DOKUZ EYLÜL UNIVERSITY GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

MODEL SELECTION METHODS FOR MULTIVARIATE LINEAR PARTIAL LEAST SQUARES REGRESSION

by

Elif BULUT

March, 2010

İZMİR

MODEL SELECTION METHODS FOR MULTIVARIATE LINEAR PARTIAL LEAST SQUARES REGRESSION

A Thesis Submitted to the
Graduate School of Natural and Applied Sciences of Dokuz Eylül University in Partial Fulfillment of the Requirements for the Degree of Doctor of Philosophy in Statistics Program

by
Elif BULUT

March, 2010
İZMİR

Ph.D. THESIS EXAMINATON RESULT FORM

We have read the thesis entitled **"MODEL SELECTION METHODS FOR MULTIVARIATE LINEAR PARTIAL LEAST SQUARES REGRESSION"** completed by **ELİF BULUT** under supervision of **PROF. DR. SERDAR KURT** and we certify that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Doctor of Philosophy.

Prof. Dr. Serdar KURT

Supervisor

Prof. Dr. Gül ERGÖR

Thesis Committee Member

Assist. Prof. Dr. Ali Rıza FİRUZAN

Thesis Committee Member

Prof. Dr. Aydın ERAR

Examining Committee Member

Assoc. Prof. Dr. Ali Kemal ŞEHİRLİOĞLU

Examining Committee Member

Prof. Dr. Mustafa SABUNCU
Director
Graduate School of Natural and Applied Sciences

ACKNOWLEDGEMENTS

I express my deepest gratitude to Prof. Dr. Serdar KURT for his valuable guidance and insightful comments, and warm support throughout the research. I would not have started my doctorate process and come to this point without him. He has helped me to overcome my problems sincerely.

I would like to extend my gratitude to Prof. Dr. Gül ERGÖR and Assist. Prof. Dr. Esin FİRUZAN for spending their precious time and their valuable contribution in the thesis committee and to Assist. Prof. Dr. Aylin ALIN for her helpful suggestions during the study.

I owe special thanks to Research Assistant Dr. Özlem GÜRÜNLÜ ALMA in her support to finish this dissertation and for her friendship.

I would also like to thank to Research Assistants Pervin BAYLAN, Dr. Özgül VUPA, and H. Okan İŞGÜDER for their friendship, encouragement and help throughout my Ph. D process.

Finally, I am grateful to my family, whom I am proud of, for their struggle, support and patience to see me in this position.

Elif BULUT

MODEL SELECTION METHODS FOR MULTIVARIATE LINEAR PARTIAL LEAST SQUARES REGRESSION

ABSTRACT

Having large numbers of predictor variables or having more predictor variables than the number of observations is a serious problem in regression analysis. When a data set contains many predictor variables, multicollinearity can become an issue. Multicollinearity arises when predictor variables measure the same concept or when there is a linear relationship among them. These problems can cause high degrees of correlation and violate the assumption of Ordinary Least Square Analysis. As a result, it causes poor estimates of parameter estimation in regression analysis. A possible solution to this problem is a statistical method called 'Partial Least Squares Regression'. PLSR allows for the study of regression in many situations that Multiple Linear Regression does not.

In this thesis, PLSR has been studied in the analysis of obtaining the number of new predictor variables called 'latent variables'. After obtaining the latent variables, this thesis is concerned with analyzing how many of these latent variables are the most relevant for describing the variability of predictor and response variables. Some model selection methods, such as two of the Multivariate Akaike Information Criterion which are studied by Bozdogan and Bedrick respectively, use PRESS values obtained from k-fold cross validation and Wold's R criterion to obtain the optimum number of latent variables. The simulation study presented in this thesis has been performed to compare the performance of these criteria. The simulation results of MAIC, PRESS and Wold's R were obtained from different number of observations and different numbers of predictor variables. These results show that for small-sized design matrices, all criteria achieved the true number of latent variables. However, the results for the other-sized design matrices varied greatly and they consistently showed different numbers of latent variables. The whole analysis, including all simulations and calculations, were done using MATLAB statistical program.

Keywords: Partial Least Squares, Partial Least Squares Regression (PLSR), Model Selection Methods, Multivariate Akaike Information Criterion (MAIC), Predicted Residual Sum of Squares (PRESS), Cross-validation.

ÇOK DEĞİŞKENLİ DOĞRUSAL KISMİ EN KÜÇÜK KARELER REGRESYONU İÇİN MODEL SEÇME YÖNTEMLERİ

ÖZ

Çok sayıda açıklayıcı değişkene veya gözlem sayısından daha fazla sayıda açıklayıcı değişkene sahip olmak regresyon analizinde ciddi bir problemdir. Veri seti birçok açıklayıcı değişken içerdiğinde çoklu doğrusal bağlantıdan söz edilebilir. Çoklu doğrusal bağlantı açıklayıcı değişkenlerin aynı kavramı ölçmelerinde veya açıklayıcı değişkenler arasında doğrusal bir bağıntı olması durumunda ortaya çıkmaktadır. Her iki durum da Sıradan En Küçük Kareler analizinin varsayımlarından sapmaya neden olmakta ve regresyon analizinde zayıf parametre tahminlerine yol açmaktadır. İstatistiksel bir yöntem olan Kısmi En Küçük Kareler Regresyonu, çoklu doğrusal bağlantı probleminin çözüm yollarından birisi olup, Çoklu Doğrusal Regresyon analizinin çalışmadığı bir çok durumda çalışma imkanı sağlamaktadır.

Bu tezde, gizli değişken denilen yeni açıklayıcı değişkenlerin sayısının saptanmasında Kısmi En Küçük Kareler Regresyon analizi çalışılmıştır. Gizli değişkenlerin saptanmasından sonra, bu değişkenlerden kaç tanesinin hem açıklayıcı hem de bağımlı değişkendeki değişimi açıklamada en ilgili olduğunun saptanması ise bu tezin amacını oluşturmaktadır. Gizli değişkenlerin optimum sayısının saptanmasında model seçme yöntemlerinden olan Bozdoğan ve Bedrick tarafından çalışılan iki çoklu Akaike Bilgi Kriteri, k blok çapraz geçerlilik ve PRESS değerleri ve Wold's R kriteri kullanılmıştır. Bu kriterlerin performansının karşılaştırılmasında bir simulasyon çalışması yapılmıştır. Simülasyon sonuçları her bir kriter için farklı sayıda gözlem genişliği ve farklı sayıda açıklayıcı değişken için verilmiştir. Sonuçlar, dizayn matrislerinden en küçüğü için kriterlerin gizli değişken sayısı için doğru sayıyı bulduğunu fakat diğer dizayn matrisleri için farklı sonuçlar verdiğini göstermektedir.

Simulasyon ve analizler MATLAB istatistik paket programında yapılmıştır.

Anahtar Sözcükler: Kısmi En Küçük Kareler, Kısmi En Küçük Kareler Regresyonu, Model Seçme Yöntemleri, Çok Değişkenli Akaike Bilgi Kriteri, Çapraz-Geçerlilik.

CONTENTS

CHAPTER ONE

INTRODUCTION

Regression analysis is commonly used as a statistical tool for analyzing the relationship among variables. Such analyses are used widely in social, behavioral and physical sciences. In statistics, regression analysis includes any techniques employed for modeling and analyzing several variables. Regression analysis is concerned with the study of the dependent variable and one or more predictor variables to construct a model that represents the relationship between these variables, the statistical analysis can be used for prediction, hypothesis testing and modeling of causal relationships. These uses of analysis depend intensively on some assumptions that must be satisfied. A failure to provide any one of these assumptions can cause a misuse of regression. This can result in a fit model that becomes a critique model.

An assumption which is the subject of this thesis and is generally considered to be a problem in regression analyses, is the dependence of the predictor variables which have linear relationship with each other. This is called multicollinearity. Multicollinearity can have severe effects on the estimation of parameters and variables selection techniques.

Various methods exist to detect multicollinearity. The most commonly used ones are Ridge Regression (RR), Principal Component Regression (PCR) and Partial Least Squares Regression (PLSR). These methods are powerful multivariate statistical tools that are widely used in quantitative analysis to overcome problems of collinearity and interactions. PLSR is a multivariate data analysis method which works with several response variables and several predictor variables. It was first studied by Herman Wold at the beginning of the 1970's in Econometrics. Soon after his son Svante Wold extended this method to Chemometrics. It intends to find the latent variables, which are the linear combinations of predictor variables, have no linear relationships among them, and model the response variables best. PLSR can be

used with many data sets that have multicollinearity and many predictor variables which are more than the number of observations. It makes a dimensional reduction by using singular value decomposition or eigenvalue decomposition. Following the dimensional reduction some methods are used to obtain the latent variables which are the most relevant variables describing the response variables. These methods are called model selection criteria. Few of these criteria are Predicted Residual Sum of Squares (PRESS), NORMPRESS, Wold's R and Akaike Information Criterion.

The purpose of this thesis is to examine PLSR and find the latent variables by using model selection criteria and to support this study with a simulation application.

The simulation study was formed in the following steps. First, data were generated according to PLS assumptions. Then MATLAB code for k fold cross-validation was written and PRESS values were obtained. Afterwards, Wold's R criterion was calculated in terms of PRESS. Additionally two different forms of Multivariate Akaike Criteria from Bedrick and Bozdogan were also calculated. Finally comparison of these model selection criteria were made according to their performance in order to obtain the optimum number of latent variables.

This thesis contains six chapters. In Chapter One, a short description of the study is given. Chapter Two introduces multiple regression analysis, multicollinearity problem, Principal Component Analysis, Principal Component Regression and Partial Least Squares Regression. In Chapter Three, PLSR is explained in detail. Chapter Four provides data splitting and model selection criteria as well as a comparison of these methods that is supported by a simulation study. Chapter Five includes the results of this simulation study. In Chapter Six, the conclusions are presented.

CHAPTER TWO

REGRESSION METHODS

2.1 Multiple Linear Regression

A regression model can serve several purposes. In process analysis and chemical engineering applications, the purpose is almost exclusively prediction. In other applications, the focus is on understanding the relationship between the predictors and response variable. Hence, many problems in applied sciences can be cast in the framework of a regression problem (Henk, et al, 2007).

Multiple Linear Regression (MLR) analysis is one of the most widely used of all statistical methods. It represents the relationship between a response variable and a set of predictor variables. The regression model for N observations and M predictor variables can be described as follows:

Multiple Linear Regression model equation is as follows:

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \ldots + \beta_m x_{im} + \varepsilon_i, \quad i = 1, \ldots, N \tag{2.1}$$

x_{mi} : value of the m^{th} predictor variable for the i^{th} observation

β_0 : regression constant

β_m : coefficient of the m^{th} parameter

M : total number of predictor variables

y_i : response in the i^{th} observation

ε_i : error terms

The MLR model in terms of the observations can be written as matrices notation by: $\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$.

$$\mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}, \quad \mathbf{X} = \begin{bmatrix} 1 & \mathbf{x}_{11} & \mathbf{x}_{12} & \cdots & \mathbf{x}_{1,(M-1)} \\ 1 & \mathbf{x}_{21} & \mathbf{x}_{22} & \cdots & \mathbf{x}_{2,(M-1)} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & \mathbf{x}_{N1} & \mathbf{x}_{N2} & \cdots & \mathbf{x}_{N,(M-1)} \end{bmatrix}$$

where $\mathbf{y}$ is an $N \times 1$ vector of observed response values, $\mathbf{X}$ is the $N \times M$ matrix of the predictor variables, $\boldsymbol{\beta}$ is the $M \times 1$, and $\boldsymbol{\varepsilon}$ is the $N \times 1$ vector of random error terms.

The aim of regression analysis is to find the estimates of unknown parameters. The regression equation is used to predict $\mathbf{Y}$ from predictors. The method of Ordinary Least Squares (OLS) is used to find the best line that, on average, is the closest to all of the points. OLS finds the best estimate of $\boldsymbol{\beta}$'s with the least squares criterion which minimizes the sum of squared distances of all of the points from the actual observation to the regression surface.

In the linear regression model $\hat{\mathbf{y}} = \mathbf{X}\hat{\boldsymbol{\beta}}$, $\hat{\mathbf{y}}$ is the vector of predicted response variable, $\mathbf{e}$ is the vector of residuals, and $\hat{\boldsymbol{\beta}}$ is the estimate of the regression coefficient. To compute $\hat{\boldsymbol{\beta}}$, the sum of the squared residuals are minimized with ordinary least squares, as shown in the following equation where $\mathbf{e_i} = \mathbf{y_i} - \mathbf{x_i'}\hat{\boldsymbol{\beta}}$, $i = 1, \ldots, N$.

$$\min_{\hat{\beta}} \sum_{i=1}^{N} \boldsymbol{\varepsilon}_i^2 \tag{2.2}$$

The OLS estimator $\hat{\boldsymbol{\beta}}$ is an unbiased estimator, which is $\mathbf{E}(\hat{\boldsymbol{\beta}}) = \boldsymbol{\beta}$ and has minimum variance, which is $\mathrm{Cov}(\hat{\boldsymbol{\beta}}) = \hat{\sigma}_2(\mathbf{X'X})^{-1}$.

The MLR is based on some assumptions. These are: no linear relationship exists among predictor variables; error terms are distributed as normal distribution with

mean zero and constant variance $\boldsymbol{\varepsilon_i} \sim N(0, \sigma^2)$, and error terms are independent of each of the predictor variables and each other.

MLR works ideally when the predictor variables are few in number and when they are not collinear. However, omitting one of the assumptions of MLR can damage an analysis and render its estimations insignificant. As with other assumptions, avoding multicollinearity is important, because the least squares estimators are very poor in the analysis in the presence of multicollinearity. The next subsection is concerned with multicollinearity and solving this problem.

2.1.1 Multicollinearity

Bowerman and O'Connell (1990) describe multicollinearity as a problem in regression analysis when the predictor variables in a regression model are intercorrelated on each other. The problem that multicollinearity poses is that it makes it difficult to separate the effects of two variables on an outcome variable. If two variables are significantly related to each other, it becomes impossible to determine which of the variables accounts for variance in the response variable.

For example, it is assumed that the MLR model is given as $y_i = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \varepsilon_i$ and $X_2 = 3X_1$ so, the correlation between two predictor variables is 1 and the MLR model is written as below:

$$\begin{aligned} y_i &= \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \varepsilon_i \\ &= \beta_0 + (\beta_1 + 3\beta_2) x_{1i} + \varepsilon_i \ . \end{aligned}$$

From the regression model, thus, only $\beta_1 + 3\beta_2$ can be estimated. It is not possible to get separate estimates of β_1 and β_2. From this example, some results can be obtained. These are: when one or more predictor variables are present, a possible problem may occur; two or more variables can explain the dependent variable well,

but they may be closely correlated. Therefore, the results suggest that it is difficult to distinguish the individual effects of both variables.

The sources of multicollinearity can be explained in many ways.

Firstly, a variable that is computed from other variables in the equation can be included. For example, a regression model of a family's income which is formed by both the husband's income and the wife's income, includes all the three measures. Also including the same or almost the same variable twice can cause multicollinearity, for example height in feet and height in inches. Constraints on the population being sampled can also cause multicollinearity; for example people with higher incomes will have more wealth and more predictor variables than the number of observations.

Multicollinearity can be a big problem when the aim is to try to understand how the variation of the predictor variable affects response variable.

Multicollinearity can be explained as the following aspect of regression model: the greater the multicollinearity, the greater the standard errors: When there is high multicollinearity, confidence intervals for coefficients tend to be very wide. The confidence intervals may even include zero, which means you cannot be confident whether an increase in the predictor variables value is associated with an increase or a decrease in the response variable. t statistics tend to be very small, therefore the estimation of regression coefficients in these cases is statistically insignificant. Even extreme multicollinearity does not violate any of the assumptions of OLS regression, OLS estimates are still unbiased and OLS estimators are the best linear unbiased estimators. Although the t-ratio of one or more coefficients is statistically insignificant, R^2 the overall measure of goodness of fit can be very high. The OLS estimators can be sensitive to small changes in the data. Collinear variables contribute redundant information and can cause other variables to appear to be less important than they are. Overestimating the effect of one parameter will tend to

underestimate the effect of the other. Hence coefficient estimates tend to be very weak from one sample to the other.

Some classical signs of multicollinearity are;

- having a significant F, but no significant t-ratios and high R^2.
- widely changing coefficients when an additional variable is included.
- high pairwise correlations among predictors.
- the tolerances or Variance Inflation Factor is probably superior for examining the bivariate correlations.

Sometimes eigenvalues, condition index and then condition number will be referred to when examining multicollinearity.

2.1.2 Detecting Methods for Multicollinearity

Multicollinearity on a data set can be determined with some methods. The most commonly used methods are given below.

2.1.2.1 Condition Index

The condition number (CN) is the condition index (CI) with the largest eigenvalue and it equals the square root of the largest eigenvalue (λ_{max}) divided by the smallest eigenvalue (λ_{min}).

$$CN = \frac{\lambda_{max}}{\lambda_{min}}, \tag{2.3}$$

and the CI is defined as:

$$CI = \sqrt{\frac{\lambda_{max}}{\lambda_{min}}} = \sqrt{CN}\ .$$

When there is no collinearity the eigenvalues, condition index, the condition number will all be equal to one. An informal rule of thumb is that if the condition number is 15, multicollinearity is a concern. If it is greater than 30, multicollinearity is a very serious concern.

2.1.2.2 Variance Inflation Factor and Tolerance

VIF and tolerance are the classical tests for diagnosing collinearity problems. They can be explained by the help of variance of the sampling distribution for OLS coefficients. The variance of the sampling distribution for OLS coefficients can be expressed as:

$$Var(\beta_i) = \frac{1}{1-R_i^2}\frac{\sigma_e^2}{(n-1)S_i^2},\ \ i = 1, 2, \ldots, N \qquad (2.4)$$

R_i^2 is the explained variance that is obtained when regressing $\mathbf{X_i}$ on the other $\mathbf{X}$ variables in the model; S_i^2 is the variance of $\mathbf{X_i}$; σ_e^2 = MSE of the model. $Var(\beta_i)$ is increased if σ_e^2 is large, S_i^2 is small or R_i^2 is large.

The first term of the expression above is called the Variance Inflation Factor (VIF).

$$VIF = \frac{1}{1-R_i^2}.$$

If $\mathbf{X_i}$ is highly correlated with the other $\mathbf{X}$ variables, then R_i^2 will be large, making the denominator of the VIF small and hence the VIF becomes very large. This inflates the variance of β_i and makes it difficult to obtain a significant t-ratio.

The value 10 is used as a threshold which considers multicollinearity to be a problem.

Another measure to detect multicollinearity is tolerance. Tolerance which is defined as:

$$TOL_i = \left(1 - R_i^2\right) = \left(\frac{1}{VIF_i}\right)$$

$TOL_i = 1$ if $\mathbf{X_i}$ is not correlated with other predictors, whereas $TOL_i = 0$ if it is perfectly related to the predictors.

2.1.3 Solutions to Remove Multicollinearity

Several techniques have been proposed to deal with the problem of multicollinearity. The following methods have been suggested as possible solutions to the multicollinearity problem.

- Get more data: Increase the observation number by adding observations (new individuals) and extending the time period of observation. This will usually decrease standard errors.
- Drop variables: If two variables are highly correlated, leave one of them.
- Rethink of the model.
- Combine variables; for example if education and income are highly collinear, you can combine them as a "socioeconomic status".
- Use Principal Components Regression, Ridge Regression, Partial Least Squares Regression or other methods.

2.2 Principal Component Analysis

Principal Component Analysis (PCA) is the first step of the Principal Component Regression. The general objectives of Principal Component Regression are data reduction and interpretation. It is concerned with explaining the variance-covariance structure of a set of variables through a few linear combinations of these variables.

The goal of PCA is to create a new set of variables called principal components or principal variates. The principal components are linear combinations of the variables of the vector $\mathbf{Y}^*$ that are uncorrelated and the variance of the j^{th} component is maximum. $\mathbf{Y}^*_{1\times m} = [\mathbf{Y}^*_1, \mathbf{Y}^*_2, \ldots, \mathbf{Y}^*_m]$ is an observation vector with mean μ and covariance matrix $\mathbf{\Sigma}$ of full rank m.

In this analysis, m predictor variables, which are mutually collinear and have N observation, are transformed to q $(q \le m)$ new variables called principal component which are linear, orthogonal, and mutually independent.

The total variation is described by all of the m variables when m property is measured for N observation. However, the major part of the total variability can be explained by q component. Then q new component can present m variable. Thus m variables with N measure number will be reduced to q new variables without losing any information.

PCA can be defined as follows:

The first principal component $(\mathbf{Y}^*_1)$ is determined as a linear combination of $\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_m$. The first component is the component which has the maximum addition to the total variability:

$$\mathbf{Y}^*_1 = \underline{\mathbf{a}}_1' \mathbf{X} = a_{11}\mathbf{X}_1 + a_{12}\mathbf{X}_2 + \ldots + a_{1m}\mathbf{X}_m \qquad (2.5)$$

The second principal component describes the remaining maximum variation after the first principal component. These components are uncorrelated.

$$\mathbf{Y_2^* = \underline{a}_2' X = a_{21}X_1 + a_{22}X_2 + \ldots + a_{2m}X_m}$$
$$\vdots$$
$$\mathbf{Y_m^* = \underline{a}_m' X = a_{m1}X_1 + a_{m2}X_2 + \ldots + a_{mm}X_m}$$

$$\mathrm{Var}\left(\mathbf{Y_i^*}\right) = \mathbf{\underline{a}_i' \Sigma \underline{a}_i},\ \ i = 1,2,\ldots,m;\quad \mathrm{Cov}\left(\mathbf{Y_i^*, Y_q^*}\right) = \mathbf{\underline{a}_i' \Sigma \underline{a}_q} \tag{2.6}$$

The first principal component variable provides the conditions which are $\mathbf{\underline{a}_1' \underline{a}_1} = 1$ and $\max \mathrm{Var}\left(\mathbf{\underline{a}_1' X}\right)$. The second principal component provides the conditions that $\mathbf{\underline{a}_2' \underline{a}_2} = 1$ and $\max \mathrm{Var}\left(\mathbf{\underline{a}_2' X}\right)$ after the first principal component:

$$\mathrm{Cov}\left(\mathbf{\underline{a}_1' X, \underline{a}_2' X}\right) = \mathrm{Cov}\left(\mathbf{Y_1^*, Y_2^*}\right) = 0$$

The i^{th} PC satisfies $\max \mathrm{Var}\left(\mathbf{\underline{a}_i' X}\right)$, $\mathbf{\underline{a}_i' \underline{a}_i = 1}$, and for $q < i$, $\mathrm{Cov}\left(\mathbf{Y_i^*, Y_q^*}\right) = 0$.

Thus $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_m$ denote the ordered eigenvalues of $\mathbf{\Sigma}$ and $a_1, a_2, \ldots, a_m$ denote corresponding normalized eigenvectors of $\mathbf{\Sigma}$.

The variance of the j^{th} component $\mathbf{Y_j^*}$ is λ_j.

$$\mathrm{tr}(\Sigma) = \sigma_{11}^2 + \sigma_{22}^2 + \ldots + \sigma_{mm}^2 = \lambda_1 + \lambda_2 + \ldots + \lambda_m \tag{2.7}$$

The total variation accounted for by all of the principal component variables is equal to the amount of variation measured by the original variables. Therefore to measure the importance of the j^{th} principal component, the ratio of $\dfrac{\lambda_j}{\mathrm{tr}(\mathbf{\Sigma})}$ should be referred to. To achieve eigenvalues:

$$\left|\mathbf{\Sigma - \lambda I}\right| = 0 \tag{2.8}$$

$\mathbf{\Sigma}$: symmetric, nonnegative, diagonal matrix.

m eigenvectors can be achieved from this relation by using m eigenvalues. $\mathbf{a}_1$ is the first eigenvector of $(\mathbf{\Sigma}-\lambda_1\mathbf{I})\underline{\mathbf{a}}_1=0$.

If $\mathbf{Y}^*_1=\underline{\mathbf{a}}_1'\mathbf{X}$, $\mathbf{Y}^*_2=\underline{\mathbf{a}}_2'\mathbf{X}$ are the principal components obtained from the covariance matrix $\mathbf{\Sigma}$ then for k<m,

$$\rho_{Y_i,X_k}=\frac{\mathrm{Cov}(\mathbf{Y}_i^*,\mathbf{X}_k)}{\sqrt{\mathrm{Var}(\mathbf{Y}_i^*)}\sqrt{\mathrm{Var}(\mathbf{X}_k)}}=\frac{\lambda_i a_{ik}}{\sqrt{\lambda_i}\sqrt{\sigma_{kk}}}=\frac{a_{ik}\sqrt{\lambda_i}}{\sqrt{\sigma_{kk}}} \tag{2.9}$$

$$\mathbf{Y}_i^*=\underline{\mathbf{a}}_i'\mathbf{X}=a_{i1}\mathbf{X}_1+\ldots+a_{im}\mathbf{X}_m \qquad i=1,2,\ldots,m$$

$$\mathrm{Var}(\mathbf{Y}_i^*)=\underline{\mathbf{a}}_i'\mathbf{\Sigma}\underline{\mathbf{a}}_i=\lambda_i \qquad i=1,2,\ldots,m$$

$$\mathrm{Cov}(\mathbf{Y}_i^*,\mathbf{X}_q)=\underline{\mathbf{a}}_i\mathbf{\Sigma}\underline{\mathbf{a}}_q=0 \qquad i\neq q\,.$$

Principal components can also be obtained from standardized variables. Standardized variables, which are given below, are used when the variances are drastically different from each other or the measurement scale of the variables is different.

$$Z_1=\frac{(X_1-\mu_1)}{\sigma_{11}}$$

$$Z_2=\frac{(X_2-\mu_2)}{\sigma_{22}}$$

$$\vdots$$

$$Z_p=\frac{(X_p-\mu_p)}{\sigma_{pp}}$$

$$\mathbf{Z}=\left(\mathbf{V}^{1/2}\right)^{-1}(X-\mu) \tag{2.10}$$

$$\mathbf{Cov}(\mathbf{Z})=\left(\mathbf{V}^{1/2}\right)^{-1}\mathbf{\Sigma}\left(\mathbf{V}^{1/2}\right)^{-1}=\mathbf{R}\,.$$

Here **V** is the matrix of the set of all eigenvalue of covariance matrix. **R** is the correlation matrix.

The principal components of **Z** may be obtained from the eigenvectors of the correlation matrix **R** of **X**. All the other results apply to the **R**.

$$\mathbf{Y}_i^* = \underline{\mathbf{a}}_i' \mathbf{Z} = \underline{\mathbf{a}}_i' \left(\mathbf{V}^{1/2} \right)^{-1} (\mathrm{X} - \mu) \tag{2.11}$$

$$\sum_{i=1}^{m} \mathrm{Var}(\mathbf{Y}_i^*) = \sum_{i=1}^{m} \mathrm{Var}(\mathbf{Z}_i) = \mathrm{p}$$

Elements with an eigenvector are comparable to one another but elements in different eigenvectors are not comparable. To make comparisons between eigenvectors some researchers scale the eigenvectors by multiplying the elements in each vector by the square root of its corresponding eigenvalue. That is

$$\mathbf{c}_j = \sqrt{\lambda_j}\, \underline{\mathbf{a}}_j$$

The new vectors are called component loadings vector. The i^{th} element in $\mathbf{c}_j$ gives the covariance between the i^{th} original variable and the j^{th} principal component. For more details about PCA, see Johnson, 1998.

2.2.1 Determining the Number of Principal Components

There is always the question of how many components to retain. Some methods exist for determining an appropriate number of components. These are:

Method 1

The simplest way is to look at the number of eigenvalues bigger than 1 (for standardized data), or the small value of q that provides the condition $\sum_{j=1}^{q} \frac{\lambda_j}{m} \geq \frac{2}{3}$.

Method 2

Scree plot of the eigenvalues. To plot $(1, \hat{\lambda}_1), (2, \hat{\lambda}_2), \ldots, (m, \hat{\lambda}_m)$.

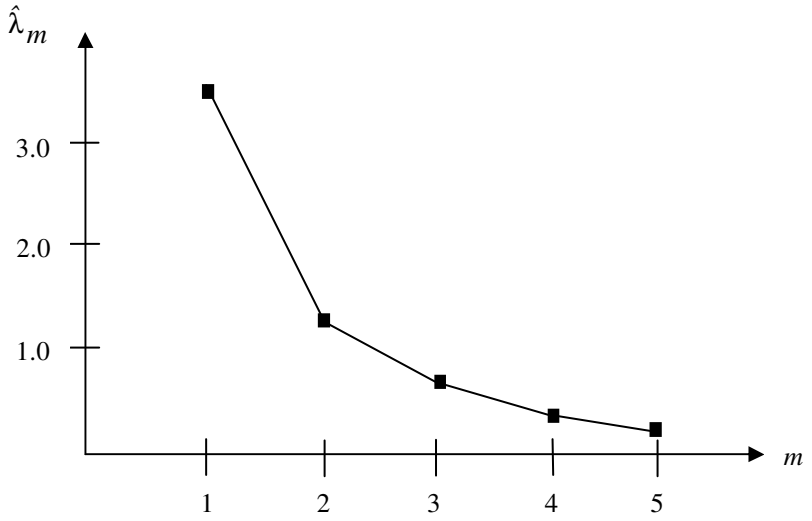


Figure 2.1. A scree plot

An elbow occurs in the plot. That is, the eigenvalues after $\hat{\lambda}_3$ are relatively small and nearly at the same size with the following eigenvalues. In this case it appears that two (or three) sample principal components effectively summarize the total variation.

2.2.2 Cautions about PCA

- If the original variables are nearly uncorrelated, nothing can be gained by carrying out a PCA. In this case, the actual dimensionality of the data is equal to the number of response variables measured.
- Any change in the measurement scale reflects the principal components.

- PCA cannot generally be used to eliminate variables, because all of the original variables are needed to score or evaluate the principal component variables for each of the individuals in a data set.

Summary of steps in PCA:

1. The data matrix which has p variable on n measurement is standardized.
2. The correlation matrix of standardized data matrix is found.
3. The eigenvalues and eigenvectors of correlation matrix is calculated.
4. The account ratio of total variation of principal component is found by the help of eigenvalues.
5. Principal component value is found by multiplying the transpose of each eigenvectors with the transpose of standardized data matrix.

2.3 Principal Component Regression

PCA selects a new set of predictor variables which are called components. These components are selected with the decreasing of variance within the predictor variables. These components are perpendicular to each other, which mean that there is no multicollinearity among them. Principal Component Regression (PCR) is used after PCA by applying MLR to the components.

PCR only deals with the variance-covariance matrix of predictor variables $(\mathbf{X}'\mathbf{X})$. It doesn't concern the relationship among the response variables. It defines all the latent variables using all of the original predictors.

2.4 Partial Least Squares Regression

There is another method, which can be used in detecting multicollinearity and which is the subject of this thesis, called Partial Least Squares Regression (PLSR). It also deals with the variation of the response variables. PLSR analysis is based on the

variance-covariance matrix of the all variables, that is $(\mathbf{X'Y})$. In particular, the method of Partial Least Squares Regression balances the two objectives, seeking latent variables that explain both response and predictor variables. The following chapter gives a brief summary about PLSR.

CHAPTER THREE

PARTIAL LEAST SQUARES REGRESSION

3.1 Literature Review of Partial Least Squares Regression

The pioneering work of PLS was done by Herman Wold at the beginning of the 1970's. After his Ph.D. on the subject of time series, he went on studying regression in econometric models. This led him to the fixed-point method. It is a method of designing path models with directly observed variables and has an algorithm which is iterative. This experience on iterative models has played an important role on later developments.

Around 1964 Herman Wold invented the NIPALS. The NIPALS method contains a number of properties that eased the path to useful PLS modelling. The NIPALS method is used to compute principal components by an iterative sequence of simple ordinary least squares regressions. Together, the combination of econometric modelling and NIPALS created the first form of PLS in the early 1970s.

PLS found its way into Chemistry in the late 1970's. Svante Wold, son of Herman Wold, had helped his father in the previous work on the NIPALS algorithm and used it on his own work. The first chemical paper to make reference to PLS was by Gerlach, Kowalski and H. Wold in 1979. Since then a growing number of chemists have used PLS to build calibration methods that seem to have superior prediction to other methods.

Many articles have been written concerning the developments of PLS. The book by Naes and Martens used statistical concepts that began to provide a theoretical basis for PLS (1989). Paul Geladi offered a review of historical development of PLS (1988). PLS regression was studied and developed from the point of view of statisticians by Agnar Höskuldsson (1988). The mathematical foundations of PLS have been discussed by Lorber, Wangen and Kowalski (1987). A tutorial for PLS

was provided by Geladi and Kowalski (1986). The most recent research was done by Inge Helland (1990), Paul Garthwaite (1994) and Svante Wold (2001).

PLS is comprised of some algorithms. These are; NIPALS algorithm, UNIPALS algorithm, KERNEL algorithm, SAMPLS algorithm and SIMPLS algorithm. Most commonly used algorithms are NIPALS, SIMPLS and KERNEL algorithms. NIPALS was the first algorithm to be studied. Then, the other algorithms were investigated based on NIPALS algorithm. SIMPLS algorithm was studied by Sijmen de Jong (1993). KERNEL algorithm was studied by Fredrik Lindgren, Paul Geladi and Svante Wold (1993). Also Cajo Ter Braak (1994) and Stefan Rännar (1994) have studies about KERNEL algorithm.

After PLS analysis, in regression part, some model selection criteria played an important role to select the best model. Baibing Li, Julian Morris and Elaine B. Martin (2002) are the major names about this subject.

3.2 Partial Least Squares Regression

PLSR is a multivariate statistical technique that allows a relationship among multiple response variables and multiple predictor variables. It is a wide class of methods which consists of regression (MLR), dimension reduction techniques (PLS), and modelling tools.

Dimension reduction is made in the PLS partition. PLS was designed to deal with multiple regression when data have missing values and multicollinearity. It is a very popular method when there is a big problem with a high number of correlated variables and a limited number of observations.

The goal of PLS is to predict Y from X while describing the common structure between the two variables. That is, PLS will give the minimum number of variables required to maximize the covariance between the predictor and predicted variables (Höskuldsson, 1988).

There are two types of PLS. PLS1 is when there is univariate response variable, PLS2 is when there are at least two response variables. PLS can be interpreted as an extension of regression problems. The predictor and response variables are each considered as a block of variables. Then PLS extracts the score vectors (latent vector or components) which serve as a new predictor representation and regresses the response variables on these new predictors. Components which are linear combinations of original predictors are mutually independent (orthogonal).

As an extension of the MLR model, PLSR shares the assumptions of Multiple regression. However, unlike MLR, it can analyze data with strongly collinear, numerous predictor variables, as well as the model several response variables.

PLSR is a latent variable based method for the linear modeling of the relationship between a set of response variables $\mathbf{Y}$ $(N \times K)$ and a set of predictor variables $\mathbf{X}$ $(N \times M)$ (Lindgren, F., et al., 1993).

Certain mathematical treatments and the working with large data sets have created some problems. Modelling large data sets limits the size of the computer memory. With the development of computer technology, this problem is constantly decreasing. Algorithms and programs have been optimized to meet the demands of today (Lindgren and Rannar, 1998).

An algorithm is a well defined procedure to solve a problem. An algorithm generally takes some input, carries out a number of effective steps in a finite amount of time, and produces some output (Algorithm, n.d.).

The choice of algorithm depends strongly on the shape of data matrices to be studied. In some studies, the number of observations is much larger than the number of variables. This leads to algorithm to work with variance-covariance, since number of variables are independent of the number of observations. For an opposite situation where the number of variables exceed the number of observations, choosing an

algorithm that works with a matrix that is independent of the number of variables will be the best choice (Lindgren and Rannar, 1998).

In multivariate studies there are three types of large data matrices:

- matrices with many observations and few variables; N large, K and M small,
- matrices with many variables and few observations; N small, K and/or M large,
- matrices with many variables and many observations; N, K and/or M large.

Several algorithms can be used in PLS regression. These algorithms use the situations that are given above. Most commonly used are NIPALS, SIMPLS, PLS-Kernel and Kernel algorithms. These are explained in next subsections.

3.2.1 NIPALS Algorithm

The NIPALS algorithm, also known as the classical algorithm, was developed by H. Wold by 1960's. It was first used for PCA and later for PLS. It is the most commonly used method for calculating the principal components of a data set. It gives more numerically accurate results when compared with Singular Value Decomposition (SVD) of the covariance matrix, but is slower to calculate. In following sections NIPALS algortihm for PCA and NIPALS algorithm for PLS will be explained, respectively.

3.2.1.1 NIPALS Algorithm for PCA

Consider the NIPALS for finding the principal components of $\mathbf{X}'\mathbf{X}$. The aim is to find the first q principal component of $\mathbf{X}'\mathbf{X}$ starting with the largest eigenvalue λ_1 and down. q must be less than or equal to m.

The algorithm starts with $j = 1$ and $\mathbf{X}_j = \mathbf{X}$ and carries on with the following iterative steps.

1. Choose $\mathbf{t}_j$ as any column of $\mathbf{X}_j$.
2. Let $\mathbf{p}_j = \dfrac{\mathbf{X}_j'\mathbf{t}_j}{\left\|\mathbf{X}_j'\mathbf{t}_j\right\|}$.
3. Let $\mathbf{t}_j = \mathbf{X}_j\mathbf{p}_j$.
4. If $\mathbf{t}_j$ equals to the one used in step 2 then continue, otherwise return step 2.
5. Let residuals $\mathbf{X}_{j+1} = \mathbf{X}_j - \mathbf{t}_j\mathbf{p}_j'$.
6. Let $j = j+1$ and repeat steps 1 to 6 by using residuals $\mathbf{X}_{j+1}$ instead of $\mathbf{X}_j$ until $j = m$.

Matrices $\mathbf{T}$ and $\mathbf{P}$ with columns $\mathbf{t}_j$ and $\mathbf{p}_j$ now satisfy $\mathbf{X} = \mathbf{T}\mathbf{P}'$.

Properties of algorithm are:

STEP 2:

Let $\lambda_j = \left|\mathbf{X}'\mathbf{t}_j\right|$. Then step 2 is written as $\mathbf{X}'\mathbf{t}_j = \lambda_j\mathbf{p}_j$

STEP 3:

$\mathbf{t}_j = \mathbf{X}_j\mathbf{p}_j$ then $\mathbf{X}'\mathbf{X}\mathbf{p}_j = \lambda_j\mathbf{p}_j$ (Eigen decomposition of $\mathbf{X}'\mathbf{X}$). Using the equation in Step 3;

$$\begin{aligned}\mathbf{t}_j'\mathbf{t}_j &= \left(\mathbf{X}\mathbf{p}_j\right)'\left(\mathbf{X}\mathbf{p}_j\right)\\ &= \mathbf{p}_j'\mathbf{X}'\mathbf{X}\mathbf{p}_j\\ &= \lambda_j\mathbf{p}_j'\mathbf{p}_j\\ &= \lambda_j\end{aligned}$$

STEP 5:

$j = 1$ gives, $\mathbf{X}_2 = \mathbf{X} - \mathbf{t}_1\mathbf{p}_1' \Rightarrow \mathbf{X} = \mathbf{X}_2 + \mathbf{t}_1\mathbf{p}_1'$

Then **X** can be written as a linear combination;

$$\begin{aligned}\mathbf{X} &= \mathbf{t}_1\mathbf{p}_1' + \mathbf{t}_2\mathbf{p}_2' + \ldots + \mathbf{t}_p\mathbf{p}_p' + \mathbf{X}_{p+1} \\ &= \mathbf{T}_q\mathbf{P}_q' + \mathbf{X}_{q+1}\end{aligned}. \qquad (3.1)$$

$\mathbf{T}_q$ and $\mathbf{P}_q$ contain the first p columns of $\mathbf{T}$ and $\mathbf{P}$. The aim is to choose q to make $\mathbf{X}_{q+1}$ is small. The relative size of the eigenvalues is expressed as a percentage of the sum of all eigenvalues. So, the percentage of variation explained by the first j component is

$$\frac{\lambda_1 + \ldots + \lambda_j}{\lambda_1 + \ldots + \lambda_q} \times 100$$

3.2.1.2 NIPALS Algorithm for PLS

The basic algorithm for PLS regression was developed by Wold in 1960's. The starting point of the algorithm is two data matrices $\mathbf{X}$ and $\mathbf{Y}$. $\mathbf{X}$ is $N \times M$, $\mathbf{Y}$ is $N \times K$ where N also represents the number of rows, M also represents the number of columns, and K is the number of response variables. Before the algorithm starts, the data matrices must be mean centered or scaled. The algorithm is as follows:

1. Start: Set $\mathbf{u}_{(N\times 1)}$ to the first column of $\mathbf{Y}$.

2. $\mathbf{w}_{(M\times 1)} = \dfrac{\mathbf{X}'_{(M\times N)}\mathbf{u}_{(N\times 1)}}{\left(\mathbf{u}'_{(1\times N)}\,\mathbf{u}_{(N\times 1)}\right)}$

3. Scale $\mathbf{w}_{(m\times 1)}$ to be of length one.

4. $\mathbf{t}_{(N\times 1)} = \mathbf{X}_{(N\times M)}\,\mathbf{w}_{(M\times 1)}$

5. $\mathbf{c}_{(K\times 1)} = \dfrac{\mathbf{Y}'_{(K\times N)}\,\mathbf{t}_{(N\times 1)}}{\left(\mathbf{t}'_{(1\times N)}\,\mathbf{t}_{(N\times 1)}\right)}$

6. Scale c to be of length one.

7. $\mathbf{u}_{(N\times 1)} = \dfrac{\mathbf{Y}_{(N\times K)}\,\mathbf{c}_{(K\times 1)}}{\left(\mathbf{c}'_{(1\times K)}\,\mathbf{c}_{(K\times 1)}\right)}$

8. If $\mathbf{t}$ in step 4 convergences to the one in the preceding iteration then go to step 9 else go to step 2.

9. X-loadings: $\mathbf{p}_{(M\times 1)} = \dfrac{\mathbf{X}'_{(M\times N)}\,\mathbf{t}_{(N\times 1)}}{\left(\mathbf{t}'_{(1\times N)}\,\mathbf{t}_{(N\times 1)}\right)}$

10. Y-loadings: $\mathbf{q}_{(K\times 1)} = \dfrac{\mathbf{Y}'_{(K\times N)}\,\mathbf{u}_{(N\times 1)}}{\left(\mathbf{u}'_{(1\times N)}\,\mathbf{u}_{(N\times 1)}\right)}$

11. Regression (u upon t): $\mathbf{b}_{(1\times 1)} = \dfrac{\mathbf{u}'_{(1\times N)}\,\mathbf{t}_{(N\times 1)}}{\left(\mathbf{t}'_{(1\times N)}\,\mathbf{t}_{(N\times 1)}\right)}$

12. Residual matrices: $\mathbf{X} \to \mathbf{X} - \mathbf{t}\mathbf{p}'$ and $\mathbf{Y} \to \mathbf{Y} - \mathbf{b}\mathbf{t}\mathbf{c}'$.

Properties of algorithm are:

STEP 2:

In PLS, the direction in the space of $\mathbf{X}$ which yields the biggest covariance between $\mathbf{X}$ and $\mathbf{Y}$ is being searched. This direction is given by a unit vector $\mathbf{w}$ (weight vector). This weight vector formed by standardizing the covariance matrix for $\mathbf{X}$ and $\mathbf{Y}$. Weights are based on the covariance between $\mathbf{X}_j$ and $\mathbf{u}_j$.

STEP 3:

$$\mathbf{w}_{(\mathbf{M}\times 1)} = \frac{\mathbf{X}'_{(\mathbf{M}\times\mathbf{N})}\mathbf{u}_{(\mathbf{N}\times 1)}}{\left(\mathbf{u}'_{(1\times\mathbf{N})}\,\mathbf{u}_{(\mathbf{N}\times 1)}\right)} \tag{3.2}$$

$\mathbf{w}$ is scaled; that is $\frac{\mathbf{w}}{\sqrt{\mathbf{w}'\mathbf{w}}}$ such that,

$$\frac{\left(\mathbf{X}'_{(\mathbf{M}\times\mathbf{N})}\mathbf{u}_{(\mathbf{N}\times 1)}\right)/\left(\mathbf{u}'_{(1\times\mathbf{N})}\mathbf{u}_{(\mathbf{N}\times 1)}\right)}{\left(\left(\frac{\mathbf{X}'_{(\mathbf{M}\times\mathbf{N})}\mathbf{u}_{(\mathbf{N}\times 1)}}{\mathbf{u}'_{(1\times\mathbf{N})}\mathbf{u}_{(\mathbf{N}\times 1)}}\right)'\left(\frac{\mathbf{X}'_{(\mathbf{M}\times\mathbf{N})}\mathbf{u}_{(\mathbf{N}\times 1)}}{\mathbf{u}'_{(1\times\mathbf{N})}\mathbf{u}_{(\mathbf{N}\times 1)}}\right)\right)^{\frac{1}{2}}} \Rightarrow \mathbf{norm}\ \mathbf{w} = \left\|\frac{\mathbf{X}'_{(\mathbf{M}\times\mathbf{N})}\mathbf{u}_{(\mathbf{N}\times 1)}}{\left(\mathbf{u}'_{(1\times\mathbf{N})}\mathbf{X}_{(\mathbf{N}\times\mathbf{M})}\mathbf{X}'_{(\mathbf{M}\times\mathbf{N})}\mathbf{u}_{(\mathbf{N}\times 1)}\right)}\right\| = 1$$

STEP 4:

The $\mathrm{N}\times 1$ latent vector $\mathbf{t}_1$ is formed as a linear combination of the columns of $\mathbf{X}$ with weights vector $\mathbf{w}_1$. The latent vectors $\mathbf{t}_\mathbf{j}$ are also called *scores*, similar to the terminology for PCA.

STEP 5:

$\mathbf{c}_{(\mathrm{K}\times 1)}$ are the weights of $\mathbf{Y}$.

STEP 8:

Convergence is tested on the change in $\mathbf{t}$. $\frac{\left\|\mathbf{t}_{\mathbf{old}} - \mathbf{t}_{\mathbf{new}}\right\|}{\left\|\mathbf{t}_{\mathbf{new}}\right\|} < \varepsilon$, $\varepsilon \cong 10^{-6}, 10^{-8}$.

STEP 9:

The vector $\mathbf{p}_{(M\times 1)}$ is the vector of regression coefficients obtained from multiple linear regression of $\mathbf{X}_j$ on $\mathbf{t}_j$. This vector is called *loadings*.

Model is, $\mathbf{X} = \mathbf{tp}'$.

STEP 10:

This step is to find the loadings for $\mathbf{Y}$.

STEP 11:

b is a scaling factor.

STEP 12:

$\mathbf{X} \rightarrow \mathbf{X} - \mathbf{tp}'$

$\hat{\mathbf{X}}$ (estimated from the algorithm)

Beginning matrix (at the beginning of the algorithm)

New matrix (Residual)

This equation can be similarly written for $\mathbf{Y}$.

NIPALS algorithm is based on the classical algorithm which was developed by Wold in 1960's. The use of NIPALS in large data structures, causes some technical problems. The calculation of score and loading vectors can be time-consuming and requires big memory. In the case of large matrices fast and powerful software is needed.

3.2.2 SIMPLS Algorithm

This algorithm was developed by Sijmen de Jong in 1993. This name was given since it's being a ***s***traightforward ***impl***ementation of a ***s***tatistically ***i***nspired ***m***odification of the PLS method (De Jong, 1993). It is much faster than the NIPALS algorithm, especially when the number of predictor variables increases, but gives slightly different results in the case of multivariate response variables. For univariate response variable, SIMPLS is equivalent to PLS1.

In both algorithms, the predictor and response variables are first mean centered. In the first stage of PLS2 the data matrix $\mathbf{X}$ is deflated in each step and the latent vectors $\mathbf{t}$ are the linear combinations of the deflated matrix not the original matrix. For that reason the interpretation of the score matrix $\mathbf{T}$ is not straightforward. SIMPLS calculates the PLS latent variables directly as linear combinations of the original variables because of deflating the covariance matrix $\mathbf{S} = \mathbf{X}'\mathbf{Y}$.

3.2.3 Kernel Algorithm

The first kernel algorithm was developed by Lindgren in 1993. It was an alternative to the classical algorithm for handling datasets where N>>M. This algorithm uses $\mathbf{X}'\mathbf{Y}\mathbf{Y}'\mathbf{X}$ $(\mathrm{M}\times\mathrm{M})$ matrix since it is independent of the number of observations. This property provides working with small matrix. This algorithm innovates to update $\mathbf{X}'\mathbf{Y}$ variance-covariance matrix by multiplication of an updating matrix $(\mathbf{I}-\mathbf{w}\mathbf{p}')$ of size $(\mathrm{M}\times\mathrm{M})$ without interfering to the original $\mathbf{X}$ and $\mathbf{Y}$ matrices.

The second kernel algorithm was presented by Rännar et al in (1994). It is similar to the first kernel algorithm but is suitable for datasets that is M>>N (many variables and fewer observations). This algorithm depends on $\mathbf{X}\mathbf{X}'\mathbf{Y}\mathbf{Y}'$ kernel matrix.

The kernel algorithms were recently modified by De Jong (1993), resulting in faster and simplified kernel algorithms. Further modifications were proposed by

Dayal et al. (1997). They utilize the fact that only one of the matrices **X** or **Y** needs to be deflated. Since the response variables are often few, deflating **Y** instead of **X** saves time.

3.2.3.1 PLS-Kernel with Many Variables and Few Objects

This is a fast PLS regression algorithm dealing with large data matrices with many variables and fewer observations. It is based on $\mathbf{XX'YY'}$ kernel matrix which is a square, non-symmetric matrix of size $(N \times N)$. This matrix is dependent on the number of observations. When the data matrices **X, Y** are large, working with these data matrices algorithm needs lots of calculation (Rännar, S., et al 1994). That is to say, the algorithm requires a multitude of multiplications of large vectors by large matrices. This requires large storage areas in computer memory. Lindgren (1995) shows that for special cases there are alternative algorithms based on small kernel matrices. These small kernel matrices requires less space than the original data, and calculations are faster than the original data matrices.

In this algorithm, it is possible to calculate:

- All score vectors
- All loading vectors
- And hence, conduct a complete PLS regression including such as R^2.

All of the vectors can be calculated by the eigen decomposition of corresponding matrices as given by Höskuldsson (1988);

$$\begin{aligned} \mathbf{w}\alpha_1 &= (\mathbf{X'YY'X})\mathbf{w} \\ \mathbf{c}\alpha_2 &= (\mathbf{Y'XX'Y})\mathbf{c} \\ \mathbf{t}\alpha_3 &= (\mathbf{XX'YY'})\mathbf{t} \\ \mathbf{u}\alpha_4 &= (\mathbf{YY'XX'})\mathbf{u} \end{aligned} \tag{3.3}$$

where $(\alpha_1, \ldots, \alpha_4)$ are the eigenvalues and **w, c, t** and **u** are the corresponding eigenvectors with unit length.

Steps of the algorithm are as follows:

Before the algorithm starts, data matrices are scaled and mean centered.

STEP 1:

Algorithm starts with creating $\mathbf{XX}'$ and $\mathbf{YY}'$ association matrices and then by the multiplication of these association matrices $\mathbf{XX}'\mathbf{YY}'$ kernel matrix is obtained.

STEP 2:

The eigenvector of the kernel matrix is calculated. This is the first **X** latent vector $\mathbf{t}_1$. Then this latent vector is used for calculating $\mathbf{u}_1$. Then these score vectors are scaled as follows;

$$\mathbf{t}_{new} = \mathbf{t}_{new} / \text{norm}(\mathbf{t}_{new})$$

But to get similar vectors as in the classical algorithm, these score vectors are rescaled as follows:

$$\begin{aligned}
\mathbf{u}_{temp} &= \frac{\mathbf{u}_a}{\left(\mathbf{t}_a'\mathbf{F}_{a-1}\mathbf{F}_{a-1}'\mathbf{t}_a\right)} \\
\mathbf{w}'\mathbf{w} &= \mathbf{u}_{temp}'\mathbf{E}_{a-1}\mathbf{E}_{a-1}'\mathbf{u}_{temp} \\
\mathbf{t}_{scaled} &= \mathbf{t}_a\sqrt{\mathbf{w}'\mathbf{w}} \\
\mathbf{u}_{scaled} &= \mathbf{u}_{temp}\sqrt{\left(\mathbf{w}'\mathbf{w}\right)}
\end{aligned} \tag{3.4}$$

Here, $\mathbf{u}_{temp}$ is a temporary vector. $a = 1,2,\ldots,A$ number of components.

STEP 3:

This step is about updating the association matrices. In kernel algorithm, $\mathbf{XX}'$ and $\mathbf{YY}'$ association matrices are reduced. $\mathbf{E}$ is the residual matrix and at the beginning of the algorithm it is equal to original $\mathbf{X}$ data matrix i.e. $\mathbf{E_0} = \mathbf{X}$. For the first component, $\mathbf{E_1}$ residual matrix will be defined on $\mathbf{E_0}$.

$$\begin{aligned}
&\mathbf{E_a} = \mathbf{E_{a-1}} - \mathbf{t_a p_a'} \\
&\qquad\qquad \left(\mathbf{p_a'} = \mathbf{t_a' E_{a-1}}\right) \rightarrow \mathbf{p_1'} = \mathbf{t_1' E_0} = \mathbf{t_1' X} \\
&\mathbf{E_a} = \mathbf{E_{a-1}} - \mathbf{t_a t_a' E_{a-1}} \\
&\mathbf{E_a} = \left(\mathbf{I} - \mathbf{t_a t_a'}\right)\mathbf{E_{a-1}} = \mathbf{G_a E_{a-1}} \\
&\mathbf{E_a'} = \mathbf{E_{a-1}'} \mathbf{G_a'} \\
&\mathbf{G_a} = \mathbf{G_a'}
\end{aligned} \tag{3.5}$$

Here $\mathbf{G_a} = \mathbf{I} - \mathbf{t_a t_a'}$.

In this case, $\mathbf{E_1 E_1'} = \mathbf{G_1}\, \mathbf{XX'} \mathbf{G_1}$

And for the component a residual is equal to; $\mathbf{E_a E_a'} = \mathbf{G_a}\, \mathbf{E_{a-1} E_{a-1}'}\, \mathbf{G_a}$.

The same calculations can be made for $\mathbf{Y}$. In this case, $\mathbf{E_1 E_1'} = \mathbf{G_1}\, \mathbf{XX'} \mathbf{G_1}$

And for the component a residual is equal to; $\mathbf{E_a E_a'} = \mathbf{G_a}\, \mathbf{E_{a-1} E_{a-1}'}\, \mathbf{G_a}$.

The same calculations can be made for $\mathbf{Y}$.

$$\begin{aligned}
&\mathbf{F_0} = \mathbf{Y} \\
&\mathbf{F_a} = \mathbf{F_{a-1}} - \mathbf{t_a c_a'} \\
&\qquad\qquad \left(\mathbf{c_a'} = \mathbf{t_a' F_{a-1}}\right) \rightarrow \mathbf{c_1'} = \mathbf{t_1' Y} \\
&\mathbf{F_a} = \mathbf{G_a F_{a-1}} \\
&\mathbf{F_a'} = \mathbf{F_{a-1}'} \mathbf{G_a}
\end{aligned} \tag{3.6}$$

And for the component a residual is equal to; $\mathbf{F_a F_a'} = \mathbf{G_a}\, \mathbf{F_{a-1} F_{a-1}'}\, \mathbf{G_a}$.

Thus the association matrices are updated by left and right multiplication by the updating matrix $\mathbf{G_a}$.

STEP 4:

In this step, weight **W** and loading matrices **P**, **C** are calculated.

$$\begin{aligned}\mathbf{W} &= \mathbf{X}'\mathbf{U} \\ \mathbf{P} &= (\mathbf{T}'\mathbf{X})(\mathbf{T}'\mathbf{T})^{-1} \\ \mathbf{C} &= (\mathbf{T}'\mathbf{Y})(\mathbf{T}'\mathbf{T})^{-1}\end{aligned} \qquad (3.7)$$

All the columns in **W** are normalized to have length 1.

$$\left.\begin{aligned}\mathbf{w}_1' &= \mathbf{u}_1'\mathbf{E}_0 \\ \mathbf{w}_2' &= \mathbf{u}_2'\mathbf{E}_1 \\ &\vdots \\ \mathbf{w}_a' &= \mathbf{u}_a'\mathbf{E}_{a-1}\end{aligned}\right\} \quad \begin{aligned}\mathbf{w}_1' &= \mathbf{u}_1'\mathbf{X} \\ \mathbf{w}_2' &= \mathbf{u}_2'(\mathbf{I} - \mathbf{t}_1\mathbf{t}_1')\mathbf{X} \\ &= \mathbf{u}_2'\mathbf{I}\mathbf{X} - \mathbf{u}_2'\mathbf{t}_1\mathbf{t}_1'\mathbf{X} \\ &= \mathbf{u}_2'\mathbf{X}\end{aligned}$$

Here, $\mathbf{E}_1 = \mathbf{G}_1\mathbf{X} = (\mathbf{I} - \mathbf{t}_1\mathbf{t}_1')\mathbf{X}$ and $\mathbf{u}_2'\mathbf{t}_1 = 0$.

Finally the PLS regression coefficients $\mathbf{B}_{PLS}$ are obtained.

$$\mathbf{B}_{PLS} = \mathbf{W}(\mathbf{P}'\mathbf{W})^{-1}\mathbf{C}'$$

- **Some properties of vectors**

  - $\mathbf{t}_i'\mathbf{u}_j = 0 \quad \text{for} \quad j > i$

$$\begin{aligned}\mathbf{u}_1 &= \mathbf{F}_o\mathbf{c}_1 \\ \mathbf{u}_2 &= \mathbf{F}_1\mathbf{c}_2 \\ &\vdots\end{aligned}$$

$$\mathbf{F}_a = (\mathbf{I} - \mathbf{t}_a'\mathbf{t}_a)(\mathbf{I} - \mathbf{t}_{a-1}'\mathbf{t}_{a-1})(\mathbf{I} - \mathbf{t}_{a-2}'\mathbf{t}_{a-2})\ldots(\mathbf{I} - \mathbf{t}_1'\mathbf{t}_1)\mathbf{F}_0$$

The orthogonality property for $\mathbf{t}_1$ and $\mathbf{u}_2$ becomes;

$$\begin{aligned}\mathbf{t}_1'\mathbf{u}_2 &= \mathbf{t}_1'\mathbf{F}_1\mathbf{c}_2 \\ &= \mathbf{t}_1'(\mathbf{I}-\mathbf{t}_1\mathbf{t}_1')\mathbf{F}_0\mathbf{c}_2 \\ &= \underbrace{(\mathbf{t}_1'\mathbf{I}-\mathbf{t}_1'\mathbf{t}_1\mathbf{t}_1')}_{0}\mathbf{F}_0\mathbf{c}_2\end{aligned}$$

Since $\mathbf{t}_1'\mathbf{t}_1 = 1$

$$\begin{aligned}&= (\mathbf{t}_1'\mathbf{I}-\mathbf{t}_1)\mathbf{F}_0\mathbf{c}_2 \\ &= 0\end{aligned}$$

This makes $\mathbf{T}'\mathbf{U}$ a lower triangular.

- $\mathbf{t}_i'\mathbf{t}_j = \mathbf{0}$ for j>i

$$\mathbf{t}_i = \mathbf{E}_{i-1}\mathbf{w}_i$$

$$\mathbf{t}_j = \mathbf{E}_{j-1}\mathbf{w}_j$$

Then for i=1, j=2

$$\begin{aligned}\mathbf{t}_1'\mathbf{t}_2 &= \mathbf{t}_1'(\mathbf{I}-\mathbf{t}_1\mathbf{t}_1')\mathbf{E}_0\mathbf{w}_2 \\ &= (\mathbf{t}_1'\mathbf{I}-\mathbf{t}_1'\mathbf{t}_1\mathbf{t}_1')\mathbf{E}_0\mathbf{w}_2 \\ &\quad \mathbf{t}_1'\mathbf{t}_1 = \mathbf{1} \quad \text{then;} \\ &= (\mathbf{t}_1'-\mathbf{t}_1')\mathbf{E}_0\mathbf{w}_2 \\ &= \mathbf{0}\end{aligned}$$

- $\mathbf{t}_1'\mathbf{t}_1 = 1$ because **t** vectors are scaled.

3.2.3.2 PLS-Kernel with Many Observations and Few Variables

This algorithm was developed by Lindgren et al. (1995) to handle datasets where N>>M. The novelty of this algorithm that it updates the variance/covariance matrices directly without interfering with the original **X** and **Y** matrices. By multiplication of an updating matrix $(\mathbf{I}-\mathbf{wp}')$ of size $(\mathrm{M}\times\mathrm{M})$, explained variance is removed from the variance/covariance matrices: $(\mathbf{I}-\mathbf{wp}')'\mathbf{X}'\mathbf{YY}'\mathbf{X}(\mathbf{I}-\mathbf{wp}')$ (Lindgren et al, 1998).

3.2.4 SAMPLS Algorithm

SAMPLS (SAMple-distance Partial Least Squares) was presented by Bush et al. in 1993, and has been focused on the special case of many predictor variables and few observations M>>N. However, the algorithm handles only one **y** response variable, which is a limiting factor compared to other algorithms (Lindgren et al, 1998). It works with the association matrix $\mathbf{XX}'$ and the response vector **y** in order to calculate the latent vector without iteration.

3.2.5 UNIPALS Algorithm

UNIPALS (UNIversal Partial Least Squares) was presented by Glen in 1989. It is based on the matrix $\mathbf{Y}'\mathbf{XX}'\mathbf{Y}$ with size $(\mathrm{K}\times\mathrm{K})$. The largest eigenvector of this matrix corresponds to the first weight vector for the **Y** block and by the help of this vector all other PLS vectors can be calculated without iteration.

CHAPTER FOUR

MODEL SELECTION METHODS

Model selection and validation are critical subjects in predicting the performance of the regression models. In model selection, a statistical model is chosen from a set of potential models. Selecting the best model depends on the correct selection of variables, so the model prediction error is minimized and the model is prevented from redundant variables. There are several variable selection techniques. Some of them are explained in the next subsections.

Suppose that there is a data set with N observations and M predictor variables such as **X** and a response variable **y**. The problem of variable selection arises when one wants to model the relationship between **y** and a subset of predictor variables, but there is uncertainty about which subset to use (Baumann, 2003). The variable selection problem is often defined as selecting K<M variables that allow the construction of the best predictor.

There can be many reasons for selecting only a subset of the variables. It is cheaper to measure less variables and knowing which components are relevant can give insight into the nature of the prediction problem. So, the predictor to be built is usually simpler and potentially faster when less components are used, Also, prediction accuracy is improved through exclusion of irrelevant components.

This situation is difficult when N is small and M is big and the predictor variables are thought to contain many redundant or irrelevant variables. For M potential predictor variables, there are $2^{\mathbf{M}} - 1$ possible regression equations. For large M, it is not practical to consider all possible subsets. Therefore, a search algorithm that evaluates only a small portion of all possible subsets is needed.

Variable selection algorithms need two theme: a mathematical modelling procedure and an objective function guiding for the search. Some of the mathematical modelling techniques combined with variable selection are MLR, PCR,

PLSR and neural networks. In PCR and PLSR, predictor variables are reduced to fewer latent variables by the help of algorithms. But determining the correct number of latent variables is still one of the most difficult part.

The objective function is used for assessing the temporarily selected variable subsets during the search for the best model. The objective function should provide an estimate of the prediction error.

As more and more latent variables are calculated, they are ordered by the degree of importance for the model. The previous latent variables in the model are the most possible ones related to both variables. Latent variables that come later generally have less information that is useful for predicting response variable. If the model contains these latent variables, the predictions can be worse than if they were omitted together.

Various methods for choosing significant latent variables are used in the literature. Some of them are from simple to complex, scree plot and likelihood ratio tests. In this paper cross-validation which is a practical approach to guide the search or the selection process will be given.

In component selection, the aim is usually to find a small subset of the latent variables that enables the construction of accurate predictors. Consequently, the accuracies of the predictor to be built need to be estimated in order to know whether a good subset has been found.

4.1 Cross-Validation

One of the most important issues in any regression modelling is a concept of its predictive ability (prediction) power. This concept is essential as one needs to estimate the optimal number of latent variables in order to avoid the risk of obtaining models with over-fitting or under-fitting. This risk is reduced by using validation

procedures to determine the number of Latent variables that minimizes the prediction error. One of this validation procedures is known as cross-validation (CV) (Barros and Rutledge, 2004).

The glossary meaning of CV is "the division of data into two approximately equal sized subsets, one of which is used to estimate the parameters in some model of interest, and the other is used to assess whether the model with these parameter values fits adequately."

CV is a very popular technique for model selection and model validation. It is used for investigating the predictive validity of a linear regression equation. It is conceptually very simple to understand, but the most calculationally intensive method of optimizing a model. Besides, it is the most common approach to estimating the true accuracy of a given model and it is based on splitting the available sample between a training set and a validation set (Last, 2006).

As mentioned above, there are two sets of CV. Training set is a portion of a data set to fit (train) a model for prediction or classification of values but unknown in other (future) data. The training set is used in conjunction with validation and/or test sets that are used to evaluate different models. Second is the validation set. It is a portion of a data set used in data mining to assess the performance of prediction or classification models that are fit on a separate portion of the same data set (training set). Typically both the training and validation sets are randomly selected, and the validation set is used as a more objective measure of the performance of various models that are fit to the training data (and whose performance with the training set is therefore not likely to be a good guide to their performance with data that they were not fit to).

There are some types of cross validation. These are;

Holdout validation: Observations are chosen randomly from the initial sample to form the validation data, and the remaining observations are retained as the training

data. Normally, less than a third of the initial sample is used for validation data (wikipedia.org).

K-fold cross-validation: In k fold cross-validation, the original sample is partitioned into k subsamples that are approximately in the same size. From these k subsamples, a single subsample is retained as the validation data for testing the model, and the remaining k–1 samples are used as training data. The cross-validation process is then repeated k times, with each of the k subsamples being used exactly once as the validation data. The k results from the folds can then be averaged to produce a single estimation (wikipedia.org).

Leave-one-out cross-validation: This involves using a single observation from the original sample as the validation data and the remaining observations as the training data. This is repeated such that each observation in the sample is used once as at the validation data. This is the same as k-fold cross-validation where k is equal to the number of observations in the original sample. This method can be time-consuming for large data sets because it recalculates the models as many times as there are observations (wikipedia.org).

For all types of cross validation, PRESS is being calculated. It is calculated by building a model with a number of factors, then predicting training data set with this model. The sum of the squared difference between the predicted and observed values gives the PRESS value for that model. PRESS criterion is a measure of how well the use of the fitted values for a subset model can predict the observed responses of a dependent variable.

The PRESS value for the i^{th} observation is as follows:

$$\text{PRESS} = \sum_{i=1}^{n} \left(y_i - \hat{y}_{i(i)} \right)^2 \qquad (4.1)$$

where the notation $\hat{y}_{i(i)}$ is used for the fitted value. By the first subscript i, it is shown that it is a predicted value for the i^{th} case and by the second subscript (i), it is shown that i^{th} case is omitted when the regression function was fitted. The smaller PRESS value shows that it is the best model to predict. In some situations PRESS should reach a minimum and start to rise again.

The advantageous feature of cross-validation is its ability to estimate the performance of the model. Since the predicted samples are not the same as the samples used to build the model, the calculated PRESS value is a very good indication of the error in the accuracy of the model when used to predict unknown samples in the future.

The disadvantage of cross-validation is that it is time consuming. It requires the recalculating of the models for every sample left out and this takes time.

Selecting the components based on PRESS:

To avoid building a model that is either overfit or underfit, the number of components where the PRESS value reaches a minimum would be the obvious choice for the best model. While the minimum of the PRESS may be the best choice for predicting the particular set of samples, most likely it is not the optimum choice for predicting all unknown samples in the future. That is, the optimum number of factors was determined rather than the selection of the model, which yields a minimum in PRESS; the model selected is the one with the fewest number of factors such that PRESS for that model is not significantly greater than the minimum PRESS (Niazi and Azizi, 2008). A solution to this problem has been suggested in which the PRESS values for all previous factors are compared to the PRESS value at the minimum.

The ratio between these values known as the F-ratio can be calculated and assigned a statistical significance based on the number of observations;

$$F_{ratio_a} = \frac{PRESS_a}{PRESS_{min}} \tag{4.2}$$

Hypothesis for this test statistic can be given as follows:

$$\begin{aligned} H_0 &: PRESS_a = PRESS_{min} \\ H_1 &: PRESS_a > PRESS_{min} \end{aligned} \tag{4.3}$$

This F ratio is an indicator of the relative significance of each model with the number of components at the minimum of the PRESS. An F test can be used to determine the significance of PRESS values greater than the minimum (Niazi, Azizi, 2008). The number of components where the F ratio falls below a predefined significance level determines the optimum number of factors for a model used for predicting unknowns. In some references the probability of that level falling at or below 0.75 is suggested as determining the point at which adding a new component to the model.

In addition to the statistic above, Osten (1988) proposed an F test based criterion, where the F value is given by:

$$F = \frac{PRESS(a) - PRESS(a+1)}{K} \Big/ \frac{PRESS(a+1)}{NK - (a+1)K} \tag{4.4}$$

This criterion is compared with an F value, $F_{K,NK-(m+1)K,0.95}$ (Li, Morris and Martin, 2002). Also a model selection criterion called Wold's R can be calculated from the PRESS values. It can be explained as follows:

$$R = \frac{PRESS(a+1)}{PRESS(a)}, \tag{4.5}$$

where PRESS(m) denotes the PRESS after including the first a latent variables. Wold's R criterion terminates when R is greater than unity or a given threshold and hence A=a (Li et al, 2002).

PRESS is also used to calculate *goodness of prediction* value called Q^2. This statistic is based on the proportional error reduction of the PRESS of squares residuals. It can be written as:

$$Q^2 = 1 - \left[\sum_{i=1}^{N} \left(y_i - \hat{y}_{(i)} \right)^2 \Big/ \sum_{i=1}^{N} \left(y_i - \bar{y}_{(i)} \right)^2 \right] \tag{4.6}$$

In this formula $\sum_{i=1}^{N} \left(y_i - \bar{y}_{(i)} \right)^2$ is the sum of squares difference between observed and y_i and the mean $\bar{y}_{(i)}$ when the i^{th} observation is omitted (Quan, 1988).

Briefly, Q^2 is (1.0-PRESS/SS) where SS is the residual sum of squares of the previous dimension (Wold et al., 1993). This means that Q^2 renders a measure of the final's model predictive capability. It answers the question of how good predictions on the basis of known X data can be (M. Henningsson et al, 2001).

In the presence of outliers the Q^2 statistic can be negative, because it is sensitive to the choice of regressors and the inclusion of influential observations (Quan, 1988).

4.2 Akaike Information Criterion

This was developed by Hirotsugu Akaike under the name of "an information criterion (AIC)" in 1971 and was proposed in Akaike (1974). It is a measure of the goodness of fit of an estimated statistical model. It is a way of selecting a model from a set of models. Given a data set, several competing models may be ranked according to their AIC, with the one having the lowest AIC being the best.

For problems associated with a single variable and more than one response variables, there are two types of information criterion. With a single response variable (K=1) criterion is:

$$\mathrm{AIC}(a) = N\log\left(\hat{\sigma}^2\right) + 2a \tag{4.7}$$

where a is the number of model parameters, N is the number of observations, and $\hat{\sigma}^2$ is the maximum likelihood estimate of the variance of the response variable. $N\log\left(\hat{\sigma}^2\right)$ represents model accuracy, $2a$ relates to model parsimony.

$\hat{\sigma}^2 = \dfrac{RSS}{N}$. RSS is the residual sum of squares.

$$\mathrm{AIC}(a) = N\log(RSS) + 2a \tag{4.8}$$

For more than one response variable (K>1), multivariate version of AIC was given by Bedrick and Tsai (1994),

$$\mathrm{MAIC}(a) = N\left(\log\left|\hat{\Sigma}\right| + K\right) + 2d\left[Ka + K(K+1)/2\right] \tag{4.9}$$

where $d = N/[N-(a+K+1)]$ and $\hat{\boldsymbol{\Sigma}}$ is the maximum likelihood estimate of $\boldsymbol{\Sigma}$ (variance-covariance matrix of the response variable).

The multivariate version of AIC was given by Bozdogan (2000) under the multivariate normal assumption for the multivariate regression model which are given as follows,

$$\mathrm{MAIC} = NM\log(2\pi) + N\log\left|\hat{\Sigma}\right| + NM + 2\left[aK + \frac{a(a+1)}{2}\right]. \tag{4.10}$$

CHAPTER FIVE

DESIGN OF SIMULATION STUDY AND RESULTS

Partial Least Squares Regression Analysis is partitioned into PLS and MLR. In PLS partition, dimension reduction is being done. After this reduction, latent variables, which are the new predictor variables, are used in regression partition. These latent variables are fewer than predictor variables. But, as all the latent variables can be used in regression, also fewer of them can be more sufficient in regression analysis. This sufficiency is achieved by describing the variance with both predictor variables and response variables.

To obtain the most relevant or sufficient latent variables, some model selection criteria were developed. Some of these criteria depend on describing the percentage of the variance or minimum error. In model selection criteria, k fold cross validation is used, followed by two different multivariate Akaike Information Criterion from Bozdogan, Bedrick and Wold's R criteria. The optimum latent variable number which is obtained from PRESS criterion was used.

In this thesis the model selection criteria was used for PLS model selection and their performances were evaluated by a Monte Carlo simulation study. The analysis including all simulations and calculations and all the data sets were generated randomly on MATLAB environment.

5.1 Design of Simulation Study

The framework for the simulation model was based on Li and Morris (2002) for the problem of multiple response variables. In this study the true number of latent variables is shown with A^*.

The dimensions of predictor variables is extended as $N \times 6$, $N \times 8$, $N \times 10$ and $N \times 12$. The dimension of response variables matrix, **Y**, is chosen as $N \times 4$.

Explanatory data matrix, **X**, was generated from equation (5.1):

$$\mathbf{X} = \sum_{i=1}^{A^*} \mathbf{r}_i \, \xi_i' + \mathbf{E} \qquad (5.1)$$

The components of **X** matrix are given in Table 5.1 and Table 5.2.

Table 5.1 The **R** and **E** matrices for **X** matrix**.**

Dimension of data matrix	**R**	**E**
N×6 N×8 N×10 N×12	**R=[$\mathbf{r_1}$, $\mathbf{r_2}$, $\mathbf{r_3}$, $\mathbf{r_4}$]**, i=1,2,3,4 were generated as; mutually independent normal variables with mean zero and var($\mathbf{r_1}$)=10 var($\mathbf{r_2}$)=5 var($\mathbf{r_3}$)=2 var($\mathbf{r_4}$)=0.5	**E=[$\mathbf{e_1}$, $\mathbf{e_2}$, $\mathbf{e_3}$, $\mathbf{e_4}$, $\mathbf{e_5}$, $\mathbf{e_6}$]** , j=1,…,6 were generated as; mutually independent random variables with mean zero and var($\mathbf{e_j}$)=0.01.

Table 5.2 The generated orthogonal vectors for ξ_i'.

$N \times 6$	$N \times 8$
$\begin{bmatrix} 0.4082 & 0.7071 & 0.4082 & 0.2887 \\ 0.4082 & -0.7071 & 0.4082 & 0.2887 \\ 0.4082 & 0 & -0.8165 & 0.2887 \\ 0.4082 & 0 & 0 & -0.8660 \\ 0.4082 & 0 & 0 & 0 \\ 0.4082 & 0 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 0.1612 & 0.3030 & 0.4082 & 0.4642 \\ 0.3030 & 0.4642 & 0.4082 & 0.1612 \\ 0.4082 & 0.4082 & 0 & -0.4082 \\ 0.4642 & 0.1612 & -0.4082 & -0.3030 \\ 0.4642 & -0.1612 & -0.4082 & 0.3030 \\ 0.4082 & -0.4082 & 0 & 0.4082 \\ 0.3030 & -0.4642 & 0.4082 & -0.1612 \\ 0.1612 & -0.3030 & 0.4082 & -0.4642 \end{bmatrix}$
$N \times 10$	$N \times 12$
$\begin{bmatrix} 0.1201 & 0.2305 & 0.3223 & 0.3879 \\ 0.2305 & 0.3879 & 0.4221 & 0.3223 \\ 0.3223 & 0.4221 & 0.2305 & -0.1201 \\ 0.3879 & 0.3223 & -0.1201 & -0.4221 \\ 0.4221 & 0.1201 & -0.3879 & -0.2305 \\ 0.4221 & -0.1201 & -0.3879 & 0.2305 \\ 0.3879 & -0.3223 & -0.1201 & 0.4221 \\ 0.3223 & -0.4221 & 0.2305 & 0.1201 \\ 0.2305 & -0.3879 & 0.4221 & -0.3223 \\ 0.1201 & -0.2305 & 0.3223 & -0.3879 \end{bmatrix}$	$\begin{bmatrix} 0.0939 & 0.1823 & 0.2601 & 0.3228 \\ 0.1823 & 0.3228 & 0.3894 & 0.3667 \\ 0.2601 & 0.3894 & 0.3228 & 0.0939 \\ 0.3228 & 0.3667 & 0.0939 & -0.2601 \\ 0.3667 & 0.2601 & -0.1823 & -0.3894 \\ 0.3894 & 0.0939 & -0.3667 & -0.1823 \\ 0.3894 & -0.0939 & -0.3667 & 0.1823 \\ 0.3667 & -0.2601 & -0.1823 & 0.3894 \\ 0.3228 & -0.3667 & 0.0939 & 0.2601 \\ 0.2601 & -0.3894 & 0.3228 & -0.0939 \\ 0.1823 & -0.3228 & 0.3894 & -0.3667 \\ 0.0939 & -0.1823 & 0.2601 & -0.3228 \end{bmatrix}$

ξ_i' are orthogonal and unit vectors. Response variables matrix is generated from equation (5.2).

$$\mathbf{Y} = \sum_{i=1}^{A^*} \mathbf{z}_i \boldsymbol{\eta}'_{A_i^*} + \boldsymbol{\varphi} = \sum_{i=1}^{A^*} \mathbf{r}_i \boldsymbol{\eta}'_{A_i^*} + \mathbf{F}_{A^*} \tag{5.2}$$

$$\mathbf{F} = \sum_{i=1}^{A^*} \mathbf{f}_i \boldsymbol{\eta}'_i + \boldsymbol{\varphi} \tag{5.3}$$

Table 5.3 The generated values for φ

Dimension of data matrix	φ
$N\times 6$ $N\times 8$ $N\times 10$ $N\times 12$	$\varphi=[\varphi_1, \varphi_2, \varphi_3, \varphi_4]$ was generated multivariate normal distribution with mean zero and following variance-covariance matrix; $\begin{bmatrix} 0.00010 & 0.00006 & 0.00006 & 0.00006 \\ 0.00006 & 0.00010 & 0.00006 & 0.00006 \\ 0.00006 & 0.00006 & 0.00010 & 0.00006 \\ 0.00006 & 0.00006 & 0.00006 & 0.00010 \end{bmatrix}$

Table 5.4 The generated orthogonal vectors for η_i .

$N\times 6$	$N\times 8$
$\begin{bmatrix} 0.500 & 0.500 & 0.500 & 0.500 \\ 0.7071 & -0.7071 & 0 & 0 \\ 0.4082 & 0.4082 & -0.8165 & 0 \\ 0.2887 & 0.2887 & 0.2887 & -0.8660 \end{bmatrix}$	$\begin{bmatrix} 0.2887 & 0.500 & 0.5774 & 0.500 \\ 0.500 & 0.500 & 0 & -0.500 \\ 0.5774 & 0 & -0.5774 & 0 \\ 0.500 & -0.500 & 0 & 0.500 \end{bmatrix}$
$N\times 10$	$N\times 12$
$\begin{bmatrix} 0.3717 & 0.6015 & 0.6015 & 0.3717 \\ 0.6015 & 0.3717 & -0.3717 & -0.6015 \\ 0.6015 & -0.3717 & -0.3717 & 0.6015 \\ 0.3717 & -0.6015 & 0.6015 & -0.3717 \end{bmatrix}$	$\begin{bmatrix} 0.6935 & 0.5879 & 0.3928 & 0.1379 \\ 0.5879 & -0.1379 & -0.6935 & -0.3928 \\ 0.3928 & -0.6935 & 0.1379 & 0.5879 \\ 0.1379 & -0.3928 & 0.5879 & -0.6935 \end{bmatrix}$

Table 5.5 The generated data for **Y.**

Dimension of data matrix	**F**
$N\times 6$ $N\times 8$ $N\times 10$ $N\times 12$	$\mathbf{F}=[\mathbf{f_1}, \mathbf{f_2}, \mathbf{f_3}, \mathbf{f_4}]$, i=1,2,3,4 were generated as; mutually independent normal variables with mean zero and $var(\mathbf{f_1})=0.25$ $var(\mathbf{f_2})=0.125$ $var(\mathbf{f_3})=0.05$ $var(\mathbf{f_4})=0.0125$

Table 5.1-Table 5.5 show how data matrices are generated. After generating of all data sets, the Variance Inflation Factor is calculated for $N\times 6$ design matrix in Minitab to see there is multicollinearity or not. The VIF values are shown in Table 5.6.

Table 5.6 VIF values for $N\times 6$.

Predictors	VIF
X_1	635,6
X_2	439,9
X_3	990,8
X_4	765,8
X_5	626,9
X_6	839,0

Then the frequencies of the selected number of latent variables are calculated.

Table 5.7 Relative cumulative variances of **X** and **Y** for $N \times 6$.

True Model	Blocks	Number of Latent Variables					
		1	2	3	4	5	6
A*=4	X-block	0,54196	0,88199	0,95207	0,99984	1,00000	1,00000
	Y-block	0,93226	0,96577	0,97492	0,97530	0,97530	0,97519

As seen from Table 5.7 the true number of latent variables is 4.

5.2 Results of Simulation Study

MATLAB code is written for k-fold cross validation in Modified Kernel Algorithm. k=5 is chosen and the simulation is repeated 10000 times for all design matrix. N is chosen as 100, 250 and 500. The comparison of results are shown in Figure 5.1-Figure5.6.

*(NOTE: in this study 6*4 is a design matrix represents $N \times 6$ and means that the number of predictor variables is 6 and these variables are reduced to number 4 for the number of latent variables. This is the same for 8*4, 10*4 and 12*4).*

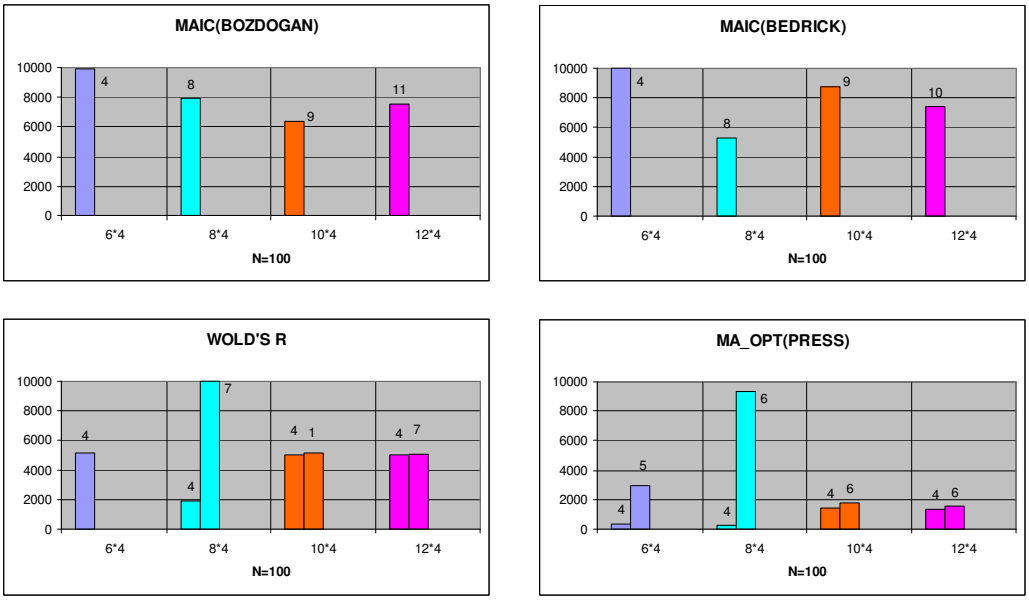


Figure 5.1 All model selection criteria for N=100.

These figures show the maximum iteration number for each design for N=100. As is shown, each criteria finds the true number of latent variables in 10000 iteration for $N \times 6$. But for other-sized design matrices, they find the number of latent variables with a higher number, and they cannot find the true number of latent variables.

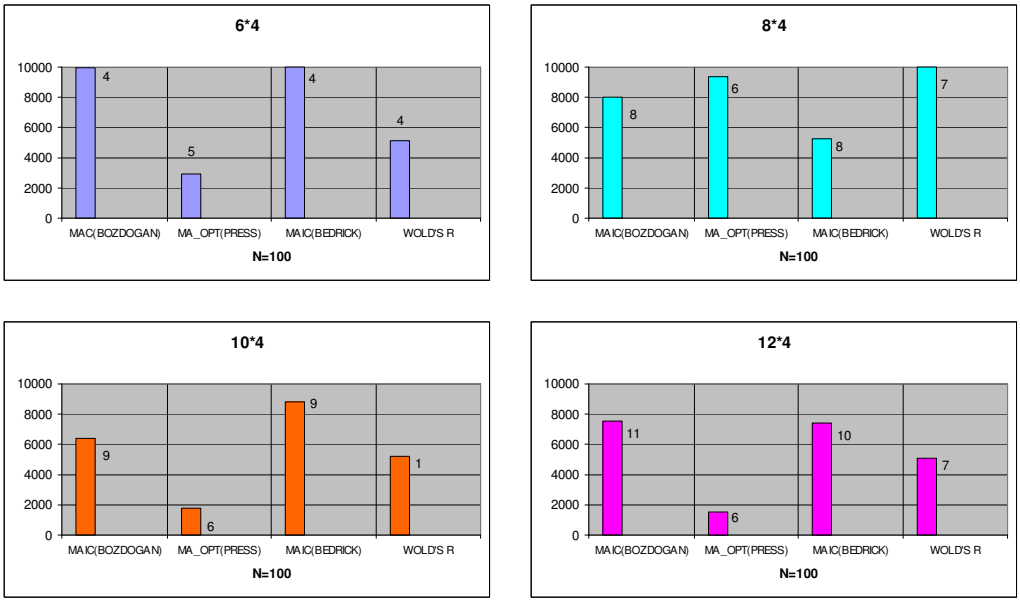


Figure 5.2 Model selection criterias for each design matrix for N=100.

These figures display the number of latent variable for each model selection criterion in each design matrix for N=100. All model selection criteria find the true number of latent variables in $N \times 6$ design matrix. But when the number of predictor variables increases, they find the number of latent variable close to the number of predictor variables.

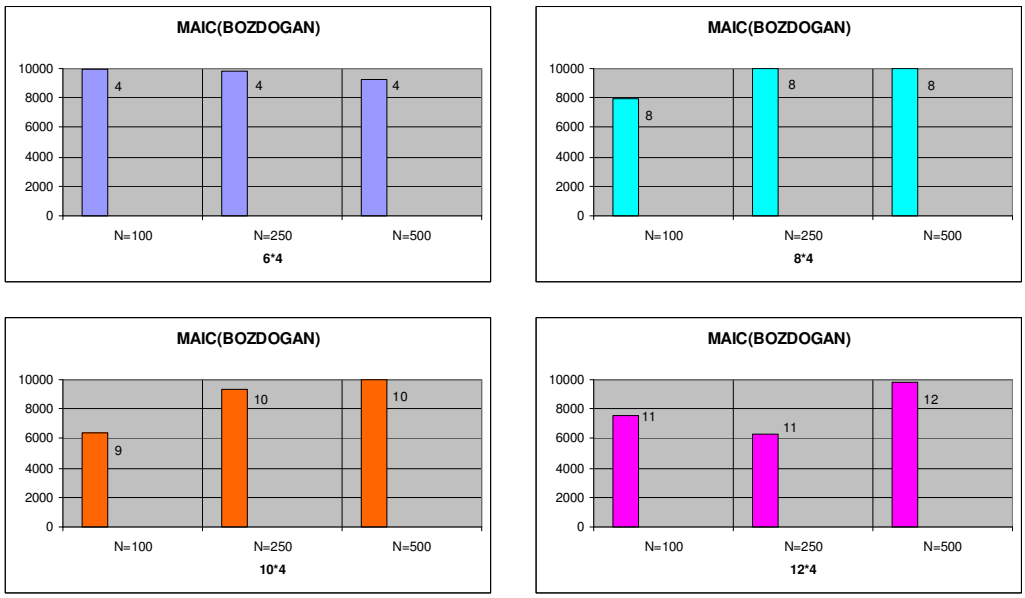


Figure 5.3 MAIC(BOZDOGAN) criterion for each design matrix for each N.

In these figures MAIC(BOZDOGAN) criterion is displayed for each design matrix for each N. In $N \times 6$ design matrix, it finds the true number of latent variables but for other design matrix, it finds the number of latent variables close to the number of predictor variables.

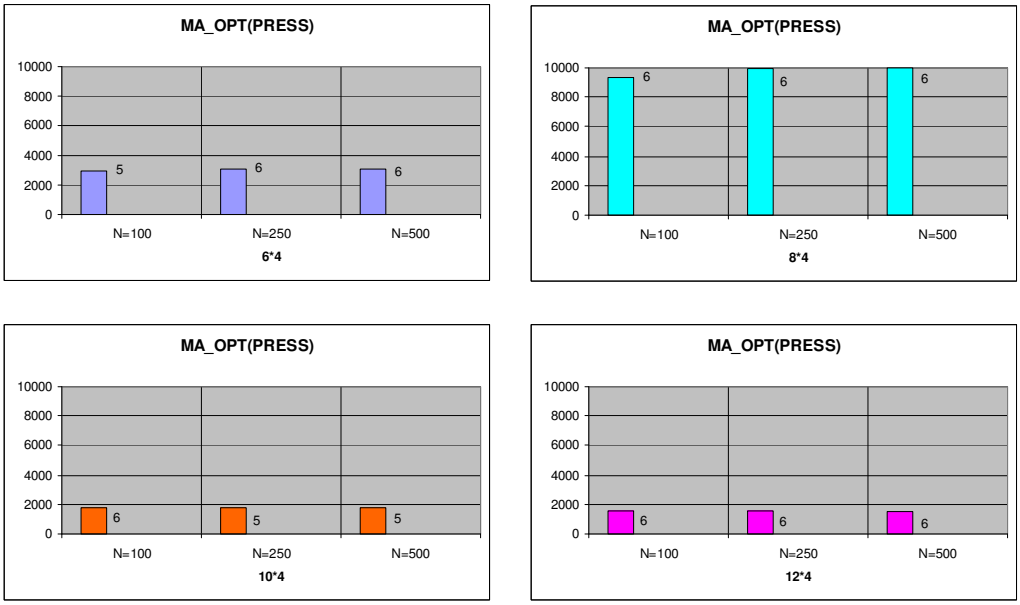


Figure 5.4 MA_OPT(PRESS) criterion for each design matrix for each N.

In these figures MA_OPT(PRESS) criterion is displayed for each design matrix for each N. As is shown, it cannot find the true number of latent variables but it finds the number of latent variables close to the true number of predictor variables.

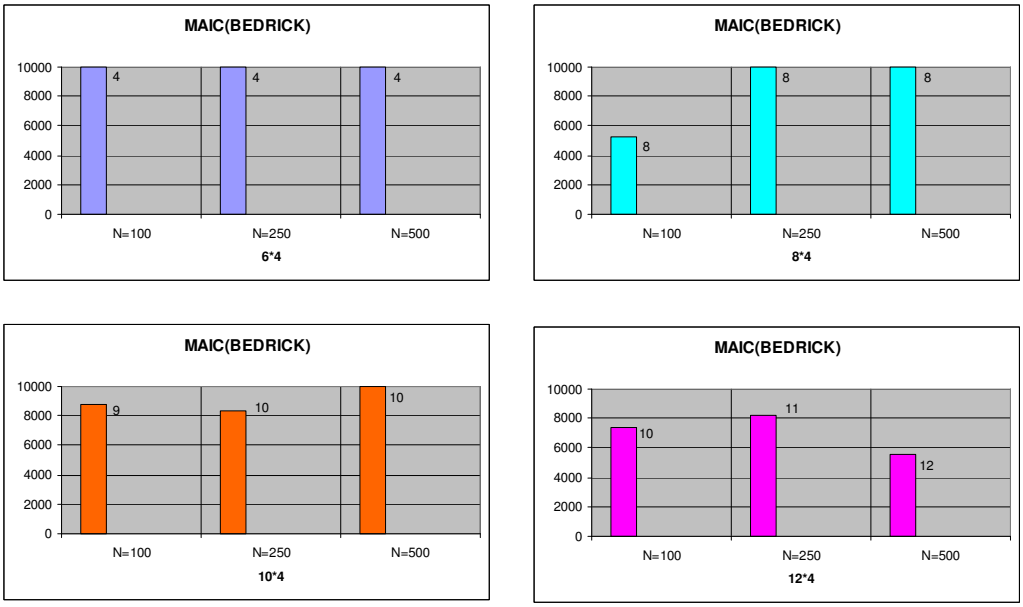


Figure 5.5 MAIC (BEDRICK) criterion for each design matrix for each N.

MAIC(BEDRICK) criterion is displayed for each of the design matrices and for each the number of observations. As is shown, it finds the true latent variable number in $N \times 6$ design matrix. When the design matrix and the number of observations get larger, it finds the number of latent variables close to number of predictor variables.

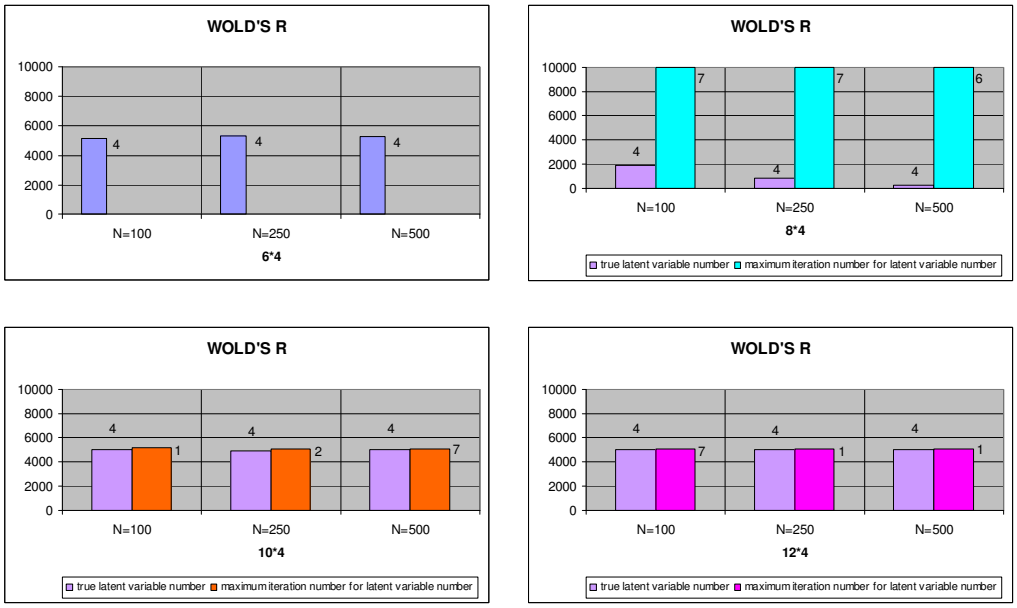


Figure 5.6 WOLD'S R criterion for each design matrix for each N.

WOLD'S R criterion is displayed for each of the design matrices and for each object number. As is shown, these figures are given with a true number of latent variables and the number of latent variable which is the most iterated.

CHAPTER SIX

CONCLUSION

In this thesis a Monte Carlo simulation study was done based on the paper of Li and Morris (2002). The paper's simulation study was extended for high dimensional data. For more information, see Li and Morris (2002).

The data was generated in MATLAB statistical program with the number 6, 8,10 and 12 predictor variables. The number of response variable was taken as 4. The observation numbers were taken as 100, 250 and 500, respectively. These data matrices were generated in terms of PLSR assumptions and according to true number of latent variables which is equal to 4. The code for k-fold cross-validation was written and put into Modified Kernel algorithm. k was taken as 5. Model selection criteria were applied to data to compare the performance of criteria in order to find the true number of latent variables. The details were given in Chapter 5.

Main contribution of this thesis is comparing the performance of criteria in order to find the true number of latent variables for high dimensional data which resembles the study of Li and Morris. Li and Morris indicated that all criteria are effective for the small-sized design matrices. Especially WOLD'S R criterion gave the best results in finding the true latent variable number. Working with high dimensional data matrices, the reaction of these criteria, especially the reaction of Wold's R was wondered by the researcher. Then the simulation study was done according to the interest of criteria's performance, especially WOLD'S R.

In the simulation study, firstly, the same results were obtained for the same sized data matrices of Li and Morris. Afterwards, data matrices were extended for larger number of predictor variables and observations. 10000 iterations were done for each design matrices. The results were given in Chapter 5 and Appendix 4.

The simulation results show that all criteria achieved the true number of latent variables for small-sized design matrices. However, the results for the other-sized

design matrices varied greatly and they consistently showed different numbers of latent variables. Generally it can be said that, when N increases, PLS creates a model with a high number of latent variables, which is statistically significant. The simulation studies also show that WOLD'S R criterion is effective for a 6*4 design matrix. That is, WOLD'S R gave the same result as in Li and Morris when the data was generated according to their paper. Also, when the data was generated with nonorthogonal vectors, 8*4, 10*4 and 12*4, as the same as Li and Morris, WOLD'S R gave the best results. Nevertheless, when the data was generated according to the assumptions of PLSR, it seems that WOLD'S R criterion did not give desirable results in high dimensional data. MAIC(BOZDOGAN) and MAIC(BEDRICK) found almost the same results as the number of latent variables but for high dimensional data they could not find the true number of latent variables. MA_OPT(PRESS) gave the same or nearly the same results with WOLD'S R criterion.

In the simulation study, it is shown that, for high dimensional data matrices, although all design matrices were generated as 4 was the true number of latent variables, all of the model selection criteria found the number of latent variables close to the number of predictor variable.

REFERENCES

Akaike, H. (1971). Autoregressive Model Fitting for Control. *Ann. Inst. Statistics Math.* Vol, 23, 163-180.

Akaike, H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19, 716–723.

Algorithm, (n.d.) Retrieved June 27, 2007 from http://www.answers.com/topic/algorithm?cat=biz-fin.

Barros, A., & Rutledge, D. (2004). Principal Component Transform-Partial Least Squares:A novel method to accelerate cross-validation in PLS regression. *Chemometrics and Intelligent Laboratory System*, 73, 245-255.

Baumann, K. (2003). Cross-validation as the Objective Function for variable Selection Techniques. *Trends in Analytical Chemistry*, 6.

Bedrick, E. J., & Tsai, C. L. (1994). Model Selection for Multivariate Regression in Small Samples. *Biometrics*, 50, 226-231.

Bowerman, B. L., & O'Connell R. T. (1990). *Linear Statistical Models*. Pws Pub Co.

Cross-validation, (n.d.) Retrieved January 06, 2010 from http://en.wikipedia.org/wiki/Cross-validation_(statistics).

Dayal, B., & MacGregor., J. (1997). Improved PLS algorithms. *Journal of Chemometrics*, Vol. 11, 73-85.

De Jong, S. (1993). SIMPLS: an alternative approach to partial least squares regression. *Chemometrics and Intelligent Laboratory Systems*, 18. 251-263.

De Jong, S., & Ter Braak, C. J. F. (1994). Comments on the PLS kernel algorithm *Journal of Chemometrics*, Vol. 8, 169 – 174.

Garthwaite, P. H. (1994). An Interpretation of Partial Least Squares. *Journal of the American Statistical Association* 89, 122-127.

Geladi, P., & Kowalski, R. (1986). Partial Least Squares Regression: A Tutorial. *Analytica Chimica Acta* 185, 1-17.

Geladi, P. (1988). Notes on the History and Nature of Partial Least Squares (PLS) Modelling. *Journal of Chemometrics* 2, 231-246.

Gerlach, R. W., Kowalski, B. R., & Wold, H. (1979). Partial Least Squares Modelling With Latent Variables. *Analytica Chimica Acta*. 417-421.

Glen, W. G., Dunn III, W. J., & Scott, D. R. (1989). Principal Components Analysis and Partial Least Squares Regression, *Tetrahedron Computational Methodology*, 2, 349–376.

Helland, I. S. (1990). Partial Least Squares Regression and Statistical Models. *Scandinavian Journal of Statistics* 17, 97-114.

Henningsson, M., Sundbom E., Armelius, B. A., & Erdberg, P. (2001). PLS Model Building: A Multivariate Approach to Personality Test Data. *Scandinavian Journal of Psychology*, 42, 399-409.

Höskuldson, A. (1988). PLS Regression Methods. *Journal of Chemometrics* 2, 211-228.

Ian, W., & Morris, J. (1993). A Test of Significance for the Least Squares Regression. *Journal of Chemometrics*, 7, 291-304.

Johnson, E. D. (1998). *Applied Multivariate Methods for Data Analysis*. Duxbury Press.

Kiers, A. L. H., & Smilde, A. K. (2007). A Comparison of Various Methods for Multivariate Regression With Highly Collinear Variables. *Statistical Methods and Applications*, 16, 193-228.

Last, M. (2006). *The Uncertainty principle of cross-validation*. Member, IEEE.

Li, B., Morris, J., & Martin, B. (2002). Model Selection for Partial Least Squares Regression. *Chemometrics and Intelligent Laboratory Systems,* 64, 79-89.

Lindgren, F., Geladi, P., & Wold, S. (1993). The Kernel Algorithm for PLS. *Journal of Chemometrics*, 7, 45-59.

Lindgren, F., & Rannar, S. (1998). Alternative Partial Least-Squares (PLS) Algorithm. *Perspective in Drug Discovery and Design*, 12/13/14. 105-113.

Lorber, A., Wangen, L. E., & Kowalski, B. R. (1987). A theoretical foundation for the PLS algorithm. *Journal of Chemometrics*, 1, 19-31.

Martens, H., & Naes, T. (1989). *Multivariate Calibration*. John Wiley & Sons.

Miller, R. G. (1974). The jackknife - A review. *Biometrika*, 61, 1-15.

Niazi, A., & Azizi, A. (2008). Orthogonal Signal Correction-Partial Least Squares Method for Simultaneous Spectrophotometric determination of Nickel, Cobalt and Zinc. *TUBITAK*, 32, 217-228.

Osten, D. W. (1988). Selection of optimal regression models via cross-validation. *Journal of Chemometrics*, 2, 39-48.

Rännar, S., Lindgren, F., Geladi, P., & Wold, S. (1994). A PLS Kernel Algorithm For Data Sets With Many Variables and Fewer Objects. Part 1: Theory and Algorithm. *Journal of Chemometrics*, Vol. 8, 111-125.

Quan, N. T. (1988). The Prediction Sum of Squares as a General Measure for Regression Diagnostics. *Journal of Business & Economic Statistics*, 6, 501-504.

Wold, H. (1985). *Partial Least Squares Encyclopedia of Statistical Sciences*. New York: Wiley, 6, 581-591.

Wold, S., Sjöström, M., Eriksson, L. (2001). PLS-regression:a basic tool of chemometrics. *Chemometrics and Intelligent Laboratory Systems* 58, 109-130.

Wold, S., & Eriksson, L. (2004). *PLS method-partial least squares projections to latent structures and its applications in industrial RDP*. Prague.

APPENDICES

Appendix 1. Notations

Matrices are denoted with bold upper letters, vectors are denoted with bold lower letters.

a → index of components (a=1,2,…,A)

A → number of components in PLS model

i → index of observations (i=1,2,…,N)

N → number of observations

M → number of predictor variables (m=1,2,…,M)

K → number of response variables (k=1,2,…,K)

$\mathbf{X}$ → matrix of predictor variables with dimension $(N \times M)$

$\mathbf{Y}$ → matrix of response variables with dimension $(N \times K)$

$\mathbf{b_m}$ → regression coefficient for the mth predictor variable.

$\mathbf{B}$ → matrix of regression coefficients of all Y's $(M \times K)$

$\mathbf{c_a}$ → PLSR Y weights of component a $(K \times 1)$

$\mathbf{C}$ → Y weight matrix $(K \times A)$

$\mathbf{E}$ → matrix of X residuals $(N \times M)$

$\mathbf{f_a}$ → residuals of a th component on y variable $(N \times 1)$

$\mathbf{F}$ → matrix of Y residuals $(N \times K)$

$\mathbf{p_a}$ → X loading vector of component a $(M \times 1)$

$\mathbf{P}$ → Loading matrix $(M \times A)$

$\mathbf{t_a}$ → X scores of component a $(N \times 1)$

$\mathbf{T}$ → latent variable (score) matrix $(N \times A)$

$\mathbf{w_a}$ → X weight of component a $(M \times 1)$

$\mathbf{W}$ → X weight matrix $(M \times A)$

Appendix 2. Abbreviations

CV	→	Cross-validation
EVD	→	Eigenvalue Decomposition
OLS	→	Ordinary Least Squares
MLR	→	Multiple Linear Regression
MSE	→	Mean Square Error
NIPALS	→	Non-Linear Iterative Partial Least Squares
PCA	→	Principal Component Analysis
PCR	→	Principal Component Regression
PLS	→	Partial Least squares
PLSR	→	Partial Least Squares Regression
PRESS	→	Predicted Residual Sum of Squares
SIMPLS	→	Straightforward Partial Least Squares
SVD	→	Singular Value Decomposition
VIF	→	Variance Inflation Factor

Appendix 3. MATLAB Code for $N \times 6$

```
clc;
clear;
maic=[];
maic1=[];
maic2=[];
MAKAIKE=[];
MBEDRICK=[];
MNEW=[];
PRESS=[];
NORMPRESS2=[];
saydir=zeros(6,1);
N=100;
Woldlar=zeros(counter,5);

for counter=1:10000

  X=[];
  Y=[];

   E=mvnrnd([0 0 0 0 0 0],[0.01 0 0 0 0 0;0 0.01 0 0 0 0;0 0 0.01 0 0 0;0 0 0 0.01 0
0;0 0 0 0 0.01 0;0 0 0 0 0 0.01],N);

   R=mvnrnd([0 0 0 0],[10 0 0 0;0 5 0 0;0 0 2 0;0 0 0 0.5],N);

  zeta =[ 0.4082    0.4082    0.4082    0.4082    0.4082   0.4082;
          0.7071   -0.7071         0         0         0        0;
          0.4082    0.4082   -0.8165         0         0        0;
          0.2887    0.2887    0.2887   -0.8660         0        0];

   X=(R*zeta)+E;
```

```
fi=mvnrnd([0 0 0 0],[0.25 0 0 0;0 0.125 0 0;0 0 0.05 0;0 0 0 0.0125],N);

 eta =[  0.5000    0.5000    0.5000    0.500;
        0.7071   -0.7071         0         0;
        0.4082    0.4082   -0.8165         0;
        0.2887    0.2887    0.2887   -0.8660];

 pisi=mvnrnd([0 0 0 0],[0.00010 0.00006 0.00006 0.00006;
          0.00006 0.00010 0.00006 0.00006;
          0.00006 0.00006 0.00010 0.00006;
          0.00006 0.00006 0.00006 0.00010],N);

F=(fi*eta)+pisi;
 Y=(R*eta)+F;
 SYY=Y'*Y;
 SXX=X'*X;

%k-FOLD CROSS-VALIDATION PROCEDURE k=5

cr=5;
PRESS=[];
latent=4;
for cr=1:5

   if cr=1

        Xd = X(bol+1:N,:);
        Yd = Y(bol+1:N,:);
        Y_cr=Yd;
        X_cr=Xd;
        SXX_cr=X_cr'*X_cr;
```

```
    SXY_cr=X_cr'*Y_cr;

  for a=1:latent;
    P_cr=[];
    R_cr=[];
    BETA_cr=[];
    C_cr=[];

          for i=1:a,
     [c_cr s_cr w_cr]=svds(SXY_cr'*SXX_cr,1);
     r_cr=w_cr;
     tt_cr=r_cr'*SXX_cr*r_cr;
    p_cr=(r_cr'*SXX_cr)'/tt_cr;
    c_cr=(r_cr'*SXY_cr)'/tt_cr;
    SXY_cr=SXY_cr-p_cr*c_cr'*tt_cr;
    C_cr=[C_cr c_cr];
    R_cr=[R_cr r_cr];
   P_cr=[P_cr p_cr];
          end

    BETA_cr=R_cr*C_cr';
  end

     for i=1:size(X,2)
        XXa(1:bol,i)=X(1:bol,i);
        Yacap(1:bol,i)=XXa(1:bol,i)*BETA_cr(i,m);
        Hata(1:bol,i)=Y(1:bol,m)-Yacap(1:bol,i);
   end

        HHata(1:bol,:,m)=Hata(1:bol,:);
 end
elseif cr>1 && cr<5
```

```
    Xd = [X(1:((cr-1)*bol),:) ; X((cr*bol)+1:n,:)];
    Yd = [Y(1:((cr-1)*bol),:) ; Y((cr*bol)+1:n,:)];
    SXX_cr=X_cr'*X_cr;
    SXY_cr=X_cr'*Y_cr;

for a=1:latent;

   P_cr=[];
   R_cr=[];
   BETA_cr=[];
   C_cr=[];

    for i=1:a,

       [c_cr s_cr w_cr]=svds(SXY_cr'*SXX_cr,1);
       r_cr=w_cr;

       if i>1,
          for j=1:(i-1),
             r_cr=r_cr-(P_cr(:,j)'*w_cr)*R_cr(:,j);
          end
       end

    tt_cr=r_cr'*SXX_cr*r_cr;
    p_cr=(r_cr'*SXX_cr)'/tt_cr;
    c_cr=(r_cr'*SXY_cr)'/tt_cr;
    SXY_cr=SXY_cr-p_cr*c_cr'*tt_cr;
    C_cr=[C_cr c_cr];
    R_cr=[R_cr r_cr];
    P_cr=[P_cr p_cr];
    end
```

```
    BETA_cr=R_cr*C_cr';
  end

  for m=1:size(Y,2)

      HHata((cr-1)*bol+1:cr*bol,:,m)=Hata((cr-1)*bol+1:cr*bol,:);
  end

elseif cr=5

    Xd = X(1:(cr-1)*bol,:);
    Yd = Y(1:(cr-1)*bol,:);
        SXX_cr=X_cr'*X_cr;
    SXY_cr=X_cr'*Y_cr;

    for a=1:latent;
    P_cr=[];
    R_cr=[];
    BETA_cr=[];
    C_cr=[];

    for i=1:a,

      [c_cr s_cr w_cr]=svds(SXY_cr'*SXX_cr,1);
      r_cr=w_cr;

      if i>1,
        for j=1:(i-1),
          r_cr=r_cr-(P_cr(:,j)'*w_cr)*R_cr(:,j);
        end
      end
    tt_cr=r_cr'*SXX_cr*r_cr;
```

```
        p_cr=(r_cr'*SXX_cr)'/tt_cr;
        c_cr=(r_cr'*SXY_cr)'/tt_cr;
        SXY_cr=SXY_cr-p_cr*c_cr'*tt_cr;
        C_cr=[C_cr c_cr];
        R_cr=[R_cr r_cr];
        P_cr=[P_cr p_cr];
        end

      BETA_cr=R_cr*C_cr';
    end

      for m=1:size(Y,2)

          HHata((cr-1)*bol+1:n,:,m)=Hata((cr-1)*bol+1:n,:);
      end
  end
end

% COMPARISON FOR VARIABLE SELECTION METHODS

% 1- NORMPRESS
    for p=1:size(Y,2)

        PRESS=[PRESS real(diag((HHata(:,:,p))'*(HHata(:,:,p))))];
    end

        NORMPRESS=[];

        for p=1:size(PRESS,1),

      NORMPRESS=[NORMPRESS norm(PRESS(p,:))];
      NORMPRESS1=NORMPRESS';
```

```
end

    minp=find(NORMPRESS1==min(NORMPRESS1(:,1)));
    NORMPRESS2(sayac,1) =minp;

 % 2- WOLD'S R

   ree=1;
   oran=0;

   for b=1:(size(NORMPRESS1,1)-1);
      oran=NORMPRESS1(b+1)/NORMPRESS1(b);
      Wold(sayac,b)=oran;

      if (oran>=1)

       Woldlar(counter,b)=Woldlar(counter,b);
      end
    b=b+1;
   end

SumWold=sum(Woldlar(:,:));
  A_opt=find(NORMPRESS1==min(NORMPRESS1));
  MA_opt(sayac,1) =A_opt;
  MA_opt(sayac,1)=MA_opt(sayac,1);
  saydir(A_opt)=saydir(A_opt)+1;
  SayNormpress2=find(4==NORMPRESS2(:,:));

% 3- MAIC

XX=[];
for i=1:size(X,2);
```

```
    XX(:,i)=X(:,i);
    q=size(XX,2);
    p=size(Y,2);
    d=n/(n-(q+p+1));
    I=eye(n);
    sigma=Y'*(I-(XX*inv(XX'*XX)*XX'))*Y*(1/n);

makaike(counter,i)=n*p*log10(2*pi)+n*log10(det(sigma))+n*p+2*(p*q+0.5*p*(p+
1));  % Akaike from Bozdogan
    mbedrick(counter,i)=n*log10(det(sigma+p))+2*d*(p*q+0.5*p*(p+1));  %Bedrick
criterion
end

    mini=find(makaike==min(makaike(sayac,:)));
    MAKAIKE(sayac,1) =mini;
    MAKAIKE(sayac,1)=MAKAIKE(sayac,1)*1/sayac;

    mini1=find(mbedrick==min(mbedrick(sayac,:)));
    MBEDRICK(sayac,1) =mini1;
    MBEDRICK(sayac,1)=MBEDRICK(sayac,1)*1/sayac;

end
```

Appendix 4. Results of Simulation Study

The figures following Chapter 5 are as follows.

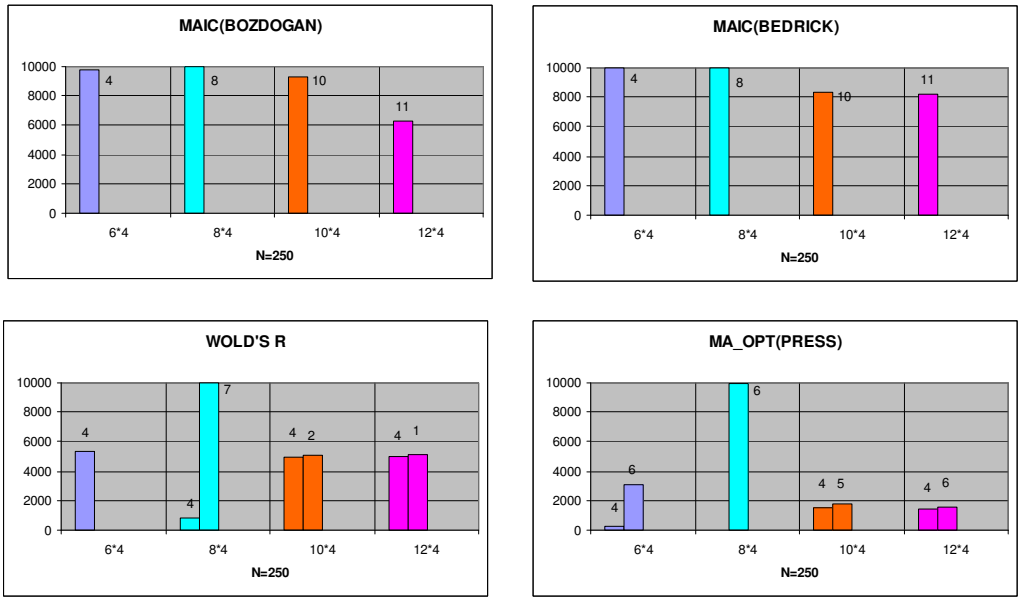


Figure A4.1 All model selection criteria for N=250.

These figures show the maximum iteration number for each design for N=250. All criteria find 4 as the true number of latent variables in 10000 iteration in $N \times 6$ except MA_OPT(PRESS). But when the predictor variable number increases, they find latent variable with a higher number and they cannot find the true number of latent variables.

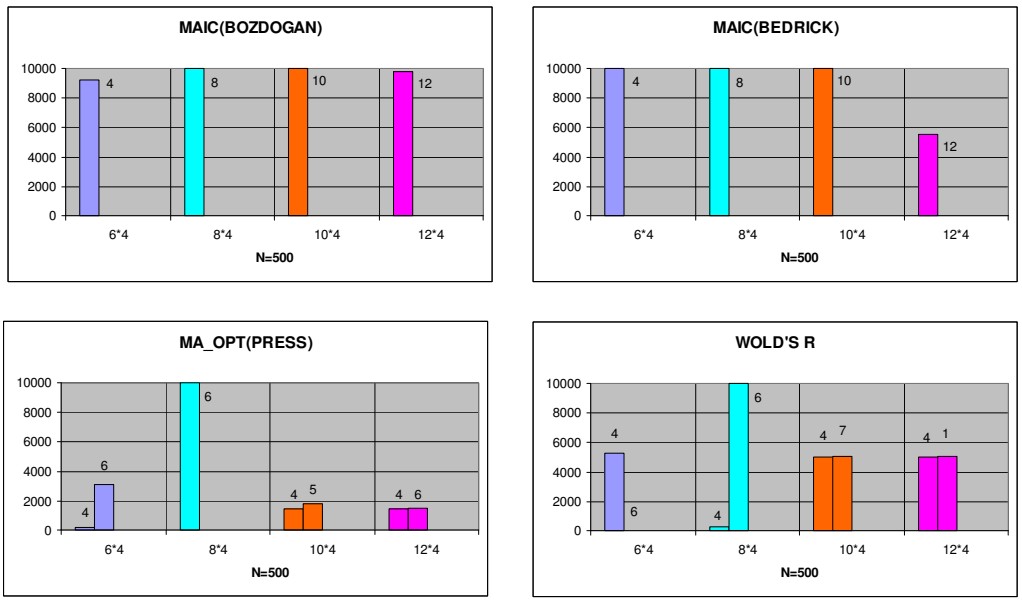


Figure A4.2 All model selection criteria for N=500.

These figures show the maximum iteration number for each design for N=500. All criteria find 4 as the true number of latent variables in 10000 iteration in $N \times 6$ except MA_OPT(PRESS). But when the predictor variable number increases, they find latent variable with a higher number and they cannot find the true number of latent variables.

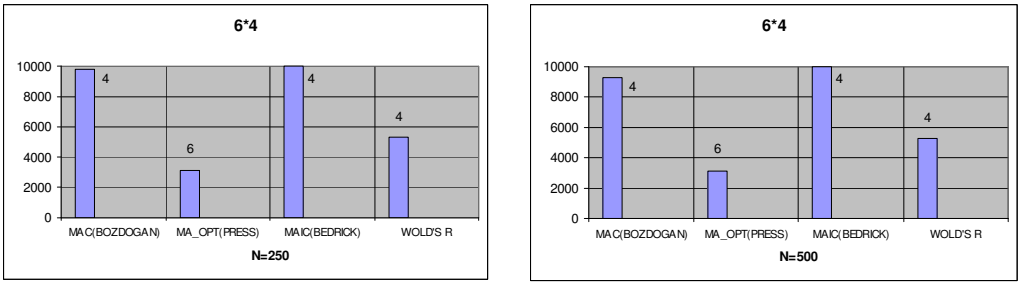


Figure A4.3 All model selection criteria for 6*4.

In these figures each model selection criterion is displayed for each object number for $N \times 6$. As it can be seen, each criterion find true number of latent variables for each number of observation. Only MA_OPT(PRESS) gives a different result for number of latent variables.

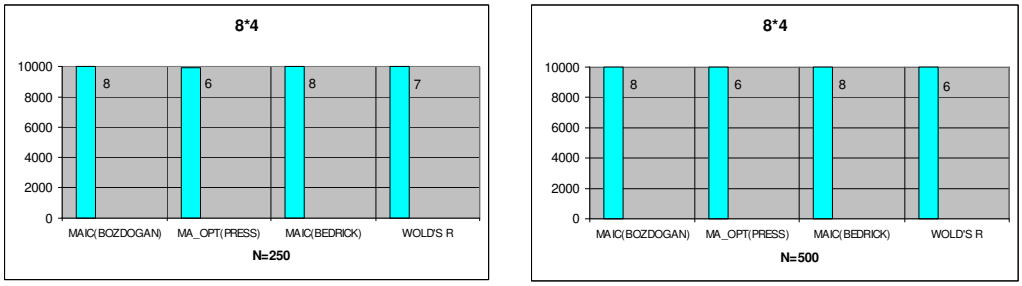


Figure A4.4 All model selection criteria for 8*4

In these figures each model selection criterion is displayed for each object number for $N \times 8$. Each criterion finds true number of latent variables close to the number predictor variables.

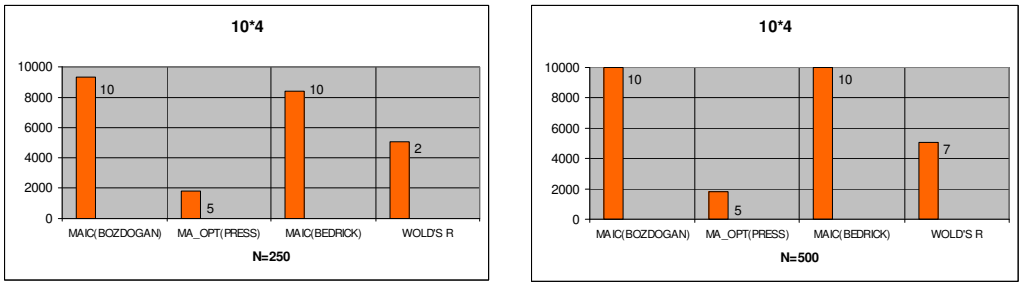


Figure A4.5 All model selection criteria for 10*4

In these figures each model selection criterion is displayed for each object number for $N \times 10$. MAIC(BOZDOGAN) and MAIC(BEDRICK) criteria find number of latent variables close to the number of predictor variables, MA_OPT(PRESS) and WOLD'S R criteria find number of latent variables in a small number.

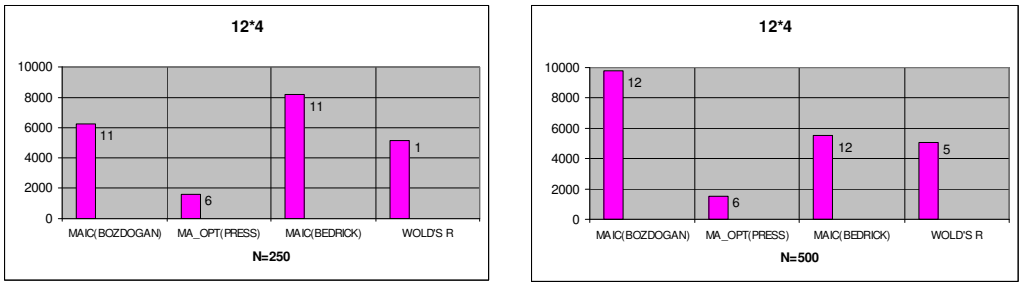
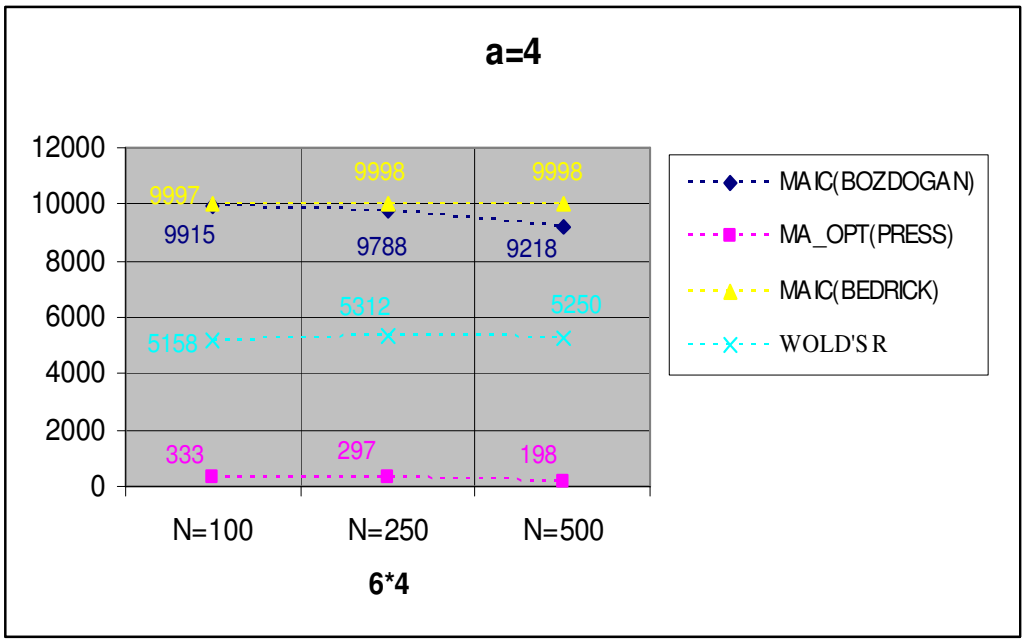
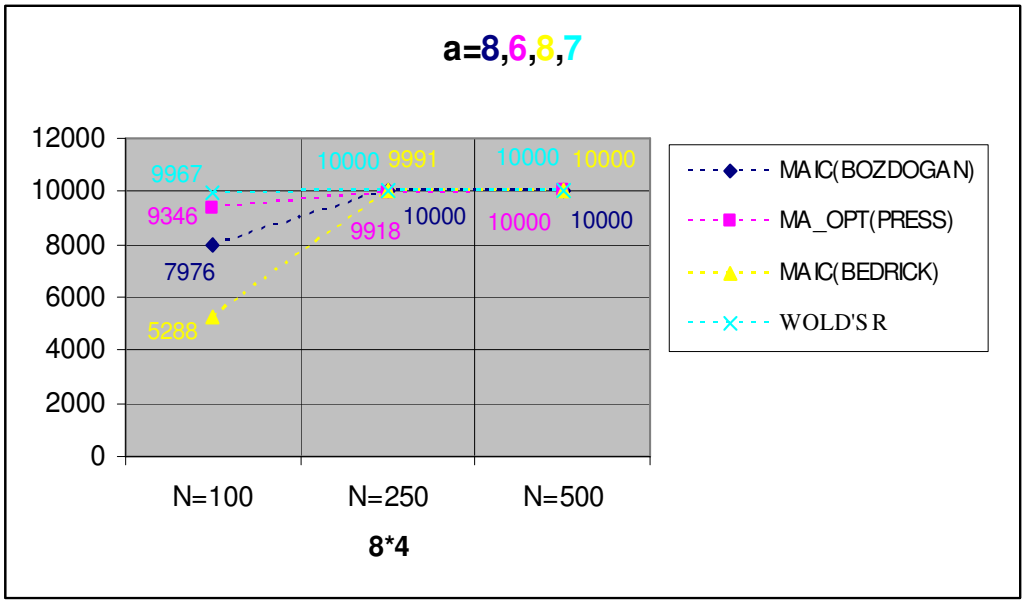
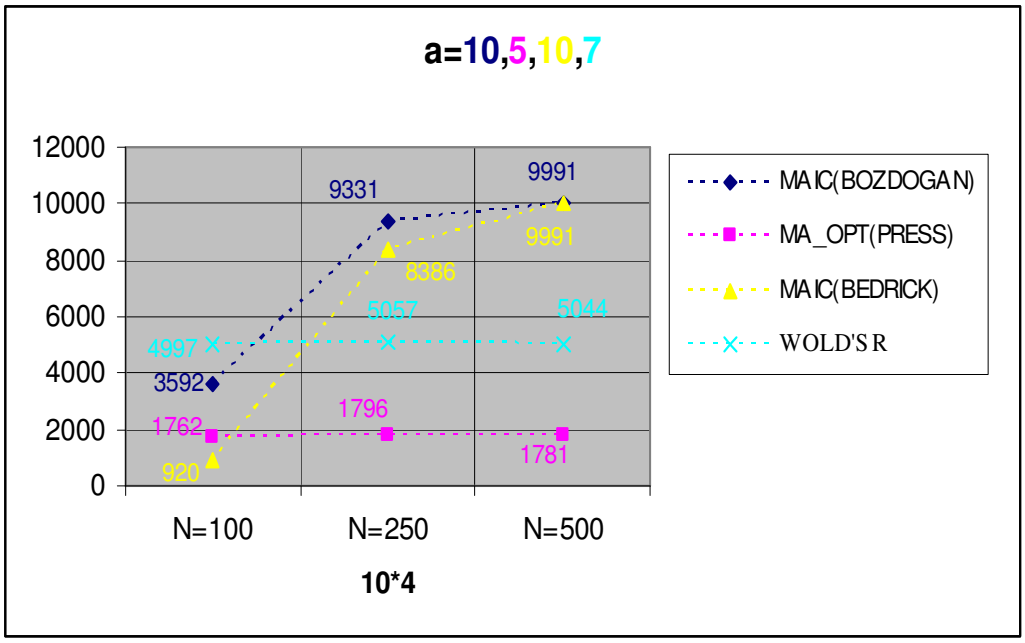


Figure A4.6 All model selection criteria for 12*4

In these figures each model selection criterion is displayed for each object number for $N \times 10$. MAIC(BOZDOGAN) and MAIC(BEDRICK) criteria find number of latent variables close to the number of predictor variables, MA_OPT(PRESS) and WOLD'S R criteria find the number of latent variables in a small number.







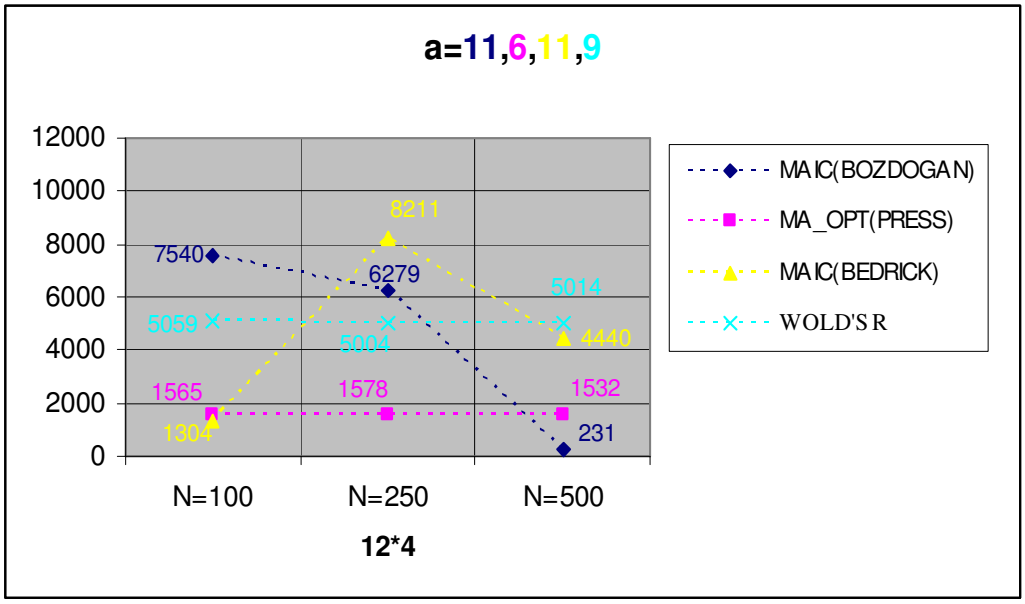
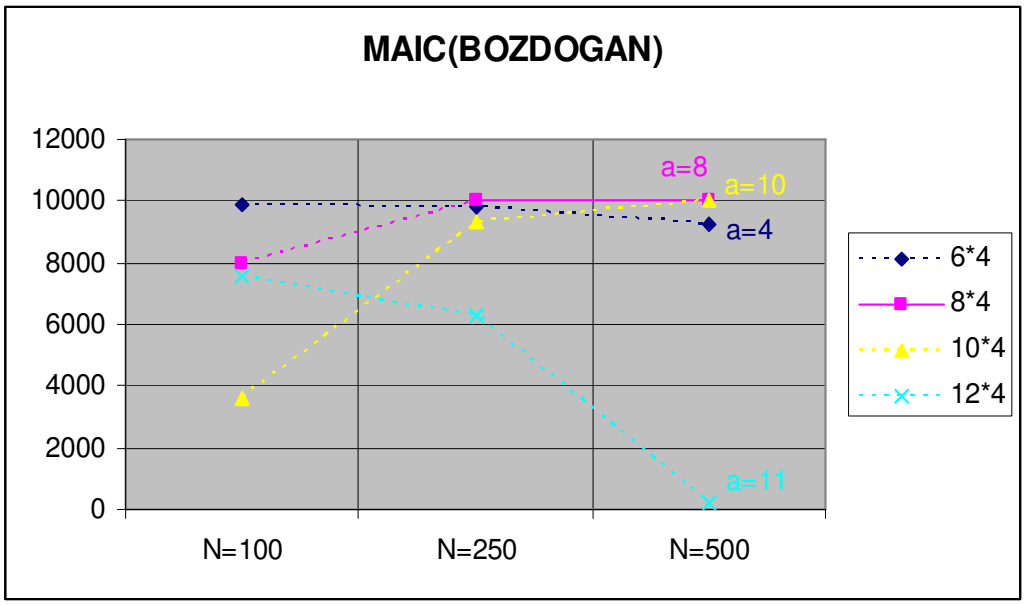
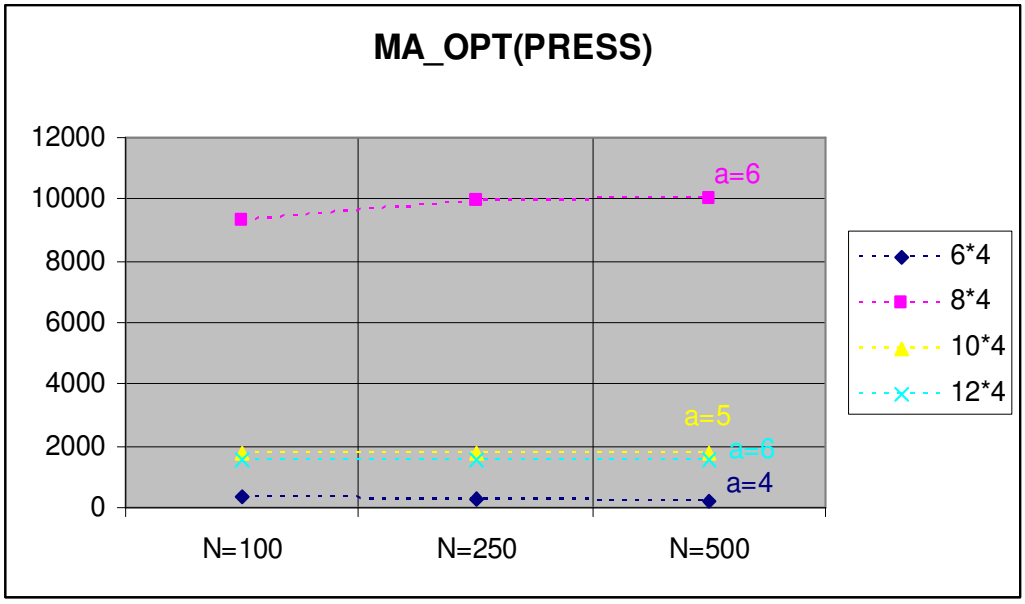


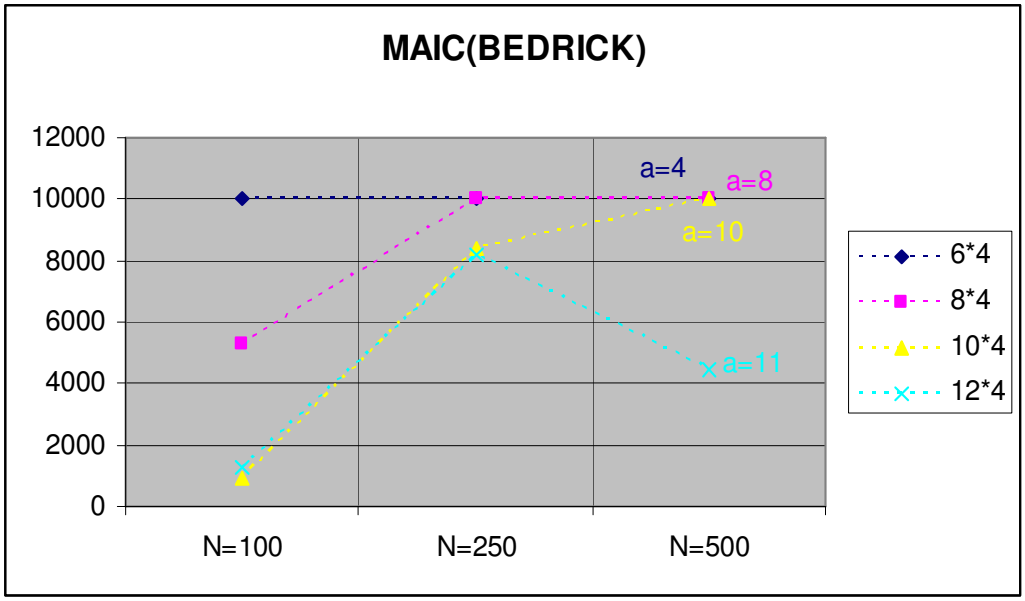
Figure A4.7 Trends for each criterion.

These figures illustrates the trends with the latent variable number for all criteria. These trends are shown for each design matrices and for each number of observations with iteration number. In the first figure for $N \times 6$, true number of latent variables is found in each number of observation with all criteria. In the other-sized design matrices, the transition for nearly most iterated number of latent

variables is given and shown in different colors according to the model selection criteria.







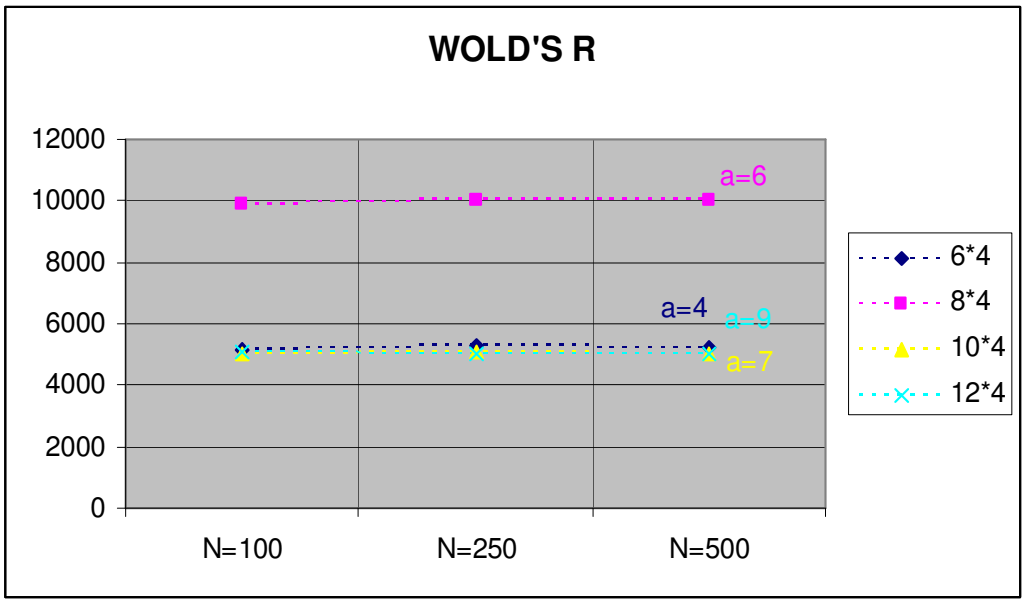


Figure A4.8 Trends for each design matrix.

In these figures the transition for the most iterated number of latent variables is shown for each design matrices according to the number of observations,

respectively for all criteria. Numbers of latent variables are given in different colors according to the design matrices.

Appendix 5. Results of Simulation Study

Table A5.1 Simulation results for N=100.

MAIC(BOZDOGAN):MBoz MAIC(BEDRICK):MB MA_OPT(PRESS) WOLD'S R	**m=6**				**m=8**				**m=10**				**m=12**			
	MBoz	MA-opt PRESS	MB	Wold's R	MBoz	MA-opt PRESS	MB	Wold's R	MBoz	MA-opt PRESS	MB	Wold's R	MBoz	MA-opt PRESS	MB	Wold's R
a=4	9915	333	9997	5025(1)	0	282	0	597(1)	0	1407	0	5167(1)	0	1387	0	5026(1)
a=5	85	2949	3	4810(2)	0	16	0	6866(2)	0	1762	0	5000(2)	0	1426	0	5040(2)
a=6	0	2908	0	5119(3)	0	9346	0	4366(3)	0	1807	0	4967(3)	0	1565	0	5061(3)
a=7				5158(4)	2024	0	4712	1899(4)	0	1407	0	5000(4)	0	1444	0	4990(4)
a=8				5046(5)	7976	0	5288	23(5)	7	1146	0	5000(4)	0	1003	0	4931(5)
a=9								9896(6)	6401	484	313	5015(5)	0	846	1308	4988(6)
a=10								9967(7)	3592	208	8767	5036(6)	1002	447	7383	5070(7)
a=11												4895(8)	7540	139	1304	5016(8)
a=12												4792(9)	1458	42	5	5059(9) 5042(10) 4808(11)

Table A5.2 Simulation results for N=250.

MAIC(BOZDOGAN):MBoz MAIC(BEDRICK):MB MA_OPT(PRESS) WOLD'S R	**m=6**				**m=8**				**m=10**				**m=12**			
	MBoz	MA-opt PRESS	MB	Wold's R	MBoz	MA-opt PRESS	MB	Wold's R	MBoz	MA-opt PRESS	MB	Wold's R	MBoz	MA-opt PRESS	MB	Wold's R
a=4	9788	297	9998	50261)	0	32	0	29(1)	0	1483	0	4986(1)	0	1440	0	5110(1)
a=5	199	2976	2	4813(2)	0	0	0	7861(2)	0	1796	0	5079(2)	0	1516	0	4985(2)
a=6	13	3105	0	4806(3)	0	9918	0	3972(3)	0	1710	0	4967(3)	0	1578	0	4966(3)
a=7				5312(4)	0	0	9	840(4)	0	1401	0	4920(4)	0	1361	0	4989(4)
a=8				4938(5)	10000	0	9991	0(5)	0	1144	0	5064(5)	0	1017	0	5002(5)
a=9								9998(6)	669	454	1614	4958(6)	0	787	0	4972(6)
a=10								10000(7)	9331	139	8386	5057(7)	3721	469	754	5037(7)
a=11												4986(8) 4912(9)	6279	119	8211	5039(8)
a=12													0	20	1035	5004(9) 5000(10) 4840(11)

Table A5.3 Simulation results for N=500.

MAIC(BOZDOGAN):MBoz MAIC(BEDRICK):MB MA_OPT(PRESS) WOLD'S R	**m=6**				**m=8**				**m=10**				**m=12**			
	MBoz	MA-opt PRESS	MB	Wold's R	MBoz	MA-opt PRESS	MB	Wold's R	MBoz	MA-opt PRESS	MB	Wold's R	MBoz	MA-opt PRESS	MB	Wold's R
a=4	9218	198	9998	4915(1)	0	0	0	2(1)	0	1410	0	5021(1)	0	1447	0	5082(1)
a=5	715	3060	2	4964(2)	0	0	0	8707(2)	0	1781	0	4988(2)	0	1440	0	4994(2)
a=6	67	3107	0	4935(3)	0	10000	0	3436(3)	0	1764	0	5026(3)	0	1532	0	4995(3)
a=7				5250(4)	0	0	0	255(4)	0	1461	0	4994(4)	0	1430	0	5011(4)
a=8				5022(5)	10000	0	10000	0(5)	0	1140	0	4962(5)	0	1064	0	5041(5)
a=9								10000(6)	9	494	9	4983(6)	0	851	0	4990(6)
a=10								10000(7)	9991	121	9991	5044(7)	0	460	0	5034(7)
a=11												5001(8)	231	120	4440	4935(8)
a=12												4992(9)	9769	27	5560	5014(9) 4948(10) 4883(11)