Comments on Mallows’ $c_p$ Statistics And Multicollinearity Effects On Predictions

By

Rahim Moineddin

A thesis submitted in conformity with the requirements for the degree of MS.c.
Graduate Department of Community Health
University of Toronto

© Copyright by Rahim Moineddin January 2001
The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author’s permission.

0-612-58663-4
Comments on Mallows' $C_p$ Statistics 
And Multicollinearity Effects 
On Predictions

MS.c. January 2001

Rahim Moineddin

Graduate Department of Community Health
University of Toronto

ABSTRACT

Selecting a subset of predictors by considering all possible combinations of variables 
with corresponding values of Mallows' $C_p$, AIC and BIC is discussed. It is shown that 
when a model with the smallest $C_p$ is selected 16% of non-related predictors had the 
chance of being entered in the model.

The effect of multicollinearity on the prediction errors is studied. When multicollinearity 
is not severe the fitted model can be used for prediction. When severe multicollinearity 
exists it is still safe to interpolate but not to extrapolate.
ACKNOWLEDGMENTS

I am extremely grateful to my supervisor Professor Paul Corey for his efficient guidance and constructive criticisms throughout the course of this research. I am very happy to find an occasion to express my gratitude to my family, though whatever I express may not fully capture my feelings of love and respect for them. I dedicate this work to my wife and my children.
Table of Contents

Chapter One: Introduction 5

Chapter Two: Linear Models 8

2.1: Classical Regression Models 8

2.1.1: Ordinary Least Squares Estimation 9

2.1.2: The Least Squares Estimation for Non-independent Error Structure 11

2.2: Some Linear Time Series Models 14

2.3: Testing for Autoregression 15

Chapter Three: Subset Selection in Regression 17

3.1: All Possible Regression Procedure 18

3.1.1: Mallows’ $C_p$ Criterion 20

3.1.2: AIC and BIC Criteria 24

3.2: Discussing Mallows’ $C_p$ 26

3.3: Simulation Results 31

Chapter Four: Effects of Multicollinearity on Predictions 45

4.1: Sources of Multicollinearity 45

4.2: Simulation Results 54

Appendix: Akaike Information Criterion 61

Sample SAS Programs 64

References 82
CHAPTER ONE

INTRODUCTION

Regression analysis is perhaps the most widely used statistical method. Regression provides a way of empirically identifying how a response or outcome variable is affected by other variables usually called regressor, predictor or independent variables. Therefore regression methods have a wide range of applications in fields that include social sciences, engineering, medical research and business.

Any method of fitting a mathematical equation (model) to data may be called regression. Such mathematical equations are valuable tools for making predictions and judging the strength of the relationships.

In some applications theoretical considerations or previous experiences can help the researcher in selecting the regressors to be used in the model. However, in most practical problems the analyst has a pool of candidate predictors that should include all the influential factors. However the actual subset of regressors that should be used in the model needs to be determined.
Variable selection problems involve the determination of an appropriate subset of predictors for the regression model. In practice we should start out with a long list of predictors that we suspect have some effect on the dependent variable. We would like to cull the list for various reasons such as:

a) It is easier to interpret and work with simple models
b) Reducing number of variables often reduces multicollinearity
c) Lowering the ratio of the number of variables to the number of observations is beneficial

It is obvious that we want to delete those predictors that have little effect on the dependent variable. Generally there are types of procedures that can help the analyst to select a subset of variables among candidate variables. One of the procedures helps the analyst to choose a subset of variables by presenting all possible combinations of variables (independent variables) with corresponding values of some statistics such as Mallows' \( C_p \), AIC or BIC. In this thesis we will not consider other procedures.

There is a problem with this procedure. If we construct a set of predictors with some entirely random numbers and apply this procedure with one of these statistics for model selection then by chance some of those variables appear to be related to
the dependent variable. This problem is considered in chapter three.

When the columns of the design matrix \( X \) are nearly linearly dependent this near dependency leads to the near singularity of \( XX \). When \( XX \) is almost singular multicollinearity is present. Multicollinearity has several effects. In chapter four the effect of multicollinearity on the prediction errors is studied.
In this chapter we introduce some basic ideas of linear models and simple time series models.

2.1 CLASSICAL REGRESSION MODELS

Let $Y$ denote the dependent (response) variable that is linearly related to a set of $p$ independent (explanatory) variables $X_1, X_2, ..., X_p$ according to the following equation

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p + \epsilon$$

This is called the linear regression model. When we have $n$ observations on $Y$ and $X_i$'s this equation can be represented as follows:

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

where $Y = (y_1, ..., y_n)'$ is a $n$-vector of responses
and $P=\{P_0, P_1, \ldots, P_p\}$ is a $p+1$-vector of parameters and $E=(\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_n)'$ is a $n$-vector of error terms.

We will consider the problem of estimating and testing of hypotheses concerning $\beta$ under some realistic assumptions. For estimating $\beta$ we may use the general procedure of minimizing

$$
\sum_{i=1}^{n} Q(\varepsilon_i) = \sum_{i=1}^{n} Q(y_i - x_i' \beta)
$$

where $x_i=(x_{i1}, x_{i2}, \ldots, x_{ip})'$ and $Q(\cdot)$ is a suitably chosen function non-negative such as $Q(x)=|x|$ or $Q(x)=x^2$.

$Q(x)=x^2$ is the basis of the theory of least squares.

2.1.1 Ordinary Least Squares Estimation

Let $B$ denote the set of all possible vectors $\beta$. Also let $E(\varepsilon)=0$ and $V(\varepsilon)=\sigma^2 I_n$. We want to find a vector $b=(b_0, b_1, \ldots, b_p)' \in B$ that minimizes

$$
X_{nxp+1} = \begin{bmatrix}
1 & x_{11} & x_{21} & \ldots & x_{p1} \\
1 & x_{12} & x_{22} & \ldots & x_{p2} \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
1 & x_{1n} & x_{2n} & \ldots & x_{pn}
\end{bmatrix}
$$
$$Q(\beta) = \sum_{i=1}^{n} \varepsilon_i^2 = (Y - X\hat{\beta})(Y - X\hat{\beta})$$

given the observed vector $Y$ and matrix $X$.

It is well known that $b_{OLS} = (XX)^{-1}XY$ is the unique solution when $XX$ is non-singular. When $XX$ is not full rank, $Q(\beta)$ attains its minimum at an infinite number of points

$$b_{OLS} = (XX)^{-1}XY + (I - (XX)^{-1}XX)\omega$$

where $\omega$ is an arbitrary vector and $(XX)^{-1}$ is the generalized inverse of $XX$.

**Definition:** Let $A$ be any $m \times n$ matrix. Then the matrix $A^{-}$ is said to be a generalized inverse of $A$ if $AA^{-}A = A$ holds.

In this thesis we will not consider the case of $XX$ not being of full rank.

$\hat{Y} = Xb_{OLS}$ is the vector of fitted $Y$ values for the $n$ individuals and $\hat{y}_i = x_i^t b_{OLS}$ is the predicted value of $Y$ for the $i$th individual.

The usual assumptions are that $E(\varepsilon) = 0$ and $E(\varepsilon'\varepsilon) = \sigma^2 I$ and $X_{nxp}$ is a matrix of full rank. Although the $X_{nxp}$ matrix may be a random variable it is conventionally assumed fixed for the purpose of inference.
It is a well-known result of classical regression theory that if $Q(\varepsilon) = \varepsilon^2$ then $\hat{\beta} = \beta_{OLS} = (XX)^{-1}XY$ is the minimum variance linear unbiased estimator of $\beta$ with variance $\sigma^2(XX)^{-1}$ and is called the least squares estimator of $\beta$.

Furthermore if we assume that the error vector $\varepsilon \sim MN(0, \sigma^2I)$ then $\hat{\beta} = \beta_{OLS} = (XX)^{-1}XY$ is also the maximum likelihood estimator of $\beta$.

Let the vector $\hat{Y} = (\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_n)'$ denote the vector of fitted values and $e = (e_1, e_2, \ldots, e_n)'$ with $e_i = y_i - \hat{y}_i$ denoting the vector of residuals. Then $\hat{Y} = X\beta_{OLS}$ and $e = Y - X\beta_{OLS}$. Now

$$e = Y - X(XX)^{-1}XY = (I - X(XX)^{-1}X')Y = (I - H)Y.$$ The matrix $H$ is called the hat matrix. The vector of residuals can be expressed as $e_{n \times 1} = (I - H)Y$ with $V(e) = \sigma^2(I - H)$. Let $SSE = e'e$, then the variance of $e$ can be estimated by $S^2(e) = \frac{SSE}{n - p}(I - H)$.

2.1.2 The Least Squares Estimation For Non-Independent Error Structure

Consider the multiple linear regression model

$$Y = X\beta + \varepsilon$$
We assume that the design matrix $X$ is full rank, $E(\varepsilon) = 0$, and $E(\varepsilon'\varepsilon) = \sigma^2 W_{n \times n}$, where $\sigma^2$ is a finite constant real number and $W$ is a $n \times n$ positive definite matrix.

An important feature of the weighted linear regression model is that the $n \times n$ matrix $W$ may involve as many as $\frac{1}{2} n(n+1)$ additional unknown parameters in the estimation problem. The sample size is fixed and we may not be able to estimate all of the parameters simultaneously. When $W$ is unknown, we may be able to reduce the number of parameters that need to be estimated by assuming a specific structure for the error term such as an autoregression structure.

We first review the estimation of $\beta$ when $W$ is known.

**Gauss-Markov-Aitken theorem:** If $Y = X\beta + \varepsilon$, where $E(\varepsilon) = 0$ and $E(\varepsilon'\varepsilon) = \sigma^2 W_{n \times n}$, the generalized least-squares estimator (GLSE), $b_{GLS} = (X'W^{-1}X)^{-1}X'W^{-1}Y$ is the best linear unbiased estimator (BLUE) for $\beta$. The variance-covariance matrix of $b_{GLS}$ is given by $V(b_{GLS}) = \sigma^2(X'W^{-1}X)^{-1}$ and the unbiased estimator of $\sigma^2$ is given by $S^2 = \frac{e'W^{-1}e}{n-p}$ where $e$ is the vector of fitted residuals.

Assume that the covariance matrix $W$ is miss-specified by a covariance matrix $A \neq W$. In this case the resulting estimator
is \( \hat{\beta} = (XA^{-1}X)^{-1}X'A^{-1}Y \) assuming that \( A^{-1} \) and \( (XA^{-1}X)^{-1} \) exist. It can be shown that \( \hat{\beta} \) is unbiased with \( V(\hat{\beta}) = \sigma^2(XA^{-1}X)^{-1}X'A^{-1}WA^{-1}X(XA^{-1}X)^{-1} \) and loss of efficiency in estimating \( \beta \) by \( (XA^{-1}X)^{-1}X'A^{-1}Y \) instead of the GLSE \( b_{GLS} = (X'WX)^{-1}X'W^{-1}Y \) is

\[
V(\hat{\beta}_{OLS}) - V(\hat{\beta}_{GLS}) = 
\sigma^2 \left( (XA^{-1}X)^{-1}X'A^{-1} - (XW^{-1}X)^{-1}X'W^{-1} \right) W \left[ (XA^{-1}X)^{-1}X'A^{-1} - (XW^{-1}X)^{-1}X'W^{-1} \right],
\]

which is non-negative definite.

Setting \( A = I \) and \( U = (XX)^{-1} \) we can calculate the increase in dispersion due to using OLS \( b_{OLS} = (XX)^{-1}XY = UXY \) instead of the GLSE \( b_{GLS} = (X'WX)^{-1}X'W^{-1}Y = S^{-1}X'W^{-1}Y \). This is given by

\[
V(\hat{\beta}_{OLS}) - V(\hat{\beta}_{GLS}) = \sigma^2 (UX' - S^{-1}X'W^{-1})W(XU' - W^{-1}XS^{-1}).
\]

It is clear that \( V(\hat{\beta}_{OLS}) = V(\hat{\beta}_{GLS}) \) if and only if \( UX' = S^{-1}X'W^{-1} \). This condition is equivalent to \( X'WZ = 0 \) where \( Z \) is a matrix of maximum rank such that \( Z'X = 0 \). Clearly an incorrect specification of \( W \) will lead to errors in estimating \( \sigma^2 \) by \( \hat{\sigma}^2 \), which is based on \( \hat{\varepsilon} \). We can show that

\[
E(\hat{\sigma}^2) = \sigma^2 + \frac{\sigma^2}{n-p} \left( p - tr[(XX)^{-1}X'WX] \right).
\]

If the disturbances are positively correlated then the bias
\( \frac{\sigma^2}{n-p} \left( p-tr[(XX)^{-1}XWX] \right) \) tends to be negative. So the true variance will be under-estimated (See Rao 1995 for details and more discussions).

2.2 Some Linear Time Series Models

In many respects the simplest kind of time series \( \{X_t\} \) is one in which the random variables \( X_t, t \in \mathbb{Z} \), are independently distributed with zero mean and variance \( \sigma^2 \).

**Definition:** The stochastic process \( \{e_t\} \) is said to be white noise with constant variance \( \sigma^2 \) if and only if \( \{e_t\} \) has zero mean and covariance function

\[
\gamma_s = \begin{cases} 
\sigma^2 & \text{if } s = 0 \\
0 & \text{if } s \neq 0 
\end{cases}
\]

where \( \gamma_s \) represents the covariance between \( X_t \) and \( X_{t+s} \).

**Definition:** The stochastic process \( \{X_t\} \) defined by

\[
X_t = \phi_1 X_{t-1} + \phi_2 X_{t-2} + \cdots + \phi_p X_{t-p} + e_t
\]

where \( \{e_t\} \) is white noise, is called an autoregressive process of order \( p \) which will be denoted by \( AR(p) \).

**Definition:** The stochastic process \( \{X_t\} \) defined by

\[
X_t = \theta_1 e_{t-1} + \theta_2 e_{t-2} + \cdots + \theta_q e_{t-q} + e_t
\]
where \( \{e_t\} \) is white noise, is called a moving average process of order \( q \) and will be denoted by \( MA(q) \).

**Definition:** The stochastic process \( \{X_t, t \in \mathbb{Z}\} \) defined by

\[
X_t - \phi_1 X_{t-1} - \cdots - \phi_p X_{t-p} = \theta_1 e_{t-1} + \cdots + \theta_q e_{t-q} + e_t
\]

where \( \{e_t\} \) is white noise, is called an autoregressive moving average process of order \((p,q)\) and will be denoted by \( ARMA(p,q) \).

### 2.3 Testing for Autoregression

In time series analysis data are typically inter-dependent. When time series data are used in regression analysis, often the error term is not independent through time. Instead the errors are serially correlated or autocorrelated. Also autocorrelation of first and higher orders in the disturbances may arise from observational errors in the included variables or from estimation of missing data by either extrapolating or averaging.

The well-known Durbin-Watson test can be used to test the assumption that the error term structure follows a first order autoregression model.

The Durbin-Watson (Durbin 1951) test statistic is
\[
d = \frac{\sum_{i=2}^{n} (\hat{\varepsilon}_i - \hat{\varepsilon}_{i-1})^2}{\sum_{i=1}^{n} \hat{\varepsilon}_i^2}
\]

where \( \hat{\varepsilon} = Y - X(X'X)^{-1}XY \). The null two-sided Durbin-Watson test for \( H_o : \rho = 0 \) against \( H_a : \rho \neq 0 \) is as follows:

Do not reject \( H_o \) if \( d > d_U \) or \( d < 4 - d_U \)

Reject \( H_o \) if \( d < d_L \) or \( d > 4 - d_L \)

No decision if \( d_L \leq d \leq d_U \) or \( 4 - d_U \leq d \leq 4 - d_L \) i.e. Test is inconclusive.

Where \( d_L \) and \( d_U \) are critical values.

If the assumption \( H_o : \rho = 0 \) is rejected then \( \hat{\rho} = \frac{\sum_{i=2}^{n} \hat{\varepsilon}_i \hat{\varepsilon}_{i-1}}{\sqrt{\sum_{i=2}^{n} \hat{\varepsilon}_i^2} \sqrt{\sum_{i=2}^{n} \hat{\varepsilon}_{i-1}^2}} \) is inserted in \( W^{-1} = \begin{bmatrix} 1 & -\rho & \ldots & 0 \\ -\rho & 1 + \rho^2 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & 1 \end{bmatrix} \) and \( \beta \) is estimated by

\( b = (X\hat{W}^{-1}X)^{-1}X\hat{W}^{-1}Y \). This procedure can be repeated through a number of iterations until \( \hat{\rho} \) and \( \hat{\beta} \) change very little.
There are many reasons for wanting to select only a subset of a large number of variables. In practice the number of explanatory (independent) variables may be very large. It would be easier to interpret a model that has only a small number of variables. Furthermore, many of these variables may be highly inter-correlated. Hence, we wish to reduce the number of explanatory variables in the final model. The identification of a subset of explanatory variables to include in the final model constitutes one of the most difficult problems in regression analysis. Since the purpose for developing a regression model may vary no one subset of explanatory variables may always be best. The choice of a good subset needs to be done with care. Elimination of key explanatory variables can seriously damage the explanatory power of the model and lead to biased estimators of the regression coefficients. Selection of too many explanatory variables can result in an over-fitted model that results in large variances for the estimators of the coefficients as well as a large variance for a predicted value $\hat{y}_i$. 
There are several approaches that can assist the statistician in reducing the number of potential explanatory variables. One approach, which is practical for dealing with a small or moderate number of variables is to consider all possible subsets of the explanatory variables that can be developed from the pool of potential explanatory variables and select a good subset according to some criterion.

The other approach employs an automatic search procedure to arrive at a single subset of explanatory variables. This approach may be recommended for dealing with a large number of explanatory variables. The forward stepwise regression or backward elimination methods are examples of automatic search procedures. These procedures for model selection will not be considered in this thesis. See Miller (1994) for other procedures and criteria.

3.1 All Possible Regression Procedure

The all-possible regressions procedure requires that all possible subsets from the pool of potential explanatory variables be identified. A regression model is fitted for each of these subsets and assessed according to some criterion.

When there are $p$ potential explanatory variables in the pool, there would be $2^p - 1$ possible regression models because the number of non-empty subsets of a set with $p$ distinct elements
is $2^p - 1$. The empty subset corresponds to the regression model that includes only the intercept. Different criteria for comparing the regression models may be used with the all-possible regressions selection procedure. We will discuss three procedures: $C_p$, AIC and BIC.

Some notation needs to be introduced.

1. $y_i$ is the value of the response variable in the ith trial or for the ith individual, or ith case.
2. $\beta_0, \beta_1, \ldots, \beta_{p-1}$ are $p$ unknown parameters
3. $X_i$ is a known constant, namely, the value of the predictor variables in the ith trial
4. $\varepsilon_i$ is a random error term with mean zero and constant variance $\sigma^2$ and $V(\varepsilon) = \sigma^2 W$.
5. The observations vector $Y$, the design matrix $X$, the vector of parameters $\beta$ and the vector of error terms are defined as follows:

$$Y = (y_1, y_2, \ldots, y_n)'$$
$$\beta = (\beta_0, \beta_1, \ldots, \beta_{p-1})'$$
$$X = \begin{bmatrix}
1 & x_{11} & \cdots & x_{p-1,1} \\
1 & x_{12} & \cdots & x_{p-1,2} \\
\vdots & \vdots & \ddots & \vdots \\
1 & x_{1n} & \cdots & x_{p-1,n}
\end{bmatrix}$$
3.1.1 Mallows' $C_p$ Criterion

When there are $p$ potential independent variables there are $2^p$ possible equations (including constant equation) to be considered. The most comprehensive approach is to do all $2^p$ possible regressions and then compare them. This is an old idea, which even before the widespread of computer several authors published such results computed by hand. (See Gorman and Toman 1966). C. Mallows first introduced a statistic, which we can use for comparing regression models. (See Mallows, C. 1964).

Let $\mu_i$ be the true mean response when the levels of the predictor (explanatory) variables are those for the $i$th case. This criterion is concerned with the total mean squared error of the $n$ fitted values for each subset regression. The mean squared error involves the total error in each fitted value. For the $i$th individual this error is given by $\hat{y}_i - \mu_i$. The bias component for the $i$th fitted value $\hat{y}_i$ is $E(\hat{y}_i) - \mu_i$. If the fitted model is not correct, $E(\hat{y}_i)$ will differ from the true mean response $\mu_i$ and the difference represents the bias of the fitted model.
The random error component for $\hat{y}_i$ is $\hat{y}_i - E(\hat{y}_i)$. This represents the deviation of the fitted value $\hat{y}_i$ for the given sample from the expected value of the fitted value.

The mean squared error for $\hat{y}_i$ is defined as the expected value of $\hat{y}_i - \mu_i$ squared.

$$E[(\hat{y}_i - \mu_i)^2] = E[(E(\hat{y}_i) - \mu_i) + (\hat{y}_i - E(\hat{y}_i))^2]$$
$$= E(E(\hat{y}_i) - \mu_i)^2 + E[(\hat{y}_i - E(\hat{y}_i))^2]$$
$$= (E(\hat{y}_i) - \mu_i)^2 + Var(\hat{y}_i)$$

So the mean squared error for the fitted value $\hat{y}_i$ is the sum of the squared bias and the variance of $\hat{y}_i$.

The total mean squared error is

$$\sum_{i=1}^{n}[(E(\hat{y}_i) - \mu_i)^2 + Var(\hat{y}_i)]$$

We normalize the total mean squared error by $\sigma^2$ and denote it by $\Gamma_p$

$$\Gamma_p = \frac{1}{\sigma^2} \left[ \sum_{i=1}^{n} [(E(\hat{y}_i) - \mu_i)^2 + Var(\hat{y}_i)] \right]$$

which is equivalent to

$$\Gamma_p = \frac{1}{\sigma^2} \sum_{i=1}^{n} [E(\hat{y}_i) - \mu_i]^2 + \frac{1}{\sigma^2} \sum_{i=1}^{n} V(\hat{y}_i)$$

Now
\[
\sum_{i=1}^{n} V(\hat{y}_i) = \text{tr}(V(\hat{Y})) \\
= \text{tr}(V(HY)) \\
= \text{tr}(HV(Y)H') \\
= \text{tr}(H\sigma^2 IH') \\
= \sigma^2 \text{tr}(H^2) \\
= p\sigma^2 \quad (\text{tr}(H^2) = \text{tr}(H) = p)
\]

and

\[
E[y_i - \hat{y}_i]^2 = E[y_i - \mu_i - (\hat{y}_i - \mu_i)]^2 \\
= E[(y_i - \mu_i)^2] + E[(\hat{y}_i - \mu_i)^2] - 2E[(y_i - \mu_i)(\hat{y}_i - \mu_i)] \\
= \sigma^2 + [E(\hat{y}_i) - \mu_i]^2 + V(\hat{y}_i) - 2E[\hat{y}_i(y_i - \mu_i)] \\
= \sigma^2 + [E(\hat{y}_i) - \mu_i]^2 + V(\hat{y}_i) - 2E \left[ \sum_{j=1}^{n} h_{ij} y_j (y_i - \mu_i) \right] \\
= \sigma^2 + [E(\hat{y}_i) - \mu_i]^2 + V(\hat{y}_i) - 2 \sum_{j=1}^{n} E(h_{ij} y_i y_j - h_{ij} y_j y_i) \\
= \sigma^2 + [E(\hat{y}_i) - \mu_i]^2 + V(\hat{y}_i) - 2 \sum_{j=1}^{n} h_{ij} E(y_i y_j) + 2 \sum_{j=1}^{n} h_{ij} \mu_i \mu_j \\
= \sigma^2 + [E(\hat{y}_i) - \mu_i]^2 + V(\hat{y}_i) - 2h_{ii} \sigma^2 - 2 \sum_{j=1}^{n} h_{ij} \mu_i \mu_j + 2 \sum_{j=1}^{n} h_{ij} \mu_i \mu_j \\
= \sigma^2 + [E(\hat{y}_i) - \mu_i]^2 + V(\hat{y}_i) - 2h_{ii} \sigma^2
\]
Let \( RSS_p = \sum_{i=1}^{n} [y_i - \hat{y}_i]^2 \) denote the error sum of squares of the regression model when subsets of \( p \) predictors are in the model. Taking the expectation we have

\[
E(RSS_p) = \sum_{i=1}^{n} E[(y_i - \hat{y}_i)^2]
\]

\[
= \sum_{i=1}^{n} \left[ \sigma^2 + [E(\hat{y}_i - \mu_i)^2] + V(\hat{y}_i) - 2h_i \sigma^2 \right]
\]

\[
= n \sigma^2 + p \sigma^2 - 2p \sigma^2 + \sum_{i=1}^{n} [E(\hat{y}_i) - \mu_i]^2
\]

\[
= (n - p) \sigma^2 + \sum_{i=1}^{n} [E(\hat{y}_i) - \mu_i]^2
\]

Substituting \( E(RSS_p) \) in \( \Gamma_p = \frac{1}{\sigma^2} \sum_{i=1}^{n} [E(\hat{y}_i) - \mu_i]^2 + \frac{1}{\sigma^2} \sum_{i=1}^{n} V(\hat{y}_i) \) we have

\[
\Gamma_p = \frac{1}{\sigma^2} \left[ E(RSS_p) - (n - p) \sigma^2 \right] + \frac{1}{\sigma^2} p \sigma^2
\]

\[
= \frac{E(RSS_p)}{\sigma^2} + (2p - n)
\]

Suppose that \( \hat{\sigma}^2 \) is a good estimate of \( \sigma^2 \). Replacing \( E(RSS_p) \) by the observed value \( RSS_p \) produces an estimate of \( \Gamma_p \) as follows

\[
C_p = \frac{RSS_p}{\hat{\sigma}^2} - n + 2p
\]

where \( \hat{\sigma}^2 = \frac{RSS_k}{n-k} \) is the estimate of \( \sigma^2 \) based on the full model.
If the p term model has negligible bias, then \( \sum_{i=1}^{n} [E(\hat{y}_i) - \mu_i]^2 = 0 \) and consequently \( E(RSS_p) = (n-p)\sigma^2 \), and

\[
E[C_p|bias = 0] = \frac{(n-p)\sigma^2}{\sigma^2} - n + 2p = p
\]

When we use \( C_p \) it is helpful to construct a plot of \( C_p \) as a function of \( p \) for each regression equation. Regression equations with small bias have values of \( C_p \) close to \( p \). Generally, small values of \( C_p \) are desirable.

### 3.1.2 AIC and BIC criteria

As we mentioned earlier a fundamental difficulty in statistics is the choice of an appropriate model. In time series analysis the Box-Jenkins approach has been a popular subjective judgment method for selecting an appropriate model. Akaike adopted an alternative approach seeking objective expressions for the subjective judgment. Then the problem of model identification is converted into an optimization problem with respect to a specific criterion over a pre-selected family of models.

Assume that a statistical model of \( p \) parameters is fitted to the data. To assess the quality of the model fitting, Akaike
introduced an information criterion. The criterion has been called AIC (Akaike Information Criterion) and is defined as

\[ AIC(p) = -2\ln \left[ \text{likelihood of a model with } p \text{ parameters} \right] + 2p \]

where \( p \) is the number of parameters in the model. A brief derivation of AIC is given in the appendix.

In several time series books (e.g. M. B. Priestly (1981) page 375 and H. Tong (1990) page 291) Akaike and Schwarz are referenced as the first statisticians who modified AIC and replaced the coefficient \( 2p \) in the AIC formula by a function of \( p \) and \( n \), where \( n \) stands for the sample size. So we can say that more recently, Akaike and Schwarz have developed a Bayesian extension of the maximum AIC procedure, called BIC, Bayesian Information Criteria. Akaike’s BIC has the following form

\[ BIC(p) = n\hat{\sigma}_e^2 - (n - p)\ln \left( \frac{1}{n} \right) + p\ln(n) + p\ln \left[ \frac{\hat{\sigma}^2_Y}{\hat{\sigma}_e^2} - 1 \right] / p \]

where \( \hat{\sigma}_e^2 \) is maximum likelihood estimate of the variance of the error term, \( \hat{\sigma}^2_Y \) is the estimate of the variance of \( Y \).

Priestley (1981) page 375 algebraically showed that this BIC is approximately equal to AIC when the second term in AIC, \( 2p \), is replaced by the term \( p + q\log(n) \). This has the effect of increasing the weight attached to the penalty term. Priestley
also mentioned that the BIC suggested by Schwarz (1978) the term $2p$ in AIC is replaced by $p \log(n)$.

AIC and BIC were originally developed for estimating the order of autoregression in time series analysis.

3.2 Discussing Mallows' $C_p$

Mallows' $C_p$ has been widely used, as a criterion in subset selection for which $C_p$ is a minimum. Most of the applications have been in situations where the predictors are random variables whereas the derivation assumes that the $X$'s are fixed or controllable variables. Colin L. Mallows in several papers discussed some problems of Mallows' $C_p$ statistics (See Mallows 1995 for further references and discussions).

Let us consider the orthogonal regression model $Y = X\beta + \varepsilon$ where $XX = nI_p$. We can simplify $\hat{\beta}$, the least squares estimate of $\beta$, as follows.

$$\hat{\beta} = (XX)^{-1}XY$$

$$= \frac{1}{n}XY$$

$$= \frac{1}{n}(X_1 \quad X_2 \quad \ldots \quad X_p)Y$$
where $X_j$ denotes the $j$th column of the design matrix $X$. Thus $\hat{\beta}_j$, the estimate of $\beta_j$, and its standard deviation are given by

$$\hat{\beta}_j = \frac{X_j'Y}{n}, \quad sd(\hat{\beta}_j) = \frac{\sigma}{\sqrt{n}}$$

Now

$$\hat{\beta} = (XX)^{-1}XY$$
$$= \frac{1}{n}XY$$
$$= \frac{1}{n}X'(X\beta + \varepsilon)$$

$$= \frac{1}{n}XX\beta + \frac{1}{n}X'\varepsilon$$
$$= \frac{n}{n}\beta + \frac{1}{n}X'\varepsilon$$
$$= \beta + \frac{1}{n}(X_1, X_2, \ldots, X_p)^T\varepsilon$$

therefore $\hat{\beta}_j = \beta_j + \frac{X_j'\varepsilon}{n}$.

From the Central Limit Theorem we have

$$\frac{\hat{\beta}_j - \beta_j}{sd(\hat{\beta}_j)} = \frac{n\frac{X_j'\varepsilon}{\sigma}}{\sigma \sqrt{n}} = Z$$

Thus

$$\hat{\beta}_j = \beta_j + \frac{\sigma Z}{\sqrt{n}}, \quad Z \sim N(0,1)$$
When $\sigma^2$ is unknown, the test statistics $t_j = \frac{\hat{\beta}_j}{\text{Sd}(\hat{\beta}_j)} = \frac{\sqrt{n}\hat{\beta}_j}{\hat{\sigma}}$ can be used.

The regression sum of squares $SSR$ can be rewritten as

$$SSR = \hat{\beta}'X'Y = \hat{\beta}'XX(X'X)^{-1}XY = \hat{\beta}'X\hat{\beta} = n\hat{\beta}'\hat{\beta} = n\sum_{j=1}^{p} \hat{\beta}_j^2$$

So $X_j$ improves fit by adding $n\hat{\beta}_j^2 = \left(\beta_j + \frac{\sigma Z}{\sqrt{n}}\right)^2$ which has a non-central $\chi^2$ distribution.

If we choose the model with smallest $C_p$ therefore $X_{k+1}$ would be added to the model if and only if $C_{k+1} < C_k$.

$$C_{k+1} < C_k \iff C_k - C_{k+1} > 0 \iff \frac{SSE_k - SSE_{k+1} + 2k - 2(k+1)}{\sigma^2} > 0 \iff \frac{RSS_{k+1} - RSS_k}{\sigma^2} - 2 > 0 \iff \frac{nb_{k+1}^2}{\sigma^2} - 2 > 0 \iff Z_{k+1}^2 - 2 > 0$$

Thus add $X_{k+1}$ when $Z_{k+1}^2 - 2 > 0 \Rightarrow |Z_{k+1}| > \sqrt{2}$. Since $\Pr\{Z > \sqrt{2}\} = 0.157$ therefore in the null case one chooses about 16% of variables.
If we choose the model with the shortest distance from the line $C_\rho = p$, the variable $X_{k+1}$ would be added to the model if $|C_{k+1} - (k+1)| < |C_k - k|$. Suppose that $C_{k+1} - (k+1)$ and $C_k - k$ are both positive.

Thus $X_{k+1}$ would be added if

$$C_{k+1} - (k+1) < C_k - k \Rightarrow C_k - C_{k+1} + 1 > 0 \Rightarrow Z_{k+1}^2 - 1 > 0$$

So add $X_{k+1}$ when $Z_{k+1}^2 - 1 > 0$ or $|Z| > 1$. Since $P(|Z| > 1) = 0.32$ therefore one choose about 32% of variables.

Let us remove the orthogonality condition and consider the general case. For a regression model with one predictor (mean corrected) the regression sum of squares is given by

$$SSR(X_1) = [B_0 \ B_1] \begin{bmatrix} n & 0 \\ 0 & S_{11} \end{bmatrix} \begin{bmatrix} B_0 \\ B_1 \end{bmatrix} = nB_0^2 + S_{11}B_1^2$$

For a regression model with two predictors (mean corrected) the regression sum of squares is

$$SSR(X_1, X_2) = [b_0 \ b_1 \ b_2] \begin{bmatrix} n & 0 & 0 \\ 0 & S_{11} & S_{12} \\ 0 & S_{21} & S_{22} \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \\ b_2 \end{bmatrix} = nb_0^2 + S_{11}b_1^2 + 2S_{12}b_1b_2 + S_{22}b_2^2$$

Thus

$$SSR(X_1, X_2) - SSR(X_1) = S_{11}b_1^2 + 2S_{12}b_1b_2 + S_{22}b_2^2 - S_{11}b_1^2$$

since $b_0 = B_0 = \bar{y}$. 

We can derive the following relations from \[
\begin{bmatrix}
    n & 0 & 0 \\
    0 & S_{11} & S_{12} \\
    0 & S_{21} & S_{22}
\end{bmatrix}
\begin{bmatrix}
    b_0 \\
    b_1 \\
    b_2
\end{bmatrix} =
\begin{bmatrix}
    \sum y_i \\
    \sum x_{i1}y_i \\
    \sum x_{i2}y_i
\end{bmatrix}
\]
and
\[
\begin{bmatrix}
    n & 0 & 0 \\
    0 & S_{11} & S_{12} \\
    0 & S_{21} & S_{22}
\end{bmatrix}
\begin{bmatrix}
    b_0 \\
    b_1 \\
    b_2
\end{bmatrix} =
\begin{bmatrix}
    \sum y_i \\
    \sum x_{i1}y_i \\
    \sum x_{i2}y_i
\end{bmatrix}
\]
for regression models with one and two predictors respectively

\[
B_1 = b_1 + b_{21}b_2, \quad b_{21} = \frac{S_{12}}{S_{11}}, \quad b_{12} = \frac{S_{21}}{S_{22}}
\]

Therefore we have

\[
SSR(X_1, X_2) - SSR(X_1) = S_{11}^2 + 2S_{12}b_1b_2 + S_{22}b_2^2 - S_{11}[b_1 + b_{21}b_2]^2
\]
\[
= 2S_{12}b_1b_2 + S_{22}b_2^2 - 2S_{11}b_1b_2b_{21} - S_{11}b_{21}^2b_2
\]
\[
= 2S_{12}b_1b_2 + S_{22}b_2^2 - 2S_{12}b_1b_2 - S_{11}b_{21}b_2^2
\]
\[
= S_{22}b_2^2 - S_{11}b_{21}b_2^2
\]

Add \( X_2 \) if \( C_2 < C_1 \), which is equivalent to

\[
\frac{SSR(X_1, X_2)}{\sigma^2} - n - 6 < \frac{SSR(X_1)}{\sigma^2} - n - 4 \Rightarrow SSR(X_1, X_2) - SSR(X_1) < 2\sigma^2.
\]

Thus add \( X_2 \) if

\[
S_{22}b_2^2 - S_{11}b_{21}b_2^2 > 2\sigma^2 \quad \text{or} \quad S_{22}b_2^2 \left[ 1 - \frac{S_{11}}{S_{22}} \times \frac{S_{12}^2}{S_{11}^2} \right] > 2\sigma^2 \quad \text{or} \quad S_{22}b_2^2 \left[ 1 - R_{12}^2 \right] > 2\sigma^2 \quad \text{or} \quad b_2^2 > \frac{2\sigma^2}{S_{22}[1 - R_{12}^2]}
\]
Therefore add $X_2$ if $b_2^2 > 2\text{Var}(b_2)$ or $\frac{b_2}{\text{Std}(b_2)} > \sqrt{2}$ which is exactly the same as the orthogonal case.

3.3 Simulation Results

In order to assess the performance of $C_p$ Mallows' statistics in identifying the true model, when we have orthogonal predictors, or multi-collinear predictors, or time series dependent error terms, the following simulation study is conducted:

For fixed sample size $n$ ($n=100$ and $n=500$) we generated 200 replicates, each of size $n$, from the regression model

$$Y = \beta_0 + \beta_1X_1 + \beta_2X_2 + \beta_3X_3 + \varepsilon$$

where $\beta_0 = 75, \beta_1 = 2, \beta_2 = -3, \beta_3 = 3.8$ and $\varepsilon \sim N(0, \sigma^2), \sigma = 2,10$. Independent variables $X_i, i=1,\ldots,5$ are normal random numbers with means 40, 50, 35, 50, 60 and standard deviations 5, 8, 6, 9, and 10 respectively. For orthogonal models $X_1, X_2, X_3$ are independent.

For multi-collinear model the correlation matrix of $X_i$'s is

$$
\begin{bmatrix}
1 & 0.8 & 0.9 \\
0.8 & 1 & 0.8 \\
0.9 & 0.8 & 1
\end{bmatrix}.
$$

For orthogonal or multi-collinear model with $AR$ error term, $\varepsilon_i = 1 - 0.9\varepsilon_{i-1} + a_i$ with $a_i \sim N(0, \sigma^2), \sigma = 2,10$. 
Similarly for orthogonal or multi-collinear model with MA error term, \( \varepsilon_t = 1 - 0.9a_{t-1} + a_t \) with \( \varepsilon_t = 1 - 0.9a_{t-1} + a_t \) and finally for models with ARMA error term, \( \varepsilon_t = 1 - 0.9\varepsilon_{t-1} + 0.9a_{t-1} + a_t \) with \( \varepsilon_t = 1 - 0.9a_{t-1} + a_t \). In all cases \( X_4 \) and \( X_5 \) are independent of \( X_i, i = 1, 2, 3 \).

After generating \( n \) observations, the SAS procedure REG with the model selection \( C_p \) option is used to select the best model once with \( X_1, X_2, \) and \( X_3 \) as potential predictors, then with \( X_1, X_2, X_3, \) and \( X_4 \) (\( X_4 \) is not related to \( Y \) and is independent of \( X_i, i = 1, 2, 3 \)) as potential predictors and finally with, \( X_1, X_2, X_3, X_4, \) and \( X_5 \) (\( X_4 \) and \( X_5 \) are not related to \( Y \) and are independent of \( X_i, i = 1, 2, 3 \)) as potential predictors. After repeating the experiment 200 times, the percentage of times that the correct model is selected, that is a model with only \( X_1, X_2, X_3 \), is calculated.

Note that for each model the proportions of selecting correct models across the two levels of variance were not statistically significantly different. P-values were greater than 0.2.

Note that \( R^2 \) reported in table 3.1 is calculated as \( R^2 = \frac{a}{a + b} \).

where
\[ a = \beta_1^2 V(X_1) + \beta_2^2 V(X_2) + \beta_3^2 V(X_3) + 2\beta_1\beta_2 \text{COV}(X_1, X_2) + 2\beta_1\beta_3 \text{COV}(X_1, X_3) + 2\beta_2\beta_3 \text{COV}(X_2, X_3) \]

and \( b = V(\varepsilon) \). The results are given in Table 3.1.
### TABLE 3.1

Percentages of correct model selection by $C_p$

<table>
<thead>
<tr>
<th>Model</th>
<th>Candidates $X_1, X_2, X_3$</th>
<th>Candidates $X_1, X_2, X_3, X_4$</th>
<th>Candidates $X_1, X_2, X_3, X_4, X_5$</th>
<th>$V(\varepsilon)$</th>
<th>$R^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Orth.</td>
<td>100</td>
<td>100</td>
<td>87</td>
<td>80</td>
<td>70</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>100</td>
<td>84.5</td>
<td>86</td>
<td>74.5</td>
</tr>
<tr>
<td>Orth-AR</td>
<td>100</td>
<td>100</td>
<td>85</td>
<td>86.5</td>
<td>70</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>100</td>
<td>87</td>
<td>87.5</td>
<td>72.5</td>
</tr>
<tr>
<td>Orth-MA</td>
<td>100</td>
<td>100</td>
<td>84</td>
<td>85</td>
<td>69.5</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>100</td>
<td>84</td>
<td>80.5</td>
<td>68.5</td>
</tr>
<tr>
<td>Orth-ARMA</td>
<td>100</td>
<td>100</td>
<td>87</td>
<td>80</td>
<td>70</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>100</td>
<td>84.5</td>
<td>86</td>
<td>74.5</td>
</tr>
<tr>
<td>Multi.</td>
<td>100</td>
<td>100</td>
<td>87</td>
<td>80</td>
<td>70</td>
</tr>
<tr>
<td></td>
<td>99</td>
<td>100</td>
<td>84</td>
<td>86</td>
<td>74</td>
</tr>
<tr>
<td>Multi-AR</td>
<td>100</td>
<td>100</td>
<td>85</td>
<td>86.5</td>
<td>70</td>
</tr>
<tr>
<td></td>
<td>67</td>
<td>100</td>
<td>60.5</td>
<td>87.5</td>
<td>50.5</td>
</tr>
<tr>
<td>Multi-MA</td>
<td>100</td>
<td>100</td>
<td>84</td>
<td>85</td>
<td>69.5</td>
</tr>
<tr>
<td></td>
<td>96</td>
<td>100</td>
<td>80.5</td>
<td>80.5</td>
<td>65</td>
</tr>
<tr>
<td>Multi-ARMA</td>
<td>100</td>
<td>100</td>
<td>87</td>
<td>80</td>
<td>70</td>
</tr>
<tr>
<td></td>
<td>99</td>
<td>100</td>
<td>84</td>
<td>86</td>
<td>74</td>
</tr>
</tbody>
</table>
Correct Model: A selected model, which is the same as simulated one.

As we increase the sample size and keep the number of candidates fixed, the performance of $C_p$ remains almost the same. (See Gilies R. Ducharme for comments on consistency of $C_p$). Note that when the sample size is 100 and the number of potential variables is 4, the variable $X_4$ which is unrelated to $Y$ is selected almost 16% of the times which is the same as theoretical value reported earlier. Also with the same sample size when two unrelated variables are added to the list of potential predictors the percentage of times that neither of the two unrelated variables was selected is close to 70% (approximately $(1-0.16) \times (1-0.16) \times 100$).

When only $X_1, X_2$, and $X_3$ are possible predictors, the performance of the Mallows’ $C_p$ statistics is perfect except for multi-collinear model with AR error term and sample size 100. When errors are highly correlated it may severely violate the basic assumption of regression analysis. Performance increased for sample size 500. The percentages of selecting the correct model decreases as the number of potential predictors increases.
Note that when the error term has time series structure and variance of \( \varepsilon \) is 100 performance of \( C_p \) decreased for sample size 100 and for sample size 500 performances increased or remained almost the same.

We repeat exactly the same simulation procedure with AIC and BIC as criterion for selecting the best model. Results are given in tables 3.2 and 3.3 respectively.
### TABLE 3.2

Percentages of correct model selection by AIC

<table>
<thead>
<tr>
<th>Model</th>
<th>$X_1, X_2, X_3$</th>
<th>$X_1, X_2, X_3, X_4$</th>
<th>$X_1, X_2, X_3, X_4, X_5$</th>
<th>$V(\varepsilon)$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$n = 100$</td>
<td>$n = 500$</td>
<td>$n = 100$</td>
<td>$n = 500$</td>
</tr>
<tr>
<td>Orth.</td>
<td>100</td>
<td>100</td>
<td>86</td>
<td>80</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>100</td>
<td>83</td>
<td>86</td>
</tr>
<tr>
<td>Orth-AR</td>
<td>100</td>
<td>100</td>
<td>83.5</td>
<td>86.5</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>100</td>
<td>86</td>
<td>87.5</td>
</tr>
<tr>
<td>Orth-MA</td>
<td>100</td>
<td>100</td>
<td>83.5</td>
<td>85</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>100</td>
<td>83.5</td>
<td>80.5</td>
</tr>
<tr>
<td>Orth-ARMA</td>
<td>100</td>
<td>100</td>
<td>86</td>
<td>80</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>100</td>
<td>83</td>
<td>86</td>
</tr>
<tr>
<td>Multi.</td>
<td>100</td>
<td>100</td>
<td>86</td>
<td>80</td>
</tr>
<tr>
<td></td>
<td>99</td>
<td>100</td>
<td>82.5</td>
<td>86</td>
</tr>
<tr>
<td>Multi-AR</td>
<td>100</td>
<td>100</td>
<td>83.5</td>
<td>86.5</td>
</tr>
<tr>
<td></td>
<td>67.5</td>
<td>100</td>
<td>60.5</td>
<td>87.5</td>
</tr>
<tr>
<td>Multi-MA</td>
<td>100</td>
<td>100</td>
<td>83.5</td>
<td>85</td>
</tr>
<tr>
<td></td>
<td>96</td>
<td>100</td>
<td>80</td>
<td>80.5</td>
</tr>
<tr>
<td>Multi-ARMA</td>
<td>100</td>
<td>100</td>
<td>86</td>
<td>80</td>
</tr>
<tr>
<td></td>
<td>99</td>
<td>100</td>
<td>82.5</td>
<td>86</td>
</tr>
</tbody>
</table>
The performance of AIC is almost similar to that of Mallows' \( C_p \) regardless of error term structure or being multicollinear. It is known that \( C_p \) and AIC are asymptotically equivalent to each other. (See Ryuei Nishii 1984).
### TABLE 3.3

Percentages of correct model selection by BIC

<table>
<thead>
<tr>
<th>Model</th>
<th>Candidates $X_1, X_2, X_3$</th>
<th>Candidates $X_1, X_2, X_3, X_4$</th>
<th>Candidates $X_1, X_2, X_3, X_4, X_5$</th>
<th>$V(\varepsilon)$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$n = 100$ $n = 500$</td>
<td>$n = 100$ $n = 500$</td>
<td>$n = 100$ $n = 500$</td>
<td></td>
</tr>
<tr>
<td>Orth.</td>
<td>100 100</td>
<td>88.5 81.5</td>
<td>73.5 69.5</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>100 100</td>
<td>86 86</td>
<td>77 72.5</td>
<td>100</td>
</tr>
<tr>
<td>Orth-AR</td>
<td>100 100</td>
<td>87.5 87</td>
<td>71.5 69.5</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>100 100</td>
<td>89 88</td>
<td>75 75</td>
<td>100</td>
</tr>
<tr>
<td>Orth-MA</td>
<td>100 100</td>
<td>86.5 85.5</td>
<td>74 71.5</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>100 100</td>
<td>86.5 81</td>
<td>72 66</td>
<td>100</td>
</tr>
<tr>
<td>Orth-ARMA</td>
<td>100 100</td>
<td>88 81.5</td>
<td>73.5 69.5</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>100 100</td>
<td>86 86</td>
<td>77 72.5</td>
<td>100</td>
</tr>
<tr>
<td>Multi.</td>
<td>100 100</td>
<td>88.5 81.5</td>
<td>73.5 76.5</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>99 100</td>
<td>85.5 86</td>
<td>69.5 72.5</td>
<td>100</td>
</tr>
<tr>
<td>Multi-AR</td>
<td>100 100</td>
<td>87.5 87</td>
<td>71.5 69.5</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>65 100</td>
<td>59.5 88</td>
<td>51 75</td>
<td>100</td>
</tr>
<tr>
<td>Multi-MA</td>
<td>100 100</td>
<td>86.5 85.5</td>
<td>74 71.5</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>94.5 100</td>
<td>81.5 81</td>
<td>67 66</td>
<td>100</td>
</tr>
<tr>
<td>Multi-ARMA</td>
<td>100 100</td>
<td>88.5 81.5</td>
<td>73.5 69.5</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>99 100</td>
<td>85.5 86.5</td>
<td>76.5 72.5</td>
<td>100</td>
</tr>
</tbody>
</table>
The performance of BIC in selecting the correct model is slightly better than AIC and $C_p$ for sample size 100 and is almost the same for sample size 500. Let $p-1$ denote number of selected predictors. We classified each selected model as being under-fitted ($p \leq 3$) correct ($p = 4$ and $X_1, X_2, X_3$ were selected) incorrect ($p = 4$ but $X_4$ or $X_5$ or both are selected) over-fitted ($p = 5$ or $p = 6$). Percentages of each class are given in tables 3.4, 3.5 and 3.6.
### Table 3.4

Percentages of classified models by $C_p$

<table>
<thead>
<tr>
<th>Candidates</th>
<th>$X_1, X_2, X_3$</th>
<th>$X_1, X_2, X_3, X_4$</th>
<th>$X_1, X_2, X_3, X_4, X_5$</th>
<th>$n$</th>
<th>$V(\mathcal{E})$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Orth</td>
<td>unde 100 0 0 0</td>
<td>unde 87 0 13</td>
<td>unde 70 0 30</td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td>corr 100 0 0 0</td>
<td>corr 80 0 20</td>
<td>corr 68 0 32</td>
<td>500</td>
<td></td>
</tr>
<tr>
<td></td>
<td>incr 100 0 0 0</td>
<td>incr 84.5 0 15.5</td>
<td>incr 74.5 0 25.5</td>
<td>100</td>
<td></td>
</tr>
<tr>
<td>Orth-AR</td>
<td>unde 100 0 0 0</td>
<td>unde 87 0 13</td>
<td>unde 72.5 0 27.5</td>
<td>100</td>
<td></td>
</tr>
<tr>
<td></td>
<td>corr 100 0 0 0</td>
<td>corr 85 0 15</td>
<td>corr 70 0 30</td>
<td>100</td>
<td></td>
</tr>
<tr>
<td></td>
<td>incr 100 0 0 0</td>
<td>incr 86 0 14</td>
<td>incr 69 0 31</td>
<td>500</td>
<td></td>
</tr>
<tr>
<td>Orth-MA</td>
<td>unde 100 0 0 0</td>
<td>unde 87 0 13</td>
<td>unde 72.5 0 27.5</td>
<td>100</td>
<td></td>
</tr>
<tr>
<td></td>
<td>corr 100 0 0 0</td>
<td>corr 85 0 15</td>
<td>corr 70 0 30</td>
<td>100</td>
<td></td>
</tr>
<tr>
<td></td>
<td>incr 100 0 0 0</td>
<td>incr 84 0 16</td>
<td>incr 69.5 0 30.5</td>
<td>100</td>
<td></td>
</tr>
<tr>
<td>Orth-ARMA</td>
<td>unde 100 0 0 0</td>
<td>unde 87 0 13</td>
<td>unde 70 0 30</td>
<td>100</td>
<td></td>
</tr>
<tr>
<td></td>
<td>corr 100 0 0 0</td>
<td>corr 84.5 0 15.5</td>
<td>corr 70 0 30</td>
<td>100</td>
<td></td>
</tr>
<tr>
<td></td>
<td>incr 100 0 0 0</td>
<td>incr 80 0 20</td>
<td>incr 68 0 32</td>
<td>500</td>
<td></td>
</tr>
<tr>
<td>Multicollinear</td>
<td>unde 100 0 0 0</td>
<td>unde 87 0 13</td>
<td>unde 72.5 0 27.5</td>
<td>100</td>
<td></td>
</tr>
<tr>
<td></td>
<td>corr 100 0 0 0</td>
<td>corr 80 0 20</td>
<td>corr 68 0 32</td>
<td>500</td>
<td></td>
</tr>
<tr>
<td></td>
<td>incr 100 0 0 0</td>
<td>incr 84 1 15</td>
<td>incr 74 1 25</td>
<td>100</td>
<td></td>
</tr>
<tr>
<td>Multi-AR</td>
<td>unde 100 0 0 0</td>
<td>unde 87 0 13</td>
<td>unde 70 0 30</td>
<td>100</td>
<td></td>
</tr>
<tr>
<td></td>
<td>corr 100 0 0 0</td>
<td>corr 85 0 15</td>
<td>corr 70 0 30</td>
<td>100</td>
<td></td>
</tr>
<tr>
<td></td>
<td>incr 100 0 33 67</td>
<td>incr 60.5 6 6.5 22.5</td>
<td>incr 50.5 6.5 18.5 100</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Multi-MA</td>
<td>unde 100 0 0 0</td>
<td>unde 87 0 13</td>
<td>unde 70 0 30</td>
<td>100</td>
<td></td>
</tr>
<tr>
<td></td>
<td>corr 100 0 0 0</td>
<td>corr 84 0 16</td>
<td>corr 69.5 0 30.5</td>
<td>100</td>
<td></td>
</tr>
<tr>
<td></td>
<td>incr 100 0 0 0</td>
<td>incr 85 0 15</td>
<td>incr 71 0 29</td>
<td>500</td>
<td></td>
</tr>
<tr>
<td>Multi-ARMA</td>
<td>unde 100 0 0 0</td>
<td>unde 87 0 13</td>
<td>unde 70 0 30</td>
<td>100</td>
<td></td>
</tr>
<tr>
<td></td>
<td>corr 100 0 49 96</td>
<td>corr 80.5 0.5 15.5 3.5</td>
<td>corr 65 0.5 31 100</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>incr 100 0 0 0</td>
<td>incr 86 0 14</td>
<td>incr 72.5 0 27.5</td>
<td>500</td>
<td></td>
</tr>
<tr>
<td></td>
<td>corr 100 0 49 96</td>
<td>corr 80.5 0.5 15.5 3.5</td>
<td>corr 65 0.5 31 100</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>incr 100 0 0 0</td>
<td>incr 86 0 14</td>
<td>incr 72.5 0 27.5</td>
<td>500</td>
<td></td>
</tr>
</tbody>
</table>
Table 3.5

Percentages of classified models by AIC

<table>
<thead>
<tr>
<th>Candidates</th>
<th>Candidates</th>
<th>Candidates</th>
<th>n</th>
<th>V(\mathcal{C})</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$X_1, X_2, X_3$</td>
<td>$X_1, X_2, X_3, X_4$</td>
<td>$X_1, X_2, X_3, X_4, X_5$</td>
<td></td>
</tr>
<tr>
<td>Orth</td>
<td>Orth-AR</td>
<td>Orth-MA</td>
<td>Orth-ARMA</td>
<td>Multicol linear</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
</tr>
</tbody>
</table>
### Table 3.5
Percentages of classified models by BIC

<table>
<thead>
<tr>
<th>Candidates</th>
<th>$X_1, X_2, X_3$</th>
<th>$X_1, X_2, X_3, X_4$</th>
<th>$X_1, X_2, X_3, X_4, X_5$</th>
<th>$n$</th>
<th>$\text{V}(\varepsilon)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Orth</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>unde</td>
<td>corr</td>
<td>incr</td>
<td>over</td>
<td>unde</td>
</tr>
<tr>
<td>Orth-AR</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>100</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Orth-MA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>100</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Orth-ARMA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>100</td>
<td>0</td>
<td>12</td>
<td>0</td>
</tr>
<tr>
<td>Multicol linear</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>100</td>
<td>0</td>
<td>14</td>
<td>0</td>
</tr>
<tr>
<td>Multi-AR</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>100</td>
<td>0</td>
<td>14</td>
<td>0</td>
</tr>
<tr>
<td>Multi-MA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>100</td>
<td>0</td>
<td>14</td>
<td>0</td>
</tr>
<tr>
<td>Multi-ARMA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>100</td>
<td>0</td>
<td>14</td>
<td>0</td>
</tr>
</tbody>
</table>

In all cases except when variance of $\varepsilon$ is 4, sample size is 100, and error term follows an AR model, the wrong selected model is over fitted. This supports the theoretical finding.
that any predictor, which is independent of $Y$, has $16\%$ chance of being entered in the selected model. When variance of $\epsilon$ is small, sample size is moderate, and error terms are highly correlated one or two predictors did not enter in the selected model. However when sample size or variance $\epsilon$ increased this problem almost disappeared.
CHAPTER FOUR

EFFECTS OF MULTICOLLINEARITY ON PREDICTIONS

Regression models are widely used for a variety of applications. Multicollinearity is a problem, which may affect the usefulness of a regression model. Multicollinearity has several effects, but here we will only consider its effects on predictions.

4.1 Sources of Multicollinearity

If there is no linear relationship between the predictors, they are said to be orthogonal. When there are near linear dependencies between the predictors, the multicollinearity problem is said to exist.

The multiple regression model is written as

\[ Y = X\beta + \epsilon \]

where \( Y, X, \beta, \) and \( \epsilon \) are defined in the previous chapters. For convenience we assume that \( Y \) and \( X \)'s are centered and re-scaled to unit length.

Centering involves taking the difference between each observation and the mean of all observations for the variable.
and scaling involves expressing the centered observations in units of the standard deviation of the observations for the variables. Thus the response variable $Y$ and the predictor variables $X$'s are standardized as follows:

$$ y_i^0 = \frac{y_i - \bar{y}}{S_{yy}^{1/2}} $$

$$ w_{ij} = \frac{x_{ij} - \bar{x}_j}{S_{ij}^{1/2}} $$

where $S_{jj} = \sum (x_{ij} - \bar{x}_j)^2$ and $S_{yy} = \sum (y_i - \bar{y})^2$. Under this transformation the vector of least squares regression coefficients is

$$ \hat{\beta} = (WW)^{-1} WY^0 $$

where $W = (w_{ij})_{n \times p}$ and $Y^0 = [y_1^0, ..., y_n^0]'$.

In the unit length scaling, $WW$ is in the form of correlation matrix; that is

$$ WW = \begin{bmatrix} 1 & r_2 & \cdots & r_p \\ r_2 & 1 & \cdots & r_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ r_p & r_{2p} & \cdots & 1 \end{bmatrix} $$

where $r_{ij} = \frac{S_{ij}}{\sqrt{S_{ii}S_{jj}}}$ and $WY^0 = [r_{1Y}, r_{2Y}, \ldots, r_{pY}]'$ with $r_{jY} = \frac{S_{jY}}{\sqrt{S_{jj}S_{YY}}}$.

**Definition:** The vectors $X_1, X_2, ..., X_p$ are linearly dependent if there is a set of constants $a_1, a_2, ..., a_p$, not all zero, such that
When \( \sum_{j=1}^{p} a_j X_j = 0 \) holds for a subset of the columns of \( X \), then \( (XX)^{-1} \) does not exist and \( \text{Rank}(X'X) < p \).

When \( \sum_{j=1}^{p} a_j X_j \) is approximately zero the problem of multicollinearity is said to exist. This problem in the literature is also known as ill-conditioning (A matrix is called ill-conditioned if it is near singularity). In fact multicollinearity is a form of ill conditioning in the \( (XX) \) matrix. Thus every data set will suffer from multicollinearity to some extent.

The characteristic roots or eigenvalues of \( (XX) \), say \( \lambda_1, \lambda_2, \ldots, \lambda_p \), can be used to measure the extent of multicollinearity in the data. If there is at least one near dependency in the data, one or more of the eigenvalues will be small. This implies that there are near dependencies among the columns of \( X' \)'s. The condition number, which can be defined as

\[ k = \frac{\lambda_{\text{max}}}{\lambda_{\text{min}}} \]

measures the spread in the value spectrum of \( (XX) \). Generally if the condition number is less than 100, there is no serious problem with multicollinearity. Condition numbers between 100
and 1000 imply moderate to strong multicollinearity. If \( \frac{\lambda_{\text{max}}}{\lambda_{\text{min}}} \) exceeds 1000, severe multicollinearity is indicated.

It is well known that multicollinearity tends to produce least squares estimates that are too large in absolute value. While the method of least squares will generally produce poor estimates of the individual model parameters when strong multicollinearity is present, this does not necessarily imply that the fitted model is a poor predictor. If predictors are confined to regions of the x-space when the multicollinearity holds approximately, the fitted model often produce satisfactory predictions. This occurs because the linear combination \( \sum \beta_j x_j \) may be estimated quite well, even though the individual parameters are estimated poorly. That is, if the original data lie approximately along the hyper-plane defined by \( \sum_{j=1}^{p} a_j X_j = 0 \), then future observations that also lie near this hyper-plane can often be precisely predicted despite the inadequate estimates of the individual model parameters.

In predicting new responses at a given point \( x \), one must be careful about extrapolating beyond the region containing the original observations. It is postulated that when the point \( x \) is "inside" the original data the predicted and observed values agree closely. That is the model is adequate for
interpolation within the range of the original data in spite of poor estimates for the individual parameters. When multicollinearity is suspected and the point $x$ is "outside" the range of the original data then the model may predict very poorly. In other words when multicollinearity exists, the fitted model may not be adequate for extrapolation.

**Definition:** The smallest convex set containing all the original $n$ points $(x_{i1}, x_{i2}, ..., x_{ip}), i = 1, 2, ..., n$, is called the Regressor Variable Hull (RVH).

If the point $x_0 = (x_{01}, x_{02}, ..., x_{0p})$ lies inside or on the boundary of the RVH, then prediction involves interpolation, while if this point lies outside the RVH extrapolation is required.

The following series of graphs represent the RVH plots of $X_1$ vs $X_2$ for different regression ($\text{Corr}(X_1, X_2) = 0, 0.5, -0.5, 0.9, -0.9, 0.999, -0.999$) respectively.
Corr(X1, X2) = 0
Corr(X1, X2) = 0.5

Corr(X1, X2) = -0.5
Corr(X₁, X₂) = 0.9

Corr(X₁, X₂) = -0.9
\text{Corr}(X_1, X_2) = 0.999
Let $h_{ii}$ denote the diagonal elements of the hat matrix

$$H = X(X'X)^{-1}X'$$
and $h_{\text{max}} = \max_{i=1,2,...,n} h_{ii}$.

**Definition:** The set of points $x$ that satisfy

$$x'(X'X)^{-1}x \leq h_{\text{max}}$$

is an ellipsoid that enclose all points inside the RVH.

Thus for points $x_0 = (x_{01}, x_{02}, ..., x_{0p})$ for which $x_0'(X'X)^{-1}x_0 > h_{\text{max}}$, prediction would be considered as extrapolation and for the points $x_0 = (x_{01}, x_{02}, ..., x_{0p})$ for which $x_0'(X'X)^{-1}x_0 \leq h_{\text{max}}$, prediction would be considered interpolation. The effect of multicollinearity on the prediction is briefly discussed by Graybill (1994).

### 4.2 Simulation Results

In order to investigate the behavior of the mean and variance of the prediction errors when multicollinearity exists, the following simulation study was conducted:

For fixed sample size $n$ ($n=50$, 100, and 500) we generated 200 replications, each of size $n+500$, from the following regression model

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \varepsilon$$

where $\beta_0 = 25, \beta_1 = 3, \beta_2 = -5, \beta_3 = 7$ and $\varepsilon \sim N(0, \sigma^2)$. For the first $n$ observations predictors $X_i, i=1,2,3$ are normal random numbers
with means 20, 30, 40 and standard deviations 5, 4, and 6 respectively. For the next 500 observations means of $X_i$'s were as above when we considered interpolation and were 40, 60, and 80 when we considered extrapolation. In either case the standard deviations of $X_i$'s were set equal to 5, 4, and 6. Correlations between $X_i$'s are as follows:

$$r_{12} = Cor(X_1, X_2), \quad r_{13} = Cor(X_1, X_3), \quad r_{23} = Cor(X_2, X_3).$$

In our simulation study we considered four sets of correlation.

After generating $n+500$ observations, the SAS procedure REG was used to fit a multiple regression model with $X_1, X_2,$ and $X_3$ as predictors to the first $n$ observations. Then the fitted model was used to predict the next 500 generated observations. Each predicted values were classified as interpolation or extrapolation depending on the location of observed $X_i$ values. That is $X_i$'s being inside or outside the ellipsoid that enclosed all points inside the RVH. Let $\mu_i, i = 1, 2, \ldots, 500$ and $\hat{y}_i, i = 1, 2, \ldots, 500$ denote the $i$th true mean and predicted values. The prediction error is calculated as $\hat{y}_i - \mu_i$. In each replication we had 500 prediction errors. We calculated the sample mean of these 500 prediction errors squared. The reported MSE is the average of 200 sample means of prediction
errors squared. The condition number in table 4.1 is the average of 200 condition numbers. Simulation results are given in table 4.1.

Table 4.1

<table>
<thead>
<tr>
<th>$r_{12}$</th>
<th>$r_{13}$</th>
<th>$r_{23}$</th>
<th>Condition Number</th>
<th>MSE</th>
<th>Sample size</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1.571</td>
<td>1.914</td>
<td>61.9</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1.354</td>
<td>0.956</td>
<td>28.6</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1.152</td>
<td>0.196</td>
<td>6.4</td>
</tr>
<tr>
<td>0.5</td>
<td>0.7</td>
<td>0.4</td>
<td>8.023</td>
<td>1.914</td>
<td>47.5</td>
</tr>
<tr>
<td>0.5</td>
<td>0.7</td>
<td>0.4</td>
<td>7.426</td>
<td>0.967</td>
<td>21.2</td>
</tr>
<tr>
<td>0.5</td>
<td>0.7</td>
<td>0.4</td>
<td>7.288</td>
<td>0.196</td>
<td>3.9</td>
</tr>
<tr>
<td>-0.95</td>
<td>0.97</td>
<td>-0.94</td>
<td>114.0</td>
<td>1.924</td>
<td>915</td>
</tr>
<tr>
<td>-0.95</td>
<td>0.97</td>
<td>-0.94</td>
<td>104.3</td>
<td>0.964</td>
<td>451</td>
</tr>
<tr>
<td>-0.95</td>
<td>0.97</td>
<td>-0.94</td>
<td>101.0</td>
<td>0.196</td>
<td>80</td>
</tr>
<tr>
<td>-0.999</td>
<td>0.95</td>
<td>-0.95</td>
<td>3276</td>
<td>1.902</td>
<td>30657</td>
</tr>
<tr>
<td>-0.999</td>
<td>0.95</td>
<td>-0.95</td>
<td>3037</td>
<td>0.965</td>
<td>18560</td>
</tr>
<tr>
<td>-0.999</td>
<td>0.95</td>
<td>-0.95</td>
<td>2935</td>
<td>0.198</td>
<td>3544</td>
</tr>
<tr>
<td>-0.99999</td>
<td>0.95</td>
<td>-0.95</td>
<td>254550</td>
<td>1.942</td>
<td>3171653</td>
</tr>
<tr>
<td>-0.99999</td>
<td>0.95</td>
<td>-0.95</td>
<td>258230</td>
<td>0.960</td>
<td>1832389</td>
</tr>
<tr>
<td>-0.99999</td>
<td>0.95</td>
<td>-0.95</td>
<td>282660</td>
<td>0.198</td>
<td>354558</td>
</tr>
</tbody>
</table>
We see that the MSE of the prediction errors is small for interpolated values regardless of the existence or nonexistence of the multicollinearity. Therefore multicollinearity had no effect on the prediction performance of the fitted model when prediction was interpolation. When the fitted model was extrapolation the MSE of the residuals increased, as correlation between $X_i$’s increased. Therefore when we use a multicollinear fitted model to extrapolate the predicted values would have a large MSE.

In practice a predictive model may often be used for prediction in a situation where there might be more heterogeneity than in the original sample used to construct the regression model or the mean of some key predictors may change. To address these situations we considered four scenarios to study the effect of changes in the mean or variances of the predictors on predictions.

Note that we used the first $n$ observations for fitting and the next 500 for prediction validation. The means and standard deviations of the first $n$ predictor variables were 20, 30, 40, 25, 16, and 36 respectively. In the four considered scenarios the means and standard deviations were as follows:

Scenario 1 :20, 30, 40, 25, 16, 36

Scenario 2 :25, 37.5, 50, 25, 16, 36

Scenario 3 :20, 30, 40, 31.25, 20, 40
Scenario 4: 25, 37.5, 50, 31.25, 20, 45

These four scenarios can be tabulated as follows:

<table>
<thead>
<tr>
<th>Same variance</th>
<th>Larger variance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Same mean</td>
<td>Scenario 1</td>
</tr>
<tr>
<td>Larger mean</td>
<td>Scenario 2</td>
</tr>
</tbody>
</table>

Scenario 1 may be considered as interpolation and other scenarios as extrapolation.

Regression model is $Y = 25 + 3X_1 - 5X_2 + 7X_3 + \varepsilon$.

Simulation results are given in table 4.2.
<table>
<thead>
<tr>
<th>$r_{12}$</th>
<th>$r_{13}$</th>
<th>$r_{23}$</th>
<th>Scenario 1 (interpolation)</th>
<th>Scenario 2 (extrapolation)</th>
<th>Scenario 3 (extrapolation)</th>
<th>Scenario 4 (extrapolation)</th>
<th>$v(E)$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>n=50 n=100 n=500</td>
<td>n=50 n=100 n=500</td>
<td>n=50 n=100 n=500</td>
<td>n=50 n=100 n=500</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.334 0.154 0.032</td>
<td>0.931 0.448 0.097</td>
<td>0.395 0.185 0.038</td>
<td>0.991 0.478 0.103</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>8.722 4.066 0.739</td>
<td>26.284 10.162 2.012</td>
<td>10.381 4.828 0.880</td>
<td>27.973 10.918 2.152</td>
<td></td>
</tr>
<tr>
<td>0.5</td>
<td>0.7</td>
<td>0.4</td>
<td>0.342 0.156 0.032</td>
<td>0.799 0.372 0.075</td>
<td>0.404 0.187 0.038</td>
<td>0.862 0.403 0.081</td>
<td>4</td>
</tr>
<tr>
<td>-0.95</td>
<td>0.97</td>
<td>-0.94</td>
<td>0.341 0.155 0.031</td>
<td>9.877 4.444 1.018</td>
<td>0.403 0.185 0.037</td>
<td>9.939 4.474 1.023</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>8.667 4.071 0.738</td>
<td>274.02 108.25 18.61</td>
<td>10.313 4.829 0.880</td>
<td>275.77 109.01 18.75</td>
<td></td>
</tr>
<tr>
<td>-0.999</td>
<td>0.95</td>
<td>-0.95</td>
<td>0.334 0.154 0.031</td>
<td>324.86 156.94 35.32</td>
<td>0.395 0.185 0.037</td>
<td>324.92 156.97 35.32</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>8.707 4.071 0.737</td>
<td>9386 4316 830</td>
<td>10.363 4.829 0.878</td>
<td>9388 4316 830</td>
<td></td>
</tr>
<tr>
<td>-0.99999</td>
<td>0.95</td>
<td>-0.95</td>
<td>0.337 0.151 0.031</td>
<td>31124 14975 3349</td>
<td>0.398 0.181 0.037</td>
<td>31124 14975 3349</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>8.696 4.078 0.739</td>
<td>911234 432213 86006</td>
<td>10.351 4.838 0.880</td>
<td>911240 432215 86006</td>
<td></td>
</tr>
</tbody>
</table>

Table 4.2

MSE of validation residuals for four scenarios
When we increased the variances of $X_i$'s by 25% the MSE of the validation residuals did not change. However when we increased the means of $X_i$'s by 25%, the MSE of the validation residuals increased for multicollinear models.
Akaike Information Criterion:

Let \( f \) and \( g \) be the probability density functions of the true and the hypothetical distributions respectively. Let \( f_N \) denote the probability density function estimate based on the conceptual random sampling of \( N \) observations from \( g \). Let \( B(f;g) \) stand for the entropy of the distribution \( f \) with respect to the distribution \( g \). This is based on Sanov’s results (1961) which justifies the definition of

\[
B(f;g) = -\int f(z) \ln(f(z)/g(z)) dz
\]

as \( \lim_{\varepsilon \downarrow 0} \lim_{N \to \infty} N^{-1} \ln \Pr(\sup_x |f_N(x) - f(x)| < \varepsilon) \). Note that \( B(f;g) = 0 \) if and only if \( f \equiv g \) and that \( B(f;g) \leq 0 \). Akaike argued that \( -B(f;g) \) may be used as a measure of discrepancy between \( f \) and \( g \).

Suppose that the data set \( X \) of \( N \) observations is given. Akaike suggests that the purpose of statistical analysis of \( X \) is the prediction of future observations \( Y \) whose distribution is identical to that for the elements of \( X \). The prediction is realized by specifying a distribution \( g(Y|X) \), the predictive distribution of \( Y \), as a function of the available data \( X \). Assume that \( f(Y) \) is the distribution of \( Y \).
Goodness of \( g(Y|X) \) as an estimate of \( f(Y) \) is measured by the entropy of \( f(Y) \), with respect to \( g(Y|X) \), that is

\[
B\{f(.);g(.|X)\} = \int \left( \frac{f(Y)}{g(Y|X)} \right) \ln \left( \frac{f(Y)}{g(Y|X)} \right) g(Y|X) dY
\]

\[
= \int f(Y) \ln g(Y|X) dY - \int f(Y) \ln f(Y) dY
\]

\[
= E_Y \ln g(Y|X) - \text{constant}
\]

So the goodness of the estimation procedure specified by \( g(Y|X) \) is measured by \( E_X E_Y \ln g(Y|X) \) which is the average over \( X \) of the expected log likelihood of the model \( g(Y|X) \), w.r.t. a future observation \( Y \). Suppose that \( X \) and \( Y \) are independent. If \( g(.|X) = g(.|\Theta) \), that is a distribution specified by a fixed parameter \( \Theta \), then \( \ln g(X|X) = \ln g(X|\Theta) \), which is the log likelihood of the parameter \( \Theta \). Akaike proposes that the log likelihood of the data-dependent model \( g(.|X) \), as distinct from the log likelihood of the parameter \( \Theta \), be defined by

\[
I\{g(.|X)\} = \ln g(X|X) + C. \]

Let \( g(Y|\Theta(X)) \) denote \( M, m = 1, \ldots, M \) competing models. Under the assumption that the true model belongs to each of these \( M \) models, Akaike showed that the constant \( C \) is equal to \(-\text{dim}_m \Theta(X)\). Note that as \( N \to \infty \), the likelihood ratio statistics \( 2\ln g(X|\Theta(X)) - 2\ln g(X|\Theta_0) \) has \( \chi^2 \) distribution. Akaike has proposed that the model that maximizes \( \ln g(X|\Theta(X)) - r \) over \( m = 1, \ldots, M \), shall be adopted. In
practice the maximization of $\ln g(X|m \hat{\Theta}(X)) - r$ is replaced by the minimization of $-2\ln g(X|m \hat{\Theta}(X)) + 2r$ or

$$AIC(m) = -2\ln(\text{maximized likelihood}) + 2p$$

where $p$ denotes number of independently adjusted parameters.
Sample SAS Programs

This SAS program generates n observations and fits different regression models to the generated data.

This program handles simulations for chapter 3.

```sas
%let nrep = 200;
%let m1 = 40; mean X1
%let m2 = 50; mean X2
%let m3 = 35; mean X3
%let m4 = 50; mean X4
%let m5 = 60; mean X5
%let s11 = 25; Var X1
%let s22 = 64; Var X2
%let s33 = 36; Var X3
%let s44 = 81; Var X4
%let s55 = 100; Var X5
%let beta0 = 75;
%let beta1 = 2;
%let beta2 = -3.0;
%let beta3 = 3.8;

%macro fit(phi0, phi1, theta1, r12, r13, r23, model);
  data temp;
  run;
  data temp;
    m1 = &m1;
    m2 = &m2;
    m3 = &m3;
    m4 = &m4;
    m5 = &m5;
    s11 = &s11;
    s22 = &s22;
    s33 = &s33;
    s44 = &s44;
    s55 = &s55;
    se = &se;
    r12 = &r12;
    r13 = &r13;
    r23 = &r23;
    s12 = r12 * sqrt(s11) * sqrt(s22);
    s13 = r13 * sqrt(s11) * sqrt(s33);
    s23 = r23 * sqrt(s22) * sqrt(s33);
    s21 = s12;
    s32 = s23;
    s31 = s13;
    seed = 10;
  do se = 4, 100;
    do n = 100, 500;
      do rep = 1 to &nrep;
```

do i = 1 to n;
  x3 = m3 + sqrt(s33) * rannor(seed) ;
  m = m2 + (s23/s33)*(x3 - m3) ;
  s = s22*(1-r23**2) ;
  x2 = m + sqrt(s) * rannor(seed) ;
  c1 = 1 / (s22*s33 - s23**2) ;
  a = c1 *( (x2 - m2 ) * ( s12 * s33 - s13 * s23 ) +
          (x3 - m3 ) * ( s13 * s22 - s12 * s23 ) ) ;
  m = m1 + a ;
  s = s11 - c1 * (s12**2 * s33 - 2 * s12 * s13 * s23 +
                    s13**2 * s22 ) ;
  x1 = m + sqrt(s) * rannor(seed) ;
  x4 = m4 + sqrt(s44) * rannor(seed) ;
  x5 = m5 + sqrt(s55) * rannor(seed) ;
  el = 0 ;
    al = 0 ;
    do ii = 1 to 10 ;
      e = sqrt(se) * rannor(seed) ;
      a = &phi0 + &phi1 * al + &theta1 * el + e ;
      al = a ;
      el = e ;
    end ;
    error = a ;
    output ;
  end ;
end ;
keep rep n x1 x2 x3 x4 x5 error r12 r13 r23 rsquare se ;
run ;
data temp ; set temp ;
  y = &beta0 + &beta1 * x1 + &beta2 * x2 + &beta3 * x3 + error ;
run ;
proc sort data=temp ; by se rep n ; run ;
**** REGRESSION X1 X2 X3 and Cp ;
proc reg data=temp outest = pl cp noprint ;
  by se rep n ;
  model y = x1 x2 x3 / selection = cp ;
run ;
data rahim ; set pl ; by se rep n ; if first.n ;
  if x1 ne . then xlin = 'yes' ; else xlin = 'no' ;
  if x2 ne . then x2in = 'yes' ; else x2in = 'no' ;
  if x3 ne . then x3in = 'yes' ; else x3in = 'no' ;
  if x1 = . or x2 = . or x3 = . then fit = 'underfitted' ;
  if x1 ne . and x2 ne . and x3 ne . then fit = 'correct' ;
  /*
     if _IN_ = 3 and x1 ne . and x2 ne . and x3 ne . then fit='correct'
     else fit = 'incorrect';
  */
run ;
proc sort data=rahim ; by se n ; run ;
proc freq data=rahim noprint ; by se n ;
  tables xlin*x2in*x3in*fit /out=outl ;
run;

title1"Cp statistics & model model";
title2' X1 X2 and X3 candidates';
proc print data=out1; run;

***** X1 X2 X3 X4 Cp;
proc reg data=temp outest = p2 cp noprint;
  by se rep n;
  model y = x1 x2 x3 x4 / selection = cp;
run;
data rahim1 ; set p2 ; by se rep n ; if first.n;
  if x1 ne . then x1in = 'yes' ; else x1in = 'no' ;
  if x2 ne . then x2in = 'yes' ; else x2in = 'no' ;
  if x3 ne . then x3in = 'yes' ; else x3in = 'no' ;
  if x4 ne . then x4in = 'yes' ; else x4in = 'no' ;
  if _IN_ lt 3 then fit = 'underfitted ';
  if _IN_ ge 3 then fit = 'overfitted ';
  if _IN_ = 3 and x1 ne . and x2 ne . and x3 ne . then fit = 'correct ';
  if _IN_ = 3 and x4 ne . then fit='incorrect ';
/*
  if _IN_ = 3 and x1 ne . and x2 ne . and x3 ne . then fit='correct ';
  else fit = 'incorrect'; */
run;

proc sort data=rahim1 ; by se n ; run;
proc freq data=rahim1 noprint; by se n;
  tables x1in*x2in*x3in*x4in*fit / out=outS;
run;

title1"Cp statistics & model model";
title2' X1 X2 X3 and X4 candidates';
proc print data=out2; run;

****** X1 X2 X3 X4 X5 Cp;
proc reg data=temp outest = p3 cp noprint;
  by se rep n;
  model y = x1 x2 x3 x4 x5 / selection = cp;
run;
data rahim2 ; set p3 ; by se rep n ; if first.n;
  if x1 ne . then x1in = 'yes' ; else x1in = 'no' ;
  if x2 ne . then x2in = 'yes' ; else x2in = 'no' ;
  if x3 ne . then x3in = 'yes' ; else x3in = 'no' ;
  if x4 ne . then x4in = 'yes' ; else x4in = 'no' ;
  if x5 ne . then x5in = 'yes' ; else x5in = 'no' ;
  if _IN_ lt 3 then fit = 'underfitted ';
  if _IN_ ge 3 then fit = 'overfitted ';
  if _IN_ = 3 and x1 ne . and x2 ne . and x3 ne . then fit = 'correct ';
  if _IN_ = 3 then do;
    if x4 ne . or x5 ne . then fit = 'incorrect ';
    end;
/*
  if _IN_ = 3 and x1 ne . and x2 ne . and x3 ne . then fit='correct ';
  else fit = 'incorrect'; */
run;
**proc sort data=rahim2 ; by se n ; run ;**
**proc freq data=rahim2 noprint; by se n ;**
  **tables x1in*x2in*x3in*x4in*fit /out=out3 ;**
*run ;
**title1"Cp statistics & model model";**
**title2' X1 X2 X3 X4 and X5 candidates';**
**proc print data=out3 ; run ;**
**%mend;**

**%fit(phi0=0, phi1=0, theta1=0, r12=0, r13=0, r23=0, model=orthogonal);**
**%fit(phi0=1, phi1=-0.9, theta1=0, r12=0, r13=0, r23=0, model=orthogonal with AR error);**
**%fit(phi0=0, phi1=0, theta1=-0.9, r12=0, r13=0, r23=0, model=orthogonal with MA error);**
**%fit(phi0=1, phi1=-0.9, theta1=0.9, r12=0, r13=0, r23=0, model=orthogonal with ARMA error);**
**%fit(phi0=0, phi1=0, theta1=0, r12=0.8, r13=0.9, r23=0.8, model=multicollinear);**
**%fit(phi0=1, phi1=-0.9, theta1=0, r12=0.8, r13=0.9, r23=0.8, model=multiplicollinear with AR error);**
**%fit(phi0=0, phi1=0, theta1=-0.9, r12=0.8, r13=0.9, r23=0.8, model=multiplicollinear with MA error);**
**%fit(phi0=1, phi1=-0.9, theta1=0.9, r12=0.8, r13=0.9, r23=0.8, model=multiplicollinear with ARMA error);**

*************** AIC ***************
**%macro fitaic(phi0, phi1, theta1, r12, r13, r23, model);**

**data temp ; run ;**
**data temp ;**
  **m1 = &m1 ;**
  **m2 = &m2 ;**
  **m3 = &m3 ;**
  **m4 = &m4 ;**
  **m5 = &m5 ;**
  **s11 = &s11 ;**
  **s22 = &s22 ;**
  **s33 = &s33 ;**
  **s44 = &s44 ;**
  **s55 = &s55 ;**
  **se = &se ;**
  **r12 = &r12 ;**
  **r13 = &r13 ;**
  **r23 = &r23 ;**
  **s12 = r12 * sqrt(s11) * sqrt(s22) ;**
  **s13 = r13 * sqrt(s11) * sqrt(s33) ;**
  **s23 = r23 * sqrt(s22) * sqrt(s33) ;**
  **s21 = s12 ;**
  **s32 = s23 ;**
  **s31 = s13 ;**
  **seed = 10 ;**
  **do se = 4, 100 ;**
  **do n = 100, 500 ;**
  **do rep = 1 to &nrep ;**
```plaintext
do i = 1 to n;
x3 = m3 + sqrt(s33) * rannor(seed);
m = m2 + (s23/s33)*(x3 - m3);
s = s22*(1-r23**2);
x2 = m + sqrt(s) * rannor(seed);
c1 = 1 / (s22*s33 - s23**2);
a = c1 * ( (x2 - m2) * ( s12 * s33 - s13 * s23 ) +
      (x3 - m3) * ( s13 * s22 - s12 * s23 ) )
m = m1 + a;
s = s11 - c1 * (s12**2 * s33 - 2 * s12 * s13 * s23 +
      s13**2 * s22 )
x1 = m + sqrt(s) * rannor(seed);
x4 = m4 + sqrt(s44) * rannor(seed);
x5 = m5 + sqrt(s55) * rannor(seed);
el = 0;
al = 0;
do ii = 1 to 10;
e = sqrt(se) * rannor(seed);
a = &phi0 + &phi1 * al + &theta1 * el + e;
al = a;
el = e;
end;
error = a;
output;
end;
end;

---
data temp ; set temp ;
y = &beta0 + &beta1 * x1 + &beta2 * x2 + &beta3 * x3 + error ;
run ;
proc sort data=temp ; by se rep n ; run ;
**** REGRESSION X1 X2 X3 and Cp ;
proc reg data=temp outest = pl aic noprint ;
   by se rep n ;
   model y = x1 x2 x3 /selection = cp;
run ;
proc sort data=pl ; by se rep n _AIC_ ; run ;
data rahim ; set pl ; by se rep n ; if first.n ;
   if x1 ne . then x1in = 'yes' ; else x1in = 'no' ;
   if x2 ne . then x2in = 'yes' ; else x2in = 'no' ;
   if x3 ne . then x3in = 'yes' ; else x3in = 'no' ;
   if x1 = . or x2 = . or x3 = . then fit = 'underfitted' ;
   else if x1 ne . and x2 ne . and x3 ne . then fit = 'correct' ;
   /*
   if _IN_ = 3 and x1 ne . and x2 ne . and x3 ne . then fit='correct' ;
   else fit = 'incorrect' ;
   */
run ;
proc sort data=rahim ; by se n ; run ;
proc freq data=rahim noprint ; by se n ;
   tables x1in*x2in*x3in*fit /out=out1 ;
run ;
```
title1"AIC statistics &model model";
title2' X1 X2 and X3 candidates';
proc print data=out1; run;

***** X1 X2 X3 X4 Cp;
proc reg data=temp outest = p2 aic noprnt;
   by se rep n;
   model y = x1 x2 x3 x4 / selection=cp;
run;
proc sort data=p2; by se rep n; if first.n;
   if x1 ne . then x1in = 'yes'; else x1in = 'no '
   if x2 ne . then x2in = 'yes'; else x2in = 'no '
   if x3 ne . then x3in = 'yes'; else x3in = 'no '
   if x4 ne . then x4in = 'yes'; else x4in = 'no '
   if _IN_ lt 3 then fit = 'underfitted '
   if _IN_ gt 3 then fit = 'overfitted '
   if _IN_ = 3 and x1 ne . and x2 ne . and x3 ne . then fit = 'correct '
   if _IN_ = 3 and x4 ne . then fit='incorrect '
*/
   if _IN_ = 3 and x1 ne . and x2 ne . and x3 ne . then fit='correct '
   else fit = 'incorrect';
*/
run;

proc sort data=rahim1; by se n; run;
proc freq data=rahim1 noprnt; by se n;
   tables x1in*x2in*x3in*x4in*fit /out=out2;
run;

title1"AIC statistics &model model";
title2' X1 X2 X3 and X4 candidates';
proc print data=out2; run;

********** X1 X2 X3 X4 X5 Cp;
proc reg data=temp outest = p3 aic noprnt;
   by se rep n;
   model y = x1 x2 x3 x4 x5 /selection=cp;
run;
proc sort data=p3; by se rep n; if first.n;
   if x1 ne . then x1in = 'yes'; else x1in = 'no '
   if x2 ne . then x2in = 'yes'; else x2in = 'no '
   if x3 ne . then x3in = 'yes'; else x3in = 'no '
   if x4 ne . then x4in = 'yes'; else x4in = 'no '
   if x5 ne . then x5in = 'yes'; else x5in = 'no '
   if _IN_ lt 3 then fit = 'underfitted '
   if _IN_ gt 3 then fit = 'overfitted '
   if _IN_ = 3 and x1 ne . and x2 ne . and x3 ne . then fit = 'correct '
   if _IN_ = 3 then do;
     if x4 ne . or x5 ne . then fit = 'incorrect '
   end;
/*
   if _IN_ = 3 and x1 ne . and x2 ne . and x3 ne . then fit='correct '
   else fit = 'incorrect';
*/
/*
 run;

 proc sort data=rahim2; by se n; run;
 proc freq data=rahim2 noprint; by se n;
   tables x1in*x2in*x3in*x4in*fit /out=out3 ;
 run;

title1"AIC statistics &model model";
title2' X1 X2 X3 X4 and X5 candidates';
 proc print data=out3 ; run ;
%mend;

%fitaic(phi0=0, phi1=0, theta1=0, r12=0, r13=0, r23=0, model=orthogonal);
%fitaic(phi0=1, phi1=-0.9, theta1=0, r12=0, r13=0, r23=0, model=orthogonal with AR error);
%fitaic(phi0=0, phi1=0, theta1=-0.9, r12=0, r13=0, r23=0, model=orthogonal with MA error);
%fitaic(phi0=1, phi1=-0.9, theta1=0.9, r12=0, r13=0, r23=0, model=orthogonal with ARMA error);
%fitaic(phi0=0, phi1=0, theta1=0, r12=0.8, r13=0.9, r23=0.8, model=multicollinear);
%fitaic(phi0=1, phi1=-0.9, theta1=0.9, r12=0.8, r13=0.9, r23=0.8, model=multicollinear with AR error);
%fitaic(phi0=0, phi1=0, theta1=-0.9, r12=0.8, r13=0.9, r23=0.8, model=multicollinear with MA error);
%fitaic(phi0=1, phi1=-0.9, theta1=0.9, r12=0.8, r13=0.9, r23=0.8, model=multicollinear with ARMA error);

*************** BIC ***************
%mmacro fitbic(phi0, phi1, theta1, r12, r13, r23, model);
data temp ; run;
data temp;
  m1 = &m1 ;
  m2 = &m2 ;
  m3 = &m3 ;
  m4 = &m4 ;
  m5 = &m5 ;
  s11 = &s11 ;
  s22 = &s22 ;
  s33 = &s33 ;
  s44 = &s44 ;
  s55 = &s55 ;
  se = &se ;
  r12 = &r12 ;
  r13 = &r13 ;
  r23 = &r23 ;
  s12 = r12 * sqrt(s11) * sqrt(s22) ;
  s13 = r13 * sqrt(s11) * sqrt(s33) ;
  s23 = r23 * sqrt(s22) * sqrt(s33) ;
  s21 = s12 ;
  s32 = s23 ;
  s11 = s13 ;
  seed = 10 ;
do se = 4, 100;
do n = 100, 500 ;
do rep = 1 to &nrep;
  do i = 1 to n;
    x3 = m3 + sqrt(s33) * rannor(seed);
    m = m2 + (s23/s33)*(x3 - m3);
    s = s22*(1-r23**2);
    x2 = m + sqrt(s) * rannor(seed);
    c1 = 1 / (s22*s33 - s23**2);
    a = c1 * ( (x2 - m2) * ( s12 * s33 - s13 * s23 ) +
               (x3 - m3) * ( s13 * s22 - s12 * s23 ) ) ;
    m = ml + a ;
    s = sll - cl * ( s12**2 * s33 - 2 * s12 * s13 * s23 +
                     s13**2 * s22 ) ;
    x1 = m + sqrt(s) * rannor(seed);
    x4 = m4 + sqrt(s44) * rannor(seed);
    x5 = m5 + sqrt(s55) * rannor(seed);
    el = 0 ;
    al = 0 ;
    do ii = 1 to 10 ;
      e = sqrt(se) * rannor(seed);
      a = &phi0 + &phil * al + &thetal * el + e ;
      al = a ;
      el = e ;
      end ;
    end ;
    output ;
  end ;
  keep rep n x1 x2 x3 x4 x5 error r12 r13 r23 rsquare se ;
run ;

data temp ; set temp ;
  y = &beta0 + &beta1 * x1 + &beta2 * x2 + &beta3 * x3 + error ;
run ;
proc sort data=temp ; by se rep ; run ;
**** REGRESSION X1 X2 X3 and Cp ;
proc reg data=temp outest = pl bic noprint ;
  by se rep n ;
  model y = x1 x2 x3 /selection = cp ;
run ;
proc sort data=pl ; by se rep n _BIC_ ; run ;
data rahim ; set pl ; by se rep n ; if first.n ;
  if x1 ne . then x1in = 'yes' ; else x1in = 'no ' ;
  if x2 ne . then x2in = 'yes' ; else x2in = 'no ' ;
  if x3 ne . then x3in = 'yes' ; else x3in = 'no ' ;
  if x1 = . or x2 = . or x3 = . then fit = 'underfitted' ;
  else if x1 ne . and x2 ne . and x3 ne . then fit = 'correct ' ;
  /*
    if _IN_ = 3 and x1 ne . and x2 ne . and x3 ne . then fit='correct ' ;
    else fit = 'incorrect';
  */
run ;

proc sort data=rahim ; by se n ; run ;
proc freq data=rahim noprint; by se n ;
tables x1in*x2in*x3in*fit /out=out1 ;
run;

title1 "BIC statistics & model model";
title2 'X1 X2 and X3 candidates';
proc print data=out1; run;

****** X1 X2 X3 X4 Cp;
proc reg data=temp outest = p2 bic noprint;
   by se rep n;
   model y = x1 x2 x3 x4 / selection=cp;
run;
proc sort data=p2 ; by se rep n _BIC_ ; run;
data rahim1 ; set p2 ; by se rep n ; if first.n ;
   if x1 ne . then x1in = 'yes' ; else x1in = 'no ' ;
   if x2 ne . then x2in = 'yes' ; else x2in = 'no ' ;
   if x3 ne . then x3in = 'yes' ; else x3in = 'no ' ;
   if x4 ne . then x4in = 'yes' ; else x4in = 'no ' ;
   if _IN_ lt 3 then fit = 'underfitted ' ;
   if _IN_ gt 3 then fit = 'overfitted ' ;
   if _IN_ = 3 and x1 ne . and x2 ne . and x3 ne . then fit = 'correct ' ;
   if _IN_ = 3 and x4 ne . then fit='incorrect ' ;
/*
   if _IN_ = 3 and x1 ne . and x2 ne . and x3 ne . then fit='correct ' ;
   else fit = 'incorrect';
*/
run;
proc sort data=rahim1 ; by se n ; run ;
proc freq data=rahim1 noprint; by se n ;
tables x1in*x2in*x3in*x4in*fit /out=out2 ;
run;

title1 "BIC statistics & model model";
title2 'X1 X2 X3 and X4 candidates';
proc print data=out2; run;

********** X1 X2 X3 X4 X5 Cp ;
proc reg data=temp outest = p3 bic noprint ;
   by se rep n ;
   model y = x1 x2 x3 x4 x5 /selection=cp;
run;
proc sort data=p3 ; by se rep n _BIC_ ; run ;
data rahim2 ; set p3 ; by se rep n ; if first.n ;
   if x1 ne . then x1in = 'yes' ; else x1in = 'no ' ;
   if x2 ne . then x2in = 'yes' ; else x2in = 'no ' ;
   if x3 ne . then x3in = 'yes' ; else x3in = 'no ' ;
   if x4 ne . then x4in = 'yes' ; else x4in = 'no ' ;
   if x5 ne . then x5in = 'yes' ; else x5in = 'no ' ;
   if _IN_ lt 3 then fit = 'underfitted ' ;
   if _IN_ gt 3 then fit = 'overfitted ' ;
   if _IN_ = 3 and x1 ne . and x2 ne . and x3 ne . then fit = 'correct ' ;
   if _IN_ = 3 then do ;
      if x4 ne . or x5 ne . then fit = 'incorrect ' ;
      end ;
/*
   if _IN_ = 3 and x1 ne . and x2 ne . and x3 ne . then fit='correct ' ;
*/
else fit = 'incorrect';
*/
run ;

proc sort data=rahim2 ; by se n ; run ;
proc freq data=rahim2 noprint; by se n ;
   tables x1in*x2in*x3in*x4in*x5in*fit /out=out3 ;
run ;

title1"BIC statistics &model model";
title2' X1 X2 X3 X4 and X5 candddidates';
proc print data=out3 ; run ;
%mend;

%fitbic(phi0=0, phi1=0,   theta1=0,     r12=0,     r13=0,     r23=0,     model=orthogonal) ;
%fitbic(phi0=1, phi1=-0.9, theta1=0,     r12=0,     r13=0,     r23=0,     model=orthogonal with AR error) ;
%fitbic(phi0=0, phi1=0,   theta1=-0.9,  r12=0,     r13=0,     r23=0,     model=orthogonal with MA error) ;
%fitbic(phi0=1, phi1=-0.9, theta1=0.9,  r12=0,     r13=0,     r23=0,     model=orthogonal with ARMA error) ;
%fitbic(phi0=0, phi1=0,   theta1=0,     r12=0.8,   r13=0.9,   r23=0.8,  model=multicollinear) ;
%fitbic(phi0=1, phi1=-0.9, theta1=0,     r12=0.8,   r13=0.9,   r23=0.8,  model=mul-collin with AR error) ;
%fitbic(phi0=0, phi1=0,   theta1=-0.9,  r12=0.8,   r13=0.9,   r23=0.8,  model=mul-collin with MA error) ;
%fitbic(phi0=1, phi1=-0.9, theta1=0.9,  r12=0.8,   r13=0.9,   r23=0.8,  model=mul-collin with ARMA error) ;
This SAS program generates \( n+m \) observations, fits a regression model to the first \( n \) observations and then uses the fitted model to predict the last \( m \) observations. Predicted values are classified as interpolation or extrapolation based on the definition given in chapter 4. This program handles simulations for chapter 4.

\[
\begin{align*}
\text{%let nrep = 200 ;} \\
\text{%let ext = 0 ;} \\
\text{%let m1 = 20 ;} \\
\text{%let m2 = 30 ;} \\
\text{%let m3 = 40 ;} \\
\text{%let s11 = 25 ;} \\
\text{%let s22 = 16 ;} \\
\text{%let s33 = 36 ;} \\
\text{%let se = 25 ;} \\
\text{%let r12 = -0.99999 ;} \\
\text{%let r13 = 0.95 ;} \\
\text{%let r23 = -0.95 ;} \\
\text{%let mp1 = 40 ;} \\
\text{%let mp2 = 60 ;} \\
\text{%let mp3 = 80 ;} \\
\text{data temp ; run ;} \\
\text{data temp ;} \\
\text{m1 = &m1 ;} \\
\text{m2 = &m2 ;} \\
\text{m3 = &m3 ;} \\
\text{s11 = &s11 ;} \\
\text{s22 = &s22 ;} \\
\text{s33 = &s33 ;} \\
\text{se = &se ;} \\
\text{r12 = &r12 ;} \\
\text{r13 = &r13 ;} \\
\text{r23 = &r23 ;} \\
\text{s12 = r12 * sqrt(s11) * sqrt(s22) ;} \\
\text{s13 = r13 * sqrt(s11) * sqrt(s33) ;} \\
\text{s23 = r23 * sqrt(s22) * sqrt(s33) ;} \\
\text{s21 = s12 ;} \\
\text{s32 = s23 ;} \\
\text{s31 = s13 ;} \\
\text{seed = 10 ;} \\
\text{mp1 = &mp1 ;} \\
\text{mp2 = &mp2 ;} \\
\text{mp3 = &mp3 ;} \\
\text{do n = 50, 100, 500 ;} \\
\text{do rep = 1 to &nrep ;} \\
\text{do i = 1 to n ;}
\end{align*}
\]
x3 = m3 + sqrt(s33) * rannor(seed) ;
m = m2 + (s23/s33)*(x3 - m3);
s = s22*(1-r23**2);
x2 = m + sqrt(s) * rannor(seed);
c1 = 1 / (s22*s33 - s23**2);
a = c1 *((x2 - m2) * (s12 * s33 - s13 * s23) +
        (x3 - m3) * (s13 * s22 - s12 * s23)) ;
m = m1 + a ;
s = s11 - c1 *(s12**2 * s33 - 2 * s12 * s13 * s23 +
        s13**2 * s22) ;
x1 = m + sqrt(s) * rannor(seed);
y = 25 + 3 * x1 - 5 * x2 + 7 * x3 + sqrt(se) * rannor(seed);
y = round(y,1);
y1 = y ;
output ;
end ;
do i = 1 to 500;
x3 = mp3 + sqrt(s33 + ext) * rannor(seed) ;
m = mp2 + (s23/s33)*(x3 - mp3);
s = s22*(1-r23**2);
x2 = m + sqrt(s) * rannor(seed);
c1 = 1 / (s22*s33 - s23**2);
a = c1 *((x2 - mp2) * (s12 * s33 - s13 * s23) +
        (x3 - mp3) * (s13 * s22 - s12 * s23)) ;
m = mp1 + a ;
s = s11 - c1 *(s12**2 * s33 - 2 * s12 * s13 * s23 +
        s13**2 * s22) ;
x1 = m + sqrt(s) * rannor(seed);
y = . ;
y1 = 25 + 3 * x1 - 5 * x2 + 7 * x3 + sqrt(se) * rannor(seed);
y1 = round(y1,1);
output ;
end ;
end ;
keep rep n x1 x2 x3 y y1 r12 r13 r23 mp1 mp2 mp3;
run ;

proc reg data=temp noprint ;
   by n rep ;
   id y1 x1 x2 x3;
   model y = x1 x2 x3 / collinoint;
   output out=out h=h p = p;
run ;

proc means data=out noprint; where y ne . ;
   by n rep ;
   var h;
   output out=out1 max=max;
run ;
data t ; merge out out1 ;
   by n rep ;
   if y ne . then delete ;
drop _FREQ_ _TYPE_ ;
   if h gt max then predic='extrapolation';
   else if h le max then predic = 'interpolation';
   error = (p-(25 + 3 * x1 - 5 * x2 + 7 * x3)) **2 ;
keep rep error predic n ;
run ;

proc sort data=t ; by n rep predic ; run ;
proc means data=t noprint ; by n rep predic ;
  var rror ;
  output out=out mean=mean ;
run ;

proc sort data=out ; by n predic ; run ;
proc means data=out ; by n predic ;
  var mean ;
run ;

/* calculating the eign values */

filename routed 'd:out1';
proc printto print= routed new ; run ;
proc reg data=temp simple noprint ;
  by n rep ;
  id y1 ;
  model y = x1 x2 x3 / collinoint ;
run ;
proc printto ; run ;

/*

data out ;
infile routed ;
  retain rep 0 ;
  input word1 $17. word2 $ word3 $ @ ;
  if word2 = 'Number' or word2 = '1' or word2='2' or word2='3' or
     substr(word3,1,2)=N=1 ;
  if substr(word1,1,1) = 'X' then delete ;
else if word2 = 'Number' then rep = rep + 1 ;
  if word2 = '1' then max = word3 ;
  if word2 = '3' then min = word3 ;
  if word2 = '2' or word2 = 'Number' then delete ;
  if substr(word3,1,2)=N=1 then n = substr(word3,3,3) ;
  drop word1 ;
run ;
proc print ; run ;
*/

data out ;
infile routed ;
  retain rep 0 ;
  input word1 $17. word2 $ word3 $ @ ;
  if word2 = 'Number' or word2 = '1' or word2='2' or word2='3' or
     substr(word3,1,2)=N=1 ;
  if substr(word1,1,1) = 'X' then delete ;
else if word2 = 'Number' then rep = rep + 1 ;
  if word2 = '1' then max = word3 ;
  if word2 = '3' then min = word3 ;
if word2 = '2' or word2 = 'Number' then delete;
if substr(word3,1,2)='N=' then n = substr(word3,3,3);
drop word1;
run;
data o1; set out;
  if word2 ne '1' then delete;
  keep rep max;
run;
data o2; set out;
  if word2 ne '3' then delete;
  keep rep min;
run;
data o3; set out;
  if substr(word3,1,2)='N=';
  rep1 = rep*1 + 1;
  n1 = n*1;
  keep rep1 n1;
run;
data o3; set o3;
  n = n1;
  rep = rep1;
  keep n rep;
run;
data out1; merge o1 o2 o3; by rep;
  min_eign = min * 1;
  max_eign = max * 1;
  ratio = max_eign / min_eign;
  keep min_eign max_eign ratio n;
run;
proc means data = out1; by n;
  var min_eign max_eign ratio;
run;
filename routed 'd:out1';
proc printto print= routed new;
run;
proc reg data=temp simple;
  by n rep;
  id y1;
  model y = x1 x2 x3 / collinoint;
run;
proc printto;
run;
/*
data out;
  infile routed;
  retain rep 0;
  input word1 $17. word2 $ word3 $ @;
  if word2 = 'Number' or word2 = '1' or word2='2' or word2='3' or
  substr(word3,1,2)='N=';
  if substr(word1,1,1) = 'X' then delete;
  else if word2 = 'Number' then rep = rep + 1;
if word2 = '1' then max = word3;
if word2 = '3' then min = word3;
if word2 = '2' or word2 = 'Number' then delete;
if substr(word3,1,2)='N=' then n = substr(word3,3,3);
drop word1;
run;
proc print; run;
*/
data out;
  infile routed;
  retain rep 0;
  input word1 $17. word2 $ word3 $ @;
  if word2 = 'Number' or word2 = '1' or word2='2' or word2='3' or
  substr(word3,1,2)="N=";
    if substr(word1,1,1) = 'X' then delete;
else if word2 = 'Number' then rep = rep + 1;
  if word2 = '1' then max = word3;
  if word2 = '3' then min = word3;
else if word2 = 'Number' then delete;
  if substr(word3,1,2)="N=" then n = substr(word3,3,3);
drop word1;
run;
data 01; set out;
  if word2 ne '1' then delete;
    keep rep max;
run;
data 02; set out;
  if word2 ne '3' then delete;
    keep rep min;
run;
data 03; set out;
  if substr(word3,1,2)="N=";
    repl = rep*1 + 1;
    n1 = n*1;
    keep repl n1;
run;
data 03; set 03;
  n = n1;
  rep = repl;
  keep n rep;
run;
data out1; merge 01 02 03; by rep;
  min_eign = min * 1;
  max_eign = max * 1;
  ratio = max_eign / min_eign;
  keep min_eign max_eign ratio n;
run;
proc means data = out1; by n;
  var min_eign max_eign ratio;
SAS codes for Table 4.2

%let nrep = 200;
%let adm = 1.250; factor for changing mean
%let adds = 1.250; factor for changing variance
%let m1 = 20; mean X1
%let m2 = 30; mean X2
%let m3 = 40; mean X3
%let s11 = 25; variance X1
%let s22 = 16; variance X2
%let s33 = 36; variance X3
%let r12 = -0.99999; 
%let r13 = 0.95;
%let r23 = -0.95;
data temp; run;
data temp;
ml = &ml;
m2 = &m2;
m3 = &m3;
s11 = &s11;
s22 = &s22;
s33 = &s33;
r12 = &r12;
r13 = &r13;
r23 = &r23;
s12 = r12 * sqrt(s11) * sqrt(s22);
s13 = r13 * sqrt(s11) * sqrt(s33);
s23 = r23 * sqrt(s22) * sqrt(s33);
s11 = 36;
s12 = s12;
s13 = s13;
seed = 10;
addm = &addm;
adds = &adds;
mp1 = &m1*addm;
mp2 = &m2*addm;
mp3 = &m3*addm;
do n = 50, 100, 500;
do se = 4, 100;
do rep = 1 to &nrep;
do i = 1 to n;
  x3 = m3 + sqrt(s33) * rannor(seed);
m = m2 + (s23/s33)*(x3 - m3);
s = s22*(1-r23**2);
x2 = m + sqrt(s) * rannor(seed);
c1 = 1 / (s22*s33 - s23**2);
a = c1 * ( x2 - m2 ) * ( s12 * s33 - s13 * s23 ) +
   ( x3 - m3 ) * ( s13 * s22 - s12 * s23 ) ;
m = m1 + a;
s = s11 - c1 * (s12**2 * s33 - 2 * s12 * s13 * s23 +
                   s13**2 * s22 ) ;
x1 = m + sqrt(s) * rannor(seed);
y = 25 + 3 * x1 - 5 * x2 + 7 * x3 + sqrt(se) * rannor(seed);
y = round(y,1);
yl = y;
output;
end;
spl1 = s11*adds;
spl2 = r12*sqrt(spl1) * sqrt(spl2);
spl3 = r13*sqrt(spl1) * sqrt(spl3);
spl3 = r23*sqrt(spl2) * sqrt(spl3);
sp12 = sp12;
sp23 = sp23;

sp13 = sp13;
do i = 1 to 500;
x3 = mp3 + sqrt(spl3) * rannor(seed);
m = mp2 + (sp23/sp33)*(x3 - mp3);
s = sp22*(1-r23**2);
x2 = m + sqrt(s) * rannor(seed);
c1 = 1 / (sp22*sp33 - sp23**2);
a = c1* ( (x2 - mp2 ) * ( sp12 * sp33 - sp13 * sp23 ) +
       (x3 - mp3 ) ) * ( sp13 * sp22 - sp12 * sp23 ) ) ;
m = mpl + a;
s = spl1 - c1 * (sp12**2 * sp33 - 2 * sp12 * sp13 * sp23 +
                   sp13**2 * sp22 );
x1 = m + sqrt(s) * rannor(seed);
y = .;
yl = 25 + 3 * x1 -5 * x2 + 7 * x3 + sqrt(se) * rannor(seed);
yl = round(y1,1);
output;
end;
end;
end;
keep rep n x1 x2 x3 y yl r12 r13 r23 mpl mp2 mp3 se;
run;
proc sort data=temp; by se n rep; run;
proc reg data=temp noprint;
   by se n rep;
id y1 x1 x2 x3;
   model y = x1 x2 x3 / collinoint;
   output out=out h=h p=p;
run;

proc means data=out noprint; where y ne . ;
   by se n rep;
   var h;
   output out=out1 max=max;
run;
data t ; merge out out1 ;
   by se n rep;
   if y ne . then delete ;
drop _FREQ_ _TYPE_ ;
true = 25 + 3 * x1 -5 * x2 + 7 * x3 ;
error = (p-true)**2 ;
keep rep error n se ;
run;

proc sort data=t; by se n rep ; run ;
proc means data=t noprint; by se n rep;
  var error;
  output out=out mean=mean;
run;

proc sort data=out; by se n; run;
proc means data=out noprint; by se n;
  var mean;
  output out=o mean=mse;
run;
title* r12=&r12  r13=&r13  r23=&r23  addm=&addm  adds=&adds;
proc print data=o ; run ;
References


