data set is obtained. The same process was repeated for all nine data sets.

Table 4.1 Simulated Data Sets Matrix

Data set 1 p = 10 N = 100	Data set 2 p = 25 N = 100	Data set 3 p = 60 N = 100
Data set 4 p = 10 N = 500	Data set 5 p = 25 N = 500	Data set 6 p = 60 N = 500
Data set 7 p = 10 N = 1500	Data set 8 p = 25 N = 1500	Data set 9 p = 60 N = 1500

The form of the simulated data sets matrix that represent nine data sets in total is provided in Table 4.1. In this table, p represents the number of independent variables and N represents the number of observations in the data set. For the first 3 data sets (the first row of the simulated data sets matrix), the number of observations was determined as 100 and the number of variables changed as 10, 25 and 60, respectively. For the next 3 data sets (the second row of the simulated data sets matrix), the number of observations is fixed to 500, and the number of variables is again 10, 25 and 60, respectively. For the last 3 data sets (the third row of the simulated data sets matrix), while the number of variables changed as 10, 25 and 60, respectively, the number of observations was determined as 1500.

For the data sets in this table, multicollinearity is exist for five independent variables among ten independent variables in the first column of this table, twelve independent variables among twenty-five independent variables in the second column of this table, and thirty independent variables among sixty independent variables in the last column of this table.

Condition number (C) and condition index (CI) criteria are used to determine the multicollinearity among the independent variables in the data set. As mentioned in Section 2.2.1.5., if the calculated C value is greater than 100 or the CI value is greater than 10, it is concluded that there is a multicollinearity problem. For the multicollinearity determination, the C and CI values for all simulated data sets are calculated and the corresponding values are presented in Table 4.2 and Table 4.3 respectively. When the C and CI values are examined, it is observed that there is multicollinearity among the independent variables for each data set.

Table 4.2 Condition Numbers of Simulated Data Sets

Data Set 1	Data Set 2	Data Set 3	Data Set 4	Data Set 5	Data Set 6	Data Set 7	Data Set 8	Data Set 9
233.8598	809.2002	1861.694	210.6095	483.4761	497.9871	173.9535	387.9826	435.8148

Table 4.3 Condition Indexes of Simulated Data Sets

Data Set 1	Data Set 2	Data Set 3	Data Set 4	Data Set 5	Data Set 6	Data Set 7	Data Set 8	Data Set 9
15.2925	28.4464	43.1473	14.5124	21.9881	22.3156	13.1891	19.6973	20.8762

4.2. Real World Data Sets

The same application and comparison procedure will be followed for three real world data sets. The aim of using the real world data sets in applications is to show that the performance of the shrinkage methods obtained for the simulated data sets can also be valid for the real world data sets. This will contribute to the future use of shrinkage methods for different data sets with different characteristics. Among the simulated data sets, three data sets, which are considered as small, medium and large in size, were decided and the representative real world data set was selected from various data environments.

The first real world data set is the "Daily Demand Forecasting Orders Data Set" obtained from the UCI: Machine Learning Repository [41]. The data set, which is a real database from a large logistics company in Brazil, was collected during 60 days. Therefore, the observation number of the data set is 60. It is aimed to estimate the total number of orders per day using twelve predictive attributes [42], [43]. The second real world data set, “Hitters: Baseball Data”, is included in the ISLR package in R Studio software and has also been used in the applications of ridge regression and lasso in the "An Introduction to Statistical Learning - with Applications in R" book. The purpose of using this data set is to make salary estimations of 322 players using nineteen independent variables [8], [44]. The third and final data set is the "Superconductivity Data Set" obtained from the UCI: Machine Learning Repository [41]. This data set contains 81 features extracted from 21263 superconductors along with the critical temperature. The goal in this data set is to predict the critical temperature using the determined features [45], [46].

Before using the data to be used in machine learning algorithms, it is an important step to pre-process the data in order to obtain good performance results [47]. For this reason, all data sets have undergone some pre-processing and then all the independent variables and response values have been normalized. Normalization is rearranging the data into a smaller range to give equal weight to all attributes [48]. Normalization is usually done so that this range is 0 from 1 or -1 from 1. The relationships between the original data values are maintained when applying min-max normalization [48]. Among the many normalization methods, min-max normalization given in Equation (21) was chosen to apply in real world data sets.

$$new\ x_i = \frac{x_i - min(E)}{max(E) - min(E)}(new\ max(E) - new\ min(E)) + newmin(E) \qquad (21)$$

where $min(E)$ and $max(E)$ are the minimum and maximum values in the column of variable E, $[new\ min(E),\ new\ max(E)]$ is the normalization range, x_i is the ith value in the column of variable E and $new\ x_i$ is the normalized value of x_i in the range $[new\ min(E),\ new\ max(E)]$.

When the normalization range is accepted as [0,1], Equation (22) is obtained, which is also used in the normalization of real world data sets [49].

$$new\ x_i = \frac{x_i - min(E)}{max(E) - min(E)} \qquad (22)$$

Independent variable and observation numbers of real world data sets after pre-processing and normalization steps are given in Table 4.4.

Table 4.4 Real World Data Sets Matrix

Data set 1	Data set 2	Data set 3
p = 9	p = 16	p = 81
n = 60	n = 263	n = 21263

After pre-processing the data sets and applying the normalization process, condition number and condition index values were calculated, as in simulated data sets, in order to check whether there is multicollinearity among the independent variables.

When the condition number values given in Table 4.5 are examined, it is seen that all of them are greater than 100, especially the condition number value of the third data set is very high (high multicollinearity). Likewise, the condition index values given in Table 4.6 provide the same result. Therefore, it can be concluded that all three real world data sets are suitable for the applications of the shrinkage methods.

Table 4.5 Condition Numbers of Real World Data Sets

Data set 1	Data set 2	Data set 3
302.2709	6030.793	635822.5

Table 4.6 Condition Indexes of Real World Data Sets

Data set 1	Data set 2	Data set 3
17.38594	77.65818	797.3848

CHAPTER 5

COMPARISON MEASURES AND METHODS

5.1. Performance Measures

Three shrinkage methods mentioned in this thesis study were applied to nine simulated data sets and three real-world data sets. Some performance measures were used to determine which of these methods give more accurate results in which data set. These performance measures, described below, are: mean squared error (MSE), mean squared error based on number of independent variables (PMSE), coefficient of determination (R^2), mean absolute error (MAE) and explained variance.

5.1.1. Mean Squared Error (MSE)

The mean squared error is a performance metric that calculates the mean of the squares of the residuals. That is, it is calculated by averaging the squares of the differences between the actual values and the estimated values, as the formula is given in equation (23).

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \tag{23}$$

where y_i is the actual response values and $\hat{y}_i$ is the estimated response values in a data set of N observations.

Because mean squared error represents overall estimation accuracy, and a large MSE indicates bad estimation, least squares estimation models may perform poorly if the data set has multicollinearity [50].

5.1.2. Mean Squared Error Based on Number of Independent Variables (PMSE)

PMSE is another performance measure like mean squared error. The only difference is that this performance criterion also takes into account the number of independent variables and is therefore considered a complexity measure. The formula of PMSE is as given in equation (24).

$$PMSE = \frac{1}{N+p}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \qquad (24)$$

where y_i is the actual response values and $\hat{y}_i$ is the estimated response values in a data set of N observations and p independent variables.

As in mean squared error, it can be said that the smaller the calculated value, the higher the prediction accuracy in PMSE.

5.1.3. Coefficient of Determination (R^2)

As mentioned previously in Section 2.2.1.2, R-squared (R^2) is a performance measure that can predict the rate of variation in the dependent variable from the independent variables.

$$R^2 = 1 - \frac{SS_{Regression}}{SS_{Total}} \qquad (25)$$

The values of this performance measure range from 0 to 1, and the closer the R^2 value is to 1, the more the model explains the variability in the response.

5.1.4. Mean Absolute Error (MAE)

The mean absolute error is a performance statistic that evaluates the average amount of residuals in a group of estimates without taking into account their sign. It is an evaluation technique for the accuracy.

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i - \widehat{y_i}| \tag{26}$$

where y_i is the actual response values and $\widehat{y_i}$ is the estimated response values in a data set of N observations.

It can be interpreted that the estimation reliability of the one with the smallest value among the MAE values calculated for each model and method is higher.

5.1.5. Explained Variance

The explained variance score is a very useful performance metric that looks at the ratio between the variance difference of the actual and predicted response values and the variance of the actual response values.

$$Explained\ Variance(y, \hat{y}) = 1 - \frac{Var(y) - Var(\hat{y})}{Var(y)} \tag{27}$$

The highest possible and desired score is 1 (it means all variation is explained), and thereafter the score decreases.

5.2. Multi Criteria Decision Making (MCDM)

Multi Criteria Decision Making (MCDM) can be defined as the selection of the best alternative among several potential candidates in a decision-making process, based on various criteria, which can be qualitative or quantitative [51].

In this study, MCDM methods were used to make the final decision. Two MCDM methods were taken into account to choose the best alternative among Ridge Regression, Lasso and Elastic Net based on the five performance measures described above. Since the effect of each performance criteria can be different for the model (that is, for some criteria, the highest value within the range gives a better model performance, for some criteria the opposite is true), Technique for Order Preference by Similarity to Ideal Solution (TOPSIS) and Multi-Objective Optimization by Ratio

Analysis Method (MOORA) were chosen among the multi-criteria decision making methods that can interpret these effects in the most accurate way and decide which shrinkage method is better to use in which data set.

In both methods, performance criteria are separated as cost-benefit and these features are taken into account in the decision-making process. It can be said that; for the criteria determined as cost, the smaller the better, and for the criteria determined as benefit, the bigger the better. Among the performance measures, R-squared and explained variance were determined as benefit, while the others were determined as cost.

5.2.1. Technique for Order Preference by Similarity to Ideal Solution (TOPSIS)

TOPSIS method, first proposed by Hwang and Yoon [52], aims to select the best one among many alternatives by considering some criteria. The chosen alternative should have the minimum geometrical distance from the positive ideal solution and the maximum geometrical distance from the negative ideal solution, according to the main idea of this technique. TOPSIS has the benefit of being able to quickly determine the best alternative [53]. The steps of the TOPSIS method are as follows [54].

Step 1: The decision matrix is created.

The decision matrix is an nxm dimensional matrix created by the decision maker after the decision options and evaluation criteria are determined. Here, n is the number of decision options and m is the and evaluation criteria.

Step 2: The standard decision matrix (normalized matrix) is created.

The standard decision matrix is obtained by taking the square root of the sum of the squares of the values of each criterion of the decision matrix (the sum of the squares of the column values) and dividing the relevant element of the column by this resulting value.

Step 3: A weighted standard decision matrix is created.

First of all, the weight values for the evaluation criteria are determined. By multiplying the elements of the standard decision matrix with their respective weight values, a weighted standard decision matrix is formed.

Step 4: Positive ideal and negative ideal solution values are obtained.

By using the weighted standard decision matrix, positive ideal and negative ideal solution sets are obtained for each criterion according to the purpose of the evaluation criterion of interest. If the evaluation criteria are in terms of benefit, the positive ideal solution is the largest of the columns of the matrix, and the negative ideal solution is the smallest of the columns of the matrix. If the evaluation criteria are in terms of cost, the positive ideal solution is the smallest values of the columns of the matrix, and the negative ideal solution is the largest values of the columns of the matrix.

Step 5: The distance values to the positive ideal and negative ideal solution values are obtained.

The Euclidean approach is used to find the deviations of the evaluation criteria for each decision option from the positive ideal and negative ideal solution values.

Step 6: The relative closeness coefficients to the ideal solution are calculated.

The distances from the positive ideal and negative ideal solution values are used to calculate the relative closeness coefficients of each decision option to the ideal solution.

5.2.2. Multi-Objective Optimization by Ratio Analysis Method (MOORA)

The Multi-Objective Optimization by Ratio Analysis Method (MOORA) is also one of the multi criteria decision making methods and it was introduced to the literature by Brauers and Zavadskas in 2006 [55]. It is preferred due to its features such as simplicity of application, short computation time, few mathematical operations and good reliability.

As in TOPSIS, the method starts with the creation of data in the form of a matrix, where alternatives form rows and criteria are columns. After that, normalization, ratio method, reference point method and exact product form are performed respectively. Finally, the MULTIMOORA step is applied [56]. However, in fact, MULTIMOORA is not a method or model on its own; it provides a final evaluation by evaluating the rankings made as a result of different MOORA steps according to their latest dominance.

CHAPTER 6

APPLICATION AND RESULTS

All application procedures were conducted by using MATLAB and R softwares. The simulated data sets used in the applications were produced using the Copula function in MATLAB software. Shrinkage method applications, calculation of performance metrics and applications of multi-criteria decision-making methods were carried out by R Studio software. The R Studio software used for model development and model comparison was preferred because it is a free software, its coding language is simple, and its interface is user-friendly.

6.1. Application and Results Based on Simulated Data Sets

In this part of the application, the k-fold cross validation method was preferred. With the k-fold cross validation method, the data set is divided into k parts and one of these k parts is used as the test data set, while the remaining (k-1) parts used for the train data set [57]. It is an estimation process recommended for the predicting a variable in a data set and trying to find out how accurately this model works in practice. The purpose of this approach is to test the model's ability to predict the response using new data that is not used in the prediction, while making prediction models. The separation of train and test data sets is made according to the determined k number. The forecasting model is first created using the train data set, and then the performance of the created model is validated on the test data set. For example, in the 5-fold cross validation method, the data set is divided into 5 equal parts as much as possible and one of these parts is called the test data set, while the remaining four are called the train data set. Afterwards, while the train data set is used to create the model, the test data set is used for the validation of the model.

In this thesis study, 10-fold (k=10) cross validation method has been applied to all simulated data sets. Each data set was first randomly divided into 10 sub-folds, and

9 of these sub-folds were used to construct the models, while the remaining one were used to test the model. For each data set, a linear model was first obtained, and then ridge regression, lasso and elastic net models were constructed. All of these model applications are done for all train data, and finally the original full data set. For all simulated data sets, the performance of the models was obtained based on several performance measures described in Chapter 5 for each sub-fold and for both train and test data sets. The coefficient estimates of the models created for 10 train data sets are given in Appendix A. As mentioned before, when the coefficient estimates are examined, it will be seen that the coefficient estimates of the independent variables with multicollinearity in the ridge regression models are very close to zero, but they are never zero. In addition, it is observed that some of the lasso and elastic net coefficient estimates are directly zero. In all tables in Appendix A, if the estimated coefficients are zero, they are demonstrated with dots "." for better visualization.

In order to make comparisons and comments between shrinkage methods, a single performance table was obtained by taking the average of each performance criteria obtained from the models that constructed for each sub-fold. The performance tables for the nine data sets are given in Table 6.1 to Table 6.9, respectively.

Table 6.1 Performance Measures of Data Set 1

		MSE	**PMSE**	**Rsq**	**MAE**	**Exp.Var.**
Test	**RR**	0.0816	0.0408	0.7918	0.2433	0.8567
	LASSO	0.0792	0.0440	0.7958	0.2371	0.8668
	EN	0.0796	0.0440	0.7926	0.2371	0.8902
Train	**RR**	0.0672	0.0604	0.8864	0.2214	0.8187
	LASSO	0.0653	0.0600	0.8894	0.2168	0.8332
	EN	0.0650	0.0596	0.8900	0.2159	0.8522
Full Data	**RR**	0.0679	0.0617	0.8857	0.2228	0.8183
	LASSO	0.0655	0.0606	0.8898	0.2161	0.8648
	EN	0.0655	0.0606	0.8898	0.2161	0.8656

Table 6.2 Performance Measures of Data Set 2

		MSE	PMSE	Rsq	MAE	Exp.Var.
Test	**RR**	0.0977	0.0279	0.9086	0.2535	0.9573
	LASSO	0.1028	0.0306	0.9035	0.2589	0.9672
	EN	0.1005	0.0299	0.9060	0.2598	0.9844
Train	**RR**	0.0570	0.0446	0.9623	0.1921	0.9152
	LASSO	0.0565	0.0448	0.9627	0.1914	0.9226
	EN	0.0546	0.0433	0.9639	0.1885	0.9386
Full Data	**RR**	0.0589	0.0471	0.9611	0.1954	0.9140
	LASSO	0.0558	0.0452	0.9632	0.1922	0.9494
	EN	0.0561	0.0455	0.9630	0.1923	0.9444

Table 6.3 Performance Measures of Data Set 3

		MSE	PMSE	Rsq	MAE	Exp.Var.
Test	**RR**	0.2019	0.0288	0.9450	0.3623	0.9391
	LASSO	0.2776	0.0423	0.9227	0.4256	0.9664
	EN	0.2612	0.0394	0.9270	0.4204	0.9879
Train	**RR**	0.0513	0.0308	0.9909	0.1803	0.9500
	LASSO	0.0461	0.0285	0.9918	0.1695	0.9599
	EN	0.0424	0.0261	0.9925	0.1624	0.9697
Full Data	**RR**	0.0566	0.0354	0.9900	0.1911	0.9489
	LASSO	0.0478	0.0307	0.9915	0.1735	0.9729
	EN	0.0481	0.0308	0.9915	0.1747	0.9714

Table 6.4 Performance Measures of Data Set 4

		MSE	PMSE	Rsq	MAE	Exp.Var.
Test	**RR**	0.0896	0.0747	0.8363	0.2601	0.8092
	LASSO	0.0896	0.0772	0.8361	0.2601	0.8216
	EN	0.0891	0.0768	0.8368	0.2597	0.8384
Train	**RR**	0.0857	0.0839	0.8537	0.2549	0.7985
	LASSO	0.0852	0.0837	0.8547	0.2542	0.8106
	EN	0.0848	0.0833	0.8554	0.2540	0.8272
Full Data	**RR**	0.0136	0.0134	0.8853	0.2232	0.8134
	LASSO	0.0131	0.0129	0.8899	0.2153	0.8789
	EN	0.0131	0.0129	0.8899	0.2153	0.8780

Table 6.5 Performance Measures of Data Set 5

		MSE	PMSE	Rsq	MAE	Exp.Var.
Test	**RR**	0.0966	0.0644	0.9707	0.2664	0.9159
	LASSO	0.0932	0.0640	0.9718	0.2619	0.9458
	EN	0.0922	0.0623	0.9720	0.2615	0.9590
Train	**RR**	0.0873	0.0827	0.9743	0.2523	0.9143
	LASSO	0.0836	0.0796	0.9754	0.2475	0.9434
	EN	0.0826	0.0784	0.9757	0.2470	0.9559
Full Data	**RR**	0.0878	0.0836	0.9742	0.2532	0.9142
	LASSO	0.0826	0.0790	0.9757	0.2477	0.9654
	EN	0.0825	0.0788	0.9758	0.2477	0.9658

Table 6.6 Performance Measures of Data Set 6

		MSE	PMSE	Rsq	MAE	Exp.Var.
Test	**RR**	0.1061	0.0482	0.9726	0.2744	0.9475
	LASSO	0.1076	0.0502	0.9720	0.2745	0.9410
	EN	0.1037	0.0480	0.9730	0.2707	0.9618
Train	**RR**	0.0822	0.0726	0.9803	0.2420	0.9446
	LASSO	0.0812	0.0720	0.9805	0.2388	0.9405
	EN	0.0782	0.0692	0.9812	0.2356	0.9587
Full Data	**RR**	0.0834	0.0745	0.9800	0.2438	0.9447
	LASSO	0.0783	0.0702	0.9812	0.2367	0.9735
	EN	0.0783	0.0701	0.9812	0.2367	0.9732

Table 6.7 Performance Measures of Data Set 7

		MSE	PMSE	Rsq	MAE	Exp.Var.
Test	**RR**	0.0864	0.0810	0.9719	0.2502	0.9044
	LASSO	0.0827	0.0775	0.9731	0.2470	0.9455
	EN	0.0823	0.0771	0.9732	0.2468	0.9547
Train	**RR**	0.0852	0.0845	0.9725	0.2484	0.9054
	LASSO	0.0813	0.0807	0.9738	0.2452	0.9465
	EN	0.0810	0.0804	0.9739	0.2449	0.9557
Full Data	**RR**	0.0852	0.0847	0.9725	0.2485	0.9053
	LASSO	0.0809	0.0804	0.9739	0.2450	0.9619
	EN	0.0809	0.0804	0.9739	0.2450	0.9613

Table 6.8 Performance Measures of Data Set 8

		MSE	PMSE	Rsq	MAE	Exp.Var.
Test	**RR**	0.0907	0.0777	0.9808	0.2600	0.9263
	LASSO	0.0872	0.0752	0.9815	0.2550	0.9534
	EN	0.0862	0.0743	0.9817	0.2538	0.9649
Train	**RR**	0.0871	0.0856	0.9819	0.2547	0.9269
	LASSO	0.0836	0.0822	0.9826	0.2493	0.9536
	EN	0.0827	0.0812	0.9828	0.2482	0.9650
Full Data	**RR**	0.0873	0.0859	0.9818	0.2550	0.9269
	LASSO	0.0826	0.0813	0.9828	0.2481	0.9716
	EN	0.0826	0.0813	0.9828	0.2482	0.9715

Table 6.9 Performance Measures of Data Set 9

		MSE	PMSE	Rsq	MAE	Exp.Var.
Test	**RR**	0.0904	0.0646	0.9814	0.2551	0.9450
	LASSO	0.0909	0.0649	0.9813	0.2545	0.9456
	EN	0.0882	0.0630	0.9819	0.2525	0.9637
Train	**RR**	0.0841	0.0805	0.9829	0.2458	0.9444
	LASSO	0.0836	0.0800	0.9830	0.2441	0.9451
	EN	0.0809	0.0775	0.9836	0.2419	0.9624
Full Data	**RR**	0.0844	0.0811	0.9828	0.2465	0.9446
	LASSO	0.0804	0.0773	0.9837	0.2418	0.9743
	EN	0.0804	0.0773	0.9837	0.2418	0.9741

When the values of the performance measures in the tables are examined, it is seen that the performances are generally good. As desired, MSE, PMSE and MAE values are low in general, while R-squared and explained variance values are high. The reason for this could be related with the linear models that used for simulation. In other words, the data set were produced according to the linear model. Overall, there doesn't seem to be any significant difference in performance in all three methods. But of course, performance values change as the data set changes. In general, the lowest MSE and PMSE values among the performance measures belong to the lasso model, which was created using the full data of data set 4. The highest R-squared value and the lowest MAE value were obtained in the elastic net model created with the train data of data set 3. And finally, the highest explained variance value comes from the elastic net model obtained using the test data of data set 3. Moreover, Table 6.10 summarizes the best performance results for each measurement in terms of test, train and full data sets.

Table 6.10 Best Performance According to Each Measurement

	MSE	**PMSE**	**Rsq**	**MAE**	**Exp.Var.**
Test	Data set 1 Lasso	Data set 2 Ridge	Data set 8 Ridge	Data set 1 Elastic Net	Data set 3 Elastic Net
Train	Data set 3 Elastic Net	Data set 3 Elastic Net	Data set 3 Elastic Net	Data set 3 Elastic Net	Data set 3 Elastic Net
Full Data	Data set 4 Lasso	Data set 4 Lasso	Data set 3 Lasso	Data set 3 Lasso	Data set 9 Lasso

It can be seen from this table that, although shrinkage methods applied to smaller data sets generally give better results, the most efficient results among train data are seen in elastic net models created using data set 3. Similarly, it is obvious that the best performance measures are calculated with the lasso method in models created using full data.

Afterwards, in the light of all the performance tables created, the TOPSIS method is applied in order to select which shrinkage method has the best performance for all data sets. TOPSIS method applications are made with the help of MCDM package in R Studio software [58]. For the use of the function in the package, first of all, decision matrices containing shrinkage methods and performance measurements for test, train and full data sets were created. Then, a weight of 0.2 was assigned to each of the five identified performance measures, with a total of 1. At the same time, performance criteria are separated according to cost-benefit features. MSE, PMSE and MAE are defined as "min" because they have the cost feature and are intended to be minimized, while R-squared and explained variance are defined as "max" because they are intended to be maximized because they have the benefit feature. After these assignments, TOPSIS evaluation was made for test, train and full data performance results belonging each data set. As a result of each evaluation, a preference order was obtained from the shrinkage method with the highest score to the lowest score. Tables with TOPSIS results are given in Table 6.11 to Table 6.19 for each data set, respectively.

Table 6.11 TOPSIS Results of Data Set 1

		Score	Rank
Test	**RR**	0.5722271	1
	LASSO	0.3456333	3
	EN	0.4140281	2
Train		**Score**	**Rank**
	RR	0	3
	LASSO	0.6214759	2
	EN	1	1
Full Data		**Score**	**Rank**
	RR	0	3
	LASSO	0.9875548	2
	EN	1	1

Table 6.12 TOPSIS Results of Data Set 2

		Score	Rank
Test	**RR**	1	1
	LASSO	0.00624351	3
	EN	0.05676452	2
Train		**Score**	**Rank**
	RR	0.06738739	3
	LASSO	0.18716981	2
	EN	1	1
Full Data		**Score**	**Rank**
	RR	0	3
	LASSO	1	1
	EN	0.8721039	2

Table 6.13 TOPSIS Results of Data Set 3

		Score	Rank
Test	**RR**	0.90028228	1
	LASSO	0.05828281	3
	EN	0.18273515	2
Train		**Score**	**Rank**
	RR	0	3
	LASSO	0.5107087	2
	EN	1	1
Full Data		**Score**	**Rank**
	RR	0	3
	LASSO	1	1
	EN	0.9560512	2

Table 6.14 TOPSIS Results of Data Set 4

		Score	Rank
Test	**RR**	0.4783831	2
	LASSO	0.2773924	3
	EN	0.5660921	1
Train		**Score**	**Rank**
	RR	0	3
	LASSO	0.4332017	2
	EN	1	1
Full Data		**Score**	**Rank**
	RR	0	3
	LASSO	1	1
	EN	0.9895627	2

Table 6.15 TOPSIS Results of Data Set 5

		Score	Rank
Test	**RR**	0	3
	LASSO	0.6114169	2
	EN	1	1
Train		**Score**	**Rank**
	RR	0	3
	LASSO	0.7396961	2
	EN	1	1
Full Data		**Score**	**Rank**
	RR	0	3
	LASSO	0.9721209	2
	EN	1	1

Table 6.16 TOPSIS Results of Data Set 6

		Score	Rank
Test		**Score**	**Rank**
	RR	0.5819525	2
	LASSO	0	3
	EN	1	1
Train		**Score**	**Rank**
	RR	0.05462647	3
	LASSO	0.24771751	2
	EN	1	1
Full Data		**Score**	**Rank**
	RR	0	3
	LASSO	0.9850303	2
	EN	0.9967503	1

Table 6.17 TOPSIS Results of Data Set 7

Test		**Score**	**Rank**
	RR	0	3
	LASSO	0.8628499	2
	EN	1	1
Train		**Score**	**Rank**
	RR	0	3
	LASSO	0.8763089	2
	EN	1	1
Full Data		**Score**	**Rank**
	RR	0	3
	LASSO	1	1
	EN	0.9933624	2

Table 6.18 TOPSIS Results of Data Set 8

Test		**Score**	**Rank**
	RR	0	3
	LASSO	0.7417494	2
	EN	1	1
Train		**Score**	**Rank**
	RR	0	3
	LASSO	0.7619066	2
	EN	1	1
Full Data		**Score**	**Rank**
	RR	0	3
	LASSO	1	1
	EN	0.995532	2

Table 6.19 TOPSIS Results of Data Set 9

		Score	Rank
Test		**Score**	**Rank**
	RR	0.1458047	2
	LASSO	0.04946189	3
	EN	1	1
Train		**Score**	**Rank**
	RR	0	3
	LASSO	0.1800004	2
	EN	1	1
Full Data		**Score**	**Rank**
	RR	0	3
	LASSO	1	1
	EN	0.9972938	2

Based on these results, first of all, it can be commented that lasso and elastic net, created by using full data set, got very close scores in some TOPSIS results, that is, the performance measurements of both models are very close to each other and one of them was chosen as a result of the evaluation. However, in such cases, it can be said that both models can be used since both of them will give good results in estimation.

Another point that draws attention on the TOPSIS results is that when the model scores of the test data (i.e. in small-sized data sets) are examined, it is observed that the ridge regression model generally gives good results and even gives the best results in the first three data sets. However, as the data sets get larger (train and full data sets), it is observed that the performance of the ridge regression models decreases as the complexity increases and the models tend to make variable selection, therefore the scores of the lasso and elastic net models are better.

As it done in the performance tables, if the overall TOPSIS results for the test, train and full data are examined, Table 6.20 is obtained.

Table 6.20 Overall TOPSIS Results for Simulated Data Sets

	Data set 1	**Data set 2**	**Data set 3**	**Data set 4**	**Data set 5**	**Data set 6**	**Data set 7**	**Data set 8**	**Data set 9**
Test	Ridge	Ridge	Ridge	Elastic Net	Elastic Net	Elastic Net	Elastic Net	Elastic Net	Elastic Net
Train	Elastic Net	Elastic Net	Elastic Net	Elastic Net	Elastic Net	Elastic Net	Elastic Net	Elastic Net	Elastic Net
Full Data	Elastic Net	Lasso	Lasso	Lasso	Elastic Net	Elastic Net	Lasso	Lasso	Lasso

Considering this table, for the test data, it is seen that ridge regression gives better results in small size data sets, while elastic net models give better results as the number of observations increases. When the performance values calculated for the models of all data sets are compared among the models created using train data, it is seen that the model that makes the best prediction for each data set is elastic net. Looking at the TOPSIS evaluation results for models created using full data, it is seen that lasso or elastic net models give the best results since their scores are very close to each other. In other words, for each data set, both lasso and elastic net can make good estimations and it can be interpreted that they can be used interchangeably.

In addition to TOPSIS evaluations, MOORA, which is another multi-criteria decision-making method, was applied to all data sets by using same performance measurements. Exactly same results were obtained with TOPSIS. The result tables of MOORA are provided in Appendix B.

As a result, it can be said that ridge regression models generally give better results in small size data sets. Since the complexity of the data set increases as the size of the data or the number of independent variables increases, shrinkage methods tend to make some coefficients zero in order to reduce the increasing variances of the coefficients. Therefore, it can be deduced that lasso or elastic net models generally give the best results in medium or large size data sets.

6.2. Application and Results Based on Real World Data Sets

In this part of the thesis study, real world data sets mentioned in Section 4.2. were used for the applications of shrinkage method. The real world data sets were tried to be selected to represent data sets 1, 5 and 9, among the simulated data sets, respectively. Before applying shrinkage methods to real world data sets, the hold-out method was applied in order to divide the data as train-test. Afterwards, ridge regression, lasso and elastic net models were obtained for these real world data sets. The estimated coefficients of the applied models for all three data sets are given in Appendix C. Considering the estimated coefficients given in Appendix C, for example, for real world data set 1, all variable coefficients of the ridge regression model (for the model created with full data) are different from zero, while the value of the last four variables for the model created with the lasso method is zero. For the model of elastic net, while the coefficient of the first five variables is different from zero as in lasso, the coefficient of the eighth variable is also different from zero, unlike lasso. These results are also suitable for the nature of the shrinkage methods mentioned. While ridge regression created a model using all variables, elastic net created a model between ridge regression and lasso.

The five previously used performance measures were also calculated for all models and are given in Table 6.21, Table 6.22 and Table 6.23 for each data set, respectively.

Table 6.21 Performance Measures of Real World Data Set 1

		MSE	PMSE	Rsq	MAE	Exp.Var.
Test	**RR**	0.00014	0.0001	0.9889	0.0096	0.9342
	LASSO	0.00013	0.0001	0.9893	0.0105	0.8764
	EN	0.00008	0.0001	0.9937	0.0077	0.9176
Train	**RR**	0.00029	0.0002	0.9923	0.0106	0.9390
	LASSO	0.0002	0.0002	0.9949	0.0106	0.8809
	EN	0.00014	0.0001	0.9963	0.0080	0.9192
Full Data	**RR**	0.00028	0.0002	0.9916	0.0103	0.9343
	LASSO	0.00003	0.0000	0.9990	0.0045	0.9465
	EN	0.00004	0.0000	0.9988	0.0040	0.9613

Table 6.22 Performance Measures of Real World Data Set 2

		MSE	PMSE	Rsq	MAE	Exp.Var.
Test	**RR**	0.0262	0.0201	0.3786	0.1114	0.6507
	LASSO	0.0258	0.0210	0.3898	0.1093	0.5181
	EN	0.0257	0.0210	0.3901	0.1088	0.5688
Train	**RR**	0.0168	0.0156	0.5004	0.0888	0.4549
	LASSO	0.0181	0.0171	0.4612	0.0909	0.3649
	EN	0.0179	0.0170	0.4661	0.0899	0.4060
Full Data	**RR**	0.0178	0.0168	0.4962	0.0915	0.4529
	LASSO	0.0172	0.0164	0.5149	0.0905	0.4791
	EN	0.0172	0.0165	0.5138	0.0906	0.4766

Table 6.23 Performance Measures of Real World Data Set 3

		MSE	PMSE	Rsq	MAE	Exp.Var.
Test	**RR**	0.0110	0.0108	0.6842	0.0801	0.6380
	LASSO	0.0137	0.0134	0.6076	0.0909	0.5035
	EN	0.0126	0.0124	0.6370	0.0872	0.5568
Train	**RR**	0.0103	0.0103	0.6971	0.0785	0.6444
	LASSO	0.0129	0.0128	0.6227	0.0885	0.5092
	EN	0.0119	0.0118	0.6518	0.0850	0.5640
Full Data	**RR**	0.0105	0.0104	0.6946	0.0789	0.6421
	LASSO	0.0091	0.0090	0.7359	0.0720	0.7333
	EN	0.0091	0.0090	0.7356	0.0721	0.7320

Looking at the tables created from the performance measurements of real world data sets, it is seen that the first data set gives the best results in terms of all performance measurements. It is also seen that the first data set and mostly the elastic net regression model give the best results in terms of all performance measurements.

After creating the performance tables, the same steps applied for TOPSIS in simulated data sets are applied for real world data sets. Tables containing the TOPSIS results for each real world data set are given in Table 6.24, Table 6.25 and Table 6.26, respectively. It should be noted that both TOPSIS and MOORA have the same results as in the previous subsection. MOORA results are given in Appendix D.

Table 6.24 TOPSIS Results of Real World Data Set 1

		Score	Rank
Test		**Score**	**Rank**
	RR	0.207134	2
	LASSO	0.031779	3
	EN	0.973552	1
Train		**Score**	**Rank**
	RR	0.076681	3
	LASSO	0.400602	2
	EN	0.972566	1
Full Data		**Score**	**Rank**
	RR	0	3
	LASSO	0.923928	1
	EN	0.893815	2

Table 6.25 TOPSIS Results of Real World Data Set 2

Test		**Score**	**Rank**
	RR	0.833355	1
	LASSO	0.156842	3
	EN	0.400843	2
Train		**Score**	**Rank**
	RR	1	1
	LASSO	0	3
	EN	0.359797	2
Full Data		**Score**	**Rank**
	RR	0	3
	LASSO	1	1
	EN	0.918769	2

Table 6.26 TOPSIS Results of Real World Data Set 3

Test		**Score**	**Rank**
	RR	1	1
	LASSO	0	3
	EN	0.356006	2
Train		**Score**	**Rank**
	RR	1	1
	LASSO	0	3
	EN	0.363172	2
Full Data		**Score**	**Rank**
	RR	0	3
	LASSO	1	1
	EN	0.990807	2

When looking at the overall TOPSIS results given in Table 6.27, it is observed that the difference between the scores of the test and train models increases significantly as the data set grows. Moreover, it is seen that there is a consistency between both test and train scores in terms of model preference for each data set. For example, elastic net stands out for both train and test in the first data set, while ridge regression gives the best results in the second and third data sets. On the other hand, lasso method gives the best results when the full data is modeled for each real world data set. Finally, if the results of simulated and real world data sets are compared, it turns out that the results are consistent when we considering the use of full data. Thus, in general, the methods lasso or elastic net can be recommended. It should be noted that the simulated data sets are constructed according to the linear models. In addition to that it should be taken into account that the real world data sets used in this thesis may have a non-linear structure in nature.

Table 6.27 Overall TOPSIS Results for Real World Data Sets

	Data set 1	**Data set 2**	**Data set 3**
Test	Elastic Net	Ridge	Ridge
Train	Elastic Net	Ridge	Ridge
Full Data	Lasso	Lasso	Lasso

CHAPTER 7

CONCLUSION

The aim of this thesis study is to apply some shrinkage methods to simulated and real world data sets with different characteristics in case of multicollinearity problem between the independent variables, and to compare the results and create a preference order among these methods.

For this purpose, first of all, the way of applying linear models and their aims are mentioned. The least squares estimation method, which is the most used parameter estimation method while creating the linear model, is explained. Afterwards, the multicollinearity problem, which causes the performance of linear models to be low, is mentioned. Then, the methods of how to identify the multicollinearity problem and how to solve it are discussed one by one. In the case that many independent variables in the data set are related to each other, the large variance of the estimated coefficients in the linear models created using this data set prevents obtaining accurate and reliable results. The most powerful and common methods used to eliminate this problem are the biased regression models called the shrinkage methods.

Shrinkage methods reduce the variance by increasing the bias of parameter estimations of linear regression models whose estimation accuracy decreases in the face of multicollinearity problem. For this purpose, shrinkage methods set the estimated coefficients to values very close to zero in the ridge regression, while they set the estimated coefficient values very close to or directly to zero in the lasso and elastic net regression.

In order to perform these biased regression models, first of all, nine data sets with different characteristics and multicollinearity among their independent variables were simulated using the Copula function. Using the 10-fold cross validation method, all simulated data sets are split into train-test data and then shrinkage methods are applied to all of them. Performance measurements obtained from each sub-fold were averaged for each data set in order to make comparisons between data sets and shrinkage methods. TOPSIS method, which is one of the multi-criteria decision making methods, was applied by using the final performance tables.

In real world data sets, it was determined that there was multicollinearity among the variables and they were suitable for shrinkage method applications. Afterwards, the hold-out method was used because of its simplicity and the data was randomly divided into two as 80% train and 20% test data. All shrinkage methods were applied to train, test and full data, and performance measurements were calculated for each data set and each model. The obtained performance tables were used to apply the TOPSIS method and it was observed that which shrinkage method gave better results in which data set.

The results obtained from mentioned shrinkage methods were examined for each train, test and full data sets applications. Based on these results, in general, there is a consistency between both test and train scores in terms of model preference. Moreover, if we consider the applications of ridge regression, lasso and elastic net methods applied to the full data both for the simulated and real world data sets, the lasso and elastic net models generally give the best results, looking at the majority. To conclude from all the applications, if a ranking is made among all methods, the order of preference of shrinkage methods can be determined as lasso, elastic net and ridge regression.

As a future work, other shrinkage methods (for example, fused lasso or liu) that were not used in this thesis study can be applied on the same simulated and real world data sets and the results can be compared in more detail. Another future work may be to use classification type data sets instead of the prediction type data sets used in this thesis study. In case classification type data sets are used, the corresponding performance measures should be used and evaluated in TOPSIS, which is a method of comparing alternatives, is updated according to these performance measures.

REFERENCES

[1] M. I. S. Oliveira and B. F. Lóscio, “What is a data ecosystem?,” in *19th Annual International Conference on Digital Government Research*, 2018, pp. 1–9.

[2] O. Toka, “A Comparative Study on Regression Methods in the presence of Multicollinearity,” *J. Stat. Stat. Actuar. Sci.*, vol. 9, no. 2, pp. 47–53, 2016.

[3] G. K. Uyanık and N. Güler, “A Study on Multiple Linear Regression Analysis,” *Procedia - Soc. Behav. Sci.*, vol. 106, pp. 234–240, 2013.

[4] E. Kalkan, “Lineer Regresyonda Küçültme Yöntemlerinin Karşılaştırılması,” Yüksek Lisans Tezi, Dicle Üniversitesi, Türkiye, 2020.

[5] Z. Y. Algamal, “Shrinkage parameter selection via modified cross-validation approach for ridge regression model,” *Commun. Stat. Simul. Comput.*, vol. 49, no. 7, pp. 1922–1930, 2020.

[6] M. I. Alheety and B. M. Golam Kibria, “A generalized stochastic restricted ridge regression estimator,” *Commun. Stat. - Theory Methods*, vol. 43, no. 20, pp. 4415–4427, 2014.

[7] F. S. M. Batah, M. R. Özkale, and S. D. Gore, “Combining unbiased ridge and principal component regression estimators,” *Commun. Stat. - Theory Methods*, vol. 38, no. 13, pp. 2201–2209, 2009.

[8] R. James, G., Witten, D., Hastie, T., Tibshirani, *An Introduction to Statistical Learning - with Applications in R*. New York: Springer, 2013.

[9] S. Mohammadi, “A test of harmful multicollinearity: A generalized ridge regression approach,” *Commun. Stat. - Theory Methods*, vol. 51, no. 3, pp. 724–743, 2020.

[10] D. A. Belsley, E. Kuh, and R. E. Welsch, *Regression Diagnostics: Identifying Influential Data and Sources of Collinearity*. New York: John Wiley & Sons, 1980.

[11] D. Aydın, *Uygulamalı Regresyon Analizi - Kavramlar ve R Hesaplamaları*. Ankara: Nobel Akademik Yayıncılık, 2014.

[12] B. M. G. Kibria, “Performance of Some New Ridge Regression Estimators,” *Commun. Stat. Part B Simul. Comput.*, vol. 32, no. 2, pp. 419–435, 2003.

[13] A. E. Hoerl and R. W. Kennard, “Ridge Regression: Biased Estimation for Nonorthogonal Problems,” *Technometrics*, vol. 12, no. 1, pp. 55–67, 1970.

[14] R. Tibshirani, “Regression Shrinkage and Selection via the Lasso,” *J. R. Stat. Soc. Ser. B*, vol. 58, no. 1, pp. 267–288, 1996.

[15] H. Zou and T. Hastie, “Regularization and variable selection via the elastic net,” *J. R. Stat. Soc. Ser. B Stat. Methodol.*, vol. 67, no. 2, pp. 301–320, 2005.

[16] D. Jato-Espino, E. Castillo-Lopez, J. Rodriguez-Hernandez, and J. C. Canteras-Jordana, “A review of application of multi-criteria decision making methods in construction,” *Autom. Constr.*, vol. 45, pp. 151–162, 2014.

[17] G.-H. Tzeng and J.-J. Huang, *Multiple Attribute Decision Making: Methods and Applications*. New York: Springer-Verlag, 2011.

[18] W. L. Martinez and A. R. Martinez, *Computational Statistics Handbook with MATLAB*. London: Chapman and Hall, CRC, 2002.

[19] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2nd ed. New York: Springer, 2001.

[20] D. W. Marquardt and R. D. Snee, “Ridge Regression in Practice,” *Am. Stat.*, vol. 29, no. 1, pp. 3–20, 1975.

[21] S. H. Choi, H. Y. Jung, and H. Kim, “Ridge Fuzzy Regression Model,” *Int. J. Fuzzy Syst.*, vol. 21, no. 7, pp. 2077–2090, 2019.

[22] S. Lipovetsky, “Enhanced ridge regressions,” *Math. Comput. Model.*, vol. 51, no. 5–6, pp. 338–348, 2010.

[23] H. D. Hamedani and S. Sadat Moosavi, “Dealing with big data: comparing dimension reduction and shrinkage regression methods,” *J. Appl. Stat.*, vol. 44, no. 3, pp. 511–532, 2017.

[24] S. Çankaya, S. Eker, and S. H. Abacı, “Comparison of Least Squares, Ridge Regression and Principal Component Approaches in the Presence of Multicollinearity in Regression Analysis,” *Turkish J. Agric. - Food Sci. Technol.*, vol. 7, no. 8, pp. 1166–1172, 2019.

[25] D. C. Montgomery and G. C. Runger, *Applied Statistics and Probability for Engineers*. New York: John Wiley & Sons, Inc., 2011.

[26] M. Topal, E. Eyduran, A. M. Yağanoğlu, Y. A. Sönmez, and S. Keskin, “Çoklu Doğrusal Bağlantı Durumunda Ridge ve Temel Bileşenler Regresyon Analiz Yöntemlerinin Kullanımı,” *J. Agric. Fac. Atatürk Univ.*, vol. 41, no. 1, pp. 53–57, 2010.

[27] J. Neter, W. Wasserman, and M. H. Kutner, *Applied Linear Regression Models*. Boston: Irwın Inc., 1989.

[28] G. Khalaf, K. Mansson, and G. Shukur, “Modified ridge regression estimators,” *Commun. Stat. - Theory Methods*, vol. 42, no. 8, pp. 1476–1487, 2013.

[29] D. N. Gujarati, *Basic Econometrics*, 4th ed. New Delhi; London: Tata McGraw-Hill, 2004.

[30] S. Albayrak, “Çoklu Doğrusal Bağlantı Halinde En Küçük Kareler Tekniğinin Alternatifi Yanlı Tahmin Teknikleri ve Bir Uygulama,” *ZKÜ Sos. Bilim. Derg. Cilt*, vol. 1, no. 1, pp. 106–125, 2005.

[31] N. A. M. R. Senaviratna and T. M. J. A. Cooray, “Diagnosing Multicollinearity of Logistic Regression Model,” *Asian J. Probab. Stat.*, vol. 5, no. 2, pp. 1–9, 2019.

[32] H. Midi, S. K. Sarkar, and S. Rana, “Collinearity diagnostics of binary logistic regression model,” *J. Interdiscip. Math.*, vol. 13, no. 3, pp. 253–267, 2013.

[33] D. W. Marquardt, “Generalized Inverses, Ridge Regression, Biased Linear Estimation, and Nonlinear Estimation,” *Technometrics*, vol. 12, no. 3, pp. 591–612, 1970.

[34] A. L. Webster, *Applied Statistics for Business and Economics*. New York: McGraw-Hill/Irwin, 1995.

[35] Y. Wu, "Can't Ridge Regression Perform Variable Selection?," *Technometrics*, vol. 63, no. 2, pp. 263–271, 2020.

[36] A. E. Hoerl and R. W. Kennard, "Ridge Regression: Applications to Nonorthogonal Problems," *Technometrics*, vol. 12, no. 1, pp. 69–82, 1970.

[37] H. Zou, "Comment: Ridge Regression—Still Inspiring After 50 Years," *Technometrics*, vol. 62, no. 4, pp. 456–458, 2020.

[38] P. J. García-Nieto, E. García-Gonzalo, and J. P. Paredes-Sá Nchez, "Prediction of the critical temperature of a superconductor by using the WOA/MARS, Ridge, Lasso and Elastic-net machine learning techniques," *Neural Comput. Appl.*, vol. 33, pp. 17131–17145, 2021.

[39] "Copulas: Generate Correlated Samples - MATLAB & Simulink." https://www.mathworks.com/help/stats/copulas-generate-correlated-samples.html (accessed Nov. 03, 2021).

[40] A. G. Asuero, A. Sayago, and A. G. González, "The Correlation Coefficient: An Overview," *Crit. Rev. Anal. Chem.*, vol. 36, no. 1, pp. 41–59, 2006.

[41] "UCI: Machine Learning Repository." http://archive.ics.uci.edu/ml/ (accessed Jun. 13, 2021).

[42] R. P. Ferreira, A. Martiniano, A. Ferreira, A. Ferreira, and R. J. Sassi, "Study on Daily Demand Forecasting Orders Using Artificial Neural Network," *IEEE Lat. Am. Trans.*, vol. 14, no. 3, pp. 1519–1525, 2016.

[43] "UCI Machine Learning Repository: Daily Demand Forecasting Orders Data Set." https://archive.ics.uci.edu/ml/datasets/Daily+Demand+Forecasting+Orders (accessed Jun. 13, 2021).

[44] "Hitters: Baseball Data in ISLR: Data for an Introduction to Statistical Learning with Applications in R." https://rdrr.io/cran/ISLR/man/Hitters.html (accessed Jun. 13, 2021).

[45] K. Hamidieh, “A data-driven statistical model for predicting the critical temperature of a superconductor,” *Comput. Mater. Sci.*, vol. 154, pp. 346–354, 2018.

[46] “UCI Machine Learning Repository: Superconductivty Data Set.” https://archive.ics.uci.edu/ml/datasets/superconductivty+data (accessed Jun. 13, 2021).

[47] D. Singh and B. Singh, “Investigating the impact of data normalization on classification performance,” *Appl. Soft Comput.*, vol. 97, 2020.

[48] J. Han, J. Pei, and M. Kamber, *Data Mining: Concepts and Techniques*. San Francisco: Morgan Kaufmann Publishers, 2001.

[49] A. Jain, K. Nandakumar, and A. Ross, “Score normalization in multimodal biometric systems,” *Pattern Recognit.*, vol. 38, pp. 2270–2285, 2005.

[50] W. J. Fu, “A Statistical Shrinkage Model and Its Applications,” Ph.D. Thesis, University of Toronto, Canada, 1998.

[51] K. T. Cho, “Multicriteria Decision Methods: An Attempt to Evaluate and Unify,” *Math. Comput. Model.*, vol. 37, no. 9–10, pp. 1099–1119, 2003.

[52] C.-L. Hwang and K. Yoon, *Multiple Attribute Decision Making: Methods and Applications: A State of the Art Survey*. Berlin: Springer-Verlag, 1981.

[53] C. Parkan and M. L. Wu, “On the equivalence of operational performance measurement and multiple attribute decision making,” *Int. J. Prod. Res.*, vol. 35, no. 11, pp. 2963–2988, 1997.

[54] S. Eraslan, “A Decision Making Method via TOPSIS on Soft Sets,” *J. New Results Sci.*, vol. 4, no. 8, pp. 57–70, 2015.

[55] W. K. M. Brauers and E. K. Zavadskas, “The MOORA method and its application to privatization in a transition economy,” *Control Cybern.*, vol. 35, no. 2, pp. 445–469, 2006.

[56] W. K. M. Brauers and E. K. Zavadskas, “Project management by multimoora as an instrument for transition economies,” *Technol. Econ. Dev. Econ.*, vol. 16, no. 1, pp. 5–24, 2010.

[57] J. He and X. Fan, “Evaluating the Performance of the K-fold Cross-Validation Approach for Model Selection in Growth Mixture Modeling,” *Struct. Equ. Model. A Multidiscip. J.*, vol. 26, no. 1, pp. 66–79, 2019.

[58] “CRAN - Package MCDM.” https://cran.r-project.org/web/packages/MCDM/index.html (accessed Apr. 25, 2022).

APPENDIX A

Estimated Coefficients of Simulated Data Sets

Table A.1 Estimated Coefficients of Data Set 1

	Data set 1 - 1				Data set 1 - 2		
	RR	**LASSO**	**EN**		**RR**	**LASSO**	**EN**
Intercept	1.3408	1.7318	1.5378	**Intercept**	1.3138	1.7530	1.5315
Beta1	0.1986	.	0.0484	**Beta1**	0.2734	.	0.1113
Beta2	0.7229	0.9943	0.8275	**Beta2**	0.6977	0.9665	0.8048
Beta3	0.4924	0.3926	0.4953	**Beta3**	0.5573	0.5151	0.5671
Beta4	0.3717	0.1923	0.3414	**Beta4**	0.3315	0.2299	0.3191
Beta5	0.4121	0.3204	0.3655	**Beta5**	0.4585	0.3534	0.4057
Beta6	0.8408	0.7127	0.7785	**Beta6**	0.8434	0.7081	0.7774
Beta7	0.3642	0.3052	0.3037	**Beta7**	0.3172	0.2230	0.2532
Beta8	0.6635	0.5775	0.6207	**Beta8**	0.7320	0.5901	0.6625
Beta9	0.6884	0.5470	0.6144	**Beta9**	0.6915	0.5396	0.6150
Beta10	0.7250	0.6043	0.6646	**Beta10**	0.6550	0.5063	0.5827
	Data set 1 - 3				**Data set 1 - 4**		
	RR	**LASSO**	**EN**		**RR**	**LASSO**	**EN**
Intercept	1.3529	1.7911	1.5675	**Intercept**	1.3678	1.7547	1.5627
Beta1	0.2393	.	0.0713	**Beta1**	0.1990	.	0.0414
Beta2	0.6971	0.7735	0.7835	**Beta2**	0.7569	0.9722	0.8614
Beta3	0.5677	0.5600	0.5851	**Beta3**	0.4945	0.4097	0.5024
Beta4	0.4058	0.5631	0.4402	**Beta4**	0.3624	0.2737	0.3444
Beta5	0.3816	0.2651	0.3246	**Beta5**	0.4130	0.3141	0.3634
Beta6	0.8316	0.7096	0.7742	**Beta6**	0.8333	0.7322	0.7838
Beta7	0.2632	0.0130	0.1610	**Beta7**	0.3115	0.2092	0.2447
Beta8	0.6675	0.5337	0.6016	**Beta8**	0.6570	0.5593	0.6069
Beta9	0.6878	0.5620	0.6258	**Beta9**	0.6719	0.5458	0.6062
Beta10	0.6970	0.5396	0.6198	**Beta10**	0.6875	0.5375	0.6133
	Data set 1 - 5				**Data set 1 - 6**		
	RR	**LASSO**	**EN**		**RR**	**LASSO**	**EN**
Intercept	1.4809	1.8744	1.6731	**Intercept**	1.3163	1.7129	1.5172
Beta1	0.3059	.	0.1954	**Beta1**	0.1931	.	0.0321
Beta2	0.6662	0.9686	0.7456	**Beta2**	0.7256	0.9803	0.8367
Beta3	0.4890	0.4759	0.4770	**Beta3**	0.5735	0.4817	0.5864
Beta4	0.3192	0.1806	0.3005	**Beta4**	0.3290	0.1651	0.2967
Beta5	0.3975	0.2985	0.3485	**Beta5**	0.4736	0.3768	0.4233
Beta6	0.8057	0.6956	0.7501	**Beta6**	0.8321	0.7251	0.7797
Beta7	0.2914	0.1947	0.2301	**Beta7**	0.3284	0.2736	0.2726
Beta8	0.6592	0.5478	0.6070	**Beta8**	0.6930	0.5757	0.6351
Beta9	0.6248	0.4905	0.5602	**Beta9**	0.6458	0.5228	0.5802
Beta10	0.6474	0.5252	0.5877	**Beta10**	0.7165	0.5777	0.6476

Table A.1 (continued)

Data set 1 - 7				Data set 1 - 8			
	RR	LASSO	EN		RR	LASSO	EN
Intercept	1.3504	1.7597	1.5572	**Intercept**	1.2919	1.6926	1.4950
Beta1	0.1417	.	.	**Beta1**	0.1680	.	.
Beta2	0.6944	0.9839	0.8013	**Beta2**	0.7251	0.8640	0.8198
Beta3	0.6211	0.4786	0.6217	**Beta3**	0.5702	0.5035	0.5949
Beta4	0.3679	0.1232	0.3251	**Beta4**	0.4223	0.3817	0.4136
Beta5	0.4608	0.3774	0.4168	**Beta5**	0.4566	0.3488	0.4007
Beta6	0.8290	0.7030	0.7694	**Beta6**	0.8806	0.7482	0.8168
Beta7	0.3777	0.3630	0.3279	**Beta7**	0.3061	0.1895	0.2343
Beta8	0.6622	0.5258	0.5951	**Beta8**	0.7079	0.6024	0.6543
Beta9	0.6265	0.4997	0.5569	**Beta9**	0.6818	0.5569	0.6158
Beta10	0.7053	0.5594	0.6335	**Beta10**	0.6643	0.5468	0.6066
Data set 1 - 9				**Data set 1 - 10**			
	RR	LASSO	EN		RR	LASSO	EN
Intercept	1.2895	1.6908	1.4975	**Intercept**	1.3826	1.8473	1.6093
Beta1	0.1951	.	0.0096	**Beta1**	0.0883	.	.
Beta2	0.7592	1.0014	0.8767	**Beta2**	0.8077	0.9020	0.8757
Beta3	0.5359	0.4620	0.5712	**Beta3**	0.6203	0.4831	0.5842
Beta4	0.3735	0.1620	0.3207	**Beta4**	0.3731	0.4329	0.3871
Beta5	0.4883	0.3987	0.4406	**Beta5**	0.4361	0.3498	0.3920
Beta6	0.8439	0.7181	0.7833	**Beta6**	0.8370	0.6920	0.7691
Beta7	0.3674	0.3479	0.3208	**Beta7**	0.2816	0.0936	0.1961
Beta8	0.6685	0.5724	0.6175	**Beta8**	0.6327	0.4873	0.5606
Beta9	0.6482	0.5263	0.5825	**Beta9**	0.6463	0.4884	0.5692
Beta10	0.7093	0.5556	0.6298	**Beta10**	0.6731	0.5102	0.5949

Table A.2 Estimated Coefficients of Data Set 2

Data set 2 - 1				Data set 2 - 2			
	RR	LASSO	EN		RR	LASSO	EN
Intercept	0.9599	2.8286	2.2339	**Intercept**	1.0590	2.8512	2.2585
Beta1	0.3850	0.6322	0.5280	**Beta1**	0.5401	1.1850	0.8018
Beta2	0.8224	0.7519	0.7881	**Beta2**	0.8437	0.7852	0.8165
Beta3	0.3726	0.0901	0.2801	**Beta3**	0.3736	.	0.2417
Beta4	0.6556	0.5411	0.6015	**Beta4**	0.5164	0.1170	0.3877
Beta5	0.1258	.	.	**Beta5**	0.0934	.	.
Beta6	0.3157	0.2097	0.2287	**Beta6**	0.3502	0.3015	0.2771
Beta7	-0.0691	.	.	**Beta7**	-0.1195	.	.
Beta8	0.3741	0.0072	0.1647	**Beta8**	0.4049	0.0521	0.2171
Beta9	0.6473	0.5612	0.5967	**Beta9**	0.6207	0.4106	0.5170
Beta10	0.0752	.	.	**Beta10**	0.0730	.	.
Beta11	0.6170	0.9643	0.7813	**Beta11**	0.5650	0.5979	0.6217
Beta12	0.6205	0.2839	0.4850	**Beta12**	0.6338	0.2820	0.4753

Table A.2 (continued)

Beta13	0.0380	.	.	**Beta13**	0.0103	.	.
Beta14	0.2169	.	0.0098	**Beta14**	0.2700	.	0.0574
Beta15	0.8135	0.4622	0.6193	**Beta15**	0.7807	0.5182	0.6185
Beta16	0.3595	0.0948	0.2857	**Beta16**	0.3607	0.5322	0.3821
Beta17	0.6656	0.3733	0.4700	**Beta17**	0.6549	0.3658	0.4575
Beta18	0.2283	.	0.0088	**Beta18**	0.1712	.	.
Beta19	0.2057	0.1614	0.2314	**Beta19**	0.1386	0.0205	0.1142
Beta20	1.0311	0.7538	0.8862	**Beta20**	0.9248	0.7255	0.8394
Beta21	0.4436	.	0.0186	**Beta21**	0.4157	.	0.0203
Beta22	0.4336	0.6140	0.4751	**Beta22**	0.3715	0.0709	0.3146
Beta23	0.1764	.	.	**Beta23**	0.2445	.	0.0334
Beta24	0.5211	0.1520	0.2893	**Beta24**	0.6980	0.6957	0.5715
Beta25	0.3016	.	0.0841	**Beta25**	0.2467	.	0.0463

	Data set 2 - 3				**Data set 2 - 4**		
	RR	**LASSO**	**EN**		**RR**	**LASSO**	**EN**
Intercept	0.8500	2.8487	2.1772	**Intercept**	0.9729	2.8830	2.2753
Beta1	0.4482	0.6109	0.5633	**Beta1**	0.4605	0.8158	0.6047
Beta2	0.8390	0.7857	0.8271	**Beta2**	0.9030	0.8187	0.8519
Beta3	0.3413	0.0031	0.2032	**Beta3**	0.3308	0.0038	0.2361
Beta4	0.5713	0.4795	0.5100	**Beta4**	0.5796	0.3645	0.5005
Beta5	0.1151	.	.	**Beta5**	0.1108	.	.
Beta6	0.4143	0.3124	0.3372	**Beta6**	0.3531	0.3167	0.2953
Beta7	-0.1073	.	.	**Beta7**	-0.0041	.	.
Beta8	0.3875	0.0163	0.1936	**Beta8**	0.4131	0.0156	0.2183
Beta9	0.7308	0.6521	0.6466	**Beta9**	0.6341	0.4713	0.5462
Beta10	0.0239	.	.	**Beta10**	0.0687	.	.
Beta11	0.5870	1.0825	0.8184	**Beta11**	0.6026	0.9001	0.7638
Beta12	0.6420	0.2948	0.4895	**Beta12**	0.6078	0.2332	0.4447
Beta13	0.0493	.	.	**Beta13**	0.0331	.	.
Beta14	0.2902	.	0.0598	**Beta14**	0.2817	.	0.0729
Beta15	0.7702	0.3741	0.5546	**Beta15**	0.6934	0.3239	0.4697
Beta16	0.3825	0.3980	0.3839	**Beta16**	0.5192	0.6240	0.5100
Beta17	0.6669	0.3617	0.4743	**Beta17**	0.6682	0.3609	0.4663
Beta18	0.2561	.	0.0123	**Beta18**	0.1947	.	.
Beta19	0.1809	0.1216	0.1958	**Beta19**	0.1881	0.1079	0.1932
Beta20	0.9088	0.4392	0.7201	**Beta20**	0.9389	0.5285	0.7508
Beta21	0.5438	.	0.0756	**Beta21**	0.3970	.	0.0054
Beta22	0.4166	0.1952	0.3507	**Beta22**	0.3807	0.1789	0.3805
Beta23	0.2020	.	0.0070	**Beta23**	0.1739	.	.
Beta24	0.6244	0.4369	0.4244	**Beta24**	0.5149	0.4146	0.3729
Beta25	0.2717	.	0.0595	**Beta25**	0.2619	.	0.0065

Table A.2 (continued)

	Data set 2 - 5				Data set 2 - 6		
	RR	**LASSO**	**EN**		**RR**	**LASSO**	**EN**
Intercept	1.0568	2.9504	2.3500	**Intercept**	0.8679	2.8810	2.2798
Beta1	0.4962	1.1941	0.7475	**Beta1**	0.3832	0.5453	0.4622
Beta2	0.8487	0.7659	0.8173	**Beta2**	0.8503	0.7760	0.8074
Beta3	0.3205	0.0132	0.2470	**Beta3**	0.3513	0.1056	0.3163
Beta4	0.5268	0.0412	0.3831	**Beta4**	0.6030	0.5586	0.5857
Beta5	0.1787	.	0.0412	**Beta5**	0.1518	.	.
Beta6	0.2638	0.2573	0.2044	**Beta6**	0.3763	0.1870	0.2124
Beta7	-0.0516	.	.	**Beta7**	-0.0637	.	.
Beta8	0.4343	0.0378	0.2363	**Beta8**	0.4357	0.0039	0.1494
Beta9	0.6982	0.5360	0.6468	**Beta9**	0.7179	0.6073	0.6430
Beta10	0.1019	.	.	**Beta10**	0.0615	.	.
Beta11	0.6069	0.8496	0.7572	**Beta11**	0.6681	1.0138	0.8525
Beta12	0.6452	0.2949	0.5022	**Beta12**	0.6040	0.2304	0.4391
Beta13	0.0242	.	.	**Beta13**	-0.0233	.	.
Beta14	0.2359	.	0.0120	**Beta14**	0.2837	.	0.0729
Beta15	0.7660	0.3963	0.5434	**Beta15**	0.7952	0.3892	0.5467
Beta16	0.3937	0.5091	0.3813	**Beta16**	0.3827	0.3690	0.3495
Beta17	0.5872	0.2705	0.3729	**Beta17**	0.6781	0.4067	0.5033
Beta18	0.1711	.	.	**Beta18**	0.1649	.	.
Beta19	0.2134	0.1840	0.2537	**Beta19**	0.1531	0.1373	0.2014
Beta20	0.8842	0.4828	0.6718	**Beta20**	0.9067	0.4700	0.6889
Beta21	0.4302	.	0.0178	**Beta21**	0.4838	.	0.0310
Beta22	0.3729	0.1909	0.3709	**Beta22**	0.3611	0.2737	0.3661
Beta23	0.1954	.	.	**Beta23**	0.2276	.	.
Beta24	0.5723	0.3816	0.3750	**Beta24**	0.6669	0.4514	0.4785
Beta25	0.2439	.	0.0007	**Beta25**	0.3316	.	0.0224

	Data set 2 - 7				Data set 2 - 8		
	RR	**LASSO**	**EN**		**RR**	**LASSO**	**EN**
Intercept	0.8199	2.9032	2.2063	**Intercept**	0.9651	2.7618	2.1634
Beta1	0.4550	0.8017	0.6212	**Beta1**	0.4558	0.5541	0.4857
Beta2	0.8963	0.8254	0.8754	**Beta2**	0.9331	0.8569	0.9001
Beta3	0.3598	.	0.2082	**Beta3**	0.3280	0.0129	0.2018
Beta4	0.5127	0.4203	0.5189	**Beta4**	0.6261	0.5508	0.5680
Beta5	0.1227	.	.	**Beta5**	0.1098	.	.
Beta6	0.3594	0.2018	0.2527	**Beta6**	0.3083	0.1982	0.2011
Beta7	-0.1298	.	.	**Beta7**	-0.0720	.	.
Beta8	0.3820	.	0.1429	**Beta8**	0.2785	.	0.0675
Beta9	0.6760	0.5888	0.6099	**Beta9**	0.6848	0.6254	0.6426
Beta10	0.1142	.	.	**Beta10**	-0.0370	.	.
Beta11	0.6713	1.0206	0.8679	**Beta11**	0.5490	0.7513	0.6552
Beta12	0.5544	0.2236	0.4313	**Beta12**	0.7101	0.3831	0.5783
Beta13	0.0171	.	.	**Beta13**	-0.0357	.	.

Table A.2 (continued)

Beta14	0.3489	0.0183	0.1405	**Beta14**	0.3637	0.0082	0.1487
Beta15	0.8289	0.4086	0.5659	**Beta15**	0.7645	0.4497	0.5731
Beta16	0.3835	0.0985	0.3117	**Beta16**	0.4446	0.4910	0.4549
Beta17	0.6877	0.3303	0.4689	**Beta17**	0.6426	0.3577	0.4758
Beta18	0.1944	.	.	**Beta18**	0.2847	.	0.0679
Beta19	0.1843	0.1031	0.1902	**Beta19**	0.1687	0.1744	0.2135
Beta20	0.8834	0.6873	0.7691	**Beta20**	0.9444	0.6288	0.8059
Beta21	0.4669	.	0.0264	**Beta21**	0.4186	.	0.0176
Beta22	0.3748	0.4760	0.3745	**Beta22**	0.3688	0.2285	0.3295
Beta23	0.3018	.	0.0466	**Beta23**	0.1378	.	.
Beta24	0.6103	0.2416	0.3524	**Beta24**	0.5850	0.4309	0.4150
Beta25	0.3358	.	0.0512	**Beta25**	0.3052	.	0.0753
	Data set 2 - 9				**Data set 2 - 10**		
	RR	**LASSO**	**EN**		**RR**	**LASSO**	**EN**
Intercept	1.0156	3.0362	2.3514	**Intercept**	0.9027	2.8301	2.1632
Beta1	0.5002	0.9555	0.7321	**Beta1**	0.4593	0.5882	0.4871
Beta2	0.8273	0.7355	0.7767	**Beta2**	0.8437	0.8928	0.8938
Beta3	0.3167	.	0.1402	**Beta3**	0.3419	0.0591	0.3084
Beta4	0.5169	0.1329	0.3484	**Beta4**	0.5582	0.4394	0.4702
Beta5	0.1012	.	.	**Beta5**	0.2080	.	.
Beta6	0.3942	0.2314	0.3103	**Beta6**	0.3267	0.3585	0.3478
Beta7	-0.0402	.	.	**Beta7**	-0.1075	.	.
Beta8	0.4136	0.0022	0.1621	**Beta8**	0.4668	0.0459	0.2447
Beta9	0.7509	0.6617	0.6776	**Beta9**	0.7092	0.5427	0.5867
Beta10	0.0329	.	.	**Beta10**	-0.0155	.	.
Beta11	0.7317	1.4871	1.0051	**Beta11**	0.6112	1.0129	0.8180
Beta12	0.6276	0.2115	0.4530	**Beta12**	0.5911	0.2642	0.4503
Beta13	-0.0421	.	.	**Beta13**	0.0644	.	.
Beta14	0.1970	.	0.0033	**Beta14**	0.3202	.	0.1226
Beta15	0.7858	0.3762	0.5488	**Beta15**	0.7681	0.4086	0.5597
Beta16	0.4600	0.5074	0.4613	**Beta16**	0.3956	0.4060	0.3652
Beta17	0.5868	0.2956	0.3913	**Beta17**	0.6835	0.3868	0.4929
Beta18	0.2016	.	0.0015	**Beta18**	0.1884	.	.
Beta19	0.1904	0.0472	0.1813	**Beta19**	0.1792	0.1060	0.1840
Beta20	0.8681	0.0800	0.5838	**Beta20**	0.9406	0.4929	0.7494
Beta21	0.4775	.	0.0669	**Beta21**	0.3821	.	.
Beta22	0.3575	0.1290	0.3213	**Beta22**	0.4009	0.1456	0.3270
Beta23	0.1848	.	.	**Beta23**	0.2244	.	.
Beta24	0.5180	0.3294	0.3408	**Beta24**	0.6022	0.4633	0.4263
Beta25	0.2626	.	0.0398	**Beta25**	0.3127	.	0.0924

Table A.3 Estimated Coefficients of Data Set 3

	Data set 3 - 1				Data set 3 - 2		
	RR	**LASSO**	**EN**		**RR**	**LASSO**	**EN**
Intercept	1.5460	9.5446	6.7379	**Intercept**	1.4855	9.1812	6.5698
Beta1	0.7286	1.3325	1.0890	**Beta1**	0.7326	1.3150	1.0953
Beta2	0.8381	0.3893	0.8570	**Beta2**	0.8305	0.2762	0.7138
Beta3	0.5520	.	0.2622	**Beta3**	0.5891	.	0.2423
Beta4	0.1989	.	.	**Beta4**	0.1877	.	.
Beta5	0.4582	.	.	**Beta5**	0.5474	.	0.0413
Beta6	0.6704	.	0.0146	**Beta6**	0.5314	.	0.1678
Beta7	0.6107	1.1983	0.7240	**Beta7**	0.7405	1.1395	0.7679
Beta8	0.3646	.	.	**Beta8**	0.3878	.	.
Beta9	0.2746	.	0.0161	**Beta9**	0.2896	0.3000	0.3663
Beta10	0.4810	0.0780	0.3092	**Beta10**	0.5821	0.0538	0.3034
Beta11	0.7915	0.0518	0.3037	**Beta11**	0.7616	0.2261	0.4344
Beta12	0.3069	.	.	**Beta12**	0.4082	.	.
Beta13	0.9396	0.2896	0.6427	**Beta13**	0.9562	0.2251	0.5995
Beta14	0.3240	.	0.2085	**Beta14**	0.3894	.	0.1412
Beta15	0.5297	.	0.3066	**Beta15**	0.4433	.	.
Beta16	0.3270	.	0.2944	**Beta16**	0.4675	0.7420	0.6374
Beta17	0.9218	0.5675	0.7911	**Beta17**	0.9307	0.5684	0.7704
Beta18	0.5201	0.1594	0.4702	**Beta18**	0.3715	0.0980	0.2977
Beta19	0.6080	0.5773	0.3221	**Beta19**	0.6642	0.8022	0.6082
Beta20	0.4314	.	0.2128	**Beta20**	0.5051	.	0.3601
Beta21	0.3713	.	.	**Beta21**	0.3901	.	0.0028
Beta22	0.7225	0.2205	0.6230	**Beta22**	0.6993	.	0.3018
Beta23	-0.1449	.	.	**Beta23**	-0.2721	-0.0373	-0.1635
Beta24	0.7747	0.8191	0.7683	**Beta24**	0.7692	1.1175	1.0494
Beta25	0.6794	0.6620	0.5726	**Beta25**	0.5685	0.2935	0.4307
Beta26	1.0143	0.3743	0.7061	**Beta26**	0.9636	0.4754	0.7451
Beta27	0.6946	0.2638	0.4051	**Beta27**	0.7242	0.2608	0.4512
Beta28	0.5808	.	0.2191	**Beta28**	0.4092	.	0.0162
Beta29	0.2696	.	.	**Beta29**	0.2989	.	0.0253
Beta30	0.8684	1.7523	1.3996	**Beta30**	0.7117	0.9406	0.8960
Beta31	0.7312	0.0712	0.4567	**Beta31**	0.7252	0.0502	0.3900
Beta32	0.5898	1.5127	0.8924	**Beta32**	0.5345	1.1383	0.7034
Beta33	0.2692	.	0.0867	**Beta33**	0.1805	.	0.1358
Beta34	0.5309	.	.	**Beta34**	0.5024	.	.
Beta35	0.2367	0.0543	0.1071	**Beta35**	0.3626	0.0049	0.1702
Beta36	0.6349	0.0407	0.3725	**Beta36**	0.6293	0.1810	0.4748
Beta37	0.8512	.	0.7931	**Beta37**	0.8039	.	0.7893
Beta38	0.1601	.	.	**Beta38**	0.3024	.	.
Beta39	0.7044	1.1063	0.6570	**Beta39**	0.7794	0.9589	0.6692
Beta40	0.5590	0.0513	0.4990	**Beta40**	0.6354	0.3563	0.6433
Beta41	0.8607	0.1830	0.4737	**Beta41**	0.7451	0.0217	0.3990
Beta42	0.5149	0.5904	0.5632	**Beta42**	0.4899	0.9097	0.6486

Table A.3 (continued)

Beta43	0.3826	.	.	**Beta43**	0.4973	.	0.1501
Beta44	0.9054	0.2996	0.6249	**Beta44**	0.8645	0.4476	0.7955
Beta45	0.7719	1.1925	1.2499	**Beta45**	0.6885	1.3232	1.1349
Beta46	0.2818	.	.	**Beta46**	0.1726	.	.
Beta47	0.7391	0.4332	0.6535	**Beta47**	0.8515	0.1397	0.4418
Beta48	0.2307	.	.	**Beta48**	0.4086	.	.
Beta49	0.5012	.	0.0428	**Beta49**	0.3974	.	0.1139
Beta50	0.0393	.	.	**Beta50**	0.0852	.	.
Beta51	0.1740	.	.	**Beta51**	0.2307	.	.
Beta52	0.8561	0.0877	0.3905	**Beta52**	0.7794	0.4233	0.5944
Beta53	0.1557	.	.	**Beta53**	0.2293	.	.
Beta54	1.0238	0.3896	0.9531	**Beta54**	0.9595	0.6529	0.9539
Beta55	0.0820	.	.	**Beta55**	0.0904	.	.
Beta56	0.6217	.	0.0658	**Beta56**	0.5974	.	0.0354
Beta57	0.3616	.	.	**Beta57**	0.2813	.	.
Beta58	-0.1429	.	.	**Beta58**	-0.0207	.	.
Beta59	0.1108	.	.	**Beta59**	0.1384	.	0.1177
Beta60	0.4744	.	.	**Beta60**	0.4471	.	.

	Data set 3 - 3				**Data set 3 - 4**		
	RR	**LASSO**	**EN**		**RR**	**LASSO**	**EN**
Intercept	1.2316	8.1976	5.7724	**Intercept**	1.4870	8.9241	6.2962
Beta1	0.6895	1.6396	1.5307	**Beta1**	0.7790	1.2016	1.0089
Beta2	0.8742	0.8872	0.9215	**Beta2**	0.7527	0.1964	0.7242
Beta3	0.6199	0.1473	0.3864	**Beta3**	0.5166	.	0.2069
Beta4	0.2128	.	.	**Beta4**	0.2163	.	.
Beta5	0.6823	.	0.0800	**Beta5**	0.5438	.	0.1781
Beta6	0.7341	.	0.1113	**Beta6**	0.6686	.	0.2352
Beta7	0.6146	0.3656	0.4385	**Beta7**	0.6066	1.3019	0.7271
Beta8	0.3442	.	.	**Beta8**	0.3899	.	.
Beta9	0.2774	0.4823	0.4490	**Beta9**	0.4099	0.2152	0.4291
Beta10	0.5253	0.1804	0.3710	**Beta10**	0.4865	0.1683	0.2826
Beta11	0.7974	0.2037	0.3749	**Beta11**	0.8324	0.2820	0.4261
Beta12	0.3093	.	.	**Beta12**	0.3240	.	.
Beta13	0.8856	0.2192	0.4867	**Beta13**	0.8629	0.2424	0.5563
Beta14	0.5068	.	0.0209	**Beta14**	0.2618	.	.
Beta15	0.3477	.	0.0398	**Beta15**	0.5259	.	0.2242
Beta16	0.0741	.	0.0418	**Beta16**	0.2849	.	0.2042
Beta17	0.9641	0.5554	0.7087	**Beta17**	0.8697	0.4642	0.6756
Beta18	0.5898	0.3545	0.5492	**Beta18**	0.5860	0.1865	0.5088
Beta19	0.6377	1.2243	0.6867	**Beta19**	0.6271	0.7236	0.5264
Beta20	0.5315	0.0824	0.4118	**Beta20**	0.4395	.	0.3433
Beta21	0.4258	.	0.1710	**Beta21**	0.3685	.	.
Beta22	0.7648	.	0.5733	**Beta22**	0.6894	.	0.2892
Beta23	-0.0998	-0.2374	-0.1830	**Beta23**	-0.1268	-0.0036	-0.0752

Table A.3 (continued)

Beta24	0.6447	0.6312	0.5289	**Beta24**	0.8416	0.9194	0.9417
Beta25	0.9824	0.5806	0.8043	**Beta25**	0.6731	0.5336	0.5927
Beta26	1.0455	0.6625	0.8787	**Beta26**	1.0161	0.3127	0.6074
Beta27	0.6937	0.0771	0.3873	**Beta27**	0.7005	0.5393	0.6387
Beta28	0.5489	.	0.1740	**Beta28**	0.4728	.	0.1445
Beta29	0.2419	.	.	**Beta29**	0.2328	.	.
Beta30	0.7904	1.2741	1.0682	**Beta30**	0.8386	1.7473	1.3994
Beta31	0.8567	0.5748	0.7690	**Beta31**	0.8305	0.2669	0.6303
Beta32	0.6285	1.5846	0.9285	**Beta32**	0.6772	1.1570	0.9132
Beta33	0.1052	.	0.0797	**Beta33**	0.2237	.	0.0521
Beta34	0.5248	.	.	**Beta34**	0.5682	.	0.0038
Beta35	0.3762	.	0.0908	**Beta35**	0.3315	.	0.0901
Beta36	0.6290	0.2329	0.3779	**Beta36**	0.5745	.	0.2944
Beta37	0.7350	.	0.4453	**Beta37**	0.8222	0.1130	0.6743
Beta38	0.0068	.	.	**Beta38**	0.1815	.	.
Beta39	0.7159	0.5651	0.6070	**Beta39**	0.6704	0.7581	0.4998
Beta40	0.5665	0.6748	0.6658	**Beta40**	0.6386	0.3127	0.7984
Beta41	0.9433	0.3676	0.6891	**Beta41**	0.7266	0.1058	0.3056
Beta42	0.5408	0.4856	0.5536	**Beta42**	0.5599	1.0205	0.6494
Beta43	0.3988	0.0224	0.1771	**Beta43**	0.3901	.	.
Beta44	0.7814	0.3607	0.7056	**Beta44**	0.8269	1.1022	0.9924
Beta45	0.8967	1.5992	1.2097	**Beta45**	0.7597	0.9779	1.0036
Beta46	0.1636	.	.	**Beta46**	0.3257	.	.
Beta47	0.6604	0.3200	0.5060	**Beta47**	0.6140	.	0.2911
Beta48	0.1722	.	.	**Beta48**	0.3076	.	0.0719
Beta49	0.5360	0.0087	0.2701	**Beta49**	0.5310	.	0.2421
Beta50	0.1556	.	.	**Beta50**	0.0505	.	.
Beta51	0.1543	.	.	**Beta51**	0.1616	.	.
Beta52	0.8875	0.5079	0.6838	**Beta52**	0.8566	0.4939	0.7362
Beta53	0.2561	.	.	**Beta53**	0.1564	.	.
Beta54	0.8729	0.3716	0.6994	**Beta54**	0.9838	0.5160	0.9210
Beta55	-0.0628	.	.	**Beta55**	-0.0867	.	.
Beta56	0.6717	.	0.3358	**Beta56**	0.6382	.	0.1278
Beta57	0.5123	.	0.0257	**Beta57**	0.3287	.	.
Beta58	-0.1357	0.0036	0.0004	**Beta58**	-0.0868	.	.
Beta59	0.3290	0.5467	0.5970	**Beta59**	0.2758	0.1991	0.1927
Beta60	0.4008	.	.	**Beta60**	0.4788	.	.
	Data set 3 - 5				**Data set 3 - 6**		
	RR	**LASSO**	**EN**		**RR**	**LASSO**	**EN**
Intercept	1.8591	8.8390	6.5065	**Intercept**	1.5550	8.5109	6.0779
Beta1	0.7423	1.3702	1.2493	**Beta1**	0.7197	1.4836	1.0757
Beta2	0.8810	0.9693	1.0173	**Beta2**	0.8587	0.5263	0.8364
Beta3	0.4454	.	0.0676	**Beta3**	0.7345	0.2803	0.5333
Beta4	0.0799	.	.	**Beta4**	0.1950	.	.

Table A.3 (continued)

Beta5	0.4452	.	0.0098	**Beta5**	0.5836	.	0.2371
Beta6	0.7444	.	0.2671	**Beta6**	0.7052	0.0066	0.3219
Beta7	0.5493	0.5400	0.4207	**Beta7**	0.5817	0.9884	0.7727
Beta8	0.2780	.	.	**Beta8**	0.4145	.	.
Beta9	0.3396	0.3148	0.3217	**Beta9**	0.2850	.	0.0105
Beta10	0.5477	0.1015	0.3915	**Beta10**	0.4812	0.1566	0.3048
Beta11	0.8486	0.1744	0.4310	**Beta11**	0.6844	0.0232	0.2096
Beta12	0.3296	.	.	**Beta12**	0.2928	.	.
Beta13	1.0067	0.3610	0.7756	**Beta13**	0.8513	0.1814	0.4978
Beta14	0.3933	.	0.1090	**Beta14**	0.4388	.	0.2335
Beta15	0.3378	.	.	**Beta15**	0.4579	.	0.1162
Beta16	0.3596	0.1752	0.3888	**Beta16**	0.3199	.	0.2461
Beta17	0.8250	0.5322	0.6646	**Beta17**	0.8185	0.3854	0.5541
Beta18	0.5322	0.0948	0.3961	**Beta18**	0.5739	0.4985	0.6471
Beta19	0.6670	1.4139	0.7674	**Beta19**	0.6453	1.0922	0.4963
Beta20	0.5543	0.4044	0.6086	**Beta20**	0.4878	0.0729	0.4732
Beta21	0.3492	.	.	**Beta21**	0.4002	.	.
Beta22	0.7393	.	0.5033	**Beta22**	0.7254	.	0.5726
Beta23	-0.1057	.	-0.0463	**Beta23**	-0.1233	.	-0.0344
Beta24	0.8366	0.7385	0.7667	**Beta24**	0.9462	1.0557	0.9348
Beta25	0.6604	0.5966	0.6436	**Beta25**	0.4524	0.0211	0.2605
Beta26	0.8873	0.2360	0.5099	**Beta26**	0.9924	0.6876	0.8231
Beta27	0.6373	0.2826	0.4509	**Beta27**	0.6306	0.4560	0.6563
Beta28	0.4500	.	0.0535	**Beta28**	0.5326	.	0.1087
Beta29	0.2149	.	.	**Beta29**	0.3274	.	0.0255
Beta30	0.6566	0.7881	0.7186	**Beta30**	0.9444	1.7481	1.2946
Beta31	0.7229	0.1498	0.4844	**Beta31**	0.6879	0.2547	0.5284
Beta32	0.5393	1.1507	0.6041	**Beta32**	0.4832	0.6020	0.4471
Beta33	0.2084	.	0.0402	**Beta33**	0.2475	.	0.0904
Beta34	0.4336	.	.	**Beta34**	0.5589	.	0.0535
Beta35	0.3417	0.0296	0.1851	**Beta35**	0.2479	.	.
Beta36	0.6987	0.0875	0.4638	**Beta36**	0.6043	0.0849	0.3214
Beta37	0.7957	0.3943	0.7715	**Beta37**	0.8740	.	0.8349
Beta38	0.2115	.	.	**Beta38**	0.1308	.	.
Beta39	0.7278	0.5319	0.5510	**Beta39**	0.7743	1.8495	1.2428
Beta40	0.7134	0.7828	0.9000	**Beta40**	0.6132	.	0.3699
Beta41	0.7940	0.3605	0.6287	**Beta41**	0.7526	0.1734	0.4111
Beta42	0.3536	0.1740	0.2185	**Beta42**	0.6214	0.9108	0.5888
Beta43	0.2618	.	.	**Beta43**	0.2564	.	.
Beta44	0.8152	0.4073	0.6948	**Beta44**	0.8505	0.6375	0.8286
Beta45	0.9613	1.7251	1.5396	**Beta45**	0.6806	1.3678	1.0043
Beta46	0.3889	.	0.0360	**Beta46**	0.3603	.	0.0675
Beta47	0.5675	0.1837	0.3406	**Beta47**	0.5093	0.0524	0.2735
Beta48	0.3022	.	.	**Beta48**	0.3218	.	.

Table A.3 (continued)

Beta49	0.4888	.	0.0786	**Beta49**	0.5541	0.0940	0.4424
Beta50	0.0330	.	.	**Beta50**	0.0278	.	.
Beta51	0.0968	.	.	**Beta51**	0.1301	.	.
Beta52	0.8284	0.3828	0.5736	**Beta52**	0.9448	0.3631	0.6517
Beta53	0.1798	.	.	**Beta53**	0.1861	.	.
Beta54	0.9222	0.5268	0.8658	**Beta54**	0.9513	0.6432	0.9696
Beta55	0.1074	0.1914	0.0079	**Beta55**	-0.0860	.	.
Beta56	0.5654	.	0.0450	**Beta56**	0.6705	.	0.2229
Beta57	0.2755	.	.	**Beta57**	0.4550	.	.
Beta58	-0.0642	.	.	**Beta58**	-0.0467	.	0.0822
Beta59	0.3382	0.0874	0.4368	**Beta59**	0.0791	.	.
Beta60	0.4967	.	.	**Beta60**	0.4390	.	.
	Data set 3 - 7				**Data set 3 - 8**		
	RR	**LASSO**	**EN**		**RR**	**LASSO**	**EN**
Intercept	1.2795	7.7198	5.4800	**Intercept**	1.3846	8.1326	6.1166
Beta1	0.7889	1.3710	1.0584	**Beta1**	0.7564	1.4691	1.2143
Beta2	0.8127	0.7568	0.9341	**Beta2**	0.7524	0.0254	0.5908
Beta3	0.5724	0.1320	0.3649	**Beta3**	0.5920	0.3491	0.5588
Beta4	0.1782	.	.	**Beta4**	0.1100	.	.
Beta5	0.5635	.	0.1153	**Beta5**	0.5810	.	.
Beta6	0.7212	0.0139	0.3373	**Beta6**	0.7407	.	.
Beta7	0.5395	0.7849	0.6221	**Beta7**	0.5692	1.3698	0.7899
Beta8	0.4064	.	.	**Beta8**	0.3936	.	.
Beta9	0.3954	0.2308	0.1640	**Beta9**	0.3211	.	0.0794
Beta10	0.4318	0.0134	0.1700	**Beta10**	0.4866	.	0.0339
Beta11	0.8756	0.3346	0.4965	**Beta11**	0.8074	.	0.0716
Beta12	0.3601	.	.	**Beta12**	0.2382	.	0.0331
Beta13	0.9595	0.4445	0.6811	**Beta13**	0.9701	0.4427	0.7066
Beta14	0.4331	.	0.2342	**Beta14**	0.3764	.	0.1940
Beta15	0.4649	.	0.1090	**Beta15**	0.5000	.	0.2894
Beta16	0.1121	.	.	**Beta16**	0.2937	0.1118	0.2916
Beta17	0.9139	0.7203	0.8293	**Beta17**	0.9993	0.3417	0.5746
Beta18	0.4715	0.3112	0.4517	**Beta18**	0.5404	.	0.1302
Beta19	0.7479	1.2150	0.6269	**Beta19**	0.6304	0.7424	0.3321
Beta20	0.5502	0.3761	0.7787	**Beta20**	0.4927	0.4301	0.6237
Beta21	0.4001	.	0.0560	**Beta21**	0.3543	.	.
Beta22	0.7233	0.1357	0.7641	**Beta22**	0.6692	.	0.4909
Beta23	-0.1235	-0.0531	-0.0870	**Beta23**	-0.1219	-0.0353	-0.0659
Beta24	0.8429	0.8915	0.7077	**Beta24**	0.8474	1.2621	1.0110
Beta25	0.7387	0.4967	0.6389	**Beta25**	0.6373	0.6186	0.5686
Beta26	0.9088	0.5618	0.7147	**Beta26**	1.0413	0.5430	0.7688
Beta27	0.6436	0.4816	0.6092	**Beta27**	0.5889	0.6903	0.7024
Beta28	0.5109	0.0137	0.2524	**Beta28**	0.4849	0.1969	0.4432
Beta29	0.2865	.	0.0221	**Beta29**	0.3427	.	.

Table A.3 (continued)

Beta30	0.7150	1.0486	0.8642	**Beta30**	0.8007	0.9961	0.8795
Beta31	0.7907	0.3028	0.5712	**Beta31**	0.7672	0.1674	0.4605
Beta32	0.3904	0.6803	0.3961	**Beta32**	0.5766	0.3426	0.4834
Beta33	0.1357	.	.	**Beta33**	0.2500	.	0.0455
Beta34	0.4876	.	.	**Beta34**	0.5715	.	.
Beta35	0.1912	0.0358	0.0878	**Beta35**	0.2770	0.3928	0.4411
Beta36	0.7287	0.2863	0.5849	**Beta36**	0.6386	0.0427	0.3327
Beta37	0.9637	0.6015	0.9517	**Beta37**	0.8977	1.0668	1.2374
Beta38	0.1836	.	.	**Beta38**	0.2117	.	.
Beta39	0.7866	0.5760	0.7334	**Beta39**	0.7040	.	0.0879
Beta40	0.4917	0.6982	0.6006	**Beta40**	0.5774	0.2134	0.4721
Beta41	0.8119	0.2174	0.5728	**Beta41**	0.7829	1.0783	1.0338
Beta42	0.6587	0.9703	0.7677	**Beta42**	0.5215	0.5642	0.3527
Beta43	0.3118	.	0.0275	**Beta43**	0.3492	.	0.0619
Beta44	0.9878	0.6035	0.8801	**Beta44**	0.8572	1.0206	0.9584
Beta45	0.7638	1.4310	1.1807	**Beta45**	0.7871	0.7114	0.9323
Beta46	0.3246	.	.	**Beta46**	0.2227	.	.
Beta47	0.7404	0.3539	0.5620	**Beta47**	0.6369	0.3738	0.6136
Beta48	0.1599	.	.	**Beta48**	0.2866	.	0.0123
Beta49	0.5947	.	0.1704	**Beta49**	0.5803	0.7400	0.7379
Beta50	0.0410	.	.	**Beta50**	0.0273	.	.
Beta51	0.1004	.	.	**Beta51**	0.1784	.	.
Beta52	0.9398	0.5614	0.8065	**Beta52**	0.8950	0.5424	0.7645
Beta53	0.1699	.	.	**Beta53**	0.3103	.	.
Beta54	1.0058	0.7047	0.9589	**Beta54**	0.9359	0.4991	0.8259
Beta55	0.0511	.	.	**Beta55**	0.0119	.	.
Beta56	0.6742	0.0321	0.3764	**Beta56**	0.6117	.	0.1471
Beta57	0.3244	.	.	**Beta57**	0.3847	.	.
Beta58	-0.0991	.	.	**Beta58**	-0.1790	.	.
Beta59	0.1663	0.0386	0.1300	**Beta59**	0.1814	0.1786	0.2454
Beta60	0.5508	.	.	**Beta60**	0.5675	.	.
	Data set 3 - 9				**Data set 3 - 10**		
	RR	**LASSO**	**EN**		**RR**	**LASSO**	**EN**
Intercept	1.6897	9.5116	7.1079	**Intercept**	1.5155	8.6074	6.0362
Beta1	0.7268	1.2934	1.2575	**Beta1**	0.7311	0.9058	1.1505
Beta2	0.8762	0.6547	0.8984	**Beta2**	0.9038	1.2440	1.0385
Beta3	0.5517	.	0.1651	**Beta3**	0.4712	.	0.1559
Beta4	0.1823	.	.	**Beta4**	0.1098	.	.
Beta5	0.4900	.	.	**Beta5**	0.5799	.	0.0558
Beta6	0.6846	0.0215	0.2828	**Beta6**	0.6207	.	.
Beta7	0.6108	1.1397	0.7444	**Beta7**	0.6470	0.8419	0.6432
Beta8	0.4106	.	.	**Beta8**	0.4329	.	.
Beta9	0.2960	.	0.1860	**Beta9**	0.2995	.	0.0175
Beta10	0.4639	0.0050	0.2094	**Beta10**	0.3665	0.0410	0.2302

Table A.3 (continued)

Beta11	0.7762	0.3623	0.5591	**Beta11**	0.8319	0.0692	0.3634
Beta12	0.3092	.	.	**Beta12**	0.3615	.	0.0733
Beta13	0.8947	0.2901	0.6331	**Beta13**	0.8991	0.3575	0.7132
Beta14	0.4188	.	0.0902	**Beta14**	0.3558	.	0.0898
Beta15	0.4403	.	0.0209	**Beta15**	0.5428	.	0.3050
Beta16	0.3303	.	0.2960	**Beta16**	0.3458	.	0.3348
Beta17	0.8480	0.4039	0.6066	**Beta17**	0.9165	0.4775	0.7468
Beta18	0.4941	.	0.2014	**Beta18**	0.4680	0.0882	0.3461
Beta19	0.5999	1.0876	0.6002	**Beta19**	0.5682	0.7440	0.4440
Beta20	0.4788	0.0072	0.3550	**Beta20**	0.4073	.	0.1908
Beta21	0.3796	.	.	**Beta21**	0.3574	.	.
Beta22	0.7062	.	0.3327	**Beta22**	0.7245	0.0139	0.4984
Beta23	-0.1229	.	.	**Beta23**	-0.0974	.	.
Beta24	0.7974	0.9269	0.8789	**Beta24**	0.6717	0.6455	0.6110
Beta25	0.7172	0.5615	0.6632	**Beta25**	0.5809	0.5856	0.4794
Beta26	0.9983	0.3999	0.7823	**Beta26**	0.9621	0.3424	0.6714
Beta27	0.6261	0.2239	0.3791	**Beta27**	0.5620	0.1359	0.2515
Beta28	0.4472	.	.	**Beta28**	0.6214	0.0811	0.3247
Beta29	0.2857	.	0.0003	**Beta29**	0.3538	0.0586	0.2087
Beta30	0.8267	1.4176	1.1862	**Beta30**	0.9777	1.7742	1.3811
Beta31	0.7196	0.0039	0.2493	**Beta31**	0.6858	0.5972	0.7443
Beta32	0.5779	0.7181	0.6686	**Beta32**	0.6263	0.8073	0.7973
Beta33	0.1914	.	.	**Beta33**	0.1778	.	0.0063
Beta34	0.5104	.	0.0017	**Beta34**	0.6434	.	0.1851
Beta35	0.2780	0.0006	0.0938	**Beta35**	0.2684	.	.
Beta36	0.5913	.	0.0133	**Beta36**	0.6986	0.1064	0.4656
Beta37	0.8123	.	0.4296	**Beta37**	0.8931	1.1482	1.1550
Beta38	0.2014	.	.	**Beta38**	0.1613	.	.
Beta39	0.7403	0.5181	0.6041	**Beta39**	0.5823	.	0.2469
Beta40	0.6674	0.6171	0.8492	**Beta40**	0.5031	0.6663	0.6735
Beta41	0.7177	.	0.2270	**Beta41**	0.7883	0.1692	0.4437
Beta42	0.5579	1.4501	1.0677	**Beta42**	0.5585	0.6500	0.4678
Beta43	0.3015	.	.	**Beta43**	0.3785	.	.
Beta44	0.8777	0.6607	0.8082	**Beta44**	0.7413	0.5993	0.8116
Beta45	0.8354	1.3155	1.3194	**Beta45**	0.8819	1.2718	1.3122
Beta46	0.2330	.	.	**Beta46**	0.2762	.	.
Beta47	0.6883	0.0241	0.4179	**Beta47**	0.6471	0.0813	0.4019
Beta48	0.2548	.	.	**Beta48**	0.3084	.	0.0066
Beta49	0.4948	.	0.1094	**Beta49**	0.4534	.	.
Beta50	0.0455	.	.	**Beta50**	0.0590	.	.
Beta51	0.1746	.	.	**Beta51**	0.2120	.	0.0326
Beta52	0.8945	0.3629	0.6012	**Beta52**	0.9102	0.8619	0.9209
Beta53	0.1911	.	.	**Beta53**	0.2263	.	.
Beta54	0.9157	0.3800	0.8045	**Beta54**	1.0896	0.6389	0.9988

Table A.3 (continued)

Beta55	0.0421	.	.	**Beta55**	0.0972	0.1586	.
Beta56	0.6041	.	0.0104	**Beta56**	0.5903	.	0.0733
Beta57	0.4164	.	.	**Beta57**	0.2383	.	.
Beta58	-0.0853	.	0.0002	**Beta58**	-0.0200	0.0120	0.1003
Beta59	0.1635	.	0.1031	**Beta59**	0.0609	0.3183	0.3437
Beta60	0.4391	.	.	**Beta60**	0.6122	.	0.1417

Table A.4 Estimated Coefficients of Data Set 4

	Data set 4 - 1				**Data set 4 - 2**		
	RR	**LASSO**	**EN**		**RR**	**LASSO**	**EN**
Intercept	0.7691	1.1551	0.9863	**Intercept**	0.7764	1.170837	0.9983
Beta1	0.6888	0.5529	0.6231	**Beta1**	0.7002	0.57206	0.6420
Beta2	0.0861	.	.	**Beta2**	0.0593	.	.
Beta3	0.0443	.	.	**Beta3**	0.0806	.	0.0035
Beta4	0.2181	0.0428	0.1545	**Beta4**	0.1526	.	0.0765
Beta5	0.3565	0.1894	0.2766	**Beta5**	0.3049	0.1311	0.2214
Beta6	0.8105	0.6743	0.7479	**Beta6**	0.7846	0.6495	0.7214
Beta7	0.4304	0.7443	0.4753	**Beta7**	0.4200	0.6239	0.4492
Beta8	0.5475	0.4767	0.5301	**Beta8**	0.5760	0.5262	0.5751
Beta9	0.4615	0.0702	0.3856	**Beta9**	0.5043	0.2110	0.4416
Beta10	0.8047	0.9100	0.8111	**Beta10**	0.8351	0.9109	0.8409
	Data set 4 - 3				**Data set 4 - 4**		
	RR	**LASSO**	**EN**		**RR**	**LASSO**	**EN**
Intercept	0.7462	1.1601	0.9754	**Intercept**	0.7323	1.1115	0.9429
Beta1	0.7010	0.5756	0.6433	**Beta1**	0.7157	0.5893	0.6581
Beta2	0.1040	.	0.0198	**Beta2**	0.0460	.	.
Beta3	0.0962	.	0.0179	**Beta3**	0.0844	.	0.0068
Beta4	0.1731	.	0.0867	**Beta4**	0.1597	.	0.0807
Beta5	0.3161	0.1489	0.2367	**Beta5**	0.3433	0.1783	0.2647
Beta6	0.8162	0.6851	0.7566	**Beta6**	0.8353	0.7014	0.7737
Beta7	0.4094	0.6356	0.4449	**Beta7**	0.4344	0.6487	0.4665
Beta8	0.5781	0.5371	0.5834	**Beta8**	0.6019	0.5558	0.6051
Beta9	0.4805	0.1732	0.4179	**Beta9**	0.4887	0.1905	0.4246
Beta10	0.8048	0.8859	0.8089	**Beta10**	0.7803	0.8593	0.7849
	Data set 4 - 5				**Data set 4 - 6**		
	RR	**LASSO**	**EN**		**RR**	**LASSO**	**EN**
Intercept	0.7406	1.1415	0.9682	**Intercept**	0.7821	1.1803	1.0009
Beta1	0.7012	0.5690	0.6397	**Beta1**	0.6874	0.5663	0.6299
Beta2	0.0767	.	.	**Beta2**	0.0717	.	0.0012
Beta3	0.0689	.	.	**Beta3**	0.0978	.	0.0267
Beta4	0.1935	.	0.1138	**Beta4**	0.1906	0.0003	0.1227
Beta5	0.3306	0.1582	0.2487	**Beta5**	0.2960	0.1291	0.2166
Beta6	0.8302	0.6936	0.7676	**Beta6**	0.8086	0.6841	0.7523

Table A.4 (continued)

Beta7	0.4320	0.6112	0.4539	**Beta7**	0.4485	0.7889	0.4956
Beta8	0.5714	0.5797	0.5842	**Beta8**	0.5799	0.5143	0.5664
Beta9	0.5088	0.2582	0.4520	**Beta9**	0.4385	0.0100	0.3529
Beta10	0.7622	0.7973	0.7564	**Beta10**	0.7904	0.9139	0.8039
	Data set 4 - 7				**Data set 4 - 8**		
	RR	**LASSO**	**EN**		**RR**	**LASSO**	**EN**
Intercept	0.7325	1.1249	0.9538	**Intercept**	0.7504	1.1678	0.9834
Beta1	0.6981	0.5715	0.6400	**Beta1**	0.6775	0.5483	0.6189
Beta2	0.0672	.	.	**Beta2**	0.1003	.	0.0194
Beta3	0.0721	.	.	**Beta3**	0.0827	.	0.0042
Beta4	0.2205	0.0377	0.1526	**Beta4**	0.1802	.	0.0937
Beta5	0.3371	0.1747	0.2594	**Beta5**	0.3019	0.1315	0.2204
Beta6	0.8364	0.6985	0.7723	**Beta6**	0.8144	0.6751	0.7501
Beta7	0.3975	0.5487	0.4053	**Beta7**	0.4310	0.6387	0.4643
Beta8	0.5734	0.5629	0.5743	**Beta8**	0.5828	0.5408	0.5832
Beta9	0.5217	0.2905	0.4745	**Beta9**	0.5128	0.2341	0.4534
Beta10	0.7895	0.8440	0.7913	**Beta10**	0.7819	1.1678	0.7867
	Data set 4 - 9				**Data set 4 - 10**		
	RR	**LASSO**	**EN**		**RR**	**LASSO**	**EN**
Intercept	0.7346	1.1399	0.9610	**Intercept**	0.7525	1.1537	0.9822
Beta1	0.7181	0.6000	0.6618	**Beta1**	0.7078	0.5759	0.6455
Beta2	0.0731	.	.	**Beta2**	0.0789	.	.
Beta3	0.0975	.	0.0208	**Beta3**	0.0727	.	.
Beta4	0.1908	.	0.1136	**Beta4**	0.1837	.	0.1082
Beta5	0.3268	0.1580	0.2474	**Beta5**	0.3240	0.1567	0.2451
Beta6	0.8147	0.6809	0.7531	**Beta6**	0.8158	0.6830	0.7547
Beta7	0.4144	0.5870	0.4341	**Beta7**	0.4664	0.7338	0.5060
Beta8	0.5780	0.5674	0.5798	**Beta8**	0.5515	0.5028	0.5472
Beta9	0.5027	0.2694	0.4536	**Beta9**	0.4919	0.1482	0.4188
Beta10	0.7838	0.8231	0.7802	**Beta10**	0.7775	0.8644	0.7833

Table A.5 Estimated Coefficients of Data Set 5

	Data set 5 - 1				**Data set 5 - 2**		
	RR	**LASSO**	**EN**		**RR**	**LASSO**	**EN**
Intercept	1.1607	1.6044	1.3780	**Intercept**	1.1561	1.5954	1.3711
Beta1	0.8693	1.2017	0.9533	**Beta1**	0.8997	1.2163	0.9998
Beta2	0.3039	0.0119	0.3000	**Beta2**	0.2760	.	0.2262
Beta3	0.1576	0.0698	0.1154	**Beta3**	0.1469	0.0541	0.1022
Beta4	0.0649	.	.	**Beta4**	0.0500	.	.
Beta5	0.1218	0.0248	0.0746	**Beta5**	0.1206	0.0070	0.0645
Beta6	0.7773	0.7179	0.7481	**Beta6**	0.7812	0.7200	0.7512
Beta7	0.2425	0.5369	0.2434	**Beta7**	0.2459	0.5169	0.2840
Beta8	0.4528	0.1763	0.3684	**Beta8**	0.4863	0.1965	0.3836

Table A.5 (continued)

Beta9	0.9525	0.9420	0.9391	**Beta9**	0.9568	0.9505	0.9443
Beta10	0.1434	0.0574	0.1015	**Beta10**	0.1363	0.0389	0.0886
Beta11	0.4495	0.0680	0.3355	**Beta11**	0.4377	0.0830	0.3228
Beta12	0.0647	.	0.0220	**Beta12**	0.0680	.	0.0210
Beta13	0.2463	0.1813	0.2134	**Beta13**	0.2476	0.1642	0.2054
Beta14	0.6310	0.5871	0.6085	**Beta14**	0.6315	0.5804	0.6055
Beta15	0.2006	0.1258	0.1641	**Beta15**	0.2092	0.1294	0.1707
Beta16	0.6187	0.5561	0.5908	**Beta16**	0.6776	0.7362	0.6949
Beta17	0.5300	0.5587	0.4955	**Beta17**	0.4418	0.3831	0.3821
Beta18	0.1789	0.1298	0.1552	**Beta18**	0.1845	0.1228	0.1551
Beta19	0.6963	0.4968	0.6219	**Beta19**	0.6815	0.3977	0.5719
Beta20	0.6543	0.6136	0.6339	**Beta20**	0.6382	0.5922	0.6157
Beta21	0.6623	0.4941	0.6415	**Beta21**	0.6877	0.5794	0.6830
Beta22	0.0245	.	.	**Beta22**	0.0199	.	.
Beta23	0.0885	.	0.0071	**Beta23**	0.0953	.	0.0149
Beta24	0.3203	0.2521	0.2867	**Beta24**	0.3308	0.2588	0.2956
Beta25	0.1353	0.0594	0.0994	**Beta25**	0.1358	0.0603	0.0996
	Data set 5 - 3				**Data set 5 - 4**		
	RR	**LASSO**	**EN**		**RR**	**LASSO**	**EN**
Intercept	1.1378	1.6109	1.3661	**Intercept**	1.1317	1.5935	1.3544
Beta1	0.8398	1.2012	0.9219	**Beta1**	0.8461	1.1296	0.8946
Beta2	0.2926	.	0.2709	**Beta2**	0.2929	.	0.2626
Beta3	0.1610	0.0644	0.1142	**Beta3**	0.1588	0.0642	0.1130
Beta4	0.0257	.	.	**Beta4**	0.0132	.	.
Beta5	0.1235	0.0124	0.0690	**Beta5**	0.1205	0.0098	0.0673
Beta6	0.7927	0.7361	0.7647	**Beta6**	0.7758	0.7278	0.7518
Beta7	0.2375	0.5185	0.2511	**Beta7**	0.3041	0.6248	0.3388
Beta8	0.4547	0.1741	0.3580	**Beta8**	0.4695	0.1545	0.3613
Beta9	0.9895	0.9200	0.9546	**Beta9**	0.9880	1.0404	1.0053
Beta10	0.1432	0.0404	0.0928	**Beta10**	0.1396	0.0427	0.0911
Beta11	0.4784	0.0887	0.3613	**Beta11**	0.4763	0.0295	0.3288
Beta12	0.0622	.	0.0176	**Beta12**	0.0625	.	0.0104
Beta13	0.2472	0.1629	0.2056	**Beta13**	0.2395	0.1594	0.1993
Beta14	0.6239	0.5776	0.6002	**Beta14**	0.6238	0.5697	0.5952
Beta15	0.2026	0.1331	0.1689	**Beta15**	0.1952	0.1114	0.1548
Beta16	0.6901	0.7166	0.6927	**Beta16**	0.6613	0.6880	0.6664
Beta17	0.4880	0.4736	0.4650	**Beta17**	0.5151	0.4997	0.4838
Beta18	0.1834	0.1258	0.1559	**Beta18**	0.1766	0.1208	0.1492
Beta19	0.6648	0.4033	0.5540	**Beta19**	0.7017	0.4549	0.5986
Beta20	0.6468	0.5976	0.6222	**Beta20**	0.6489	0.5927	0.6208
Beta21	0.6631	0.5316	0.6253	**Beta21**	0.6220	0.4490	0.5644
Beta22	0.0186	.	.	**Beta22**	0.0150	.	.

Table A.5 (continued)

Beta23	0.1325	.	0.0533	**Beta23**	0.1056	0.0016	0.0440
Beta24	0.3372	0.2586	0.2988	**Beta24**	0.3268	0.2720	0.3012
Beta25	0.1349	0.0489	0.0936	**Beta25**	0.1272	0.0385	0.0831
	Data set 5 - 5				**Data set 5 - 6**		
	RR	**LASSO**	**EN**		**RR**	**LASSO**	**EN**
Intercept	1.1354	1.5909	1.3574	**Intercept**	1.1818	1.6332	1.4029
Beta1	0.8147	1.0668	0.8604	**Beta1**	0.8408	1.1025	0.9253
Beta2	0.2900	0.0177	0.2868	**Beta2**	0.2677	.	0.2047
Beta3	0.1603	0.0795	0.1213	**Beta3**	0.1541	0.0649	0.1111
Beta4	0.0407	.	.	**Beta4**	0.0344	.	.
Beta5	0.1243	0.0164	0.0717	**Beta5**	0.1191	0.0227	0.0724
Beta6	0.7781	0.7269	0.7529	**Beta6**	0.7804	0.7244	0.7528
Beta7	0.2981	0.5869	0.3033	**Beta7**	0.3187	0.5675	0.3600
Beta8	0.4618	0.1667	0.3658	**Beta8**	0.4294	0.1340	0.3215
Beta9	1.0029	1.0106	0.9945	**Beta9**	0.9702	1.0267	0.9841
Beta10	0.1331	0.0296	0.0822	**Beta10**	0.1365	0.0465	0.0933
Beta11	0.4837	0.1531	0.3880	**Beta11**	0.4556	0.0855	0.3226
Beta12	0.0708	.	0.0203	**Beta12**	0.0681	0.0004	0.0265
Beta13	0.2484	0.1683	0.2077	**Beta13**	0.2507	0.1736	0.2124
Beta14	0.6261	0.5827	0.6046	**Beta14**	0.6289	0.5780	0.6037
Beta15	0.2079	0.1281	0.1691	**Beta15**	0.2040	0.1267	0.1667
Beta16	0.6469	0.6788	0.6562	**Beta16**	0.6573	0.6950	0.6646
Beta17	0.5141	0.5470	0.4945	**Beta17**	0.5415	0.4850	0.5034
Beta18	0.1863	0.1316	0.1591	**Beta18**	0.1925	0.1451	0.1697
Beta19	0.6810	0.4205	0.5775	**Beta19**	0.6687	0.4139	0.5628
Beta20	0.6454	0.5981	0.6220	**Beta20**	0.6436	0.5960	0.6205
Beta21	0.6514	0.4556	0.6072	**Beta21**	0.5864	0.4918	0.5563
Beta22	0.0268	.	.	**Beta22**	0.0254	.	.
Beta23	0.1143	.	0.0283	**Beta23**	0.1254	.	0.0524
Beta24	0.3269	0.2585	0.2934	**Beta24**	0.3247	0.2639	0.2946
Beta25	0.1352	0.0516	0.0956	**Beta25**	0.1339	0.0602	0.0984
	Data set 5 - 7				**Data set 5 - 8**		
	RR	**LASSO**	**EN**		**RR**	**LASSO**	**EN**
Intercept	1.1289	1.5979	1.3581	**Intercept**	1.1553	1.6324	1.3876
Beta1	0.8626	1.1761	0.9450	**Beta1**	0.8653	1.1495	0.9065
Beta2	0.3198	.	0.2606	**Beta2**	0.3017	.	0.2635
Beta3	0.1645	0.0723	0.1199	**Beta3**	0.1562	0.0581	0.1093
Beta4	0.0322	.	.	**Beta4**	-0.0179	.	.
Beta5	0.1224	0.0188	0.0717	**Beta5**	0.1220	0.0220	0.0736
Beta6	0.7806	0.7291	0.7549	**Beta6**	0.7809	0.7364	0.7592
Beta7	0.2804	0.6178	0.3385	**Beta7**	0.2786	0.5432	0.2978
Beta8	0.4506	0.1377	0.3381	**Beta8**	0.4954	0.2520	0.4088
Beta9	1.0045	1.0020	0.9927	**Beta9**	0.9490	0.9283	0.9333
Beta10	0.1464	0.0420	0.0954	**Beta10**	0.1386	0.0392	0.0905

Table A.5 (continued)

Beta11	0.4242	0.0165	0.2900	**Beta11**	0.4643	0.0320	0.3181
Beta12	0.0683	.	0.0210	**Beta12**	0.0700	.	0.0231
Beta13	0.2367	0.1582	0.1978	**Beta13**	0.2477	0.1719	0.2095
Beta14	0.6276	0.5801	0.6035	**Beta14**	0.6289	0.5802	0.6045
Beta15	0.1997	0.1231	0.1627	**Beta15**	0.1967	0.1155	0.1572
Beta16	0.6597	0.7114	0.6712	**Beta16**	0.6944	0.7458	0.7039
Beta17	0.5390	0.5680	0.5235	**Beta17**	0.5167	0.4738	0.4767
Beta18	0.1771	0.1173	0.1485	**Beta18**	0.1889	0.1281	0.1589
Beta19	0.6921	0.3880	0.5705	**Beta19**	0.6708	0.4022	0.5632
Beta20	0.6463	0.6050	0.6262	**Beta20**	0.6414	0.5875	0.6155
Beta21	0.6106	0.4028	0.5492	**Beta21**	0.5741	0.4793	0.5583
Beta22	0.0257	.	.	**Beta22**	0.0301	.	.
Beta23	0.1257	0.0427	0.0633	**Beta23**	0.1590	.	0.0615
Beta24	0.3308	0.2652	0.2986	**Beta24**	0.3309	0.2654	0.2986
Beta25	0.1417	0.0690	0.1063	**Beta25**	0.1345	0.0612	0.0994
	Data set 5 - 9				**Data set 5 - 10**		
	RR	**LASSO**	**EN**		**RR**	**LASSO**	**EN**
Intercept	1.1422	1.6050	1.3657	**Intercept**	1.1475	1.5951	1.3659
Beta1	0.8618	1.1580	0.9369	**Beta1**	0.8122	0.9235	0.8443
Beta2	0.2774	.	0.2359	**Beta2**	0.3206	0.3065	0.3633
Beta3	0.1631	0.0673	0.1164	**Beta3**	0.1612	0.0715	0.1179
Beta4	0.0338	.	.	**Beta4**	0.0487	.	.
Beta5	0.1242	0.0274	0.0767	**Beta5**	0.1154	0.0062	0.0616
Beta6	0.7901	0.7301	0.7602	**Beta6**	0.7772	0.7239	0.7507
Beta7	0.2621	0.4592	0.2626	**Beta7**	0.2647	0.3323	0.2392
Beta8	0.4885	0.2721	0.4178	**Beta8**	0.4702	0.2574	0.3924
Beta9	1.0135	0.9783	0.9887	**Beta9**	0.9862	0.9877	0.9727
Beta10	0.1357	0.0445	0.0914	**Beta10**	0.1344	0.0367	0.0873
Beta11	0.4217	0.0543	0.3053	**Beta11**	0.4851	0.1791	0.3755
Beta12	0.0701	.	0.0225	**Beta12**	0.0662	.	0.0206
Beta13	0.2500	0.1722	0.2107	**Beta13**	0.2322	0.1462	0.1888
Beta14	0.6256	0.5830	0.6042	**Beta14**	0.6267	0.5695	0.5978
Beta15	0.2036	0.1213	0.1638	**Beta15**	0.2040	0.1323	0.1690
Beta16	0.6272	0.7600	0.6680	**Beta16**	0.6285	0.6228	0.6224
Beta17	0.5320	0.3925	0.4716	**Beta17**	0.5576	0.6325	0.5478
Beta18	0.1807	0.1284	0.1558	**Beta18**	0.1759	0.1231	0.1509
Beta19	0.6833	0.3708	0.5539	**Beta19**	0.6609	0.3877	0.5557
Beta20	0.6515	0.6066	0.6288	**Beta20**	0.6544	0.6090	0.6324
Beta21	0.6426	0.6073	0.6291	**Beta21**	0.6556	0.4320	0.6039
Beta22	0.0246	.	.	**Beta22**	0.0256	.	.
Beta23	0.1311	.	0.0593	**Beta23**	0.062252	.	.
Beta24	0.3344	0.2713	0.3033	**Beta24**	0.32563	0.2622	0.2947
Beta25	0.1377	0.0576	0.0983	**Beta25**	0.13352	0.0462	0.0912

Table A.6 Estimated Coefficients of Data Set 6

	Data set 6 - 1				Data set 6 - 2		
	RR	LASSO	EN		RR	LASSO	EN
Intercept	1.6994	6.9820	4.6305	**Intercept**	1.7299	6.8628	4.6439
Beta1	0.2841	0.1544	0.2376	**Beta1**	0.2802	0.1094	0.2142
Beta2	0.6378	0.2518	0.4513	**Beta2**	0.6271	0.2671	0.4485
Beta3	0.9528	0.6608	0.8010	**Beta3**	0.9635	0.6268	0.7912
Beta4	0.6825	1.2031	0.8558	**Beta4**	0.6611	0.9012	0.6979
Beta5	0.7972	0.4754	0.6250	**Beta5**	0.7898	0.4938	0.6274
Beta6	0.1140	.	0.0274	**Beta6**	0.0941	.	.
Beta7	0.4673	.	0.2759	**Beta7**	0.4322	.	0.2612
Beta8	0.6571	0.3838	0.5546	**Beta8**	0.6823	0.5894	0.6700
Beta9	0.5363	0.1099	0.3226	**Beta9**	0.5382	0.1136	0.3243
Beta10	0.3899	0.8355	0.5911	**Beta10**	0.4403	0.9042	0.6663
Beta11	0.3061	.	0.1468	**Beta11**	0.2835	.	0.1140
Beta12	0.9379	1.0077	0.9835	**Beta12**	0.9320	1.0008	0.9902
Beta13	0.5769	0.2592	0.4221	**Beta13**	0.5473	0.1729	0.3576
Beta14	0.6189	0.1872	0.4432	**Beta14**	0.6359	0.2929	0.4915
Beta15	0.7238	0.4132	0.5646	**Beta15**	0.7663	0.4420	0.5991
Beta16	0.7194	1.0945	0.9556	**Beta16**	0.6885	1.1740	0.8789
Beta17	0.2276	.	0.0272	**Beta17**	0.2034	.	.
Beta18	0.1635	.	0.0310	**Beta18**	0.1636	.	0.0288
Beta19	0.8241	0.6282	0.7822	**Beta19**	0.8555	0.6074	0.8207
Beta20	0.8252	0.5848	0.7174	**Beta20**	0.7721	0.5235	0.6516
Beta21	0.5101	0.1758	0.3217	**Beta21**	0.4918	0.1366	0.2954
Beta22	0.1242	.	.	**Beta22**	0.0933	.	.
Beta23	0.3413	0.3999	0.3504	**Beta23**	0.3612	0.3447	0.3198
Beta24	0.3269	0.0634	0.1928	**Beta24**	0.2924	.	0.1184
Beta25	0.3643	.	0.1441	**Beta25**	0.3497	.	0.1457
Beta26	0.1261	.	0.0447	**Beta26**	0.1174	.	.
Beta27	0.0986	.	.	**Beta27**	0.1121	.	.
Beta28	0.7216	0.5109	0.6450	**Beta28**	0.7449	0.5605	0.7045
Beta29	0.2517	.	0.0740	**Beta29**	0.2971	.	0.1773
Beta30	0.3766	0.3131	0.3705	**Beta30**	0.3339	0.0976	0.2561
Beta31	0.3474	0.0295	0.1748	**Beta31**	0.3807	0.1332	0.2520
Beta32	0.8129	1.2065	0.9199	**Beta32**	0.7720	0.9915	0.8479
Beta33	0.8871	0.7606	0.8677	**Beta33**	0.9193	0.9162	0.9187
Beta34	0.5537	0.2099	0.3922	**Beta34**	0.5582	0.2673	0.4173
Beta35	0.4802	0.1504	0.4041	**Beta35**	0.4929	0.3987	0.4675
Beta36	0.7721	0.5167	0.6358	**Beta36**	0.7813	0.5477	0.6533
Beta37	0.3363	0.0859	0.2127	**Beta37**	0.3128	.	0.1422
Beta38	0.3986	0.0157	0.2208	**Beta38**	0.4217	0.1073	0.2788
Beta39	0.8234	0.9670	0.8579	**Beta39**	0.8027	0.7779	0.7941
Beta40	0.8394	0.6929	0.7738	**Beta40**	0.7949	0.6769	0.7361
Beta41	0.1894	.	0.0361	**Beta41**	0.1583	.	0.0026
Beta42	0.3675	0.0869	0.1971	**Beta42**	0.4532	0.3251	0.3646

Table A.6 (continued)

Beta43	0.1404	0.0804	0.1074	**Beta43**	0.1986	0.1129	0.1705
Beta44	0.5574	0.2859	0.4591	**Beta44**	0.5436	0.3425	0.4874
Beta45	0.0790	.	.	**Beta45**	0.1079	.	.
Beta46	0.4579	0.1546	0.3018	**Beta46**	0.4544	0.1840	0.3188
Beta47	0.0854	.	0.0526	**Beta47**	0.0688	.	0.0352
Beta48	0.8560	0.8554	0.7781	**Beta48**	0.8774	0.8778	0.8004
Beta49	0.1866	.	.	**Beta49**	0.1769	.	.
Beta50	0.1927	.	.	**Beta50**	0.1660	.	.
Beta51	0.3026	0.1224	0.2269	**Beta51**	0.2556	0.0870	0.1817
Beta52	0.9027	0.8574	0.9372	**Beta52**	0.9308	0.9221	1.0251
Beta53	0.1940	.	0.0812	**Beta53**	0.1038	.	.
Beta54	0.1488	.	.	**Beta54**	0.1901	.	.
Beta55	0.8310	0.5798	0.7139	**Beta55**	0.8391	0.5747	0.7110
Beta56	0.6161	0.2446	0.4355	**Beta56**	0.6312	0.2777	0.4630
Beta57	0.8024	0.4434	0.6156	**Beta57**	0.7887	0.4684	0.6157
Beta58	0.7561	0.4319	0.6008	**Beta58**	0.7626	0.4471	0.6138
Beta59	0.6455	0.2141	0.4184	**Beta59**	0.6456	0.2244	0.4246
Beta60	0.0594	0.0368	0.0653	**Beta60**	0.1202	.	0.0768
	Data set 6 - 3				**Data set 6 - 4**		
	RR	**LASSO**	**EN**		**RR**	**LASSO**	**EN**
Intercept	1.7454	7.0992	4.7475	**Intercept**	1.6819	6.6024	4.4373
Beta1	0.2384	0.0907	0.1910	**Beta1**	0.2712	0.1551	0.2315
Beta2	0.6293	0.2695	0.4621	**Beta2**	0.6588	0.3354	0.5089
Beta3	0.9903	0.7376	0.8645	**Beta3**	0.9563	0.7068	0.8272
Beta4	0.6811	1.2156	0.8492	**Beta4**	0.6742	1.0692	0.7958
Beta5	0.7665	0.4182	0.5777	**Beta5**	0.7768	0.4465	0.5960
Beta6	0.1091	.	0.0062	**Beta6**	0.1037	.	0.0323
Beta7	0.4895	.	0.3766	**Beta7**	0.4623	.	0.2993
Beta8	0.6592	0.5192	0.6033	**Beta8**	0.6813	0.5806	0.6639
Beta9	0.5499	0.0721	0.3081	**Beta9**	0.5643	0.0982	0.3297
Beta10	0.3784	0.8256	0.4784	**Beta10**	0.4684	0.9684	0.6650
Beta11	0.3344	0.0295	0.2281	**Beta11**	0.3147	.	0.1758
Beta12	0.9362	0.9710	0.9571	**Beta12**	0.9281	0.9176	0.9221
Beta13	0.5508	0.1283	0.3320	**Beta13**	0.5645	0.2453	0.3940
Beta14	0.6121	0.1844	0.4408	**Beta14**	0.5923	0.2214	0.4240
Beta15	0.7548	0.4011	0.5686	**Beta15**	0.7815	0.4608	0.6111
Beta16	0.7106	1.3012	0.8959	**Beta16**	0.6762	1.1240	0.9582
Beta17	0.2326	.	0.0240	**Beta17**	0.1952	.	0.0009
Beta18	0.1600	.	0.0198	**Beta18**	0.1784	0.0004	0.0620
Beta19	0.8215	0.4468	0.7573	**Beta19**	0.8403	0.7422	0.8340
Beta20	0.7961	0.5525	0.6796	**Beta20**	0.7920	0.5985	0.7008
Beta21	0.4811	0.1716	0.3144	**Beta21**	0.5023	0.2576	0.3674
Beta22	0.0399	.	.	**Beta22**	0.1403	.	.
Beta23	0.3533	0.4557	0.3523	**Beta23**	0.3409	0.2824	0.2835

Table A.6 (continued)

Beta24	0.2906	.	0.1109	**Beta24**	0.2856	0.0006	0.1091
Beta25	0.3491	.	0.1219	**Beta25**	0.3592	0.0016	0.1513
Beta26	0.1239	.	0.0002	**Beta26**	0.1403	.	0.0560
Beta27	0.0893	.	.	**Beta27**	0.1109	.	.
Beta28	0.7453	0.4336	0.6651	**Beta28**	0.7620	0.5708	0.6933
Beta29	0.3424	.	0.2083	**Beta29**	0.2712	.	0.0945
Beta30	0.3465	0.3267	0.3143	**Beta30**	0.3396	0.1344	0.2857
Beta31	0.3598	0.0147	0.1756	**Beta31**	0.3786	0.1132	0.2350
Beta32	0.7324	0.6019	0.7247	**Beta32**	0.8006	1.0762	0.8703
Beta33	0.9353	1.0087	0.9642	**Beta33**	0.9407	0.9171	0.9453
Beta34	0.5392	0.2293	0.3946	**Beta34**	0.5531	0.2504	0.4119
Beta35	0.5289	0.6724	0.5393	**Beta35**	0.4799	0.2497	0.4204
Beta36	0.8244	0.5832	0.6841	**Beta36**	0.7639	0.5258	0.6328
Beta37	0.2997	.	0.1192	**Beta37**	0.3214	0.0016	0.1560
Beta38	0.4327	0.1150	0.2910	**Beta38**	0.4344	0.1573	0.3141
Beta39	0.8170	0.8127	0.8365	**Beta39**	0.8092	0.8546	0.8195
Beta40	0.8008	0.6565	0.7339	**Beta40**	0.8039	0.7410	0.7719
Beta41	0.1767	.	0.0252	**Beta41**	0.1728	.	0.0334
Beta42	0.4497	0.1468	0.3116	**Beta42**	0.4197	0.2149	0.2863
Beta43	0.1705	0.0915	0.1291	**Beta43**	0.2135	0.1606	0.1975
Beta44	0.5459	0.3775	0.4783	**Beta44**	0.5847	0.3178	0.4990
Beta45	0.0921	.	.	**Beta45**	0.0705	.	.
Beta46	0.4415	0.1045	0.2755	**Beta46**	0.4277	0.1508	0.2887
Beta47	0.0833	0.0188	0.0721	**Beta47**	0.0131	.	.
Beta48	0.8707	0.8703	0.8264	**Beta48**	0.8929	0.9034	0.8229
Beta49	0.1881	.	0.0179	**Beta49**	0.1754	.	0.0240
Beta50	0.2009	.	.	**Beta50**	0.1866	.	0.0005
Beta51	0.3007	0.0416	0.1768	**Beta51**	0.2433	0.0587	0.1617
Beta52	0.9232	0.8814	0.9963	**Beta52**	0.8671	0.8976	0.9995
Beta53	0.0979	.	.	**Beta53**	0.1163	.	.
Beta54	0.1805	.	.	**Beta54**	0.2219	.	.
Beta55	0.8308	0.5754	0.7065	**Beta55**	0.8422	0.6055	0.7294
Beta56	0.6125	0.2281	0.4319	**Beta56**	0.6292	0.2640	0.4474
Beta57	0.7695	0.4291	0.6011	**Beta57**	0.7864	0.4763	0.6254
Beta58	0.7474	0.4458	0.6076	**Beta58**	0.7522	0.5017	0.6392
Beta59	0.6247	0.1074	0.3568	**Beta59**	0.6606	0.2300	0.4329
Beta60	0.0961	.	0.0704	**Beta60**	0.0810	0.0002	0.0603
	Data set 6 - 5				**Data set 6 - 6**		
	RR	**LASSO**	**EN**		**RR**	**LASSO**	**EN**
Intercept	1.6530	6.9182	4.5628	**Intercept**	1.6066	6.9867	4.6527
Beta1	0.2796	0.1124	0.2082	**Beta1**	0.2675	0.1223	0.2132
Beta2	0.6095	0.2694	0.4488	**Beta2**	0.6503	0.3377	0.5167
Beta3	0.9714	0.689555	0.831652	**Beta3**	0.9624	0.6859	0.8256
Beta4	0.7055	1.337781	0.933011	**Beta4**	0.6851	1.3025	0.9050

Table A.6 (continued)

Beta5	0.7688	0.428736	0.586548	**Beta5**	0.8042	0.4126	0.5875
Beta6	0.0710	.	.	**Beta6**	0.1447	.	0.0059
Beta7	0.4554	.	0.2996	**Beta7**	0.4179	.	0.3063
Beta8	0.6221	0.4721	0.5665	**Beta8**	0.6485	0.2779	0.4878
Beta9	0.5483	0.0975	0.3289	**Beta9**	0.5470	0.0536	0.2962
Beta10	0.4055	0.8232	0.5355	**Beta10**	0.4498	0.8985	0.5978
Beta11	0.3127	0.0244	0.2010	**Beta11**	0.3319	0.0444	0.2154
Beta12	0.9129	0.8606	0.8834	**Beta12**	0.9508	0.9156	0.9357
Beta13	0.5637	0.2705	0.4138	**Beta13**	0.5635	0.1760	0.3724
Beta14	0.6450	0.0717	0.4137	**Beta14**	0.6290	0.2453	0.4703
Beta15	0.7496	0.3860	0.5659	**Beta15**	0.7466	0.3996	0.5699
Beta16	0.6607	0.9417	0.8720	**Beta16**	0.6709	1.2106	0.9257
Beta17	0.2263	.	0.0233	**Beta17**	0.2002	.	.
Beta18	0.2009	.	0.0567	**Beta18**	0.1902	.	0.0363
Beta19	0.8691	0.8820	0.9120	**Beta19**	0.8398	0.6521	0.8864
Beta20	0.8164	0.5540	0.6924	**Beta20**	0.7850	0.4864	0.6436
Beta21	0.4947	0.2266	0.3472	**Beta21**	0.5091	0.2396	0.3617
Beta22	0.1417	.	.	**Beta22**	0.1500	.	.
Beta23	0.3172	0.2146	0.2400	**Beta23**	0.3097	0.3844	0.2152
Beta24	0.3248	0.0436	0.1836	**Beta24**	0.2754	.	0.1210
Beta25	0.3695	.	0.1512	**Beta25**	0.3515	.	0.1230
Beta26	0.1192	.	0.0566	**Beta26**	0.1497	.	0.1021
Beta27	0.1210	.	.	**Beta27**	0.0867	.	.
Beta28	0.8087	0.6703	0.7533	**Beta28**	0.7211	0.4268	0.5950
Beta29	0.2838	.	0.0946	**Beta29**	0.2967	.	0.1600
Beta30	0.3338	0.3805	0.3357	**Beta30**	0.4369	0.6122	0.4917
Beta31	0.3670	0.0616	0.2077	**Beta31**	0.3490	0.0461	0.1908
Beta32	0.7725	0.7920	0.8111	**Beta32**	0.8061	1.0765	0.9246
Beta33	0.8937	0.8478	0.8832	**Beta33**	0.8398	0.7192	0.8170
Beta34	0.5677	0.3023	0.4477	**Beta34**	0.5611	0.2528	0.4104
Beta35	0.5659	0.6699	0.5784	**Beta35**	0.4476	0.2131	0.3883
Beta36	0.7936	0.5599	0.6623	**Beta36**	0.7904	0.5010	0.6197
Beta37	0.3200	.	0.1474	**Beta37**	0.3329	.	0.1496
Beta38	0.4210	0.1115	0.2845	**Beta38**	0.4306	0.0828	0.2662
Beta39	0.7399	0.5849	0.6836	**Beta39**	0.8230	0.9483	0.8388
Beta40	0.8098	0.6982	0.7579	**Beta40**	0.8111	0.5779	0.7247
Beta41	0.2064	.	0.0386	**Beta41**	0.1820	.	.
Beta42	0.4098	0.1278	0.2804	**Beta42**	0.4408	0.0923	0.2611
Beta43	0.1251	0.0410	0.0913	**Beta43**	0.1676	0.3536	0.2893
Beta44	0.5424	0.5843	0.5675	**Beta44**	0.5281	.	0.3473
Beta45	0.0881	.	.	**Beta45**	0.1073	.	.
Beta46	0.4441	0.1497	0.2916	**Beta46**	0.4591	0.1462	0.2992
Beta47	0.1613	.	0.0990	**Beta47**	0.0914	.	0.0164
Beta48	0.8953	0.7277	0.7931	**Beta48**	0.8864	1.1636	0.9064

Table A.6 (continued)

Beta49	0.1726	.	0.0278	**Beta49**	0.2300	.	0.0538
Beta50	0.1762	.	.	**Beta50**	0.1817	.	.
Beta51	0.2602	0.0964	0.1716	**Beta51**	0.2826	.	0.1689
Beta52	0.8898	0.9319	1.0331	**Beta52**	0.9267	0.8487	0.9922
Beta53	0.1033	.	.	**Beta53**	0.1156	.	0.0126
Beta54	0.2477	.	.	**Beta54**	0.2083	.	.
Beta55	0.8180	0.5211	0.6782	**Beta55**	0.8331	0.5572	0.6964
Beta56	0.6389	0.2884	0.4673	**Beta56**	0.6123	0.2302	0.4226
Beta57	0.7945	0.4329	0.5977	**Beta57**	0.8027	0.4445	0.6141
Beta58	0.7572	0.4910	0.6299	**Beta58**	0.7561	0.4861	0.6297
Beta59	0.6268	0.1628	0.3865	**Beta59**	0.6452	0.1166	0.3672
Beta60	0.1054	.	0.0741	**Beta60**	0.0784	.	0.0363
	Data set 6 - 7				**Data set 6 - 8**		
	RR	**LASSO**	**EN**		**RR**	**LASSO**	**EN**
Intercept	1.6872	6.9959	4.7144	**Intercept**	1.6217	6.9206	4.5719
Beta1	0.2900	0.1217	0.2248	**Beta1**	0.2685	0.0676	0.1802
Beta2	0.6145	0.2515	0.4329	**Beta2**	0.6605	0.3260	0.5014
Beta3	0.9849	0.6758	0.8319	**Beta3**	0.9804	0.7466	0.8673
Beta4	0.6422	0.9910	0.7394	**Beta4**	0.6415	1.0395	0.7466
Beta5	0.7949	0.4565	0.6148	**Beta5**	0.7973	0.4060	0.5853
Beta6	0.1049	.	0.0014	**Beta6**	0.1281	.	0.0234
Beta7	0.5074	.	0.3862	**Beta7**	0.4343	.	0.3280
Beta8	0.6261	0.5899	0.6067	**Beta8**	0.6842	0.4634	0.5883
Beta9	0.5540	0.0910	0.3193	**Beta9**	0.5835	0.1213	0.3520
Beta10	0.4054	0.9269	0.5419	**Beta10**	0.4258	0.9600	0.5937
Beta11	0.3613	0.0459	0.2608	**Beta11**	0.3319	.	0.1896
Beta12	0.9101	0.9313	0.9048	**Beta12**	0.9180	0.9567	0.9449
Beta13	0.5594	0.1529	0.3422	**Beta13**	0.5402	0.2242	0.3778
Beta14	0.6212	0.1855	0.4387	**Beta14**	0.6573	0.1962	0.4733
Beta15	0.7746	0.4462	0.5992	**Beta15**	0.7676	0.4021	0.5793
Beta16	0.7641	0.9433	0.9879	**Beta16**	0.6472	1.1676	0.9220
Beta17	0.2216	.	.	**Beta17**	0.1871	.	.
Beta18	0.1785	.	0.0632	**Beta18**	0.1767	.	0.0379
Beta19	0.8349	1.0412	0.9222	**Beta19**	0.8209	0.5887	0.7932
Beta20	0.7641	0.5588	0.6674	**Beta20**	0.7915	0.5733	0.6872
Beta21	0.4790	0.1646	0.3014	**Beta21**	0.4751	0.1952	0.3208
Beta22	0.0965	.	.	**Beta22**	0.1219	.	.
Beta23	0.3477	0.0584	0.2229	**Beta23**	0.3673	0.4155	0.3227
Beta24	0.2937	.	0.1222	**Beta24**	0.3160	0.0193	0.1637
Beta25	0.3502	.	0.1121	**Beta25**	0.3321	.	0.1239
Beta26	0.1431	0.0487	0.0883	**Beta26**	0.1158	.	0.0531
Beta27	0.0990	.	.	**Beta27**	0.1174	.	.
Beta28	0.7398	0.5231	0.6393	**Beta28**	0.7587	0.5031	0.6617
Beta29	0.2222	.	.	**Beta29**	0.3034	.	0.1230

Table A.6 (continued)

Beta30	0.4192	0.3909	0.3981	**Beta30**	0.3891	0.1737	0.3251
Beta31	0.3687	0.0705	0.2056	**Beta31**	0.3710	0.0496	0.2071
Beta32	0.8438	1.0992	0.9232	**Beta32**	0.7983	1.1360	0.9142
Beta33	0.8342	0.7040	0.7961	**Beta33**	0.9037	0.7802	0.8575
Beta34	0.5500	0.2470	0.4056	**Beta34**	0.5448	0.2879	0.4275
Beta35	0.4570	0.2400	0.3972	**Beta35**	0.4602	0.3201	0.4431
Beta36	0.7502	0.5300	0.6168	**Beta36**	0.7930	0.4647	0.6155
Beta37	0.3472	.	0.1517	**Beta37**	0.3434	.	0.1537
Beta38	0.4462	0.0943	0.2782	**Beta38**	0.4290	0.1078	0.2812
Beta39	0.7997	0.8632	0.8077	**Beta39**	0.8139	0.7870	0.7815
Beta40	0.8136	0.6602	0.7551	**Beta40**	0.8218	0.7308	0.7836
Beta41	0.1736	.	.	**Beta41**	0.1777	.	0.0322
Beta42	0.4264	0.2177	0.3181	**Beta42**	0.4244	0.2943	0.3267
Beta43	0.1450	0.0264	0.0927	**Beta43**	0.2204	0.1567	0.1776
Beta44	0.6257	0.4547	0.5700	**Beta44**	0.5071	0.1577	0.4097
Beta45	0.0597	.	.	**Beta45**	0.1034	.	.
Beta46	0.4740	0.2029	0.3394	**Beta46**	0.4589	0.1196	0.2825
Beta47	0.0361	.	0.0113	**Beta47**	0.0723	.	0.0439
Beta48	0.8766	0.8205	0.7755	**Beta48**	0.8543	1.0631	0.8531
Beta49	0.2003	.	0.0149	**Beta49**	0.2160	.	0.0479
Beta50	0.1642	.	.	**Beta50**	0.1651	.	.
Beta51	0.2689	0.1321	0.2217	**Beta51**	0.3117	0.0511	0.2111
Beta52	0.8096	0.8565	0.9580	**Beta52**	0.8864	0.9029	0.9861
Beta53	0.1214	.	.	**Beta53**	0.1033	.	.
Beta54	0.2767	.	.	**Beta54**	0.1845	.	.
Beta55	0.8697	0.6107	0.7442	**Beta55**	0.8278	0.6032	0.7307
Beta56	0.6170	0.2322	0.4233	**Beta56**	0.6093	0.3127	0.4631
Beta57	0.7775	0.3972	0.5789	**Beta57**	0.8140	0.4608	0.6278
Beta58	0.7493	0.4440	0.6021	**Beta58**	0.7577	0.4570	0.6167
Beta59	0.6392	0.1376	0.3735	**Beta59**	0.6505	0.1516	0.3945
Beta60	0.1241	0.1163	0.1763	**Beta60**	0.1145	.	0.0773
	Data set 6 - 9				**Data set 6 - 10**		
	RR	**LASSO**	**EN**		**RR**	**LASSO**	**EN**
Intercept	1.7261	6.9468	4.6687	**Intercept**	1.7477	7.2505	4.8347
Beta1	0.3068	0.1379	0.2424	**Beta1**	0.2764	0.1095	0.2145
Beta2	0.6273	0.2675	0.4537	**Beta2**	0.6375	0.2597	0.4618
Beta3	0.9720	0.7527	0.8670	**Beta3**	0.9988	0.6896	0.8448
Beta4	0.7044	1.1491	0.8355	**Beta4**	0.7101	1.3640	0.9582
Beta5	0.7474	0.3533	0.5286	**Beta5**	0.7736	0.3725	0.5607
Beta6	0.1290	.	0.0132	**Beta6**	0.1562	.	0.0794
Beta7	0.4892	.	0.3631	**Beta7**	0.4372	.	0.2281
Beta8	0.6172	0.4223	0.5269	**Beta8**	0.6350	0.3669	0.5578
Beta9	0.5846	0.1399	0.3652	**Beta9**	0.5201	0.0576	0.2862
Beta10	0.3909	0.9222	0.5615	**Beta10**	0.4264	0.8252	0.6197

Table A.6 (continued)

Beta11	0.3574	.	0.2193	**Beta11**	0.2956	0.0979	0.2029
Beta12	0.9137	0.8908	0.8924	**Beta12**	0.9134	0.8812	0.8944
Beta13	0.5411	0.1938	0.3591	**Beta13**	0.5592	0.2135	0.3847
Beta14	0.5862	0.2564	0.4699	**Beta14**	0.6062	0.0639	0.3621
Beta15	0.7465	0.3926	0.5608	**Beta15**	0.7503	0.3526	0.5327
Beta16	0.6717	1.0967	0.9408	**Beta16**	0.6713	1.2722	0.9157
Beta17	0.2074	.	0.0071	**Beta17**	0.1785	.	.
Beta18	0.1735	.	0.0418	**Beta18**	0.1823	.	0.0529
Beta19	0.8131	0.6404	0.7580	**Beta19**	0.7943	0.4328	0.7394
Beta20	0.8095	0.6158	0.7183	**Beta20**	0.8225	0.5869	0.7057
Beta21	0.4755	0.1286	0.2873	**Beta21**	0.4975	0.1841	0.3208
Beta22	0.1372	.	0.0166	**Beta22**	0.1558	.	.
Beta23	0.3976	0.3760	0.3910	**Beta23**	0.3610	0.4323	0.3120
Beta24	0.2980	.	0.1133	**Beta24**	0.2831	.	0.0781
Beta25	0.3430	.	0.1461	**Beta25**	0.3254	.	0.1192
Beta26	0.0913	.	0.0002	**Beta26**	0.1091	.	0.0277
Beta27	0.1118	.	.	**Beta27**	0.1286	.	.
Beta28	0.7169	0.4945	0.6458	**Beta28**	0.7700	0.4591	0.6735
Beta29	0.2677	.	0.0668	**Beta29**	0.2920	.	0.1895
Beta30	0.3377	0.2071	0.3239	**Beta30**	0.4214	0.7677	0.4934
Beta31	0.3659	0.0683	0.2083	**Beta31**	0.3500	0.0014	0.1626
Beta32	0.7931	1.0751	0.8682	**Beta32**	0.7728	0.6533	0.8030
Beta33	0.9026	0.8759	0.9030	**Beta33**	0.8704	0.7139	0.8269
Beta34	0.5787	0.2457	0.4199	**Beta34**	0.5516	0.2888	0.4263
Beta35	0.4540	0.2386	0.4180	**Beta35**	0.5565	0.7878	0.5967
Beta36	0.7851	0.5895	0.6743	**Beta36**	0.7517	0.5003	0.6163
Beta37	0.3447	0.0386	0.1983	**Beta37**	0.3187	.	0.1521
Beta38	0.4170	0.1107	0.2823	**Beta38**	0.4174	0.0421	0.2445
Beta39	0.8203	0.8027	0.7832	**Beta39**	0.8315	0.7929	0.8204
Beta40	0.8397	0.7073	0.7842	**Beta40**	0.7497	0.4047	0.6133
Beta41	0.1755	.	0.0213	**Beta41**	0.1655	.	0.0089
Beta42	0.4614	0.2926	0.3407	**Beta42**	0.4014	0.0199	0.2324
Beta43	0.1475	0.1481	0.1297	**Beta43**	0.1798	0.1479	0.1907
Beta44	0.4961	0.2560	0.4124	**Beta44**	0.6180	0.4497	0.5799
Beta45	0.0977	.	.	**Beta45**	0.0897	.	.
Beta46	0.4615	0.1408	0.2939	**Beta46**	0.4353	0.1346	0.2779
Beta47	0.1299	.	0.0996	**Beta47**	0.0547	.	.
Beta48	0.9151	0.9476	0.8692	**Beta48**	0.8036	0.7926	0.7199
Beta49	0.1756	.	.	**Beta49**	0.2084	.	0.0221
Beta50	0.1669	.	.	**Beta50**	0.2031	.	.
Beta51	0.2748	0.1004	0.1921	**Beta51**	0.3015	.	0.1679
Beta52	0.9173	0.9384	1.0461	**Beta52**	0.8215	0.8550	0.9559
Beta53	0.1342	.	.	**Beta53**	0.1375	.	.
Beta54	0.2033	.	.	**Beta54**	0.2378	.	.

Table A.6 (continued)

Beta55	0.8199	0.5766	0.7053	**Beta55**	0.7963	0.5187	0.6665
Beta56	0.5954	0.2392	0.4189	**Beta56**	0.6164	0.2062	0.4146
Beta57	0.8328	0.4877	0.6553	**Beta57**	0.7934	0.4517	0.6150
Beta58	0.7555	0.4569	0.6072	**Beta58**	0.7639	0.4490	0.6159
Beta59	0.5840	0.0912	0.3250	**Beta59**	0.6636	0.2578	0.4480
Beta60	0.0526	.	0.0342	**Beta60**	0.0971	0.0024	0.0961

Table A.7 Estimated Coefficients of Data Set 7

	Data set 7 - 1				**Data set 7 - 2**		
	RR	**LASSO**	**EN**		**RR**	**LASSO**	**EN**
Intercept	1.5165	1.6851	1.5998	**Intercept**	1.5088	1.6790	1.5929
Beta1	0.8791	0.8262	0.8569	**Beta1**	0.9005	0.8618	0.8844
Beta2	0.5420	0.6161	0.5480	**Beta2**	0.5613	0.6599	0.5735
Beta3	0.5722	0.5195	0.5464	**Beta3**	0.5760	0.5227	0.5498
Beta4	0.7289	0.6847	0.7073	**Beta4**	0.7271	0.6817	0.7049
Beta5	0.4292	0.3637	0.3972	**Beta5**	0.4238	0.3574	0.3914
Beta6	0.7334	0.6864	0.7104	**Beta6**	0.7296	0.6819	0.7062
Beta7	0.9916	0.8804	0.9448	**Beta7**	0.9553	0.8225	0.9010
Beta8	0.4138	0.2164	0.3483	**Beta8**	0.4092	0.1893	0.3393
Beta9	0.5482	0.4925	0.5072	**Beta9**	0.5499	0.4944	0.5059
Beta10	0.7388	0.6920	0.7160	**Beta10**	0.7395	0.6916	0.7162
	Data set 7 - 3				**Data set 7 - 4**		
	RR	**LASSO**	**EN**		**RR**	**LASSO**	**EN**
Intercept	1.5168	1.6877	1.6012	**Intercept**	1.5130	1.6825	1.5966
Beta1	0.9205	0.8972	0.9093	**Beta1**	0.9064	0.8792	0.8953
Beta2	0.5447	0.6034	0.5476	**Beta2**	0.5586	0.6763	0.5750
Beta3	0.5706	0.5134	0.5426	**Beta3**	0.5760	0.5228	0.5499
Beta4	0.7286	0.6829	0.7063	**Beta4**	0.7267	0.6811	0.7044
Beta5	0.4246	0.3571	0.3917	**Beta5**	0.4277	0.3624	0.3959
Beta6	0.7222	0.6732	0.6982	**Beta6**	0.7270	0.6807	0.7043
Beta7	0.9518	0.8247	0.8985	**Beta7**	0.9409	0.7880	0.8793
Beta8	0.4088	0.2283	0.3474	**Beta8**	0.4192	0.1728	0.3427
Beta9	0.5375	0.4590	0.4873	**Beta9**	0.5454	0.5109	0.5089
Beta10	0.7397	0.6918	0.7164	**Beta10**	0.7359	0.6893	0.7133
	Data set 7 - 5				**Data set 7 - 6**		
	RR	**LASSO**	**EN**		**RR**	**LASSO**	**EN**
Intercept	1.5096	1.6816	1.5943	**Intercept**	1.5083	1.6823	1.5940
Beta1	0.9171	0.9018	0.9116	**Beta1**	0.9105	0.8686	0.8956
Beta2	0.5611	0.7071	0.5843	**Beta2**	0.5822	0.7478	0.6092
Beta3	0.5740	0.5202	0.5476	**Beta3**	0.5739	0.5197	0.5472
Beta4	0.7314	0.6867	0.7096	**Beta4**	0.7272	0.6801	0.7042
Beta5	0.4279	0.3620	0.3959	**Beta5**	0.4278	0.3622	0.3959
Beta6	0.7222	0.6723	0.6978	**Beta6**	0.7211	0.6713	0.6968

Table A.7 (continued)

Beta7	0.9318	0.7501	0.8593	**Beta7**	0.9491	0.7914	0.8870
Beta8	0.3898	0.1134	0.3065	**Beta8**	0.3866	0.0890	0.2983
Beta9	0.5715	0.5476	0.5365	**Beta9**	0.5506	0.5312	0.5159
Beta10	0.7311	0.6841	0.7083	**Beta10**	0.7379	0.6905	0.7150
	Data set 7 - 7				**Data set 7 - 8**		
	RR	**LASSO**	**EN**		**RR**	**LASSO**	**EN**
Intercept	1.5133	1.6853	1.5981	**Intercept**	1.5003	1.6676	1.5831
Beta1	0.9113	0.8592	0.8914	**Beta1**	0.8904	0.8517	0.8724
Beta2	0.5764	0.7205	0.5993	**Beta2**	0.5371	0.5803	0.5351
Beta3	0.5753	0.5212	0.5488	**Beta3**	0.5726	0.5189	0.5463
Beta4	0.7283	0.6855	0.7073	**Beta4**	0.7292	0.6851	0.7076
Beta5	0.4272	0.3613	0.3951	**Beta5**	0.4279	0.3650	0.3972
Beta6	0.7255	0.6755	0.7011	**Beta6**	0.7226	0.6740	0.6988
Beta7	0.9443	0.8030	0.8876	**Beta7**	0.9750	0.8606	0.9263
Beta8	0.3952	0.1267	0.3138	**Beta8**	0.4225	0.2552	0.3647
Beta9	0.5389	0.5052	0.5006	**Beta9**	0.5641	0.5013	0.5224
Beta10	0.7365	0.6883	0.7131	**Beta10**	0.7411	0.6958	0.7190
	Data set 7 - 9				**Data set 7 - 10**		
	RR	**LASSO**	**EN**		**RR**	**LASSO**	**EN**
Intercept	1.5153	1.6845	1.5989	**Intercept**	1.5153	1.6909	1.6019
Beta1	0.9097	0.8904	0.9002	**Beta1**	0.8876	0.8273	0.8634
Beta2	0.5431	0.6117	0.5477	**Beta2**	0.5578	0.6752	0.5755
Beta3	0.5784	0.5230	0.5512	**Beta3**	0.5752	0.5211	0.5487
Beta4	0.7278	0.6830	0.7060	**Beta4**	0.7277	0.6818	0.7053
Beta5	0.4369	0.3739	0.4061	**Beta5**	0.4236	0.3557	0.3905
Beta6	0.7224	0.6732	0.6983	**Beta6**	0.7282	0.6802	0.7047
Beta7	0.9615	0.8221	0.9039	**Beta7**	0.9749	0.8511	0.9243
Beta8	0.4087	0.2156	0.3448	**Beta8**	0.4094	0.1754	0.3358
Beta9	0.5395	0.4777	0.4954	**Beta9**	0.5438	0.4889	0.4991
Beta10	0.7386	0.6917	0.7158	**Beta10**	0.7372	0.6932	0.7159

Table A.8 Estimated Coefficients of Data Set 8

	Data set 8 - 1				**Data set 8 - 2**		
	RR	**LASSO**	**EN**		**RR**	**LASSO**	**EN**
Intercept	0.7420	1.1728	0.9524	**Intercept**	0.7643	1.1954	0.9770
Beta1	0.6689	0.6228	0.6471	**Beta1**	0.6677	0.6259	0.6479
Beta2	0.8573	0.8015	0.8298	**Beta2**	0.8588	0.7999	0.8296
Beta3	0.4326	0.4943	0.4358	**Beta3**	0.4060	0.4579	0.4260
Beta4	0.7093	0.6388	0.6753	**Beta4**	0.7111	0.6434	0.6783
Beta5	0.2349	0.1615	0.1989	**Beta5**	0.2368	0.1651	0.2019
Beta6	0.8704	0.8579	0.8784	**Beta6**	0.8603	0.8942	0.8714
Beta7	0.6842	0.5486	0.6285	**Beta7**	0.7005	0.5814	0.6530
Beta8	0.9057	0.8335	0.8662	**Beta8**	0.9187	0.7576	0.8528

Table A.8 (continued)

Beta9	0.2743	0.1937	0.2345	**Beta9**	0.2761	0.1980	0.2372
Beta10	0.7391	0.6215	0.6798	**Beta10**	0.7389	0.6666	0.6994
Beta11	0.1560	0.0728	0.1154	**Beta11**	0.1529	0.0700	0.1126
Beta12	-0.0064	.	.	**Beta12**	0.0090	.	.
Beta13	0.4888	0.4296	0.4604	**Beta13**	0.4930	0.4305	0.4629
Beta14	0.1824	0.0955	0.1396	**Beta14**	0.1846	0.0971	0.1417
Beta15	0.2084	0.0311	0.1282	**Beta15**	0.1737	0.0018	0.0846
Beta16	0.0472	.	0.0071	**Beta16**	0.0473	.	0.0073
Beta17	0.5591	0.3479	0.4745	**Beta17**	0.6080	0.4939	0.5573
Beta18	0.6340	0.7528	0.6495	**Beta18**	0.5635	0.5446	0.5411
Beta19	0.9367	0.9081	0.9205	**Beta19**	0.9099	0.8625	0.8849
Beta20	0.6625	0.4248	0.5854	**Beta20**	0.6666	0.5304	0.6120
Beta21	0.5908	0.5432	0.5542	**Beta21**	0.6114	0.5242	0.5647
Beta22	0.1890	0.1025	0.1461	**Beta22**	0.1895	0.1125	0.1510
Beta23	0.6957	0.6394	0.6680	**Beta23**	0.7036	0.6529	0.6786
Beta24	0.1278	0.0367	0.0835	**Beta24**	0.1209	0.0288	0.0767
Beta25	0.1975	0.1283	0.1636	**Beta25**	0.2000	0.1296	0.1655
	Data set 8 - 3				**Data set 8 - 4**		
	RR	**LASSO**	**EN**		**RR**	**LASSO**	**EN**
Intercept	0.7287	1.1632	0.9429	**Intercept**	0.7347	1.1856	0.9598
Beta1	0.6702	0.6222	0.6475	**Beta1**	0.6693	0.6229	0.6476
Beta2	0.8617	0.8055	0.8336	**Beta2**	0.8638	0.8067	0.8358
Beta3	0.4282	0.5029	0.4523	**Beta3**	0.4121	0.4162	0.4144
Beta4	0.7099	0.6408	0.6767	**Beta4**	0.7104	0.6396	0.6759
Beta5	0.2349	0.1559	0.1960	**Beta5**	0.2307	0.1574	0.1949
Beta6	0.8550	0.8100	0.8472	**Beta6**	0.8594	0.8846	0.8738
Beta7	0.7075	0.5980	0.6628	**Beta7**	0.7223	0.6167	0.6763
Beta8	0.9213	0.8352	0.8742	**Beta8**	0.9276	0.8193	0.8761
Beta9	0.2776	0.1986	0.2383	**Beta9**	0.2805	0.2056	0.2436
Beta10	0.7479	0.6603	0.7083	**Beta10**	0.7069	0.5970	0.6582
Beta11	0.1538	0.0677	0.1120	**Beta11**	0.1562	0.0737	0.1158
Beta12	0.0078	.	.	**Beta12**	0.0300	.	.
Beta13	0.4891	0.4297	0.4603	**Beta13**	0.4825	0.4194	0.4522
Beta14	0.1803	0.0920	0.1367	**Beta14**	0.1817	0.0900	0.1363
Beta15	0.1893	0.0001	0.0905	**Beta15**	0.1762	0.0317	0.1020
Beta16	0.0445	.	0.0019	**Beta16**	0.0448	.	0.0021
Beta17	0.6061	0.3916	0.5181	**Beta17**	0.5953	0.3912	0.5086
Beta18	0.6056	0.6602	0.5996	**Beta18**	0.5948	0.6208	0.5879
Beta19	0.9342	0.9673	0.9452	**Beta19**	0.9327	0.9554	0.9367
Beta20	0.6071	0.3864	0.5285	**Beta20**	0.6586	0.4899	0.5910
Beta21	0.6278	0.5669	0.5892	**Beta21**	0.6196	0.5156	0.5641
Beta22	0.1894	0.1071	0.1487	**Beta22**	0.1887	0.1098	0.1498
Beta23	0.7003	0.6475	0.6740	**Beta23**	0.6953	0.6425	0.6693
Beta24	0.1298	0.0360	0.0844	**Beta24**	0.1298	0.0346	0.0839
Beta25	0.1960	0.1253	0.1615	**Beta25**	0.1956	0.1237	0.1605

Table A.8 (continued)

	Data set 8 - 5				Data set 8 - 6		
	RR	LASSO	EN		RR	LASSO	EN
Intercept	0.7310	1.1665	0.9480	Intercept	0.7321	1.1736	0.9511
Beta1	0.6672	0.6219	0.6461	Beta1	0.6745	0.6250	0.6508
Beta2	0.8592	0.7996	0.8300	Beta2	0.8570	0.8031	0.8305
Beta3	0.4094	0.4720	0.4408	Beta3	0.4174	0.5388	0.4672
Beta4	0.7081	0.6396	0.6748	Beta4	0.7065	0.6415	0.6750
Beta5	0.2348	0.1634	0.2000	Beta5	0.2377	0.1662	0.2030
Beta6	0.8958	0.8821	0.8931	Beta6	0.8716	0.8599	0.8733
Beta7	0.6775	0.5360	0.6146	Beta7	0.6870	0.5447	0.6249
Beta8	0.9048	0.8131	0.8545	Beta8	0.9350	0.8214	0.8732
Beta9	0.2777	0.2025	0.2403	Beta9	0.2763	0.1980	0.2373
Beta10	0.7018	0.6234	0.6779	Beta10	0.6936	0.6225	0.6686
Beta11	0.1540	0.0685	0.1122	Beta11	0.1568	0.0723	0.1155
Beta12	0.0514	.	.	Beta12	0.0354	.	.
Beta13	0.4897	0.4328	0.4627	Beta13	0.4902	0.4320	0.4623
Beta14	0.1832	0.0950	0.1394	Beta14	0.1847	0.0974	0.1414
Beta15	0.1818	0.0401	0.1137	Beta15	0.1950	0.0021	0.1067
Beta16	0.0471	.	0.0064	Beta16	0.0481	.	0.0069
Beta17	0.5839	0.3988	0.5064	Beta17	0.6108	0.4425	0.5398
Beta18	0.5759	0.6320	0.5777	Beta18	0.5769	0.6023	0.5632
Beta19	0.9585	0.9486	0.9491	Beta19	0.9255	0.9063	0.9142
Beta20	0.6734	0.4694	0.5956	Beta20	0.6766	0.4886	0.6068
Beta21	0.6217	0.5525	0.5805	Beta21	0.6083	0.5408	0.5679
Beta22	0.1905	0.1150	0.1530	Beta22	0.1882	0.1113	0.1501
Beta23	0.7001	0.6502	0.6753	Beta23	0.6982	0.6403	0.6696
Beta24	0.1320	0.0417	0.0880	Beta24	0.1321	0.0412	0.0882
Beta25	0.1971	0.1294	0.1639	Beta25	0.1996	0.1341	0.1673
	Data set 8 - 7				Data set 8 - 8		
	RR	LASSO	EN		RR	LASSO	EN
Intercept	0.7359	1.1673	0.9498	Intercept	0.7290	1.1807	0.9517
Beta1	0.6692	0.6241	0.6475	Beta1	0.6721	0.6306	0.6525
Beta2	0.8629	0.8038	0.8335	Beta2	0.8601	0.8006	0.8308
Beta3	0.4176	0.4563	0.4207	Beta3	0.4165	0.4745	0.4195
Beta4	0.7040	0.6352	0.6708	Beta4	0.7167	0.6505	0.6844
Beta5	0.2381	0.1687	0.2045	Beta5	0.2376	0.1654	0.2024
Beta6	0.8378	0.7588	0.8225	Beta6	0.8804	0.8358	0.8811
Beta7	0.7087	0.6465	0.6816	Beta7	0.6950	0.5930	0.6504
Beta8	0.9323	0.9265	0.9126	Beta8	0.9248	0.8842	0.8894
Beta9	0.2797	0.2022	0.2413	Beta9	0.2765	0.2004	0.2389
Beta10	0.7359	0.5936	0.6738	Beta10	0.7410	0.5907	0.6719
Beta11	0.1565	0.0719	0.1151	Beta11	0.1547	0.0673	0.1122
Beta12	0.0046	.	.	Beta12	-0.0021	.	.
Beta13	0.4807	0.4220	0.4525	Beta13	0.4925	0.4312	0.4628

Table A.8 (continued)

Beta14	0.1800	0.0924	0.1369	**Beta14**	0.1791	0.0880	0.1345
Beta15	0.1930	0.0002	0.0963	**Beta15**	0.1911	.	0.1038
Beta16	0.0477	.	0.0056	**Beta16**	0.0481	.	0.0073
Beta17	0.6103	0.4315	0.5337	**Beta17**	0.5726	0.3282	0.4715
Beta18	0.5639	0.5574	0.5454	**Beta18**	0.5867	0.6596	0.5890
Beta19	0.9377	0.9396	0.9321	**Beta19**	0.9590	0.9828	0.9633
Beta20	0.6576	0.5082	0.5968	**Beta20**	0.6730	0.4692	0.6030
Beta21	0.6277	0.5366	0.5803	**Beta21**	0.6021	0.5390	0.5627
Beta22	0.1893	0.1096	0.1497	**Beta22**	0.1834	0.0994	0.1422
Beta23	0.6973	0.6437	0.6706	**Beta23**	0.6987	0.6406	0.6700
Beta24	0.1296	0.0405	0.0865	**Beta24**	0.1281	0.0331	0.0819
Beta25	0.1999	0.1291	0.1652	**Beta25**	0.1941	0.1304	0.1632
	Data set 8 - 9				**Data set 8 - 10**		
	RR	**LASSO**	**EN**		**RR**	**LASSO**	**EN**
Intercept	0.7450	1.1855	0.9643	**Intercept**	0.7438	1.1631	0.9522
Beta1	0.6681	0.6224	0.6463	**Beta1**	0.6734	0.6256	0.6507
Beta2	0.8636	0.8084	0.8365	**Beta2**	0.8563	0.7963	0.8269
Beta3	0.4205	0.4857	0.4571	**Beta3**	0.4303	0.4751	0.4558
Beta4	0.7090	0.6420	0.6768	**Beta4**	0.7142	0.6497	0.6829
Beta5	0.2377	0.1630	0.2014	**Beta5**	0.2357	0.1644	0.2007
Beta6	0.8575	0.8227	0.8484	**Beta6**	0.8672	0.8631	0.8621
Beta7	0.7022	0.5913	0.6515	**Beta7**	0.6965	0.5731	0.6439
Beta8	0.9352	0.8627	0.8933	**Beta8**	0.9203	0.8128	0.8745
Beta9	0.2842	0.2108	0.2478	**Beta9**	0.2839	0.2132	0.2487
Beta10	0.7223	0.6189	0.6818	**Beta10**	0.7350	0.6594	0.6989
Beta11	0.1568	0.0719	0.1153	**Beta11**	0.1520	0.0697	0.1119
Beta12	0.0270	.	.	**Beta12**	0.0175	.	.
Beta13	0.4866	0.4215	0.4548	**Beta13**	0.4934	0.4326	0.4640
Beta14	0.1762	0.0849	0.1310	**Beta14**	0.1810	0.0970	0.1396
Beta15	0.1787	.	0.0812	**Beta15**	0.1559	.	0.0687
Beta16	0.0504	.	0.0081	**Beta16**	0.0502	.	0.0105
Beta17	0.5759	0.3722	0.4894	**Beta17**	0.5742	0.3655	0.4854
Beta18	0.5713	0.5664	0.5547	**Beta18**	0.5818	0.6149	0.5727
Beta19	0.9443	0.9718	0.9474	**Beta19**	0.9524	0.9646	0.9536
Beta20	0.6905	0.5590	0.6368	**Beta20**	0.6842	0.4956	0.6153
Beta21	0.5930	0.4881	0.5399	**Beta21**	0.5933	0.5479	0.5631
Beta22	0.1832	0.1027	0.1433	**Beta22**	0.1881	0.1080	0.1486
Beta23	0.6967	0.6451	0.6716	**Beta23**	0.6995	0.6491	0.6746
Beta24	0.1302	0.0388	0.0859	**Beta24**	0.1254	0.0362	0.0823
Beta25	0.1944	0.1233	0.1594	**Beta25**	0.1988	0.1324	0.1663

Table A.9 Estimated Coefficients of Data Set 9

	Data set 9 - 1				Data set 9 - 2		
	RR	**LASSO**	**EN**		**RR**	**LASSO**	**EN**
Intercept	1.2646	7.1459	4.4597	**Intercept**	1.2253	7.1192	4.4119
Beta1	0.5680	1.0517	0.7265	**Beta1**	0.5607	0.7893	0.6544
Beta2	0.5139	0.1418	0.3319	**Beta2**	0.4866	0.1216	0.3095
Beta3	0.2451	.	0.1397	**Beta3**	0.2452	.	0.1144
Beta4	0.8376	0.5935	0.7470	**Beta4**	0.8337	0.7137	0.7841
Beta5	0.5089	0.4281	0.4688	**Beta5**	0.5040	0.4030	0.4596
Beta6	0.9393	0.9267	0.9155	**Beta6**	0.9401	0.9646	0.9422
Beta7	0.4662	0.3863	0.4558	**Beta7**	0.4734	0.3847	0.4522
Beta8	0.7038	0.3823	0.5501	**Beta8**	0.7057	0.4330	0.5747
Beta9	0.4825	0.1511	0.3199	**Beta9**	0.4973	0.1580	0.3334
Beta10	0.5599	0.2333	0.3949	**Beta10**	0.5577	0.2250	0.3899
Beta11	0.8344	0.4995	0.6748	**Beta11**	0.8210	0.4980	0.6659
Beta12	0.6447	0.2347	0.4701	**Beta12**	0.6712	0.3806	0.5337
Beta13	0.2110	.	0.0019	**Beta13**	0.2063	.	0.0058
Beta14	0.9011	1.4929	1.0424	**Beta14**	0.9240	1.6889	1.1274
Beta15	0.6600	0.3469	0.5052	**Beta15**	0.6610	0.3475	0.5083
Beta16	0.4511	0.0314	0.2333	**Beta16**	0.4388	0.0133	0.1923
Beta17	0.0431	.	.	**Beta17**	0.0312	.	.
Beta18	0.4698	0.2252	0.4571	**Beta18**	0.4588	0.1482	0.4177
Beta19	0.3850	0.0209	0.1909	**Beta19**	0.3977	0.0184	0.1955
Beta20	0.9250	0.6971	0.8389	**Beta20**	0.8881	0.5741	0.7725
Beta21	0.1336	.	.	**Beta21**	0.1344	.	.
Beta22	0.6181	0.2300	0.4336	**Beta22**	0.6252	0.2451	0.4443
Beta23	0.5396	0.2247	0.3764	**Beta23**	0.5413	0.2715	0.4079
Beta24	0.4502	0.3313	0.3693	**Beta24**	0.4247	0.3306	0.3538
Beta25	0.8613	0.5543	0.7058	**Beta25**	0.8713	0.5337	0.7056
Beta26	0.8594	0.6100	0.7396	**Beta26**	0.8531	0.6162	0.7376
Beta27	0.3194	0.0266	0.1713	**Beta27**	0.3211	0.0168	0.1596
Beta28	0.8867	0.7813	0.8722	**Beta28**	0.8862	0.7154	0.8455
Beta29	0.1981	.	0.0237	**Beta29**	0.1942	.	0.0373
Beta30	0.6530	0.3317	0.4912	**Beta30**	0.6600	0.3240	0.4914
Beta31	0.0744	.	.	**Beta31**	0.0574	.	.
Beta32	0.2637	.	0.0404	**Beta32**	0.2369	.	0.0156
Beta33	0.7452	0.4334	0.6681	**Beta33**	0.8120	0.5017	0.7410
Beta34	0.5127	0.6702	0.5888	**Beta34**	0.4861	0.5359	0.5043
Beta35	0.7243	0.3270	0.5233	**Beta35**	0.7511	0.3649	0.5639
Beta36	0.8505	0.5503	0.7890	**Beta36**	0.8614	0.5576	0.7744
Beta37	0.8747	0.5698	0.7292	**Beta37**	0.8826	0.5651	0.7313
Beta38	0.1658	.	0.0053	**Beta38**	0.1778	.	0.0137
Beta39	0.8833	0.5055	0.6911	**Beta39**	0.8751	0.4881	0.6764
Beta40	0.1541	.	0.0254	**Beta40**	0.1378	.	0.0374
Beta41	1.0175	1.2147	1.0013	**Beta41**	1.0148	1.2168	1.0262
Beta42	0.1745	.	0.0003	**Beta42**	0.1967	.	0.0064

Table A.9 (continued)

Beta43	0.7151	0.3237	0.5254	**Beta43**	0.7048	0.3115	0.5098
Beta44	0.3727	0.0897	0.2995	**Beta44**	0.3748	0.1171	0.2970
Beta45	0.1006	.	0.0052	**Beta45**	0.1153	.	0.0288
Beta46	0.7710	0.6042	0.7269	**Beta46**	0.7948	0.7294	0.7895
Beta47	0.3837	0.1461	0.2934	**Beta47**	0.3773	0.2984	0.3378
Beta48	0.0873	.	.	**Beta48**	0.0842	.	.
Beta49	0.3969	0.4778	0.3725	**Beta49**	0.3607	0.2500	0.2893
Beta50	0.6571	0.2859	0.4774	**Beta50**	0.6628	0.2671	0.4712
Beta51	0.8592	0.7442	0.7937	**Beta51**	0.8833	0.7233	0.8020
Beta52	0.5339	0.1963	0.3662	**Beta52**	0.5418	0.1664	0.3562
Beta53	0.8533	0.4363	0.6512	**Beta53**	0.8508	0.4626	0.6676
Beta54	0.2765	0.0904	0.2347	**Beta54**	0.2955	0.2333	0.2720
Beta55	0.7927	0.6110	0.6911	**Beta55**	0.7928	0.5793	0.6955
Beta56	0.4659	0.2238	0.4070	**Beta56**	0.4628	0.2396	0.4375
Beta57	0.7681	1.1309	0.8900	**Beta57**	0.7348	1.1271	0.8616
Beta58	0.6768	0.3216	0.5076	**Beta58**	0.6917	0.2725	0.4914
Beta59	0.1227	.	.	**Beta59**	0.1617	.	.
Beta60	0.5797	0.0281	0.3549	**Beta60**	0.5970	0.0343	0.3743
	Data set 9 - 3				**Data set 9 - 4**		
	RR	**LASSO**	**EN**		**RR**	**LASSO**	**EN**
Intercept	1.2776	7.0322	4.3859	**Intercept**	1.2213	6.9677	4.3214
Beta1	0.5637	0.8499	0.6539	**Beta1**	0.5623	0.9206	0.6728
Beta2	0.4994	0.1446	0.3303	**Beta2**	0.5144	0.1096	0.3222
Beta3	0.2506	.	0.1440	**Beta3**	0.2631	.	0.1682
Beta4	0.8564	0.6991	0.8022	**Beta4**	0.8257	0.5976	0.7298
Beta5	0.4907	0.3587	0.4176	**Beta5**	0.4702	0.3686	0.3958
Beta6	0.9536	0.9920	0.9652	**Beta6**	0.9414	0.9265	0.9320
Beta7	0.4320	0.3639	0.4195	**Beta7**	0.4636	0.3721	0.4457
Beta8	0.7267	0.4253	0.5817	**Beta8**	0.7053	0.3946	0.5620
Beta9	0.4799	0.1605	0.3246	**Beta9**	0.4891	0.1375	0.3184
Beta10	0.5634	0.2621	0.4088	**Beta10**	0.5523	0.2211	0.3824
Beta11	0.8331	0.5043	0.6751	**Beta11**	0.8223	0.5323	0.6827
Beta12	0.6375	0.3378	0.5002	**Beta12**	0.6674	0.4194	0.5626
Beta13	0.2019	.	0.0049	**Beta13**	0.2022	.	0.0053
Beta14	0.9276	1.6797	1.1335	**Beta14**	0.9202	1.4867	1.0557
Beta15	0.6596	0.3576	0.5101	**Beta15**	0.6486	0.3579	0.5023
Beta16	0.4475	0.0356	0.2342	**Beta16**	0.4458	0.0342	0.2376
Beta17	0.0373	.	.	**Beta17**	0.0343	.	.
Beta18	0.4715	0.1950	0.4453	**Beta18**	0.4455	0.2328	0.4260
Beta19	0.3955	0.0348	0.2196	**Beta19**	0.4018	0.0391	0.2193
Beta20	0.9047	0.5509	0.7742	**Beta20**	0.8996	0.5796	0.7861
Beta21	0.1336	.	.	**Beta21**	0.1467	.	0.0253
Beta22	0.6327	0.2797	0.4669	**Beta22**	0.6223	0.2845	0.4627
Beta23	0.5418	0.2406	0.3931	**Beta23**	0.5519	0.2698	0.4109

Table A.9 (continued)

Beta24	0.4418	0.3886	0.3889	**Beta24**	0.4370	0.3630	0.3714
Beta25	0.8742	0.5560	0.7159	**Beta25**	0.8649	0.5702	0.7198
Beta26	0.8429	0.5893	0.7190	**Beta26**	0.8575	0.6342	0.7482
Beta27	0.2977	0.0168	0.1449	**Beta27**	0.3098	0.0228	0.1630
Beta28	0.8592	0.6741	0.8101	**Beta28**	0.8688	0.7290	0.8322
Beta29	0.1873	.	0.0153	**Beta29**	0.2012	.	0.0355
Beta30	0.6637	0.3376	0.5031	**Beta30**	0.6576	0.3508	0.5031
Beta31	0.0636	.	.	**Beta31**	0.0784	.	.
Beta32	0.2638	.	0.0428	**Beta32**	0.2473	.	0.0307
Beta33	0.7813	0.4792	0.7043	**Beta33**	0.8093	0.5947	0.7679
Beta34	0.4615	0.4524	0.4740	**Beta34**	0.4699	0.6547	0.5128
Beta35	0.7348	0.3542	0.5441	**Beta35**	0.7385	0.3712	0.5590
Beta36	0.8646	0.5943	0.7858	**Beta36**	0.8622	0.4370	0.7758
Beta37	0.8985	0.6238	0.7692	**Beta37**	0.9001	0.5873	0.7515
Beta38	0.1574	.	0.0032	**Beta38**	0.1610	.	0.0057
Beta39	0.8824	0.5073	0.6885	**Beta39**	0.8985	0.5318	0.7057
Beta40	0.1416	.	0.0306	**Beta40**	0.1549	.	0.0337
Beta41	1.0435	1.2419	1.0574	**Beta41**	1.0291	1.3535	1.0546
Beta42	0.1836	.	0.0013	**Beta42**	0.1893	.	0.0032
Beta43	0.7145	0.3311	0.5265	**Beta43**	0.6990	0.3284	0.5149
Beta44	0.3776	0.1139	0.3052	**Beta44**	0.3483	0.0238	0.2420
Beta45	0.0845	.	0.0064	**Beta45**	0.1190	0.0028	0.0612
Beta46	0.7986	0.7485	0.7920	**Beta46**	0.7887	0.6438	0.7576
Beta47	0.4148	0.2134	0.3291	**Beta47**	0.4248	0.2643	0.3541
Beta48	0.0846	.	.	**Beta48**	0.0878	.	.
Beta49	0.3706	0.4947	0.3577	**Beta49**	0.3528	0.3796	0.3148
Beta50	0.6479	0.2537	0.4573	**Beta50**	0.6577	0.2779	0.4773
Beta51	0.8266	0.6625	0.7535	**Beta51**	0.8586	0.7268	0.7978
Beta52	0.5129	0.1305	0.3247	**Beta52**	0.5418	0.1625	0.3541
Beta53	0.8525	0.4712	0.6645	**Beta53**	0.8545	0.4905	0.6771
Beta54	0.2714	0.0208	0.2179	**Beta54**	0.2724	0.0875	0.2224
Beta55	0.8163	0.7061	0.7396	**Beta55**	0.8089	0.6408	0.7196
Beta56	0.4589	0.1761	0.3853	**Beta56**	0.4399	0.1644	0.3772
Beta57	0.7761	1.1520	0.8938	**Beta57**	0.7857	1.2040	0.9495
Beta58	0.6816	0.3155	0.5064	**Beta58**	0.7036	0.3476	0.5361
Beta59	0.1183	.	.	**Beta59**	0.1529	.	.
Beta60	0.6014	0.0546	0.3844	**Beta60**	0.5642	0.0087	0.3202
	Data set 9 - 5				**Data set 9 - 6**		
	RR	**LASSO**	**EN**		**RR**	**LASSO**	**EN**
Intercept	1.2427	7.1862	4.4488	**Intercept**	1.183596	6.9472	4.3142
Beta1	0.5717	0.9423	0.7113	**Beta1**	0.5490	0.9640	0.6910
Beta2	0.5020	0.1347	0.3236	**Beta2**	0.5057	0.1511	0.3374
Beta3	0.2520	.	0.1060	**Beta3**	0.2389	.	0.1290
Beta4	0.8338	0.6113	0.7597	**Beta4**	0.8795	0.6383	0.7905

Table A.9 (continued)

Beta5	0.4960	0.3113	0.4338	**Beta5**	0.5119	0.4241	0.4676
Beta6	0.9565	1.0292	0.9666	**Beta6**	0.9421	0.9446	0.9299
Beta7	0.4487	0.3588	0.4340	**Beta7**	0.4573	0.4273	0.4692
Beta8	0.6932	0.4196	0.5630	**Beta8**	0.7138	0.4035	0.5711
Beta9	0.4875	0.1291	0.3106	**Beta9**	0.4963	0.1392	0.3200
Beta10	0.5601	0.2360	0.3949	**Beta10**	0.5562	0.2357	0.3954
Beta11	0.8228	0.5263	0.6821	**Beta11**	0.8334	0.4977	0.6706
Beta12	0.6399	0.3316	0.4922	**Beta12**	0.6409	0.2578	0.4677
Beta13	0.1879	.	.	**Beta13**	0.2094	.	0.0073
Beta14	0.9173	1.7340	1.1222	**Beta14**	0.9055	1.4902	1.0514
Beta15	0.6711	0.3793	0.5266	**Beta15**	0.6690	0.3791	0.5276
Beta16	0.4560	0.0434	0.2436	**Beta16**	0.4585	0.0403	0.2395
Beta17	0.0198	.	.	**Beta17**	0.0419	.	.
Beta18	0.4490	0.1538	0.4043	**Beta18**	0.4551	0.2280	0.4382
Beta19	0.3961	0.0210	0.2070	**Beta19**	0.3794	0.0231	0.1856
Beta20	0.9162	0.5220	0.7823	**Beta20**	0.9226	0.6774	0.8294
Beta21	0.1500	.	0.0385	**Beta21**	0.1459	.	0.0165
Beta22	0.6317	0.2430	0.4446	**Beta22**	0.6248	0.2907	0.4697
Beta23	0.5420	0.2160	0.3753	**Beta23**	0.5487	0.2769	0.4130
Beta24	0.4520	0.4261	0.3928	**Beta24**	0.4180	0.3346	0.3520
Beta25	0.8660	0.5217	0.6939	**Beta25**	0.8524	0.5446	0.7002
Beta26	0.8594	0.5901	0.7322	**Beta26**	0.8712	0.6274	0.7520
Beta27	0.3016	0.0089	0.1341	**Beta27**	0.3200	0.0302	0.1730
Beta28	0.8679	0.6639	0.8222	**Beta28**	0.8891	0.7001	0.8402
Beta29	0.2079	.	0.0276	**Beta29**	0.1945	.	0.0190
Beta30	0.6687	0.3463	0.5075	**Beta30**	0.6537	0.2985	0.4776
Beta31	0.0753	.	.	**Beta31**	0.0676	.	.
Beta32	0.2538	.	0.0266	**Beta32**	0.2677	.	0.0486
Beta33	0.7592	0.4645	0.6921	**Beta33**	0.7824	0.5046	0.7107
Beta34	0.4544	0.5082	0.4947	**Beta34**	0.4823	0.5684	0.5332
Beta35	0.7463	0.4022	0.5805	**Beta35**	0.7496	0.4050	0.5785
Beta36	0.8644	0.5955	0.7900	**Beta36**	0.8878	0.6284	0.8171
Beta37	0.8816	0.5656	0.7347	**Beta37**	0.8759	0.5655	0.7262
Beta38	0.1656	.	0.0031	**Beta38**	0.1533	.	0.0061
Beta39	0.8833	0.4324	0.6543	**Beta39**	0.9032	0.5387	0.7162
Beta40	0.1723	.	0.0501	**Beta40**	0.1589	.	0.0261
Beta41	1.0177	1.1848	1.0075	**Beta41**	1.0159	1.1626	1.0024
Beta42	0.1772	.	0.0009	**Beta42**	0.1827	.	0.0008
Beta43	0.7242	0.3031	0.5200	**Beta43**	0.7001	0.3261	0.5138
Beta44	0.3598	0.0864	0.2887	**Beta44**	0.3647	0.1546	0.3209
Beta45	0.1066	.	0.0070	**Beta45**	0.0985	.	0.0136
Beta46	0.8069	0.7183	0.7855	**Beta46**	0.7773	0.6386	0.7371
Beta47	0.4290	0.3484	0.3933	**Beta47**	0.4319	0.2546	0.3650
Beta48	0.0862	.	.	**Beta48**	0.1079	.	.

Table A.9 (continued)

Beta49	0.3722	0.3512	0.3219	**Beta49**	0.3567	0.3348	0.3076
Beta50	0.6598	0.2644	0.4707	**Beta50**	0.6429	0.2708	0.4603
Beta51	0.8398	0.6691	0.7537	**Beta51**	0.8225	0.7502	0.7752
Beta52	0.5343	0.1641	0.3483	**Beta52**	0.5363	0.1617	0.3500
Beta53	0.8603	0.5113	0.6904	**Beta53**	0.8709	0.5119	0.6986
Beta54	0.2919	0.1180	0.2376	**Beta54**	0.2958	0.1523	0.2446
Beta55	0.7795	0.5778	0.6791	**Beta55**	0.8128	0.6084	0.7123
Beta56	0.4695	0.2032	0.4002	**Beta56**	0.4393	0.2610	0.4230
Beta57	0.7686	1.1544	0.8957	**Beta57**	0.7775	1.0534	0.8643
Beta58	0.6940	0.2818	0.4986	**Beta58**	0.6954	0.3406	0.5256
Beta59	0.1127	.	.	**Beta59**	0.1569	.	.
Beta60	0.5829	0.0294	0.3618	**Beta60**	0.5745	0.0805	0.3749
	Data set 9 - 7				**Data set 9 - 8**		
	RR	**LASSO**	**EN**		**RR**	**LASSO**	**EN**
Intercept	1.1884	6.9247	4.2846	**Intercept**	1.2622	6.9739	4.3301
Beta1	0.5765	0.9931	0.7201	**Beta1**	0.5747	0.9809	0.7079
Beta2	0.5102	0.1249	0.3239	**Beta2**	0.5110	0.1259	0.3233
Beta3	0.2273	.	0.1049	**Beta3**	0.2351	.	0.1454
Beta4	0.8412	0.6227	0.7643	**Beta4**	0.8447	0.6063	0.7557
Beta5	0.5092	0.3094	0.4383	**Beta5**	0.4969	0.3578	0.4174
Beta6	0.9381	1.0587	0.9679	**Beta6**	0.9708	1.0320	0.9846
Beta7	0.4798	0.3602	0.4536	**Beta7**	0.4589	0.3730	0.4428
Beta8	0.7043	0.3980	0.5556	**Beta8**	0.7119	0.4153	0.5701
Beta9	0.4934	0.1456	0.3227	**Beta9**	0.4799	0.1231	0.3065
Beta10	0.5433	0.2408	0.3932	**Beta10**	0.5621	0.2683	0.4182
Beta11	0.8298	0.5492	0.6931	**Beta11**	0.8153	0.5234	0.6758
Beta12	0.6498	0.2755	0.4780	**Beta12**	0.6368	0.3197	0.4954
Beta13	0.1960	.	0.0012	**Beta13**	0.2050	.	0.0042
Beta14	0.9075	1.6336	1.0851	**Beta14**	0.9409	1.7002	1.1401
Beta15	0.6673	0.3715	0.5211	**Beta15**	0.6504	0.3580	0.5052
Beta16	0.4554	0.0421	0.2443	**Beta16**	0.4449	0.0327	0.2278
Beta17	0.0411	.	.	**Beta17**	0.0279	.	.
Beta18	0.4327	0.2205	0.4330	**Beta18**	0.4450	0.1090	0.3908
Beta19	0.3897	0.0307	0.2002	**Beta19**	0.4009	0.0346	0.2143
Beta20	0.9164	0.5339	0.7698	**Beta20**	0.8950	0.5455	0.7650
Beta21	0.1463	.	0.0193	**Beta21**	0.1395	.	0.0064
Beta22	0.6272	0.2679	0.4560	**Beta22**	0.6246	0.2786	0.4616
Beta23	0.5538	0.2578	0.4035	**Beta23**	0.5427	0.2492	0.3944
Beta24	0.4478	0.4210	0.4046	**Beta24**	0.4455	0.4225	0.4098
Beta25	0.8648	0.5400	0.7022	**Beta25**	0.8550	0.5402	0.6998
Beta26	0.8627	0.6198	0.7478	**Beta26**	0.8403	0.6068	0.7305
Beta27	0.3128	0.0205	0.1556	**Beta27**	0.3226	0.0415	0.1792
Beta28	0.8707	0.6562	0.8109	**Beta28**	0.8548	0.6343	0.7950
Beta29	0.1872	.	0.0337	**Beta29**	0.1857	.	0.0351

Table A.9 (continued)

Beta30	0.6489	0.3558	0.5010	**Beta30**	0.6572	0.3485	0.5016
Beta31	0.0797	.	.	**Beta31**	0.0731	.	.
Beta32	0.2686	.	0.0504	**Beta32**	0.2639	.	0.0536
Beta33	0.7989	0.4814	0.7219	**Beta33**	0.7905	0.5072	0.7248
Beta34	0.4666	0.6417	0.5194	**Beta34**	0.4494	0.5677	0.4705
Beta35	0.7491	0.3523	0.5563	**Beta35**	0.7499	0.4096	0.5796
Beta36	0.8361	0.4795	0.7476	**Beta36**	0.8337	0.4854	0.7448
Beta37	0.8924	0.5878	0.7475	**Beta37**	0.8815	0.5887	0.7422
Beta38	0.1462	.	.	**Beta38**	0.1617	.	0.0074
Beta39	0.8873	0.5440	0.7131	**Beta39**	0.8869	0.5016	0.6872
Beta40	0.1382	0.0150	0.0636	**Beta40**	0.1918	.	0.0728
Beta41	1.0409	1.3259	1.0741	**Beta41**	1.0339	1.2637	1.0524
Beta42	0.1808	.	0.0007	**Beta42**	0.1968	.	0.0122
Beta43	0.7162	0.3180	0.5214	**Beta43**	0.7053	0.3430	0.5273
Beta44	0.3724	0.0845	0.2874	**Beta44**	0.3587	0.1060	0.2798
Beta45	0.1123	.	0.0238	**Beta45**	0.1196	.	0.0454
Beta46	0.8234	0.6466	0.7682	**Beta46**	0.7716	0.7081	0.7758
Beta47	0.4071	0.1932	0.3191	**Beta47**	0.4299	0.3068	0.3669
Beta48	0.0799	.	.	**Beta48**	0.0939	.	.
Beta49	0.3681	0.4537	0.3532	**Beta49**	0.3620	0.2342	0.2849
Beta50	0.6553	0.2873	0.4753	**Beta50**	0.6471	0.2514	0.4574
Beta51	0.8609	0.7797	0.8147	**Beta51**	0.8315	0.7479	0.7854
Beta52	0.5472	0.1910	0.3729	**Beta52**	0.5254	0.1574	0.3411
Beta53	0.8655	0.4805	0.6767	**Beta53**	0.8482	0.4648	0.6620
Beta54	0.2893	0.0840	0.2344	**Beta54**	0.2717	0.2202	0.2534
Beta55	0.7856	0.6262	0.7008	**Beta55**	0.8157	0.6335	0.7350
Beta56	0.4726	0.1675	0.3904	**Beta56**	0.4724	0.1848	0.3953
Beta57	0.7772	1.1897	0.9285	**Beta57**	0.7393	1.1202	0.8670
Beta58	0.7075	0.3429	0.5336	**Beta58**	0.6934	0.3092	0.5098
Beta59	0.1321	.	.	**Beta59**	0.1265	.	.
Beta60	0.5846	0.0267	0.3509	**Beta60**	0.6220	0.0938	0.4123
	Data set 9 - 9				**Data set 9 - 10**		
	RR	**LASSO**	**EN**		**RR**	**LASSO**	**EN**
Intercept	1.2567	7.0546	4.3994	**Intercept**	1.2519	7.1667	4.4720
Beta1	0.5499	0.7321	0.6168	**Beta1**	0.5692	0.9000	0.7005
Beta2	0.5090	0.1300	0.3250	**Beta2**	0.5192	0.1545	0.3429
Beta3	0.2220	.	0.0997	**Beta3**	0.2391	.	0.0880
Beta4	0.8600	0.7554	0.8258	**Beta4**	0.8523	0.5992	0.7575
Beta5	0.5001	0.3948	0.4668	**Beta5**	0.4967	0.4060	0.4890
Beta6	0.9453	0.9596	0.9381	**Beta6**	0.9625	1.0021	0.9630
Beta7	0.4731	0.4249	0.4678	**Beta7**	0.4481	0.4042	0.4464
Beta8	0.7195	0.4369	0.5862	**Beta8**	0.7101	0.4059	0.5623
Beta9	0.4809	0.1316	0.3111	**Beta9**	0.4847	0.0904	0.2905
Beta10	0.5374	0.2141	0.3740	**Beta10**	0.5423	0.2238	0.3819

Table A.9 (continued)

Beta11	0.8288	0.5372	0.6876	**Beta11**	0.8312	0.4655	0.6504
Beta12	0.6615	0.3853	0.5255	**Beta12**	0.6312	0.2882	0.4663
Beta13	0.2085	.	0.0024	**Beta13**	0.2067	.	0.0077
Beta14	0.9334	1.5993	1.1061	**Beta14**	0.9233	1.6988	1.1191
Beta15	0.6534	0.3498	0.5054	**Beta15**	0.6755	0.3787	0.5278
Beta16	0.4560	0.0428	0.2430	**Beta16**	0.4354	0.0247	0.2154
Beta17	0.0233	.	.	**Beta17**	0.0442	.	.
Beta18	0.4328	0.1230	0.3848	**Beta18**	0.4661	0.1878	0.4418
Beta19	0.3854	0.0230	0.1952	**Beta19**	0.3889	0.0229	0.1948
Beta20	0.9214	0.5813	0.7870	**Beta20**	0.9302	0.5593	0.7897
Beta21	0.1227	.	.	**Beta21**	0.1344	.	.
Beta22	0.6284	0.2726	0.4611	**Beta22**	0.6310	0.3007	0.4767
Beta23	0.5561	0.2831	0.4178	**Beta23**	0.5472	0.2480	0.3954
Beta24	0.4433	0.3251	0.3591	**Beta24**	0.4443	0.3841	0.3845
Beta25	0.8721	0.5390	0.7083	**Beta25**	0.8635	0.5603	0.7149
Beta26	0.8705	0.6021	0.7377	**Beta26**	0.8635	0.6014	0.7356
Beta27	0.2999	0.0115	0.1361	**Beta27**	0.2952	0.0137	0.1398
Beta28	0.8614	0.7303	0.8363	**Beta28**	0.8423	0.6315	0.7870
Beta29	0.2019	.	0.0362	**Beta29**	0.1855	.	0.0129
Beta30	0.6690	0.3470	0.5137	**Beta30**	0.6604	0.3392	0.5010
Beta31	0.0682	.	.	**Beta31**	0.0726	.	.
Beta32	0.2494	.	0.0282	**Beta32**	0.2593	.	0.0308
Beta33	0.7888	0.5740	0.7570	**Beta33**	0.7647	0.4460	0.6974
Beta34	0.4888	0.5128	0.5179	**Beta34**	0.4783	0.6320	0.5408
Beta35	0.7342	0.3527	0.5445	**Beta35**	0.7299	0.3612	0.5501
Beta36	0.8547	0.5671	0.7740	**Beta36**	0.8666	0.4799	0.7793
Beta37	0.8816	0.5672	0.7329	**Beta37**	0.8866	0.5739	0.7357
Beta38	0.1661	.	0.0092	**Beta38**	0.1671	.	0.0056
Beta39	0.8885	0.5270	0.7047	**Beta39**	0.8807	0.5068	0.6921
Beta40	0.1640	.	0.0321	**Beta40**	0.1616	.	0.0310
Beta41	1.0038	1.1959	1.0024	**Beta41**	1.0166	1.3164	1.0351
Beta42	0.1741	.	0.0025	**Beta42**	0.1859	.	.
Beta43	0.7014	0.2845	0.4931	**Beta43**	0.7179	0.2876	0.5038
Beta44	0.3868	0.1061	0.3148	**Beta44**	0.3563	0.0670	0.2905
Beta45	0.0848	.	0.0031	**Beta45**	0.1094	.	0.0108
Beta46	0.7956	0.7429	0.7916	**Beta46**	0.7882	0.6167	0.7418
Beta47	0.4469	0.2263	0.3610	**Beta47**	0.4421	0.2654	0.3590
Beta48	0.0971	.	.	**Beta48**	0.0842	.	.
Beta49	0.3536	0.5334	0.3670	**Beta49**	0.3592	0.3104	0.3053
Beta50	0.6450	0.2616	0.4604	**Beta50**	0.6445	0.2330	0.4449
Beta51	0.8229	0.6957	0.7562	**Beta51**	0.8294	0.7271	0.7748
Beta52	0.5157	0.1974	0.3592	**Beta52**	0.5195	0.1231	0.3205
Beta53	0.8549	0.4601	0.6601	**Beta53**	0.8663	0.4861	0.6789
Beta54	0.2779	0.0491	0.2311	**Beta54**	0.2817	0.2095	0.2632

Table A.9 (continued)

Beta55	0.8052	0.6250	0.6994	**Beta55**	0.8054	0.5711	0.6925
Beta56	0.4658	0.2058	0.4038	**Beta56**	0.4345	0.1924	0.3892
Beta57	0.7487	1.1425	0.9069	**Beta57**	0.7737	1.1287	0.8905
Beta58	0.6990	0.3101	0.5122	**Beta58**	0.6912	0.3370	0.5205
Beta59	0.1457	.	.	**Beta59**	0.1494	.	.
Beta60	0.5966	0.0217	0.3411	**Beta60**	0.5962	0.0970	0.4040

APPENDIX B

MOORA Results of Simulated Data Sets

Table B.1 MOORA Results of Data Set 1

		Score	Rank
Test	**RR**	8.7647	1
	LASSO	8.612041	3
	EN	8.638623	2
Train		**Score**	**Rank**
	RR	8.12093	3
	LASSO	8.773194	2
	EN	9.119149	1
Full Data		**Score**	**Rank**
	RR	7.862121	3
	LASSO	9.083433	2
	EN	9.091836	1

Table B.2 MOORA Results of Data Set 2

		Score	Rank
Test	**RR**	13.771855	1
	LASSO	5.462566	3
	EN	5.815971	2
Train		**Score**	**Rank**
	RR	8.283848	3
	LASSO	8.421292	2
	EN	9.325311	1
Full Data		**Score**	**Rank**
	RR	7.878609	3
	LASSO	9.171391	1
	EN	9.007915	2

Table B.3 MOORA Results of Data Set 3

		Score	Rank
Test		Score	Rank
	RR	15.220595	1
	LASSO	6.446754	3
	EN	7.647962	2
Train		Score	Rank
	RR	6.851033	3
	LASSO	8.863414	2
	EN	11.103042	1
Full Data		Score	Rank
	RR	6.604379	3
	LASSO	10.198866	1
	EN	10.017488	2

Table B.4 MOORA Results of Data Set 4

		Score	Rank
Test		Score	Rank
	RR	8.667487	2
	LASSO	8.513286	3
	EN	8.802505	1
Train		Score	Rank
	RR	8.414642	3
	LASSO	8.646648	2
	EN	8.922227	1
Full Data		Score	Rank
	RR	7.608623	3
	LASSO	9.239001	1
	EN	9.22954	2

Table B.5 MOORA Results of Data Set 5

		Score	Rank
Test		Score	Rank
	RR	8.002183	3
	LASSO	8.77642	2
	EN	9.256926	1
Train		Score	Rank
	RR	7.818878	3
	LASSO	8.932728	2
	EN	9.322571	1
Full Data		Score	Rank
	RR	7.598547	3
	LASSO	9.240515	2
	EN	9.279991	1

Table B.6 MOORA Results of Data Set 6

		Score	Rank
Test	**RR**	8.684421	2
	LASSO	8.15777	3
	EN	9.184664	1
Train		**Score**	**Rank**
	RR	8.190546	3
	LASSO	8.437489	2
	EN	9.42501	1
Full Data		**Score**	**Rank**
	RR	7.674025	3
	LASSO	9.218402	2
	EN	9.228707	1

Table B.7 MOORA Results of Data Set 7

		Score	Rank
Test	**RR**	7.804337	3
	LASSO	9.035554	2
	EN	9.223794	1
Train		**Score**	**Rank**
	RR	7.784267	3
	LASSO	9.058188	2
	EN	9.226598	1
Full Data		**Score**	**Rank**
	RR	7.685282	3
	LASSO	9.202392	1
	EN	9.196652	2

Table B.8 MOORA Results of Data Set 8

		Score	Rank
Test	**RR**	7.887041	3
	LASSO	8.901733	2
	EN	9.269515	1
Train		**Score**	**Rank**
	RR	7.833082	3
	LASSO	8.939146	2
	EN	9.299992	1
Full Data		**Score**	**Rank**
	RR	7.659987	3
	LASSO	9.225183	1
	EN	9.220518	2

Table B.9 MOORA Results of Data Set 9

		Score	Rank
Test		**Score**	**Rank**
	RR	8.454952	2
	LASSO	8.39378	3
	EN	9.159743	1
Train		**Score**	**Rank**
	RR	8.29093	3
	LASSO	8.458219	2
	EN	9.276832	1
Full Data		**Score**	**Rank**
	RR	7.856075	3
	LASSO	9.106182	1
	EN	9.104312	2

APPENDIX C

Estimated Coefficients of Real World Data Sets

Table C.1 Estimated Coefficients of Real World Data Set 1

	Data Set 1		
	RR	**LASSO**	**EN**
Intercept	-0.0096	0.0341	0.0161
Beta1	0.1580	0.2018	0.2150
Beta2	0.1060	0.0601	0.0875
Beta3	0.1148	0.0678	0.0970
Beta4	0.3558	0.3923	0.3791
Beta5	0.2639	0.2726	0.2758
Beta6	0.0262	.	.
Beta7	0.0519	.	.
Beta8	0.0540	.	0.0005
Beta9	0.0010	.	.

Table C.2 Estimated Coefficients of Real World Data Set 2

	Data Set 2		
	RR	**LASSO**	**EN**
Intercept	-0.0075	-0.0274	-0.0389
Beta1	-0.2052	.	.
Beta2	0.2337	0.0976	0.1113
Beta3	-0.0519	.	.
Beta4	-0.0077	.	.
Beta5	0.1377	0.0301	0.0368
Beta6	0.1023	0.0745	0.0717
Beta7	-0.1418	.	.
Beta8	0.0287	.	.
Beta9	0.3528	0.3897	0.2855
Beta10	-0.0099	.	.
Beta11	0.2632	0.0085	0.1448
Beta12	0.1947	0.1454	0.1373
Beta13	-0.0763	.	.
Beta14	0.2533	0.2199	0.2349
Beta15	0.0477	.	.
Beta16	-0.0659	.	-0.0091

Table C.3 Estimated Coefficients of Real World Data Set 3

Data Set 3			
	RR	**LASSO**	**EN**
Intercept	-0.0706	0.0052	0.0254
Beta1	0.0525	.	.
Beta2	0.0543	.	.
Beta3	-0.0117	.	.
Beta4	0.0208	.	.
Beta5	0.0244	.	.
Beta6	-0.0492	.	.
Beta7	0.1054	0.1099	0.1308
Beta8	0.1050	0.0270	0.0551
Beta9	-0.0841	.	.
Beta10	0.0651	.	.
Beta11	-0.0856	.	.
Beta12	0.0669	.	0.0104
Beta13	-0.0023	.	.
Beta14	0.1018	.	.
Beta15	0.0070	.	.
Beta16	0.0091	.	.
Beta17	0.0176	.	.
Beta18	0.0545	0.0089	0.0234
Beta19	0.0116	.	.
Beta20	-0.0360	.	.
Beta21	-0.0290	.	.
Beta22	0.0341	.	.
Beta23	0.1573	.	.
Beta24	-0.0861	.	.
Beta25	0.0379	.	.
Beta26	-0.0009	.	.
Beta27	0.0746	.	.
Beta28	0.0929	0.1095	0.1006
Beta29	-0.0950	.	-0.0023
Beta30	0.0062	.	.
Beta31	0.0460	.	.
Beta32	-0.0913	.	-0.0085
Beta33	-0.0127	.	.
Beta34	-0.0630	.	-0.0315
Beta35	0.0038	.	.
Beta36	-0.0255	.	.
Beta37	-0.0710	.	.
Beta38	-0.0154	.	.
Beta39	0.1112	.	.
Beta40	-0.0071	.	.
Beta41	-0.0733	.	.
Beta42	0.0791	.	.

Table C.3 (continued)

Beta43	0.0775	.	.
Beta44	-0.0332	-0.1159	-0.0836
Beta45	-0.2118	-0.1179	-0.1869
Beta46	-0.0760	.	.
Beta47	-0.1597	.	-0.0142
Beta48	-0.0892	.	.
Beta49	-0.1010	.	.
Beta50	0.0960	.	.
Beta51	-0.0175	.	.
Beta52	0.0452	.	.
Beta53	0.0049	.	.
Beta54	-0.0253	.	.
Beta55	0.0056	.	.
Beta56	0.0095	.	.
Beta57	0.0860	.	.
Beta58	-0.0358	.	.
Beta59	0.0611	.	.
Beta60	-0.0030	.	.
Beta61	-0.0901	-0.0202	-0.0688
Beta62	0.0811	.	.
Beta63	0.3295	0.0429	0.2538
Beta64	-0.0967	.	.
Beta65	-0.2791	.	-0.1446
Beta66	0.0477	.	.
Beta67	0.0605	.	.
Beta68	0.0120	.	.
Beta69	0.0069	.	.
Beta70	0.0025	.	.
Beta71	0.1041	0.2317	0.1419
Beta72	0.0010	-0.000050	-0.0005
Beta73	-0.0244	.	.
Beta74	0.0071	.	.
Beta75	-0.0180	.	.
Beta76	0.0051	.	.
Beta77	0.0185	.	.
Beta78	0.0128	.	.
Beta79	0.1009	.	.
Beta80	0.0032	.	.
Beta81	-0.2108	-0.1035	-0.1609

APPENDIX D

MOORA Results of Real World Data Sets

Table D.1 MOORA Results of Data Set 1

		Score	Rank
Test	RR	7.37906	2
	LASSO	5.500236	3
	EN	22.77893	1
		Score	Rank
Train	RR	4.760962	3
	LASSO	9.260718	2
	EN	24.59865	1
		Score	Rank
Full Data	RR	1.96384	3
	LASSO	282.0161	1
	EN	257.5435	2

Table D.2 MOORA Results of Data Set 2

		Score	Rank
Test	RR	9.513655	1
	LASSO	7.76336	3
	EN	8.573434	2
		Score	Rank
Train	RR	11.3907	1
	LASSO	6.948819	3
	EN	8.047779	2
		Score	Rank
Full Data	RR	7.786487	3
	LASSO	9.185423	1
	EN	9.068301	2

Table D.3 MOORA Results of Data Set 3

		Score	Rank
Test		**Score**	**Rank**
	RR	14.31422	1
	LASSO	5.723277	3
	EN	8.088001	2
Train		**Score**	**Rank**
	RR	14.21771	1
	LASSO	5.736907	3
	EN	8.132834	2
Full Data		**Score**	**Rank**
	RR	5.966259	3
	LASSO	10.5674	1
	EN	10.51945	2