Multivariate Generalization of Reduced Major Axis Regression

by

Jingjin Li

A Dissertation Presented in Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

Approved May 2012 by the
Graduate Supervisory Committee:

Dennis Young, Co-Chair
Randall Eubank, Co-Chair
Mark Reiser
Ming-Hung Kao
Yan Yang

ARIZONA STATE UNIVERSITY

August 2012

ABSTRACT

A least total area of triangle method was proposed by Teissier (1948) for fitting a straight line to data from a pair of variables without treating either variable as the dependent variable while allowing each of the variables to have measurement errors. This method is commonly called Reduced Major Axis (RMA) regression and is often used instead of Ordinary Least Squares (OLS) regression. Results for confidence intervals, hypothesis testing and asymptotic distributions of coefficient estimates in the bivariate case are reviewed. A generalization of RMA to more than two variables for fitting a plane to data is obtained by minimizing the sum of a function of the volumes obtained by drawing, from each data point, lines parallel to each coordinate axis to the fitted plane (Draper and Yang 1997; Goodman and Tofallis 2003). Generalized RMA results for the multivariate case obtained by Draper and Yang (1997) are reviewed and some investigations of multivariate RMA are given. A linear model is proposed that does not specify a dependent variable and allows for errors in the measurement of each variable. Coefficients in the model are estimated by minimization of the function of the volumes previously mentioned. Methods for obtaining coefficient estimates are discussed and simulations are used to investigate the distribution of coefficient estimates. The effects of sample size, sampling error and correlation among variables on the estimates are studied. Bootstrap methods are used to obtain confidence intervals for model coefficients. Residual analysis is considered for assessing model assumptions. Outlier and influential case diagnostics are developed and a forward selection method is proposed for subset selection of

model variables. A real data example is provided that uses the methods developed. Topics for further research are discussed.

TABLE OF CONTENTS

CHAPTER Page

LIST OF TABLES

Table Page

Table Page

Table Page

Table Page

LIST OF FIGURES

Figure Page

Figure Page

Figure Page

Figure Page

Figure Page

Figure Page

Figure Page

Figure Page

Chapter 1: INTRODUCTION

In fitting a straight line to two variables, X and Y, the ordinary least squares (OLS) method is one of the most commonly used. The OLS method requires selecting one of the variables to be the dependent and the other to be the independent variable, and only the dependent variable has additive random errors associated with it. Therefore, regressing Y on X and X on Y with OLS gives two different fitted lines. When error is present in both variables and/or when neither variable can be identified as the dependent variable, OLS is not appropriate to use.

A least total area of triangle method (figure 1.1) was proposed for fitting a straight line (Teissier 1948) (Barker, Soh and Evans 1988) without treating any one of these variables as special and each of the variables is allowed to have errors. The slope of this fitted line has been shown to be the geometric mean of the two slopes given by ordinary least squares regression and the line is called the geometric mean functional relationship (GMFR) (Teissier 1948; Barker, Soh and Evans 1988).

The fitted line obtained by minimizing the total area of the triangles which is equivalent to minimizing the sum of the geometric means of the squared deviations from the fitted line in each dimension in the two variable case (Tofallis 2002) has been studied in many disciplines through the twentieth century under different names. The method or the line is called Standard or Reduced Major Axis (RMA) in statistics, Organic Correlation (Kermack and Haldane 1950) in biology, the method of minimized areas or diagonal regression (Tofallis 2002) in

economics, and Stromberg's Impartial Line (Feigelson 1992) in astronomy. It is also classified as a type of standard weighting model (Ward, MacDonald, Thompson and Beninger, 1993) in marine research. We will refer to the method as RMA regression.

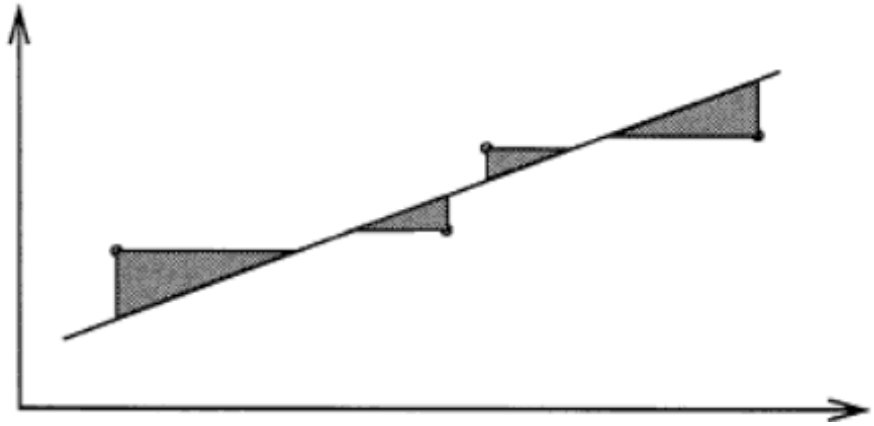

Figure 1.1 Least Area of Triangle Approach

Note: Fitting a line by minimizing the total area of the triangles defined by the data points and the line (figure from (Tofallis 2002))

If symmetry in the variables is assumed where there are no clear dependent and independent variables, one of the most commonly recommended alternatives to OLS is RMA and it is now so well-known that often it is employed with no mention of why it was selected (Smith 2009). If the axes are inverted for two RMA regressions, the slopes are exact reciprocals of each other and therefore maintain a single position with respect to the data. Thus, there is only one RMA regression line.

In the asymmetrical case where there are dependent and independent variables, RMA has been found to be more robust than the Major Axis (MA) method

(McArdle 1988) which is equivalent to Principal Components Regression. The RMA fitted line has a number of nice properties that are described in Chapter 2. Similarly as in the two variable case, a RMA fitted plane could be obtained for more than two variables by minimizing the total volumes (or hypervolumes) formed by drawing, from each data point, lines parallel to each coordinate axis to the fitted plane (Draper and Yang 1997).

The purpose of this dissertation is to investigate the RMA method in the case where there are more than two variables of interest. Specifically we will attempt to develop methods and tools that parallel those used in OLS regression. These issues will be considered: finding coefficient estimates, the distribution of coefficient estimates, inference for coefficients, model diagnostics (residual analysis, outlier detection, influential point detection) and variable selection. This dissertation will proceed as follows.

A brief literature review of research for RMA will be given in Chapter 2. The results for confidence intervals, hypothesis testing, asymptotic results with two variables, criteria of obtaining the RMA coefficient estimates in the multivariate case, and some applications will be reviewed.

In Chapter 3 some results from Draper and Yang (1997) for multivariate RMA will be revisited. A minimum objective function approach will be proposed for computing the coefficient estimates and some simulation results will be shown. Then the distribution of the RMA coefficient estimates is considered in Chapter 4. The effect of sample size, error terms of each variable, correlations between variables and zero coefficients on the distribution of estimates will be studied and

some research about the effect of transformation of coefficient estimates will be discussed.

Since there are no closed forms for estimating coefficients of the fitted plane, bootstrap methods will be used in Chapter 5 to obtain confidence intervals for the coefficients. The quality of bootstrapped confidence intervals will be evaluated by considering the hit rates and confidence interval width rank analyses.

As in OLS regression, the predicted values, residuals and residual plots obtained by RMA will be used as diagnostic tools to assess the fit of the RMA linear model in Chapter 6

In Chapter 7, residual plots will be used to detect outliers and influential points. A leave-one-out approach will be applied to coefficient estimates, fitted values, residuals and the objective function to investigate the presence of influential points.

A forward subset selection method will be proposed in Chapter 8 for selecting an appropriate model that contains a subset of the most important variables by using a criterion based on the the sum of squared residuals. The forward subset selection method will be applied to simulated data sets and the results will be compared to the corresponding true models.

In Chapter 9 a real data set, the Iris Virginica data, will be analyzed by applying RMA regression. All methods covered in Chapter 3 to Chapter 8 will be used in the analysis.

Finally, conclusions and areas for further study will be given in Chapter 10.

Chapter 2: LITERATURE REVIEW

2.1 RMA in the Bivariate Case

2.1.1 Overview

Suppose we have data (x_i, y_i), $i = 1, 2, ..., n$ for variables X and Y and that both are subject to errors, δ_i and ε_i, respectively, so that

$$\begin{aligned} y_i &= \eta_i + \varepsilon_i \\ x_i &= \xi_i + \delta_i \end{aligned},$$

and the true unobserved parameter values η_i and ξ_i have a linear relationship so that

$$\eta_i = \beta_0 + \beta_1 \xi_i ,$$

where the errors are assumed to be normally distributed as in

$$\begin{pmatrix} \delta_i \\ \varepsilon_i \end{pmatrix} \sim N\left(0, \begin{pmatrix} \sigma_\delta^2 & \sigma_{\delta\varepsilon} \\ \sigma_{\delta\varepsilon} & \sigma_\varepsilon^2 \end{pmatrix}\right).$$

In this case the maximum likelihood estimator (MLE) has been obtained (Draper and Yang 1997) in the following special case. Assume the errors ε and δ are independent ($\sigma_{\delta\varepsilon} = 0$) and assume the ratio of the error variances

$$\lambda = \sigma_\varepsilon^2 / \sigma_\delta^2$$

is known. Then the MLE of β_1 is given by

$$\hat{\beta}_{1,MLE} = \frac{S_{YY} - \lambda S_{XX} + \sqrt{\left(S_{YY} - \lambda S_{XY}\right)^2 + 4\lambda S_{XY}^2}}{2S_{XY}},$$

and the MLE of σ_ε^2 and σ_δ^2 are given by (Cheng and Van Ness 1999, chap. 1)

$$\hat{\sigma}^2_{\varepsilon,MLE} = \frac{\lambda\left(S_{YY} - 2\hat{\beta}_{1,MLE}S_{XY} + \hat{\beta}^2_{1,MLE}S_{XX}\right)^2}{2\left(\lambda + \hat{\beta}^2_{1,MLE}\right)},$$

$$\hat{\sigma}^2_{\delta,MLE} = \hat{\sigma}^2_{\varepsilon,MLE} / \lambda,$$

where S_{XX}, S_{YY}, S_{XY} are the sample variance of X, the sample variance of Y and the sample covariance of X and Y, respectively.

The RMA (or GMFR) estimate of slope is

$$\begin{aligned}\hat{\beta}_{1,RMA} &= sign\left(S_{XY}\right)\sqrt{\hat{\beta}_{Y|X,OLS}\hat{\beta}_{X|Y,OLS}} \\ &= sign\left(S_{XY}\right)\sqrt{\left(S_{XY}/S_{XX}\right)\left(S_{YY}/S_{XY}\right)}, \\ &= sign\left(S_{XY}\right)\sqrt{S_{YY}/S_{XX}}\end{aligned} \tag{2.1}$$

where $\hat{\beta}_{Y|X,OLS}$, $\hat{\beta}_{X|Y,OLS}$ are estimates of the slope from OLS when Y is regressed on X and X is regressed on Y, respectively. The RMA estimate of the intercept is

$$\hat{\beta}_{0,RMA} = \bar{y} - \hat{\beta}_{1,RMA}\bar{x}. \tag{2.2}$$

It can be shown (Sprent and Dolby 1980) that the MLE is related to certain procedures in principal component analysis and in canonical correlation analysis. If $\lambda = 1$, $\hat{\beta}_{1,MLE}$ gives the tangent of the angle between the x-axis and the first principal component direction.

Tofallis (2002) provided the following properties of RMA:

(1) The fitted line is symmetric with respect to the two variables and switching the axes does not affect the triangle areas.

(2) The fitted line is unit invariant (or scale invariant).

(3) The slope of the line is the geometric mean of the slopes of the two OLS regressions.

(4) The fitted line minimizes the sum of the geometric means of the squared deviations in the X and Y directions since minimizing the sum of the triangle areas implies that the objective function involves the product of the deviations in each dimension for each point.

(5) It is the only line for which the proportional increase in each of the mean squared error of estimation of Y, considering the OLS regression line of Y on X, or that of X for the OLS regression line of X on Y, are the same.

(6) It is the unique line that satisfies properties 1 and 2 for the set of all possible line-fitting procedures that depend on standard deviations and correlations (Samuelson 1942).

(7) A 45 degree line that bisects the two OLS regression lines is obtained by performing the RMA on standardized data.

Sprent and Dolby (1980) claimed that fitting a RMA regression line is irrelevant to the calculation of correlation coefficients or coefficients of determination; both of these are measures of the strength of a linear relationship rather than its position. However, Barker, Soh and Evans (1988) argued the connection between the RMA estimator and the data correlation coefficient is:

$$\bar{\Delta} = -|r| + 1 = \Delta L_T / \sqrt{\sum (x_i - \bar{x})^2 \sum (y_i - \bar{y})^2}$$

where $\bar{\Delta}$ is the "normalized" least total area of triangles and

$$\Delta L_T = -\left[sign\left(\sum (x_i - \bar{x})(y_i - \bar{y})\right)\right]\sum (x_i - \bar{x})(y_i - \bar{y}) + \sqrt{\sum (x_i - \bar{x})^2 \sum (y_i - \bar{y})^2}$$

is the least total area of triangles.

2.1.2 Confidence Intervals

Imbrie (1956) showed the estimated standard errors of $\hat{\beta}_{1,RMA}$ and $\hat{\beta}_{0,RMA}$ are

$$se_1 = \sqrt{\left(\frac{S_{YY}}{S_{XX}}\right)\left(\frac{1-r^2}{n}\right)}$$

and

$$se_0 = \sqrt{(S_{YY})\left(\frac{1-r^2}{n}\right)\left(1+\frac{\bar{X}^2}{S_{XX}}\right)},$$

respectively, where r is the correlation between Y and X.

Imbrie (1956) obtained a symmetrical approximate $(1-\alpha)100\%$ confidence interval on the slope by assuming that the underlying distribution of the data and $\hat{\beta}_{1,RMA}$ are normal, namely,

$$\hat{\beta}_{1,RMA} + z_{(\alpha/2)} se_1 < \beta_1 < \hat{\beta}_{1,RMA} + z_{(1-\alpha/2)} se_1$$

where $z_{(k)}$ is the $100k\%$ percentile of the standard normal distribution.

Ricker (1973) used a t-distribution in place of the standard normal distribution and obtained a similar approximate $(1-\alpha)100\%$ confidence interval, namely,

$$\hat{\beta}_{1,RMA} + t_{(\alpha/2)} se_1 \sqrt{\frac{n}{n-2}} < \beta_1 < \hat{\beta}_{1,RMA} + t_{(1-\alpha/2)} se_1 \sqrt{\frac{n}{n-2}},$$

where $t_{(k)}$ is the $100k\%$ percentile of the t-distribution with $(n-2)$ degrees of freedom. Furthermore, asymmetrical $(1-\alpha)100\%$ confidence intervals for the

slope were obtained by Jolicoeu & Mosimann (1968). Assuming an underlying bivariate normal distribution, they proposed the interval

$$\hat{\beta}_{1,RMA}\left(\sqrt{\frac{F_{(1-\alpha)}\left(1-r^2\right)}{n-2}+1}-\sqrt{\frac{F_{(1-\alpha)}\left(1-r^2\right)}{n-2}}\right)<\beta_1 \\ <\hat{\beta}_{1,RMA}\left(\sqrt{\frac{F_{(1-\alpha)}\left(1-r^2\right)}{n-2}+1}+\sqrt{\frac{F_{(1-\alpha)}\left(1-r^2\right)}{n-2}}\right),$$

where $F_{(k)}$ is the $100k\%$ percentile of the F-distribution with 1 and $(n-2)$ degrees of freedom.

Rayner (1985) gives the following $(1-\alpha)100\%$ confidence interval for β_1:

$$\hat{\beta}_{1,RMA}\sqrt{\frac{\sqrt{\frac{F_{(1-\alpha)}\left(1-r^2\right)}{(n-2)r^2}+1}}{\sqrt{\frac{F_{(1-\alpha)}\left(1-r^2\right)}{(n-2)r^2}-1}}}<\beta_1<\hat{\beta}_{1,RMA}\sqrt{\frac{\sqrt{\frac{F_{(1-\alpha)}\left(1-r^2\right)}{(n-2)r^2}-1}}{\sqrt{\frac{F_{(1-\alpha)}\left(1-r^2\right)}{(n-2)r^2}+1}}}.$$

Plotnick (1989) used a bootstrap approach to find confidence intervals of RMA coefficients in the two variable case. 1000 re-samples were generated by randomly selecting observations with replacement from the original data set repeatedly and the means and variances of the bootstrap samples were calculated and used to generate the RMA coefficient estimates (referred to as bootstrap slopes and intercepts in this paper) as described in eq. (2.1) and (2.2) for each bootstrap sample. Then the mean and standard deviation of the bootstrap slopes and intercepts were calculated as well as the estimate of the standard error to obtain a 95% confidence interval based on the 1000 iterations. Both percentile

and bias-correction methods were calculated for an example. The bootstrap methods that were applied in this paper provided results that agree well with those obtained from other available analytical methods and they were superior in capturing the asymmetry in the distribution and thus in its confidence limits.

2.1.3 Hypothesis Testing

Finney (1938) has shown how to use the distribution of $\hat{\beta}_{1,RMA}$ to conduct a hypothesis test that β_1 is a given constant b_1. However his test depends on knowing ρ (the true correlation between X and Y) and is not very sensitive if ρ has to be estimated (Clarke 1980). Kermack and Haldane (1950) showed that by straightforward methods up to the order of n^{-1},

$$\operatorname{var}\left(\sqrt{S_{YY}/S_{XX}}\right)=\left(\sigma_Y^2/\sigma_X^2\right)\left(1-\rho^2\right)n^{-1},$$

which is the same as the variance of $\hat{\beta}_{Y|X,OLS}$ in a bivariate normal population (Teissier 1948), where σ_Y^2, σ_X^2 are the variances of Y and X, respectively. The distribution of $\hat{\beta}_{1,RMA}$ is not symmetric about its mean and its variance depends on the mean. However it is shown (Clarke 1980) that the transformed value $\log\hat{\beta}_{1,RMA}$ is distributed symmetrically about its mean with variance $\left(1-\rho^2\right)n^{-1}$ to $O\left((n-1)^{-1}\right)$ and furthermore $\log\hat{\beta}_{1,RMA}$ is uncorrelated with $\left(1-r^2\right)$. Therefore a reasonable test statistic (Clarke 1980) to consider in the one-sample case is

$$T=\frac{\left|\log\hat{\beta}_{1,RMA}-\log b_1\right|}{\sqrt{\left(1-r^2\right)/n}},$$

which has an asymptotically standard normal distribution.

A test statistic to compare the slopes of lines derived from different populations has been proposed by Clarke (1980). It has the form

$$T_{12} = \frac{\left|\log \hat{\beta}_{1,RMA} - \log \hat{\beta}_{2,RMA}\right|}{\sqrt{\left(1-r_1^2\right)/n_1 + \left(1-r_2^2\right)/n_2}}.$$

and has an asymptotic standard normal distribution.

For testing equality of slopes of two lines based on two independent samples, Imbrie (1956) suggested the test statistic

$$T_{12} = \frac{\left|\hat{\beta}_{1,RMA} - \hat{\beta}_{2,RMA}\right|}{\sqrt{\hat{\beta}_{1,RMA}^2\left(1-r_1^2\right)/n_1 + \hat{\beta}_{2,RMA}^2\left(1-r_2^2\right)/n_2}}$$

which also has a standard normal limiting distribution.

2.1.4 Asymptotic Results

The sample variances and covariance, S_{XX}, S_{YY} and S_{XY}, converge in probability to their expectations (Stuart and Ord 1994, chap. 10). Thus, in this case, the functional relationship case, Cheng and Van Ness (1999, chap. 2) show that

$$S_{XX} \xrightarrow{P} S_{\xi\xi}^* + \sigma_\delta^2,$$

$$S_{YY} \xrightarrow{P} \beta_1^2 S_{\xi\xi}^* + \sigma_\varepsilon^2,$$

$$S_{XY} \xrightarrow{P} \beta_1 S_{\xi\xi}^*,$$

assuming that the following limits exist:

$$S_{\xi\xi}^* = \lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} \left(\xi_i - \bar{\xi}_n\right)^2,$$

$$\xi^* = \lim_{n\to\infty} \bar{\xi}_n ,$$

where

$$\bar{\xi}_n = \frac{1}{n}\sum_{i=1}^{n} \xi_i .$$

The convergence in probability can be made strong convergence under slightly different regularity conditions. When λ is known, the maximum likelihood estimates $\hat{\beta}_{0,MLE}$ and $\hat{\beta}_{1,MLE}$ are consistent for the parameters β_0, β_1, respectively:

$$\hat{\beta}_{1,MLE} \xrightarrow{P} \frac{\beta_1^2 S_{\xi\xi}^* - \lambda S_{\xi\xi}^* + \left[\left\{\beta_1^2 S_{\xi\xi}^* + \lambda\sigma_\delta^2 - \lambda\left(S_{\xi\xi}^* + \sigma_\delta^2\right)\right\}^2 + 4\lambda\beta_1^2\left(S_{\xi\xi}^*\right)^2\right]^{1/2}}{2\beta_1 S_{\xi\xi}^*} = \beta_1,$$

$$\hat{\beta}_{0,MLE} = \bar{Y} - \hat{\beta}_{1,MLE}\bar{X} \xrightarrow{P} \beta_0.$$

However, the MLE of the variance of ε, $\hat{\sigma}_{\varepsilon,MLE}^2$ is inconsistent when λ is known, and

$$\hat{\sigma}_{\varepsilon,MLE}^2 \xrightarrow{P} \frac{\lambda\left(\beta_1^2 S_{\xi\xi}^* + \sigma_\varepsilon^2 - 2\beta_1^2 S_{\xi\xi}^* + \beta_1^2\left(S_{\xi\xi}^* + \sigma_\delta^2\right)\right)}{2\left(\lambda + \beta_1^2\right)} = \frac{\sigma_\varepsilon^2}{2} .$$

Lindley (1947) considered this a problem of degrees of freedom and proposed consistent estimators for σ_ε^2, σ_δ^2,

$$\tilde{\sigma}_\varepsilon^2 = \frac{2n}{n-2}\hat{\sigma}_{\varepsilon,MLE}^2 = \frac{\lambda n\left(S_{YY} - 2\hat{\beta}_{1,MLE}S_{XY} + \hat{\beta}_{1,MLE}^2 S_{XX}\right)}{(n-2)\left(\lambda + \hat{\beta}_{1,MLE}^2\right)},$$

$$\tilde{\sigma}_\delta^2 = \tilde{\sigma}_\varepsilon^2 / \lambda = \frac{n\left(S_{YY} - 2\hat{\beta}_{1,MLE}S_{XY} + \hat{\beta}_{1,MLE}^2 S_{XX}\right)}{(n-2)\left(\lambda + \hat{\beta}_{1,MLE}^2\right)} .$$

This adjusted estimator has been adopted by most researchers (Cheng and Van Ness 1999, chap. 2).

The RMA estimate $\hat{\beta}_{1,RMA}$ is not consistent. Since

$$\hat{\beta}_{1,RMA} = sign\left(S_{XY}\right)\sqrt{\frac{S_{YY}}{S_{XX}}} \xrightarrow{P} sign\left(\beta_1\right)\sqrt{\frac{\beta_1^2 S_{\xi\xi}^* + \sigma_\varepsilon^2}{S_{\xi\xi}^* + \sigma_\delta^2}} = sign\left(\beta_1\right)\sqrt{\frac{\beta_1^2 + \sigma_\varepsilon^2 / S_{\xi\xi}^*}{1 + \sigma_\delta^2 / S_{\xi\xi}^*}}\ .$$

However, $\hat{\beta}_{1,RMA}$ would be approximately consistent under the conditions $\sigma_\varepsilon^2,\ \sigma_\delta^2$ are close to zero or they are small relatively to $S_{\xi\xi}^*$, i.e. $\sigma_\delta^2 << S_{\xi\xi}^*$ and $\sigma_\varepsilon^2 << S_{\xi\xi}^*$.

The OLS estimate $\hat{\beta}_{1,OLS}$ is not consistent either. Since

$$\hat{\beta}_{1,OLS} = \frac{S_{XY}}{S_{XX}} \xrightarrow{P} \frac{\beta_1 S_{\xi\xi}^*}{S_{\xi\xi}^* + \sigma_\delta^2}\ .$$

However, $\hat{\beta}_{1,OLS}$ would be approximately consistent if σ_δ^2 is close to zero or $\sigma_\delta^2 << S_{\xi\xi}^*$.

2.2 RMA in the Multivariate Case

Suppose we have data $\left(x_{i1}, x_{i2}, ..., x_{ip}, y_i\right),\ i = 1, 2, ..., n$ that are subject to errors $\delta_{i1}, \delta_{i2}, ..., \delta_{ip}, \varepsilon_i$, and we intend to fit a linear function of the form $y = \sum_{j=1}^{p} a_j x_j + a_0$ to the data. To generalize the least area of triangle procedure to higher dimensions, Tofallis (2002) considered minimizing the sum of the volumes (in three-dimension) or hypervolumes (in higher dimensions) formed by drawing, from each data point, lines parallel to each coordinate axis to the fitted plane, which are the geometric means of the squared deviations from the fitted plane in

each dimension. It is shown that for the i^{th} data point this volume deviation is proportional to

$$V_i = \left| \frac{\left(a_0 + \sum_{j=1}^{p} a_j x_{ij} - y_i \right)^{p+1}}{\Pi_{j=1}^{p} a_j} \right|$$

2.2.1 Results from Draper and Yang (1997)

Let

$$D_{il}^2 = \frac{\left(a_0 + \sum_{j=1}^{p} a_j x_{ij} - y_i \right)^2}{a_l^2}, l = 1, ..., p, i = 1, ..., n$$

$$D_{iy}^2 = \left(a_0 + \sum_{j=1}^{p} a_j x_{ij} - y_i \right)^2, i = 1, ..., n$$

Note that the geometric mean of $D_{il}^2, ..., D_{ip}^2, D_{iy}^2$ and the volume deviation are related via

$$\left(D_{iy}^2 \Pi_{l=1}^{p} D_{il}^2 \right)^{1/(p+1)} = V_i^{2/(p+1)}$$

Minimizing $\sum_{i=1}^{n} D_{il}^2$ gives the OLS solution for regressing X_l on $X_1, X_2, ..., X_{l-1}, X_{l+1}, ..., X_p$ and Y.

Draper and Yang (1997) obtained the multivariable criteria for estimating the linear coefficients or a's. Consider the case where the signs of the a's are known and it could be assumed, via reversal of the axes when necessary, that all $a_i > 0, i = 1, 2, ..., p$ (Draper and Yang 1997). Let

$$L_G^{p+1} = \sum_{i=1}^{n} V_i^{2/(p+1)} = \sum_{i=1}^{n} \frac{\left(a_0 + \sum_{j=1}^{p} a_j x_{ij} - y_i\right)^2}{\left(\Pi_{j=1}^{p} a_j^2\right)^{1/(p+1)}}$$

be the criterion; then minimizing L_G^{p+1} with respect to $a_1, a_2, ..., a_p$ gives RMA estimates $b_1, b_2, ..., b_p$, subject to $\tau_i > 0,\ i = 0,1,2,..., p$, where

$$\tau_i = \frac{a_i}{\left(a_1 a_2 ... a_p\right)^{1/(p+1)}},\ i = 1,2,..., p \text{ and } \tau_0 = \left(\tau_1 \tau_2 ... \tau_p\right)^{-1}.$$

Other possible objective functions that might be used are $\sum_{i=1}^{n} V_i^{1/(p+1)}$ or $\sum_{i=1}^{n} V_i$.

The L_G^{p+1} could be rewritten as

$$L_G^{p+1} = \left(a_1 ... a_p\right)^{-2/(p+1)} \left(a_1, ..., a_p, 1\right) \mathbf{S} \left(a_1, ..., a_p, 1\right)',$$

where

$$\mathbf{S} = \begin{pmatrix} S_{11} & \cdots & S_{1p} & -S_{1Y} \\ \vdots & \ddots & \vdots & \vdots \\ S_{p1} & \cdots & S_{pp} & -S_{pY} \\ -S_{Y1} & \cdots & -S_{Yp} & -S_{YY} \end{pmatrix} = \begin{pmatrix} \mathbf{S}_1^T \\ \vdots \\ \mathbf{S}_p^T \\ \mathbf{S}_{p+1}^T \end{pmatrix}$$

looks like the sample variance-covariance matrix of $X_1, X_2, ..., X_k$ and $-Y$ with terms

$$S_{sk} = \sum_{i=1}^{n} \left(x_{ik} - \bar{x}_k\right)\left(x_{is} - \bar{x}_s\right) / n,\ k = 1, ..., p,\ s = 1, ..., p$$

and

$$S_{kY} = \sum_{i=1}^{n} \left(x_{ik} - \bar{x}_k\right)\left(y_i - \bar{y}\right) / n,\ k = 1, ..., p.$$

Draper and Yang (1997) showed that if all of the $p+1$ OLS solutions which use in turn each of the $p+1$ variables as a dependent variable lie in the same 2^p-tant, and if the matrix $\mathbf{S}$ is non-singular, there is unique solution that minimizes

$$L_G^{p+1} = \left(a_1 ... a_p\right)^{-2/(p+1)} \left(a_1, ..., a_p, 1\right) \mathbf{S} \left(a_1, ..., a_p, 1\right)^T .$$

It lies within the simplex defined by the $p+1$ OLS solutions.

Draper and Yang (1997) provided simulated results for a 3-variable example. They generated data $\left(x_{i1}, x_{i2}, y_i\right),\ i = 1, 2, ..., 20$ with small/large errors with zero-covariance and equal/unequal error variances and considered minimizing L_G^3 as a nonlinear weighted least-squares problem with weights depending on the a's and solved the problem by using Splus where a sample code was provided.

The simulation showed that:

1. The solutions that minimize L_G^3 are always convex combinations of the three OLS solutions.
2. When the standard deviations of the measurement errors are "small", the point that minimizes L_G^3 is very close to the point whose coordinates are the geometric mean of the b_1's from the three OLS regressions, and the geometric mean of the b_2's from the three OLS regressions.

2.3 Applications of RMA

Ricker (1984) suggested that it is more appropriate to use RMA than OLS in varied fields such as morphology, physiology, life history and animal behavior. With biological data, values are expected to deviate from any line of best fit if for

no other reason than that variation is inherent in the evolutionary processes that underlie the development of traits (Smith 2009). The scatter plot of biological data in the bivariate case often resembles an ellipse rather than a straight line. It is virtually always the case with biological data that some error is present in measurements of both the x-axis and y-axis variables (McArdle 2003). To represent the general pattern of the relationship between X and Y, RMA could be used.

The slope of the line will be used to interpret the pattern of change in "shape" with change in size (Smith 2009) whether X and Y maintain an isometric relationship, or whether Y exhibits positive or negative allometry (Warton, Wright, Falsterand Westoby 2006), which will be discussed in the Allometry section.

The purpose for which the RMA has most often been used is in the study of body proportions (Ricker 1984): e.g., by Teissier (1948), Kermack and Haldane (1950), Kruskal (1953), Imbrie (1956) and Clarke (1980).

2.3.1 Allometry

The classic simple allometric equation that describes an organism's growth relating to linear variables X and Y is given by:

$$Y = bX^{a}$$

or, in logarithms,

$$\log Y = a \log X + \log b\,.$$

When $a = 1$, growth is isometric, with the two parts growing proportionally ($Y / X = b$). Given a series of measurements made on an individual at successive

time intervals, or measurements of a lot of individuals of different ages, a line can be fitted to the logarithms to estimate a and test the significance of its difference from one.

When comparing leg length with body length, there is no logical or biological reason to consider that all the variability between individuals should be assigned to the appendage and none to the body, or vice versa (Richer 1984). Instead, complete mathematical symmetry must be maintained between the two sets of measurements. Therefore, in this case, if we let

$$Y = \log Y,\ X = \log X,\ \beta_1 = a,\ \beta_0 = \log b,$$

the RMA approach could be applied. However a hidden assumption that the residual variances are assumed to be proportional to the total variances needs to be made (Kuhry and Marcus 1977).

In addition, Plotnick (1989) demonstrated the utility of the bootstrap method for RMA for fitting the allometric equation. The results were in good agreement with statistical properties determined by available analytical methods. The bootstrap also has nice properties such as it is superior to many other methods in capturing the detailed distribution of the parameters. It doesn't need to make assumptions about underlying distributions and symmetries of the distribution of estimators, and it can provide inferences when analytical solutions do not exist or have not been obtained. The bootstrap was used to determine the standard deviation and confidence intervals of the allometric equation's parameters and compare allometric curves from different taxa (Plotnick 1989).

2.3.2 Fisheries

RMA fitting has been widely used in fisheries studies (Sprent and Dolby 1980). In fisheries studies for comparing body proportions which are approximately linearly related, it is hard to argue for a cause-effect relationship between body length and body mass (Ricker 1973); neither variable is dependent upon the other and the biological interpretation of the results should be identical regardless of which variable is on each axis (Smith 2009). It has been suggested that RMA fitting is more useful than OLS.

For most fisheries data where there is approximately a straight-line relationship; deviations from an idealized straight line exhibited by the data often reflect mainly genetic or environmental variation in growth (Sprent and Dolby 1980).

2.3.3 Others

It has been suggested that RMA should be used if the prediction involves an extrapolation (Jungers 1988; Aiello 1992; Ruff 1998; Smith 2009) due to its robustness. Ricker (1973) explained that as the range of a data set increases, the Y on X and the X on Y OLS equations converge due to the increasing of the correlation between X and Y (Smith 1980, 1981). Therefore as an intermediate between the two OLS equations, the RMA equation represents the solution on which the two OLS equations would converge as the correlation approaches one. Thus, RMA is preferred in general in cases of extrapolation.

Ricker (1984) mentioned that RMA almost always provides a useful description of the population's central trend in situations where either the population or the sample, or both, are not bivariate normal in distribution. In addition, in such situations the RMA line is usually a much better fit than OLS lines for predicting Y from X or X from Y.

Chapter 3: ESTIMATING PARAMETERS

Draper and Yang (1997) consider the case that the data are $\left(x_{i1}, x_{i2}, ..., x_{ip}, y_i\right)$, $i = 1, 2, ..., n$ that are subject to errors $\delta_{i1}, \delta_{i2}, ..., \delta_{ip}, \varepsilon_i$, and intend to fit a linear function of the form $y_i = \sum_{j=1}^{p} a_j x_{ij} + a_0$ to the data. However, we would like to reformulate the results in terms of variables $X_1, ..., X_p$ without any indication of dependent and independent variables, when it is not appropriate to use OLS. Suppose we have data $\left(x_{i1}, x_{i2}, ..., x_{ip}\right)$, $i = 1, 2, ..., n$, that are subject to errors $\varepsilon_{i1}, \varepsilon_{i2}, ..., \varepsilon_{ip}$, and we intend to fit a linear function of the form $\sum_{j=1}^{p} a_j x_j = c, i = 1, ..., p$ to the data. We considered two cases:

(1) $c \neq 0$

Without loss of generality, let $c = 1$, since $\sum_{j=1}^{p} a_j x_j = c$ divided by c from both sides becomes $\sum_{j=1}^{p} \frac{a_j}{c} x_j = 1$.

(2) $c = 0$

$\sum_{j=1}^{p} a_j x_j = 0$ is not unique, since $\sum_{j=1}^{p} \frac{a_j}{t} x_j = 0, \ \forall t \neq 0$ represents the same plane. To determine a unique solution, set one of the a_j's to be one.

Here the model $\sum_{j=1}^{p} a_j x_j = 1$ is considered.

3.1 Criteria for Multivariate RMA

There were three criteria proposed to be minimized in order to find RMA estimates in the multivariate case (Tofallis 2002) and they are reformulated as the following:

$$L_V = \sum_{i=1}^{n} V_i = \sum_{i=1}^{n} \left| \frac{\left(\sum_{j=1}^{p} a_j x_{ij} - 1 \right)^p}{\prod_{j=1}^{p} a_j} \right|,$$

$$L_{Vp} = \sum_{i=1}^{n} V_i^{1/p} = \sum_{i=1}^{n} \left| \frac{\left(\sum_{j=1}^{p} a_j x_{ij} - 1 \right)}{\left(\prod_{j=1}^{p} a_j \right)^{1/p}} \right|,$$

$$L_G^p = \sum_{i=1}^{n} V_i^{2/p} = \sum_{i=1}^{n} \frac{\left(\sum_{j=1}^{p} a_j x_{ij} - 1 \right)^2}{\left(\prod_{j=1}^{p} a_j^2 \right)^{1/p}}.$$

To minimize L_V is to minimize the sum of volumes or hypervolumes (V_i's) formed by drawing, from each data point, lines parallel to each coordinate axis to the fitted plane. However, the numerator of V_i looks like an L_p norm (Tofallis 2002) with the degree p varying as the number of variables. Therefore, the L_1 norm form, L_{Vp} corresponding to taking the p^{th} root of L_V, is considered. The idea is to minimize the geometric mean of the absolute deviations in each dimension. However, it is not easy to deal with absolute values. Therefore, the problem is then considered as a nonlinear weighted least-squares problem as Draper and Yang (1997) suggested, and the criteria L_G^p is used and rewritten as

$$L_G^p = \sum_{i=1}^{n} \left(V_i^{1/p} \right)^2 = \sum_{i=1}^{n} \left(\frac{\sum_{j=1}^{p} a_j x_{ij} - 1}{\prod_{j=1}^{p} a_j^{1/p}} \right)^2 \qquad (3.1)$$

in order to apply the idea of least squares to L_G^p . We have

$$L_G^p = \frac{\sum_{i=1}^{n}\left(\sum_{j=1}^{p} a_j X_{ij} - 1\right)^2}{\left(\Pi_{j=1}^{p} a_j^2\right)^{1/p}} = \frac{\left(\mathbf{a}'\sum_{i=1}^{n}\mathbf{X}_i\mathbf{X}_i'\mathbf{a} - 2n\mathbf{a}'\bar{\mathbf{X}} + n\right)}{\left(\Pi_{j=1}^{p} a_j\right)^{2/p}},$$

where $\mathbf{a} = \left(a_1, \ldots, a_p\right)'$, $\mathbf{X}_i = \left(X_{i1}, ..., X_{ip}\right)'$ and $\bar{\mathbf{X}} = \frac{1}{n}\sum_{i=1}^{n}\mathbf{X}_i$. Then

$$\begin{aligned}
\frac{\partial L_G^p}{\partial \mathbf{a}} &= \frac{\partial}{\partial \mathbf{a}} \frac{\left(\mathbf{a}'\sum_{i=1}^{n}\mathbf{X}_i\mathbf{X}_i'\mathbf{a} - 2n\mathbf{a}'\bar{\mathbf{X}} + n\right)}{\left(\Pi_{j=1}^{p} a_j\right)^{2/p}} \\
&= \frac{\frac{\partial}{\partial \mathbf{a}}\left(\mathbf{a}'\mathbf{P}\mathbf{a} - 2n\mathbf{a}'\bar{\mathbf{X}} + n\right)\left(\Pi_{j=1}^{p} a_j\right)^{2/p} - \left(\mathbf{a}'\mathbf{P}\mathbf{a} - 2n\mathbf{a}'\bar{\mathbf{X}} + n\right)\frac{\partial}{\partial \mathbf{a}}\left(\Pi_{j=1}^{p} a_j\right)^{2/p}}{\left(\Pi_{j=1}^{p} a_j\right)^{4/p}} \\
&= \frac{\left(2\mathbf{P}\mathbf{a} - 2n\bar{\mathbf{X}}\right)\left(\Pi_{j=1}^{p} a_j\right)^{2/p} - \left(\mathbf{a}'\mathbf{P}\mathbf{a} - 2n\mathbf{a}'\bar{\mathbf{X}} + n\right)\frac{2}{p}\left(\Pi_{j=1}^{p} a_j\right)^{2/p}\left(\frac{1}{a_1}, ..., \frac{1}{a_p}\right)'}{\left(\Pi_{j=1}^{p} a_j\right)^{4/p}}
\end{aligned}$$

where $\mathbf{P} = \sum_{i=1}^{n}\mathbf{X}_i\mathbf{X}_i'$.

Set $\frac{\partial L_G^p}{\partial \mathbf{a}} = 0$, then

$$\left(2P\mathbf{a} - 2n\bar{\mathbf{X}}\right)\left(\Pi_{j=1}^{p} a_j\right)^{2/p} - \frac{2}{p}\left(\Pi_{j=1}^{p} a_j\right)^{2/p}\left(\mathbf{a}'P\mathbf{a} - 2n\mathbf{a}'\bar{\mathbf{X}} + n\right)\left(\frac{1}{a_1}, ..., \frac{1}{a_p}\right)' = 0$$

If $\mathbf{a}$ is a solution to this equation, they by multiplying $\mathbf{a}'$ from the left on both sides of the equation, we have

$$\left(\mathbf{a}'P\mathbf{a} - \mathbf{a}'n\bar{\mathbf{X}}\right) - \frac{1}{p}\left(\mathbf{a}'\left(\frac{1}{a_1}, ..., \frac{1}{a_p}\right)'\right)\left(\mathbf{a}'P\mathbf{a} - 2n\mathbf{a}'\bar{\mathbf{X}} + n\right) = 0 .$$

Therefore,

$$\mathbf{a}'\bar{\mathbf{X}} = 1,$$

written as

$$\frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{p} a_j X_{ij} = 1. \qquad (3.2)$$

3.2 One Variable Case

When there is only one variable, say X, then the fitted line is $aX = c$.

(1) $c \neq 0$, let $c = 1$.

Assuming $a \neq 0$, then the criteria becomes

$$L_G^1 = \sum_{i=1}^{n}\left(\frac{ax_i - 1}{a}\right)^2 = \sum_{i=1}^{n}\left(x_i - \frac{1}{a}\right)^2.$$

Therefore $a = 1/\bar{X}$ minimizes L_G^1.

(2) $c = 0$, then the fitted equation is $X = 0$.

3.3 Multivariate Case

3.3.1 Computing the Coefficient Estimates

(1) Three Variable Exact Solution

When there are three variables considered, the criterion to be minimized is

$$L_G^3 = \sum_{i=1}^{n}\left(\frac{\sum_{j=1}^{3} a_j x_{ij} - 1}{\Pi_{j=1}^{3} a_j^{1/3}}\right)^2.$$

Goodman and Tofallis (2003) found an exact solution for computing the coefficient estimates in the three variable case. Let

$$s_j^2 = \frac{1}{n}\sum_{i=1}^{n}\left(x_{ij} - \bar{x}_j\right)^2 \text{ and } s_{jk} = \frac{1}{n}\sum_{i=1}^{n}\left(x_{ij} - \bar{x}_j\right)\left(x_{ik} - \bar{x}_k\right),\ j,k = 1,2,3,\ j \neq k$$

where $\bar{x}_j = \frac{1}{n}\sum_{i=1}^{n} x_{ij}$, $j = 1,2,3$, and let

$$\lambda = \frac{s_{23}}{s_2 s_3},\ \mu = \frac{s_{13}}{s_1 s_3},\ \nu = \frac{s_{12}}{s_1 s_2} \text{ and } x = a_1 s_1,\ y = a_2 s_2,\ z = a_3 s_3.$$

We have

$$L_G^3 = \sum_{i=1}^{n}\left(\frac{\sum_{j=1}^{3} a_j x_{ij} - 1}{\Pi_{j=1}^{3} a_j^{1/3}}\right)^2 = f\left(a_1, a_2, a_3\right) = n\left(s_1 s_2 s_3\right)^{2/3} g\left(x, y, z\right).$$

Therefore, to minimize the L_G^3 is equivalent to minimizing

$$g\left(x, y, z\right) = \frac{x^2 + y^2 + z^2 + 2\lambda yz + 2\mu xz + 2\nu xy}{\left(xyz\right)^{2/3}}$$

over $x, y, z,\ xyz > 0$ (Goodman and Tofallis 2003).

Hence, the $\left(a_1, a_2, a_3\right)$ calculated by $\left(x / s_1, y / s_2, z / s_3\right)$ are given by the following cases:

Case 1: if $\mu = \nu = 0$ then

$\left(x, y, z\right) = \left(\sqrt{1+\lambda}, 1, 1\right)$ or $\left(\sqrt{1+\lambda}, -1, -1\right)$ for $\lambda < 0$,

$\left(x, y, z\right) = \left(-\sqrt{1-\lambda}, 1, -1\right)$ or $\left(-\sqrt{1-\lambda}, -1, 1\right)$ for $\lambda > 0$,

$\left(x, y, z\right) = \left(1,1,1\right), \left(1,-1,-1\right), \left(-1,1,-1\right)$ or $\left(-1,-1,1\right)$.

Case 2: if $\lambda = \mu = \nu > 0$ then

$\left(x, y, z\right) = \left(1+\lambda, -1, -1\right), \left(-1, 1+\lambda, -1\right)$ or $\left(-1, -1, 1+\lambda\right)$.

Case 3: if $0 \le \lambda < \mu = \nu$ or $\lambda < 0 < \mu = \nu$ then

$\left(x, y, z\right) = \left(\mu + \sqrt{\mu^2 + 4\lambda + 4}, -2, -2\right)$.

Case 4: if $0 < \lambda = \mu < \nu$ then

$(x, y, z) = (-\alpha, \beta, -1)$ or $(\beta, -\alpha, -1)$, where

$$\alpha = \frac{1}{2}\left(\sqrt{\mu^2 + \frac{4(1-\mu^2)}{1-\nu}} - \mu\right), \beta = \frac{1}{2}\left(\sqrt{\mu^2 + \frac{4(1-\mu^2)}{1-\nu}} + \mu\right).$$

Case 5: if $\mu \geq 0, \nu > 0, \lambda = -\nu$ then

$$(x, y, z) = \left(2, -\nu - \sqrt{\nu^2 - 4\mu + 4}, -2\right).$$

Case 6: if $0 \leq \lambda < \mu < \nu$ or $\lambda < 0 \leq \mu < \nu$, $\mu, \nu \neq -\lambda$ then

$(x, y, z) = (1, -\alpha, -\beta)$, where $-\alpha$ is the zero of the

$$P(t) = (1-\lambda^2)t^4 + \lambda(\mu - \lambda\nu)t^3 + 2(\lambda\mu\nu - 1)t^2 + \mu(\lambda - \mu\nu)t + (1-\mu^2)$$

and $\beta = \frac{1-\alpha^2}{\mu + \lambda\alpha}$.

In this case, if $\lambda > -\mu$ then $-\alpha$ is the unique zero of P in $(-1, 0)$ and $0 < \beta < 1$.

If $-\nu < \lambda < -\mu$ then $-\alpha$ is the unique zero of P in $(-\infty, -1)$ and $0 < \beta < 1$.

If $\lambda < -\nu$ then $-\alpha$ is the unique zero of P in $(-\infty, -1)$ and $\beta > 1$.

This three variable exact solution is used in simulations when there are three variables and when the software is able to find zeros of $P(t)$. Otherwise, a nonlinear optimization approach described in (2) is used instead.

(2) Program

The SAS procedure IML and function NLPLM for optimization of a nonlinear function are used to minimize the objective function and find the estimates of coefficients by iteration given some initial values of the coefficients, when there

are more than 3 variables. Performing the procedure repeatedly for various initial guesses of the parameter estimates, the minimum objective function value and the corresponding estimates of coefficients are recorded and these estimates are the RMA estimates. Draper and Yang (1997) considered the problem originally as a nonlinear weighted least squares problem and did simulations in S-PLUS. In our study, our simulation results show that when using criterion L_G^p all three optimization methods available in SAS proc IML (NLPQN, NLPTR and NLPLM) tend to give the same minimum objective function value and estimates of coefficients. However, the NLPQN and NLPTR in PROC IML are not as efficient as the NLPLM.

The criterion

$$L_G^p = \sum_{i=1}^{n} \left(V_i^{1/p} \right)^2 = \sum_{i=1}^{n} \left(\frac{\sum_{j=1}^{p} a_j x_{ij} - 1}{\Pi_{j=1}^{p} a_j^{1/p}} \right)^2$$

could also be considered as a least squares problem. The NLPLM method is the fastest, the most adapted to the least squares idea and the most stable.

(3) Initial Guesses

To assess the effect of the initial guess of the parameter estimates, an example data set of 4 variables, $X_1, ..., X_4$, that contains random errors is generated with 50 data points.

The values of variables X_2, X_3, X_4 are generated from the multivariate normal distribution $MN(\mathbf{0}, \mathbf{I})$. The "observed values" for X_2, X_3, X_4 are obtained by adding an error term of the form $mult * \varepsilon_{ij}$, where $mult$ is a scalar factor that is

between 0 and 1, and the errors ε_{ij} are independent, identically distributed (iid) $N(0,1)$. X_1 is obtained by equation

$$X_1 = \frac{1}{3} - \frac{2}{3}X_2 - \frac{4}{3}X_3 + \frac{1}{3}X_4 + error\,.$$

So that we have

$$3X_1 + 2X_2 + 4X_3 - X_4 = 1$$

with measurement errors in each variable and the $mult$ parameter is set to 0.15. First, the OLS regression was applied to the same data set by taking each $X_1,...,X_4$ as the dependent variable in turn. The sample data set produced these OLS fitted planes that are written in canonical form as

$$\begin{aligned} 3.028X_1 + 2.016X_2 + 4.111X_3 - 1.000X_4 &= 1 \\ 2.969X_1 + 2.406X_2 + 3.926X_3 - 0.956X_4 &= 1 \\ 2.981X_1 + 1.933X_2 + 4.219X_3 - 1.002X_4 &= 1 \\ 2.724X_1 + 1.769X_2 + 3.764X_3 - 1.606X_4 &= 1 \end{aligned} \qquad (3.3)$$

with R-squared respectively in table 3.1. As shown in eq. 3.3, the four OLS equations provide different coefficient estimates depending on which variable is the dependent variable.

Table 3.1 R-squared for the four OLS regression in eq. 3.3

Dependent Variable	X_1	X_2	X_3	X_4
R square	0.974	0.835	0.963	0.566

Then RMA was performed on the same data set. The simulation results showed that the objective function value and estimates of coefficients are very sensitive to

the starting values. Then OLS coefficient estimates associated with the largest R squared are used as a basis for initial guesses.

In this case, the R-squared associated with the model that has X_1 as the dependent variable is the largest among the four, and the corresponding OLS estimates (3.028, 2.016, 4.111, -1) are selected to be the basis of the initial guess. Then initial values are obtained by adding random terms to each OLS estimate that are generated from iid $U(-1,1)$.

50, 100 and 200 sets of initial guesses are used to obtain the minimum objective function. The simulation results (as shown in table 3.2) for the different numbers of initial guesses show that the minimum objective function value and the parameter estimates are the same for the three different numbers of initial guesses. However, when sample size is small ($n = 20$), sometimes the optimization procedure prefers more initial guesses to find a global minimum. 200 sets of initial guesses are used in our final algorithm.

Table 3.2 Parameter Estimates and Objective Function with 50, 100 and 200 initial guesses

# Initial Guesses	Min Obj Function	$\hat{a}_1$	$\hat{a}_2$	$\hat{a}_3$	$\hat{a}_4$
50	3.205	2.928	2.032	4.010	-1.135
100	3.205	2.928	2.032	4.010	-1.135
200	3.205	2.928	2.032	4.010	-1.135

(4) One or more of the variables are constants

When one of the variables is a constant, the variance covariance matrix of the original data is singular and it is not appropriate to perform OLS to obtain the initial guesses. If one or more variables are constant the optimization fails.

3.4 Simulation Results

We now consider a small simulation study to investigate the effects of sample size, magnitude of the additive error and correlation between variables on the means of the coefficient estimates. Further investigation of the distribution of the coefficient estimates is given in Chapter 4.

In simulations, 500 sample data sets of size 20, 50 and 100 are generated, with three uncorrelated variables (X_2, X_3) that are generated from $MN(\mathbf{0}, \mathbf{P})$ and X_1 that is obtained by the equation

$$3X_1 + 2X_2 + X_3 = 1,$$

with small, moderate and large additive error ($mult = 0.15, 0.5, 0.8$). Here $\mathbf{P}$ is the covariance matrix with zero, moderate and large correlation ($\rho = 0, 0.5, 0.8$) for each pair of variables; 200 initial guesses are used.

As table 3.3 shows, most of the means of the estimates are close to the true coefficients 3, 2, 1 found by the minimum objective function approach, except those data sets with moderate and large errors ($mult = 0.5, 0.8$). When the sample size is larger, the mean of the coefficient estimates are closer to the true values than when the sample size is small ($n = 20$), even when the additive errors are large ($mult = 0.5, 0.8$). As expected, the RMA regression and minimum objective approach provide better coefficient estimates for data with a large sample size and small additive errors.

Table 3.3 Simulation results of average parameter estimates and average minimum objective function values for 27 sample sets

n / ρ / *mult*	Avg Min Obj Function	$\bar{\hat{a}}_1$	$\bar{\hat{a}}_2$	$\bar{\hat{a}}_3$
20/0/0.15	0.818	3.072	2.049	1.112
20/0/0.5	8.240	3.850	2.802	1.837
20/0/0.8	18.745	**-2.399**	**2.389**	**-0.852**
20/0.5/0.15	0.819	3.069	2.002	1.122
20/0.5/0.5	8.107	**7.137**	**4.925**	**2.031**
20/0.5/0.8	18.227	**-15.035**	**-14.171**	**-11.402**
20/0.8/0.15	0.810	3.068	1.924	1.188
20/0.8/0.5	7.555	**1.046**	**3.022**	**-4.381**
20/0.8/0.8	16.754	**2.482**	**0.636**	**1.677**
50/0/0.15	2.187	3.011	2.024	1.081
50/0/0.5	21.958	**3.921**	**2.989**	**2.124**
50/0/0.8	51.991	**3.931**	**3.050**	**2.439**
50/0.5/0.15	2.170	3.009	1.972	1.101
50/0.5/0.5	21.910	3.397	2.261	1.698
50/0.5/0.8	51.672	**-0.591**	**-0.464**	**0.221**
50/0.8/0.15	2.143	3.008	1.897	1.164
50/0.8/0.5	21.372	3.210	2.159	1.367
50/0.8/0.8	49.507	3.555	2.049	1.938
100/0/0.15	4.472	3.012	2.014	1.079
100/0/0.5	44.920	3.153	2.240	1.616
100/0/0.8	107.190	2.653	2.044	1.750
100/0.5/0.15	4.456	3.008	1.967	1.094
100/0.5/0.5	45.086	3.132	2.032	1.576
100/0.5/0.8	108.102	3.721	2.726	1.898
100/0.8/0.15	4.395	3.008	1.893	1.157
100/0.8/0.5	44.477	3.158	1.992	1.507
100/0.8/0.8	105.778	3.671	2.824	1.047

Note: those estimates that are not close to the true coefficients are marked bold.

Chapter 4: DISTRIBUTION OF ESTIMATES

Unlike in OLS regression, the RMA coefficient estimates do not have closed formulas. The exact and asymptotic distributions of the coefficient estimates appear difficult to obtain. Alternatively, it is of interest to look at the distributions of coefficient estimates obtained from simulations.

4.1 Effects of Sample Size, Error and Correlation between Variables

4.1.1 Effects of Sample Size

As shown in Chapter 3, the coefficient estimates are not stable when obtained from simulated data when sample size is small and the results suggest that a sample size of 50 or above will provide much more reliable RMA estimates. Sample sizes of 20, 50 and 100 with small additive errors are used in the simulations and the histograms and normal QQplots of coefficient estimates are obtained. Several examples will provide detailed results.

500 sample data sets of size 20 are generated, with three uncorrelated variables (X_2, X_3, X_4) that are generated from $MN(\mathbf{0}, \mathbf{I})$ and X_1 that is obtained by the equation

$$3X_1 + 2X_2 + 4X_3 - 1X_4 = 1,$$

with small additive error ($mult = 0.15$).

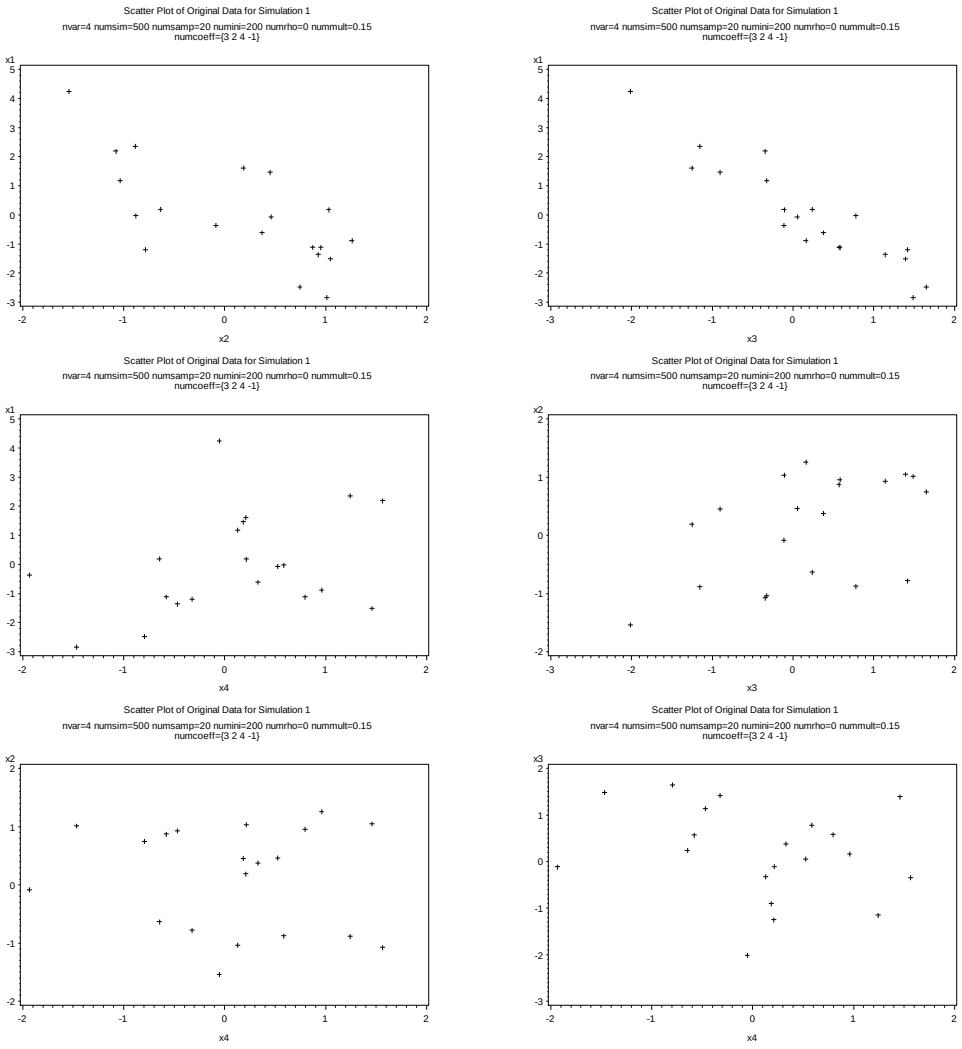

Figure 4.1 Scatter plots of original variables X_i *vs* X_j, $i < j = 1,...,4$ for sample with $n = 20$, $\rho = 0$, $mult = 0.15$ and coefficients are {3, 2, 4, -1}

As shown in figure 4.1, scatter plots of X_1 *vs* X_2 and X_1 *vs* X_3 have clear linear relationships. Other than these, there are no obvious trends in the scatter plots.

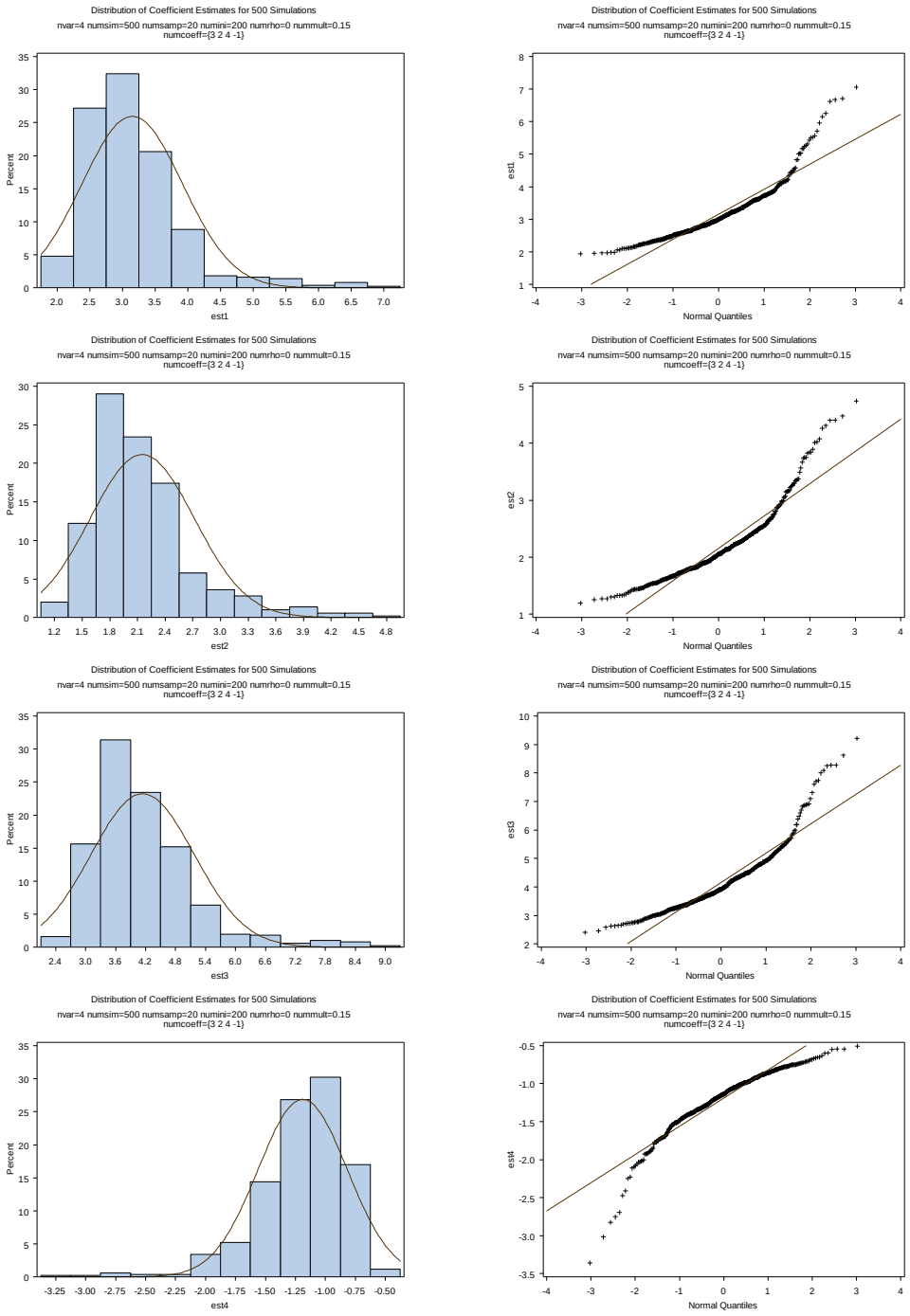

Figure 4.2 Histograms (*left*) and normal QQplots (*right*) of coefficient estimates based on 500 samples with $n = 20$, $\rho = 0$ and $mult = 0.15$

As shown in figure 4.2, both histograms and normal QQplots do not show a normal distribution of coefficient estimates. The histograms are highly skewed when the sample size is 20.

Table 4.1 Descriptive statistics of $\hat{a}_i, i = 1,...,4$ with $n = 20$, $\rho = 0$ and $mult = 0.15$

	$\hat{a}_1$	$\hat{a}_2$	$\hat{a}_3$	$\hat{a}_4$
Mean	3.15	2.15	4.14	-1.19
StdDev	0.77	0.57	1.03	0.37
Median	3.00	2.05	3.92	-1.14
Min	1.95	1.19	2.40	-3.36
Max	7.07	4.74	9.21	-0.50
Q1	2.64	1.79	3.46	-1.37
Q3	3.49	2.38	4.58	-0.93

As shown in table 4.1, the mean for each estimate is close to the true coefficient. The mean and median are somewhat close for each $\hat{a}_i, i = 1,...,4$ and the variability of each estimate is not very large.

500 sample data sets of size 50 are generated. Three uncorrelated variables (X_2, X_3, X_4) are generated from $MN(\mathbf{0}, \mathbf{I})$ and X_1 is obtained by the equation

$$3X_1 + 2X_2 + 4X_3 - 1X_4 = 1,$$

with small additive error ($mult = 0.15$).

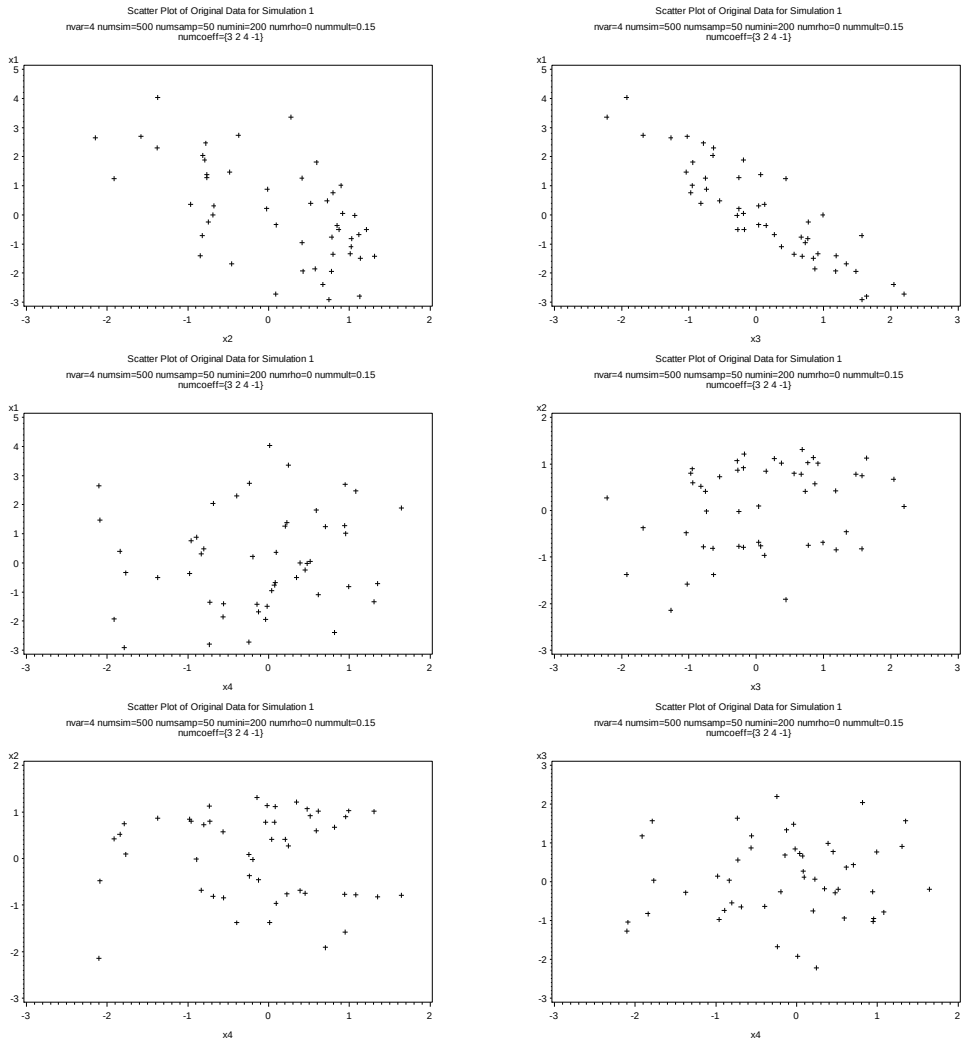

Figure 4.3 Scatter plots of original variables X_i *vs* $X_j, i < j = 1,...,4$ for sample with $n = 50$, $\rho = 0$, $mult = 0.15$ and coefficients are {3, 2, 4, -1}

As shown in figure 4.3, there are no linear trends in the plots except for the ones associated with X_1.

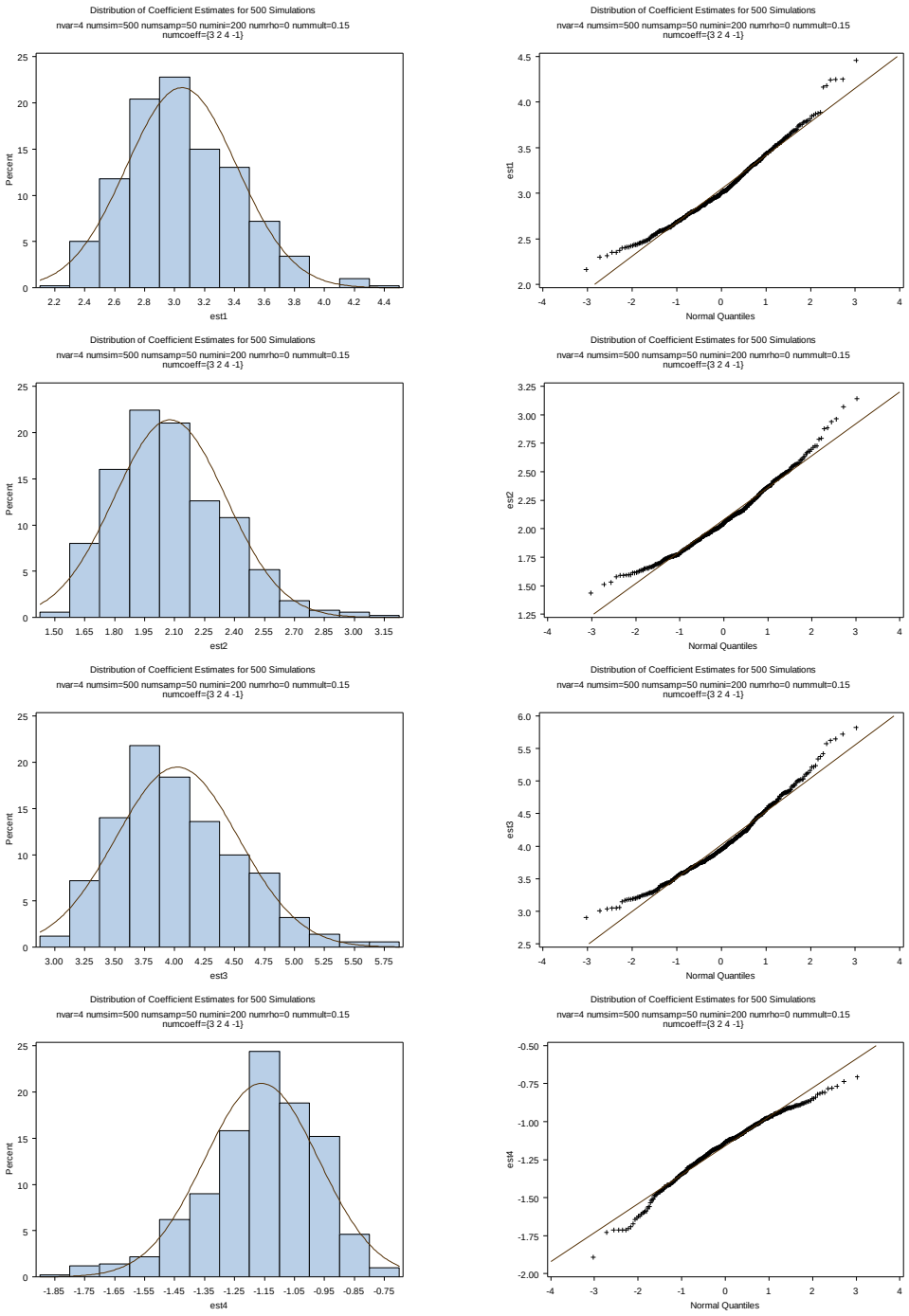

Figure 4.4 Histograms (*left*) and normal QQplots (*right*) of coefficient estimates based on 500 samples with $n = 50$, $\rho = 0$, $mult = 0.15$

As shown in figure 4.4, the histograms are less skewed than for a sample of size 20. Similarly, normal QQplots suggest that when the sample size is 50, the coefficient estimates have a skewed distribution.

Table 4.2 Descriptive statistics of $\hat{a}_i, i = 1,...,4$ with $n = 50$, $\rho = 0$ and $mult = 0.15$

	$\hat{a}_1$	$\hat{a}_2$	$\hat{a}_3$	$\hat{a}_4$
Mean	3.05	2.08	4.02	-1.16
StdDev	0.37	0.28	0.51	0.19
Median	3.01	2.04	3.95	-1.14
Min	2.17	1.43	2.90	-1.89
Max	4.46	3.14	5.82	-0.71
Q1	2.79	1.88	3.67	-1.27
Q3	3.30	2.26	4.34	-1.02

As shown in table 4.2, the mean for each estimate is closer to the true coefficient than when the sample size is 20. The mean and median are close for each $\hat{a}_i, i = 1,...,4$ and variability of each estimate is also smaller than for a sample of size 20.

500 sample data sets of size 100 are generated, with three uncorrelated variables (X_2, X_3, X_4) that are generated from $MN(\mathbf{0}, \mathbf{I})$ and X_1 that is obtained by the equation

$$3X_1 + 2X_2 + 4X_3 - 1X_4 = 1,$$

with small additive error ($mult = 0.15$).

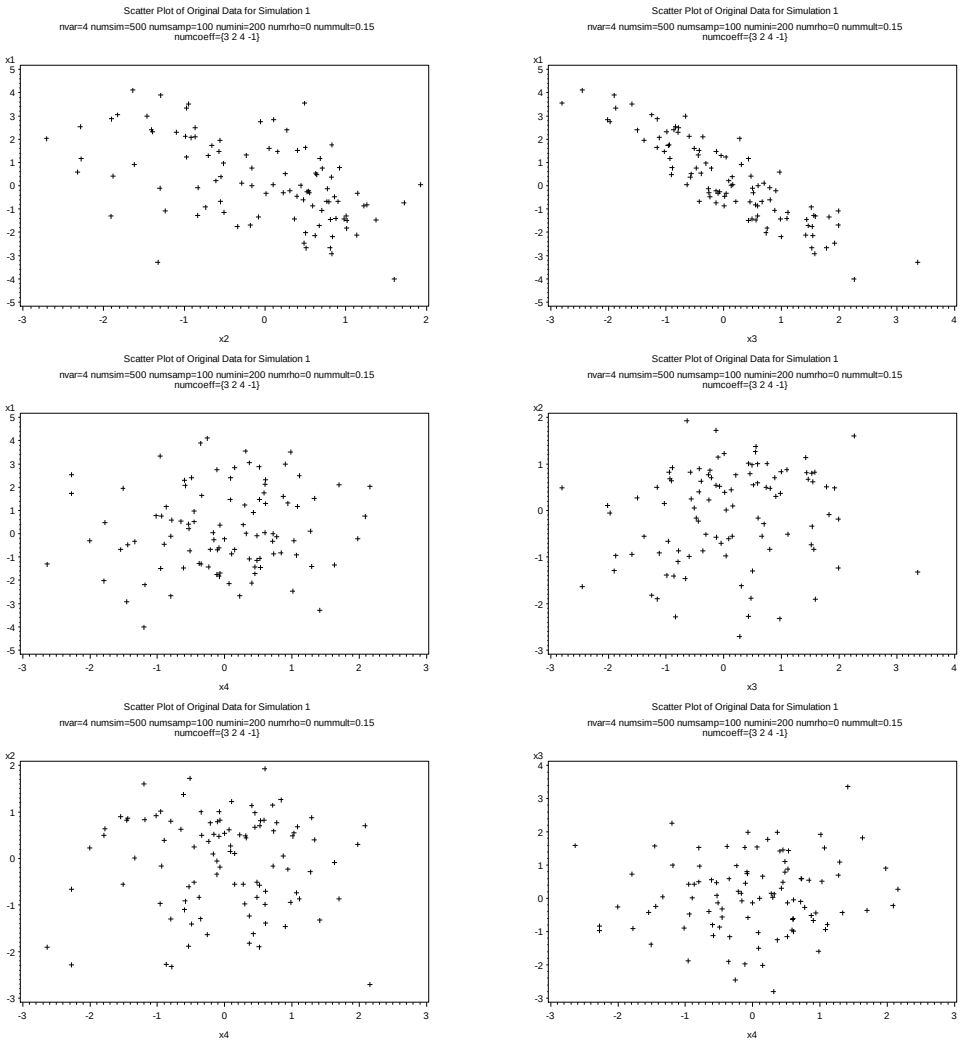

Figure 4.5 Scatter plots of original variables X_i *vs* $X_j, i < j = 1,...,4$ for sample with $n = 100$, $\rho = 0$, $mult = 0.15$ and coefficients are {3, 2, 4, -1}

As shown in figure 4.5, there are no linear trends in the plots except for the ones associated with X_1.

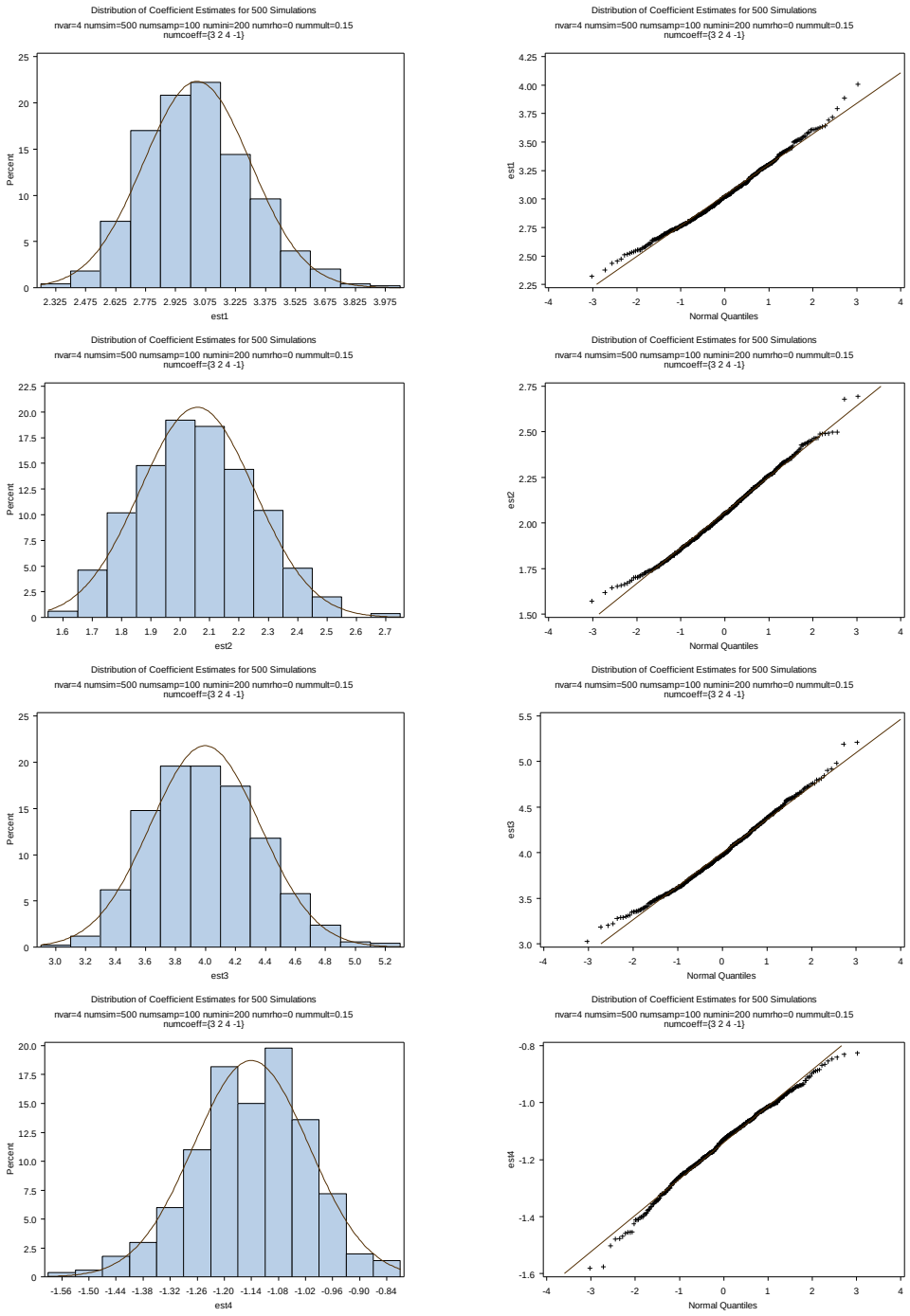

Figure 4.6 Histograms (*left*) and normal QQplots (*right*) of coefficient estimates based on 500 samples with $n = 100$, $\rho = 0$ and $mult = 0.15$

As shown in figure 4.6, the histograms are tending to be symmetric and less obviously skewed than for samples of size 20 and 50. Similarly, normal QQplots

suggest when the sample size is 100, the coefficient estimates are more closely normally distributed.

Table 4.3 Descriptive statistics of $\hat{a}_i, i = 1,...,4$ with $n = 100$, $\rho = 0$ and $mult = 0.15$

	$\hat{a}_1$	$\hat{a}_2$	$\hat{a}_3$	$\hat{a}_4$
Mean	3.03	2.06	4.00	-1.14
StdDev	0.27	0.20	0.37	0.13
Median	3.02	2.05	3.97	-1.13
Min	2.32	1.57	3.03	-1.58
Max	4.01	2.69	5.21	-0.83
Q1	2.84	1.92	3.73	-1.22
Q3	3.21	2.20	4.25	-1.05

As shown in table 4.3, the mean for each estimate is very close to the true coefficient. The mean and median are the same for each $\hat{a}_i, i = 1,...,4$ and variability of each estimate is small.

Hence, large sample size (50 or more) would yield more stable and reliable results and the distributions of coefficient estimates are slightly skewed or approximately normal.

4.1.2 Effects of Error

Next we consider the effect of the additive errors on the distribution of the coefficient estimates. We restrict consideration to a sample size of 100 in future simulations to obtain results for the distribution of coefficient estimates.

Small, moderate and large additive errors ($mult = 0.15, 0.5, 0.8$) are used in the simulations and the histograms and normal QQplots of coefficient estimates are obtained. Several examples will provide detailed results.

As shown in figure 4.6, the distribution of coefficient estimates of a sample with 100 cases and small errors ($mult = 0.15$) is approximately normal.

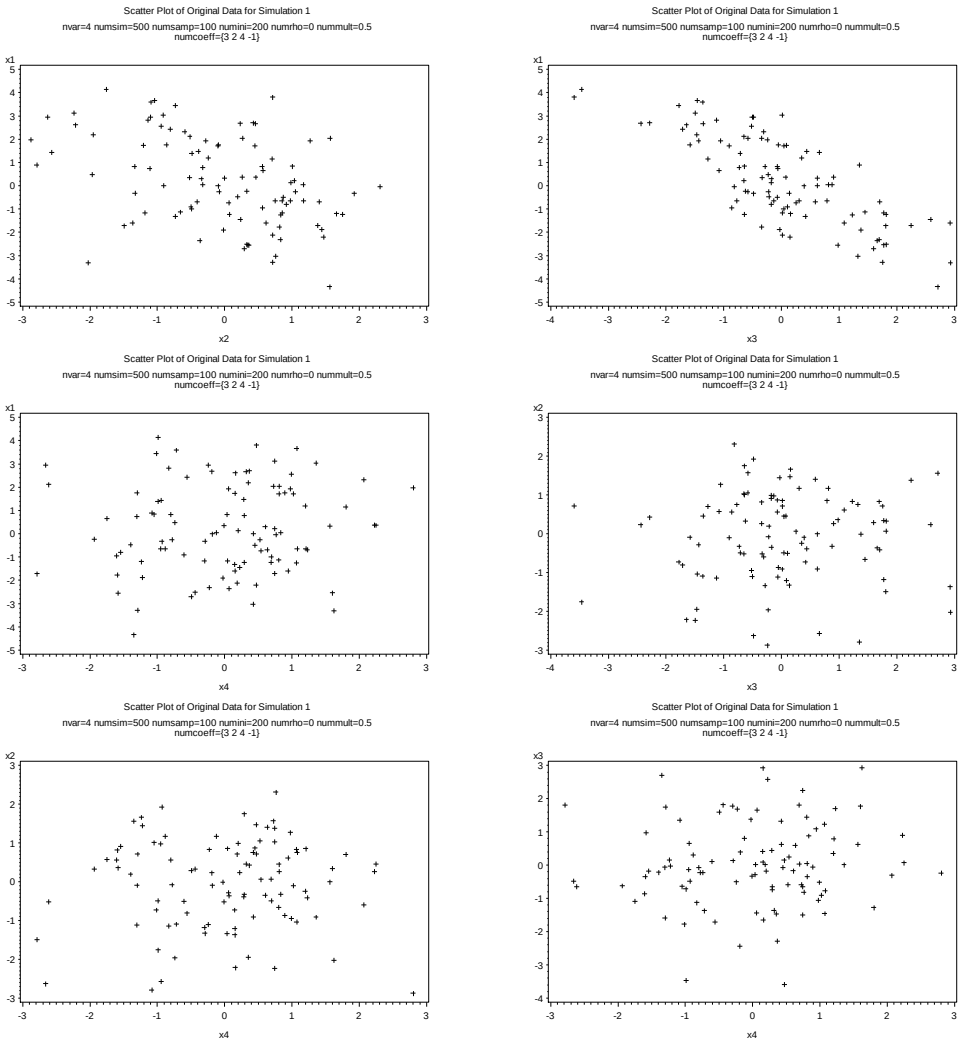

Figure 4.7 Scatter plots of original variables X_i *vs* $X_j, i < j = 1,...,4$ for sample with $n = 100$, $\rho = 0$, $mult = 0.5$ and coefficients are {3, 2, 4, -1}

500 sample data sets of size 100 are generated, with three uncorrelated variables (X_2, X_3, X_4) that are generated from $MN(\mathbf{0}, \mathbf{I})$ and X_1 that is obtained by the equation

$$3X_1 + 2X_2 + 4X_3 - 1X_4 = 1,$$

with moderate additive error ($mult = 0.5$).

As shown in figure 4.7, there are no linear trends in the plots except for the ones associated with X_1 .

As shown in figure 4.8, as the errors become larger, the histograms become much more skewed and the normal QQplots clearly show non-normality.

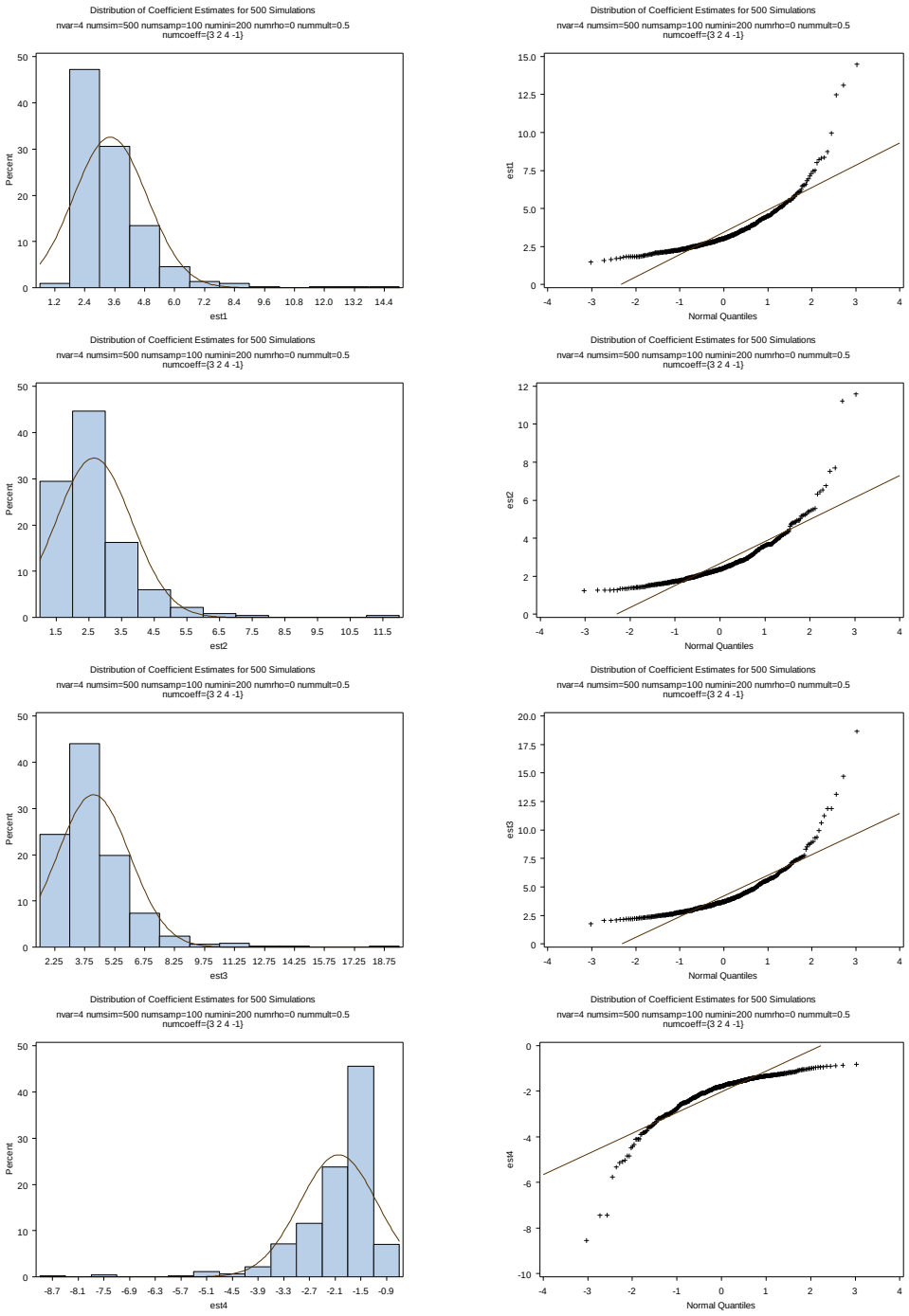

Figure 4.8 Histograms (*left*) and normal QQplots (*right*) of coefficient estimates based on 500 samples with $n = 100$, $\rho = 0$ and $mult = 0.5$

Table 4.4 Descriptive statistics of $\hat{a}_i, i = 1,...,4$ with $n = 100$, $\rho = 0$ and $mult = 0.5$

	$\hat{a}_1$	$\hat{a}_2$	$\hat{a}_3$	$\hat{a}_4$
Mean	3.43	2.66	4.20	-2.02
StdDev	1.47	1.16	1.81	0.91
Median	3.02	2.38	3.71	-1.78
Min	1.47	1.24	1.75	-8.54
Max	14.48	11.59	18.65	-0.82
Q1	2.50	1.91	3.03	-2.33
Q3	3.94	3.06	4.82	-1.43

As shown in table 4.4, the mean for each estimate is further away from the true coefficient than when the errors are small ($mult = 0.15$). The mean and median are more different for each $\hat{a}_i, i = 1,...,4$ and variability of each estimate is larger. The statistics suggest the distribution is skewed as seen from the figure 4.8.

500 sample data sets of size 100 are generated, with three uncorrelated variables (X_2, X_3, X_4) that are generated from $MN(\mathbf{0}, \mathbf{I})$ and X_1 that is obtained by the equation

$$3X_1 + 2X_2 + 4X_3 - 1X_4 = 1,$$

with large additive error ($mult = 0.8$).

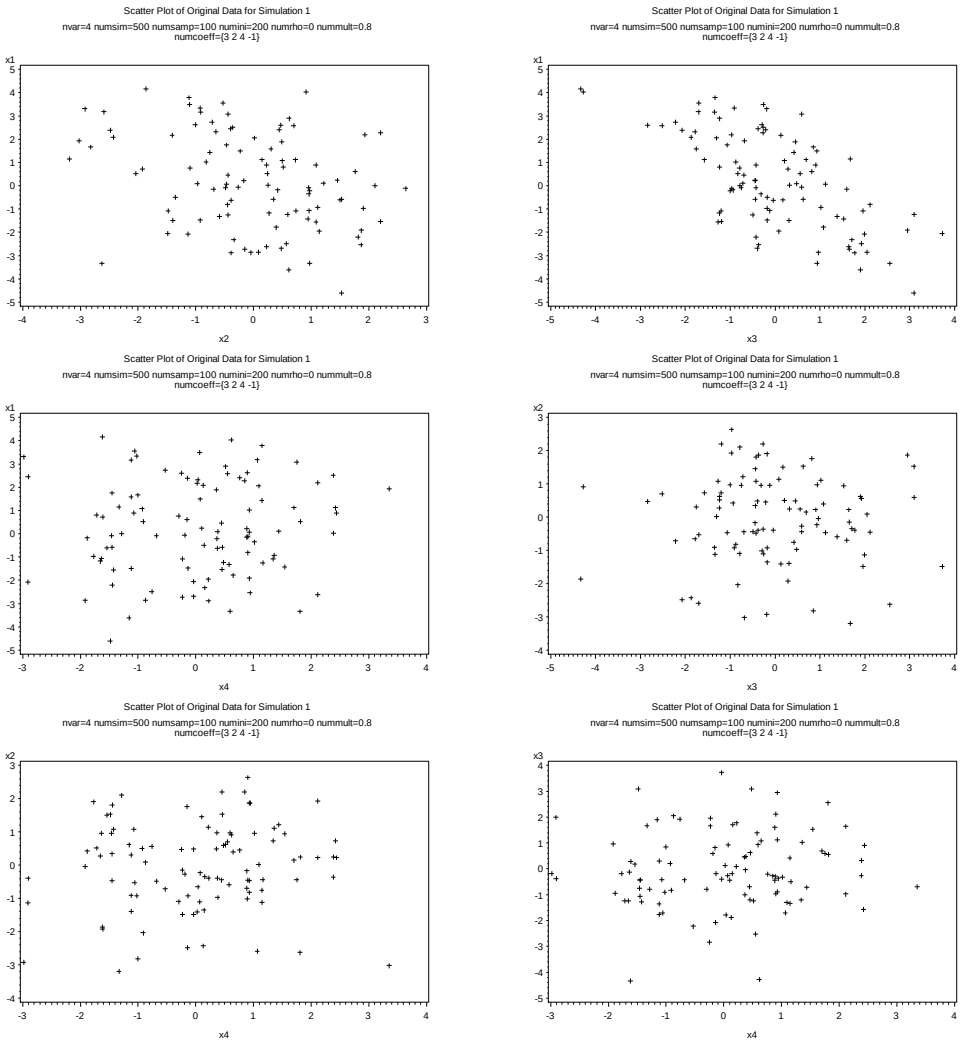

Figure 4.9 Scatter plots of original variables X_i *vs* $X_j, i < j = 1,...,4$ for sample with $n = 100$, $\rho = 0$, $mult = 0.8$ and coefficients are {3, 2, 4, -1}

As shown in figure 4.9, there are no linear trends in the plots except for the ones associated with X_1.

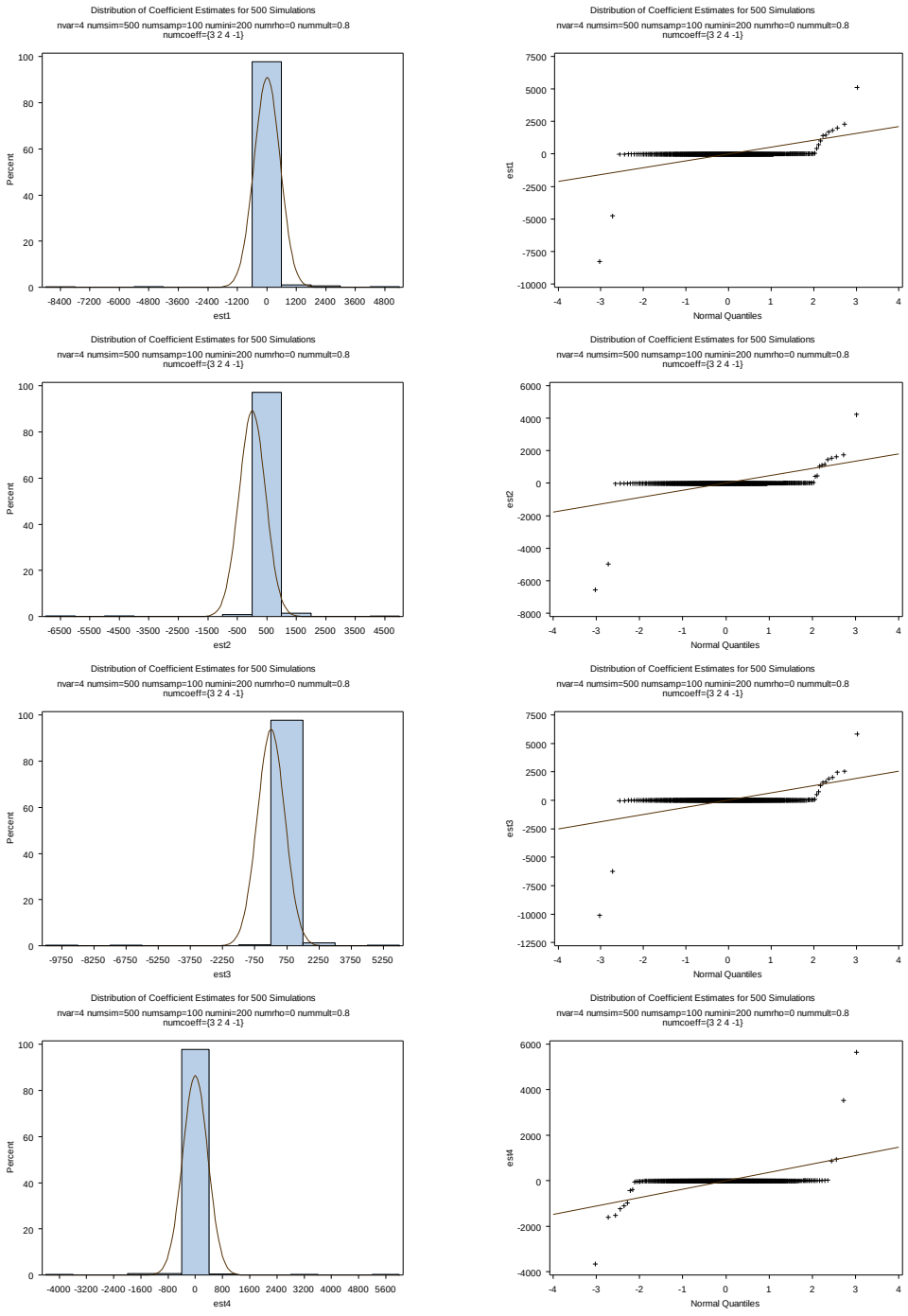

Figure 4.10 Histograms (*left*) and normal QQplots (*right*) of coefficient estimates based on 500 samples with $n = 100$, $\rho = 0$, $mult = 0.8$

As shown in figure 4.10, both histograms and normal QQplots do not show a normal distribution of coefficient estimates. When the errors are relatively large

($mult = 0.8$), RMA sometimes gives wild estimates for coefficients. Due to these wild estimates, the histograms are extremely long tailed and show a wide range of estimates.

Table 4.5 Descriptive statistics of $\hat{a}_i, i = 1,...,4$ with $n = 100$, $\rho = 0$ and $mult = 0.8$

	$\hat{a}_1$	$\hat{a}_2$	$\hat{a}_3$	$\hat{a}_4$
Mean	14.07	10.18	13.08	-2.83
StdDev	526.41	447.53	638.03	369.65
Median	3.04	2.62	3.46	-2.15
Min	-8253.13	-6559.19	-10123.88	-3657.34
Max	5120.79	4228.19	5831.05	5641.82
Q1	2.23	1.85	2.53	-3.44
Q3	4.90	4.05	5.58	-1.51

As shown in table 4.5, the mean for each estimate is no longer close to the true coefficient while the median seems to be closer than the mean. The mean and median are very different for each $\hat{a}_i, i = 1,...,4$ than those when the error is moderate ($mult = 0.5$) and the minimum or maximum of each estimate is wildly far away from the true coefficient. The statistics suggest the distribution is highly skewed and long tailed.

Samples with size of 20 and 50 were generated with moderate and large errors as well. The results show that when sample size is 20 or 50, the distribution of estimates behaves similar to those discussed in Section 4.1.1 and it is more skewed and the estimates have more variability than when the sample size is 100. By adding moderate and large errors to each variable of samples of size 20 and 50, the estimates become wild and the distributions of estimates are even more skewed and long tailed. Descriptive statistics for samples of size 20 and 50 with moderate and large errors will be found in appendix 4.1.

Hence, the simulation results show that in RMA regression, the distribution of coefficient estimates is very sensitive to the magnitude of the additive errors. The results suggest when additive errors are relatively large the distributions of estimates are long tailed.

4.1.3 Effects of Correlation between Variables

In OLS, the existence of multicolinearity reduces the precision of the estimated variable coefficients, and results in estimates with large variability and has complex impact on the regression model (Kutner, Nachtsheim, Neter and Li 2004). Here we consider the effects of correlations among the variables on the RMA coefficient estimates.

A correlation of 0, 0.5 and 0.8 between variables is used in simulations to observe the impact on the distribution of estimates.

Previously the case of zero correlation was considered. See figure 4.5 and 4.6.

500 sample data sets of size 100 are generated with three correlated variables (X_2, X_3, X_4) that are generated from $MN(\mathbf{0}, \mathbf{P})$ and X_1 that is obtained by the equation

$$3X_1 + 2X_2 + 4X_3 - 1X_4 = 1,$$

with small additive error ($mult = 0.15$) and moderate correlation ($\rho = 0.5$) between all pairs of X_2, X_3 and X_4.

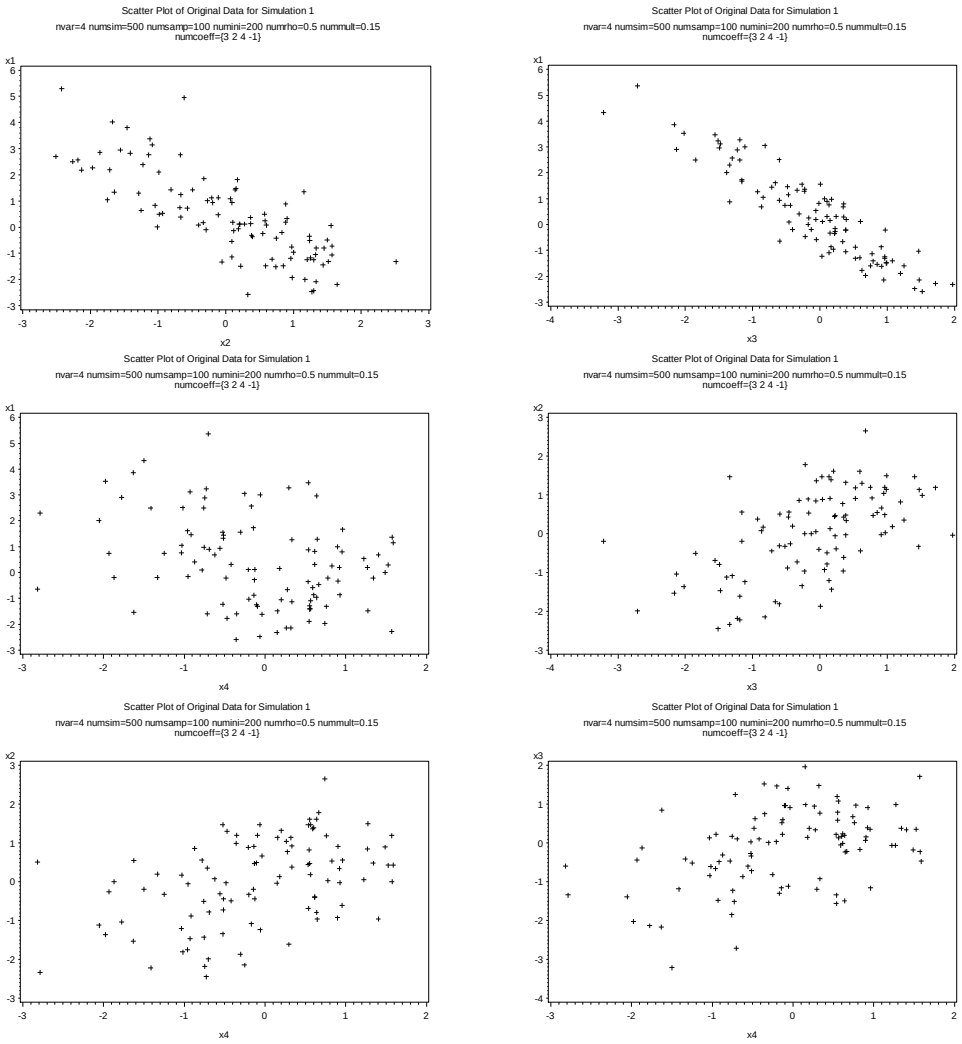

Figure 4.11 Scatter plots of original variables X_i *vs* $X_j, i < j = 1,...,4$ for sample with $n = 100$, $\rho = 0.5$, $mult = 0.15$ and coefficients are {3, 2, 4, -1}

As shown in figure 4.11, all plots show varying degrees of linear trends.

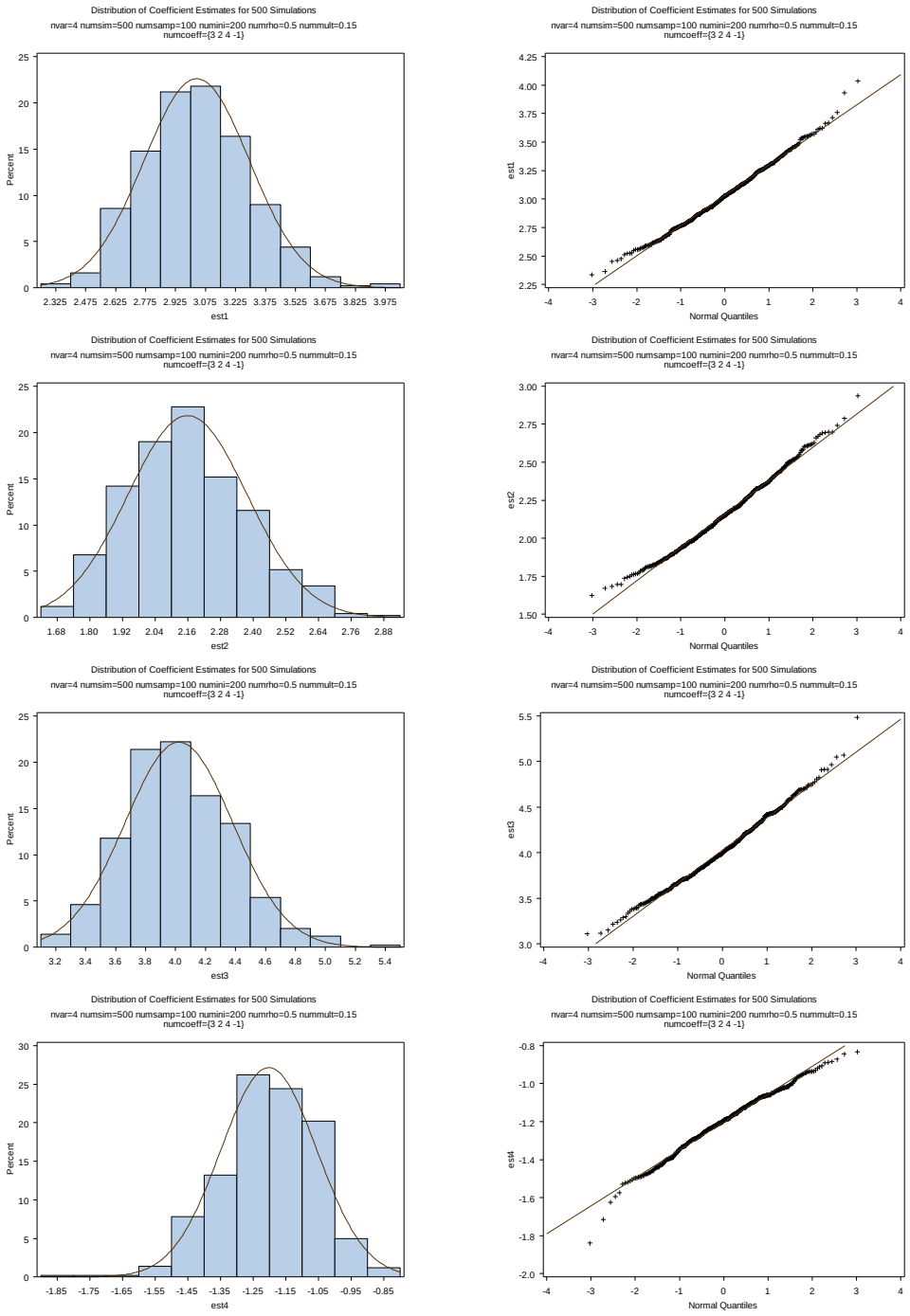

Figure 4.12 Histograms (*left*) and normal QQplots (*right*) of coefficient estimates based on 500 samples with $n = 100$, $\rho = 0.5$, $mult = 0.15$

As shown in figure 4.12, both histograms and the normal QQplots show the distributions of estimates are slightly skewed.

Table 4.6 Descriptive statistics of $\hat{a}_i, i = 1,...,4$ with $n = 100$, $\rho = 0.5$, $mult = 0.15$

	$\hat{a}_1$	$\hat{a}_2$	$\hat{a}_3$	$\hat{a}_4$
Mean	3.03	2.16	4.02	-1.20
StdDev	0.26	0.22	0.36	0.15
Median	3.03	2.15	3.99	-1.19
Min	2.34	1.62	3.11	-1.84
Max	4.03	2.94	5.48	-0.83
Q1	2.84	2.00	3.76	-1.29
Q3	3.21	2.31	4.26	-1.10

As shown in table 4.6, the mean for each estimate is close to the true coefficient. The mean and median are the same for each $\hat{a}_i, i = 1,...,4$ and the variability of each estimate is not large. The results are similar to those in table 4.3 for the case of $\rho = 0$.

500 sample data sets of size 100 are generated, with three correlated variables (X_2, X_3, X_4) that are generated from $MN(\mathbf{0}, \mathbf{P})$ and X_1 that is obtained by the equation

$$3X_1 + 2X_2 + 4X_3 - 1X_4 = 1,$$

with small additive error ($mult = 0.15$) and large correlation ($\rho = 0.8$) between all pairs of X_2, X_3 and X_4.

.

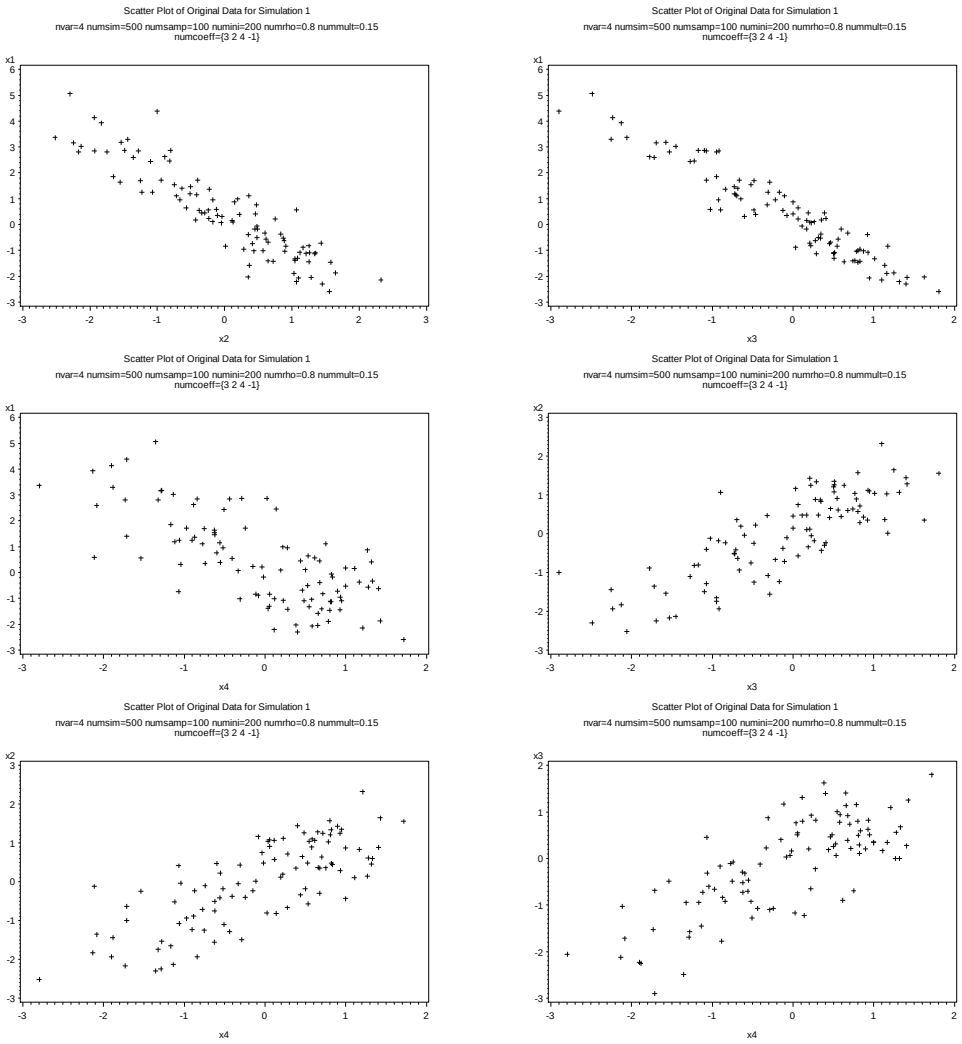

Figure 4.13 Scatter plots of original variables X_i *vs* $X_j, i < j = 1,...,4$ for sample with $n = 100$, $\rho = 0.8$, $mult = 0.15$ and coefficients are {3, 2, 4, -1}

As shown in figure 4.13, all plots show linear trends.

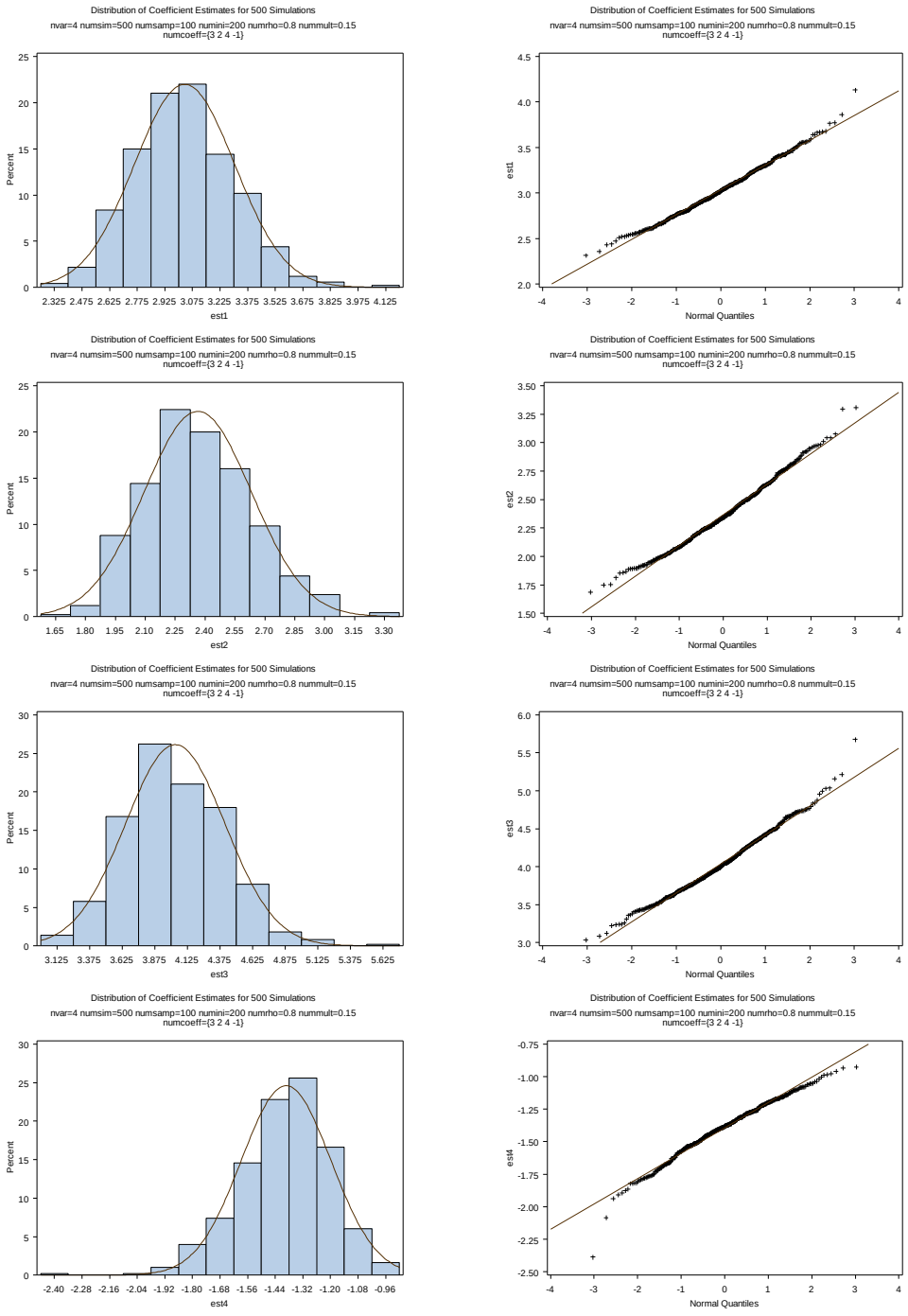

Figure 4.14 Histograms (*left*) and normal QQplots (*right*) of coefficient estimates based on 500 samples with $n = 100$, $\rho = 0.8$ and $mult = 0.15$

As shown in figure 4.14, the plots suggest slightly skewed distributions for the coefficient estimates.

Table 4.7 Descriptive statistics of $\hat{a}_i, i = 1,...,4$ with $n = 100$, $\rho = 0.8$ and $mult = 0.15$

	$\hat{a}_1$	$\hat{a}_2$	$\hat{a}_3$	$\hat{a}_4$
Mean	3.03	2.36	4.03	-1.39
StdDev	0.27	0.27	0.38	0.19
Median	3.02	2.34	4.00	-1.38
Min	2.31	1.69	3.03	-2.39
Max	4.13	3.31	5.68	-0.93
Q1	2.84	2.18	3.76	-1.51
Q3	3.22	2.54	4.29	-1.26

As shown in table 4.7, the mean for each estimate is close to the true coefficient. The mean and median are very close for each $\hat{a}_i, i = 1,...,4$ and the variability of each estimate is not large. The results are similar to those in table 4.3 and 4.6. Therefore, when variables are highly correlated and a linear relationship between each two variables is present, RMA regression does a good job to obtain the estimates for coefficients and the distributions of estimates are generally skewed. Sample data sets with size of 20 and 50 are generated with small errors and with moderate and large correlations between all pairs of X_2, X_3 and X_4 as well. The results show that when the sample size is 20 or 50, the distributions of estimates behave similar to those discussed in Section 4.1.1 and are more skewed and the estimates have more variability than when the sample size is 100. By assigning moderate and large correlations between all pairs of X_2, X_3 and X_4 of samples of size 20 and 50, the estimates behave very similar as when X_2, X_3 and X_4 are uncorrelated. Descriptive statistics for samples of size 20 and 50 with moderate and large correlations between all pairs of X_2, X_3 and X_4 are shown in the appendix 4.1.

4.2 Effects of Zero Coefficients

The previous discussion considered different combinations of sample sizes, magnitudes of errors and correlations between variables when all true coefficients that generated the data are non-zero. A model with zero coefficients will be of interest.

A sample size of 100, with small additive errors ($mult = 0.15$) and uncorrelated data will be used in the investigation of the distribution of coefficient estimates when a parameter coefficient is zero.

500 sample data sets of size 100 are generated with three uncorrelated variables (X_2, X_3, X_4) that are generated from $MN(\mathbf{0}, \mathbf{I})$ and X_1 that is obtained by the equation

$$3X_1 + 2X_2 + 0X_3 - 1X_4 = 1,$$

with small additive error ($mult = 0.15$).

As shown in figure 4.15, there are no linear trends in the plots except for the ones associated with X_1.

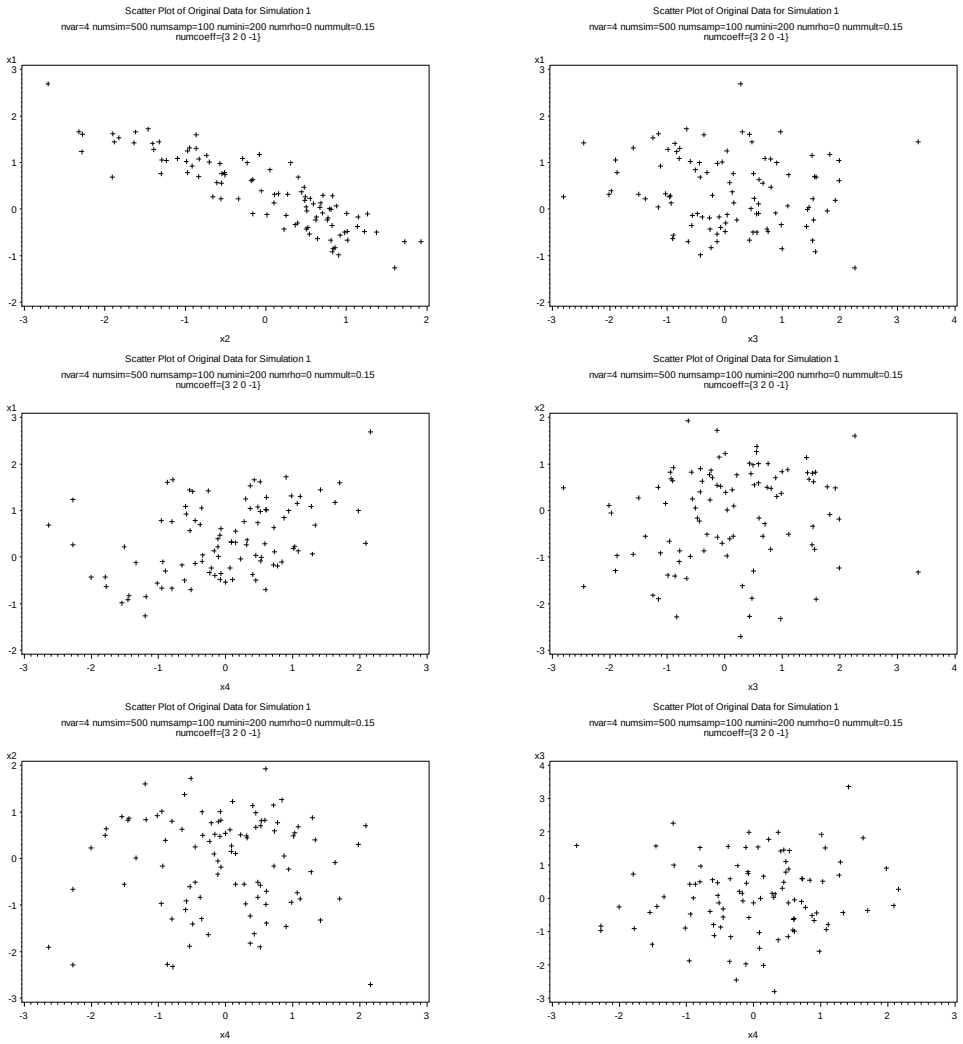

Figure 4.15 Scatter plots of original variables X_i *vs* $X_j, i < j = 1,...,4$ for sample with $n = 100$, $\rho = 0$, $mult = 0.15$ and coefficients are {3, 2, 0, -1}

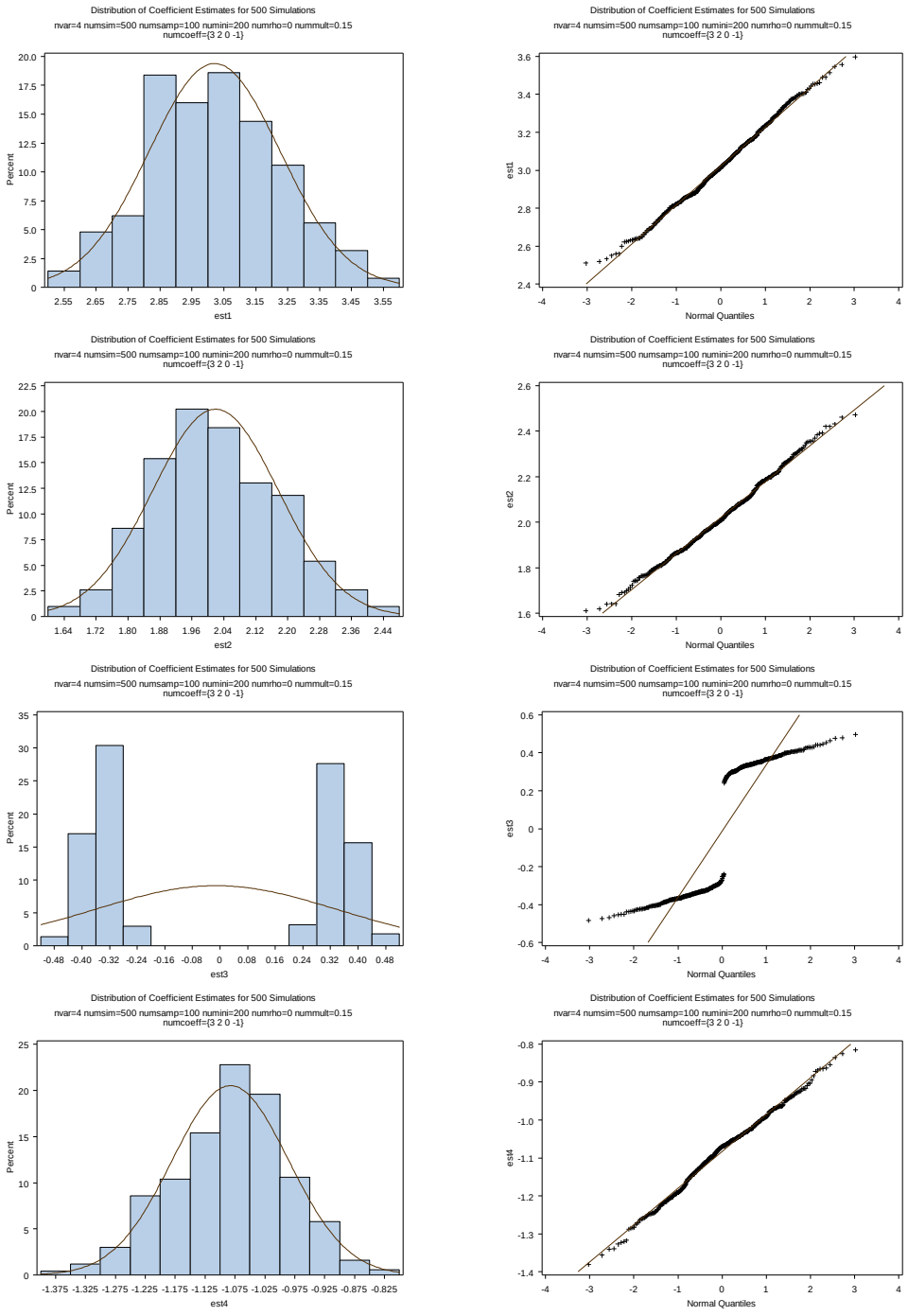

Figure 4.16 Histograms (*left*) and normal QQplots (*right*) of coefficient estimates based on 500 samples with $n = 100$, $\rho = 0$ and $mult = 0.15$

As shown in figure 4.16, for the case of a sample size of 100 and small errors, all plots suggest that the coefficient estimates for X_1, X_2 and X_4 have an

approximate normal distribution. The histogram and normal QQplot of the estimates for the coefficient associated with X_3 ($\hat{a}_3$) show that there is a "hole" at zero. The estimates for the variable having zeros as true coefficient obtained by RMA are very close to zero but never hit the value 0.

Table 4.8 Descriptive Statistics of $\hat{a}_i, i = 1,...,4$ with $n = 100$, $\rho = 0$ and $mult = 0.15$

	$\hat{a}_1$	$\hat{a}_2$	$\hat{a}_3$	$\hat{a}_4$
Mean	3.02	2.02	-0.01	-1.08
StdDev	0.21	0.16	0.35	0.10
Median	3.02	2.01	-0.27	-1.07
Min	2.51	1.61	-0.48	-1.38
Max	3.60	2.47	0.50	-0.81
Q1	2.87	1.91	-0.34	-1.14
Q3	3.16	2.12	0.34	-1.02

As shown in table 4.8, the mean for each estimate is very close to the true coefficient. The mean and median are close for each $\hat{a}_i, i = 1,...,4$ and the variability of each estimate is not large.

500 sample data sets of size 100 are generated, with three correlated variables (X_2, X_3, X_4) that are generated from $MN(\mathbf{0}, \mathbf{P})$ and X_1 that is obtained by the equation

$$3X_1 + 2X_2 + 0X_3 - 1X_4 = 1,$$

with small additive error ($mult = 0.15$) and moderate correlation ($\rho = 0.5$) between all pairs of X_2, X_3 and X_4.

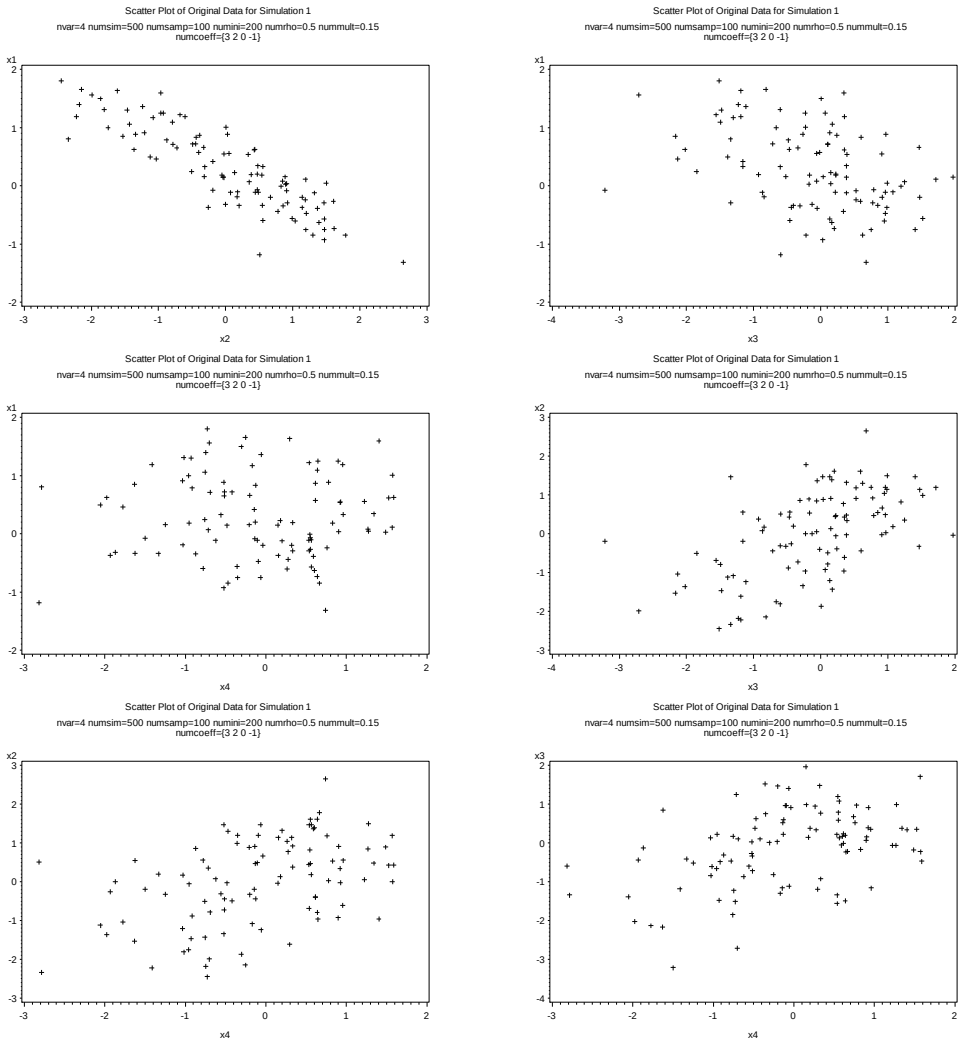

Figure 4.17 Scatter plots of original variables X_i *vs* $X_j, i < j = 1,...,4$ for sample with $n = 100$, $\rho = 0.5$, $mult = 0.15$ and coefficients are {3, 2, 0, -1}

As shown in figure 4.17, all plots show linear trends.

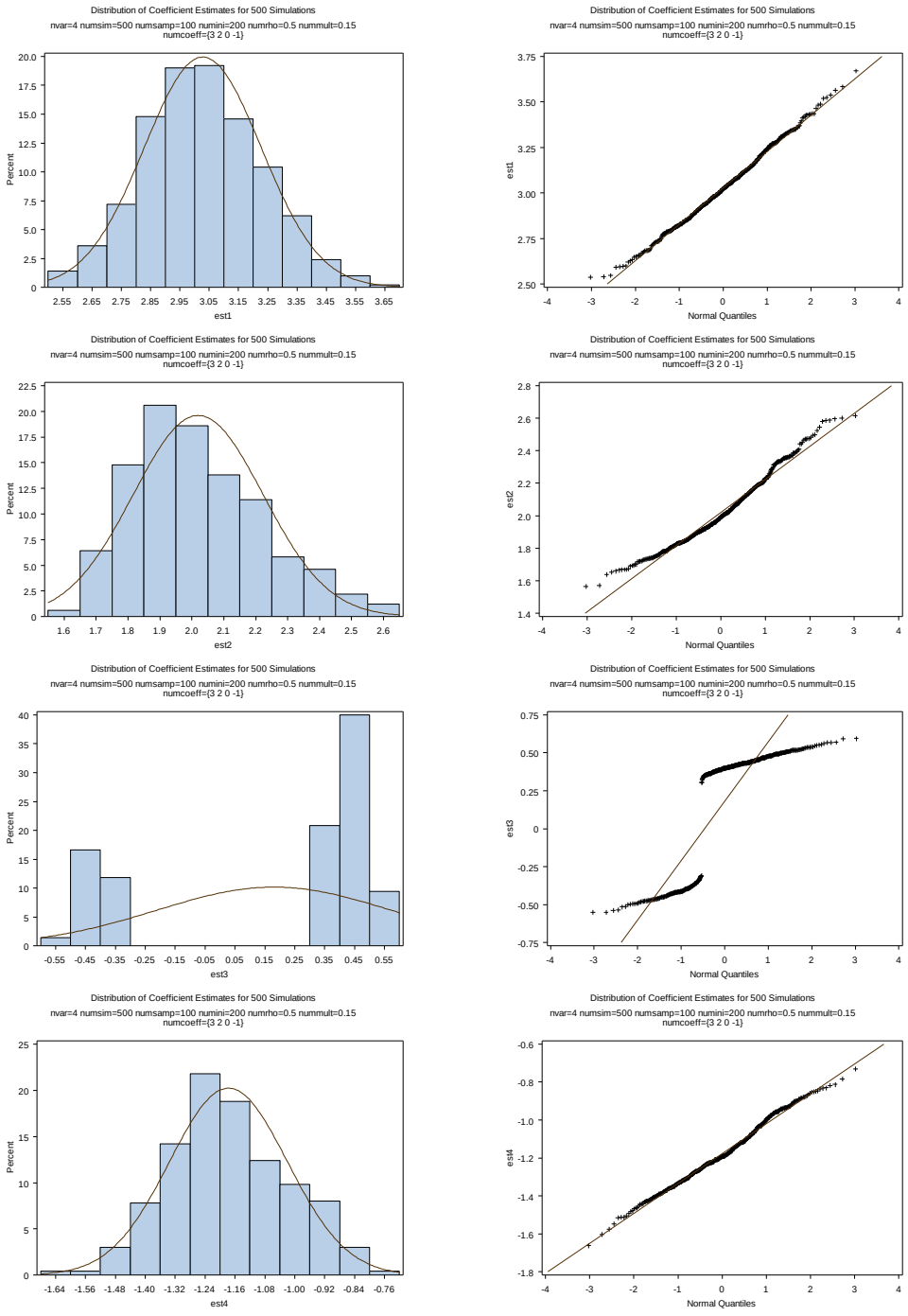

Figure 4.18 Histograms (*left*) and normal QQplots (*right*) of coefficient estimates based on 500 samples with $n = 100$, $\rho = 0.5$ and $mult = 0.15$

As shown in figure 4.18, the histogram and normal QQplot for $\hat{a}_3$ show a "hole" at zero and the histograms for the other coefficients are more skewed than when the variables are uncorrelated.

Table 4.9 Descriptive Statistics of $\hat{a}_i, i = 1,...,4$ with $n = 100$, $\rho = 0.5$ and $mult = 0.15$

	$\hat{a}_1$	$\hat{a}_2$	$\hat{a}_3$	$\hat{a}_4$
Mean	3.03	2.02	0.18	-1.18
StdDev	0.20	0.20	0.39	0.16
Median	3.02	1.99	0.40	-1.19
Min	2.54	1.57	-0.55	-1.66
Max	3.67	2.61	0.59	-0.73
Q1	2.89	1.87	-0.36	-1.28
Q3	3.15	2.15	0.45	-1.07

As shown in table 4.9, the mean for each estimate is close or very close to the true coefficient. The mean and median are close for each $\hat{a}_i, i = 1,...,4$ and the variability of each estimate is not large.

500 sample data sets of size 100 are generated, with three correlated variables (X_2, X_3, X_4) that are generated from $MN(\mathbf{0}, \mathbf{P})$ and X_1 that is obtained by the equation

$$3X_1 + 2X_2 + 0X_3 - 1X_4 = 1,$$

with small additive error ($mult = 0.15$) and large correlation ($\rho = 0.8$) between all pairs of X_2, X_3 and X_4.

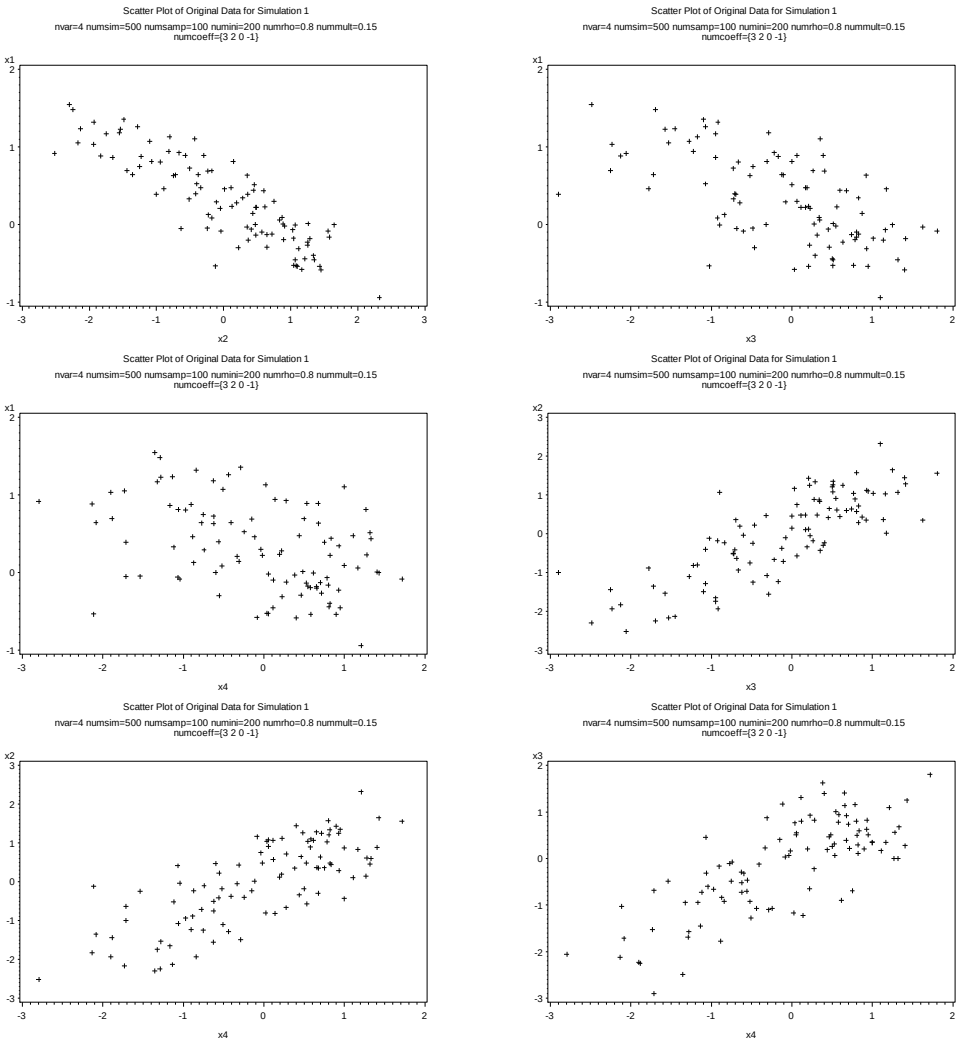

Figure 4.19 Scatter plots of original variables X_i *vs* $X_j, i < j = 1,...,4$ for sample with $n = 100$, $\rho = 0.8$, $mult = 0.15$ and coefficients are {3, 2, 0, -1}

As shown in figure 4.19, all plots show linear trends.

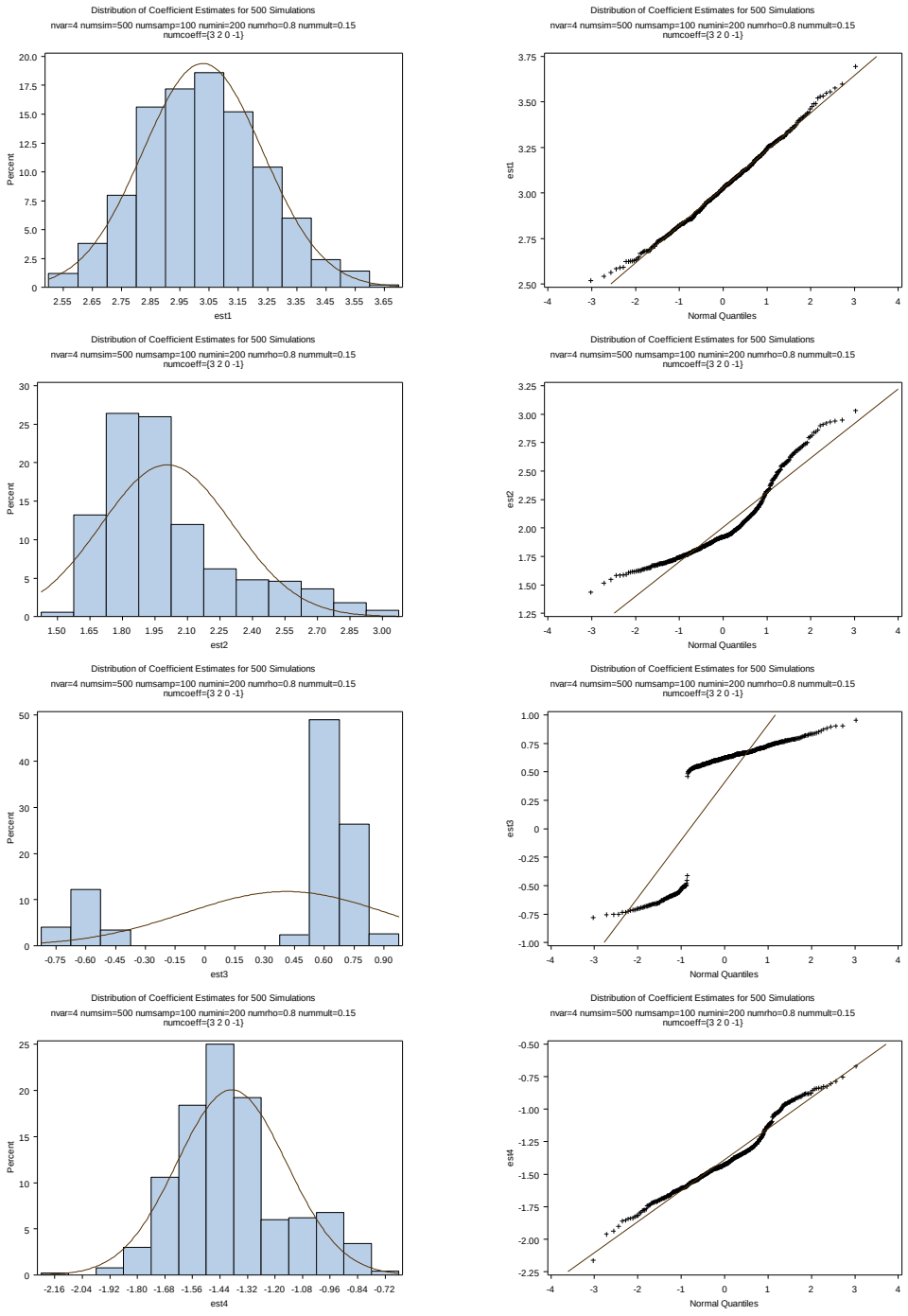

Figure 4.20 Histograms (*left*) and normal QQplots (*right*) of coefficient estimates based on 500 samples with $n = 100$, $\rho = 0.8$ and $mult = 0.15$

As shown in figure 4.20, the histogram and normal QQplot for $\hat{a}_3$ show a "hole" at zero and histograms for the other coefficients seem to be right skewed.

Samples of size 100 with moderate and large errors were also analyzed to investigate the distribution of estimates. The histogram for $\hat{a}_3$ still shows a "hole" at zero. All distributions of other estimates are highly skewed and long-tailed due to the magnitudes of additive errors.

Table 4.10 Descriptive Statistics of $\hat{a}_i, i = 1,...,4$ with $n = 100$, $\rho = 0.8$ and $mult = 0.15$

	$\hat{a}_1$	$\hat{a}_2$	$\hat{a}_3$	$\hat{a}_4$
Mean	3.03	2.01	0.41	-1.39
StdDev	0.21	0.30	0.51	0.24
Median	3.02	1.92	0.62	-1.42
Min	2.52	1.43	-0.78	-2.16
Max	3.69	3.03	0.95	-0.67
Q1	2.87	1.80	0.54	-1.55
Q3	3.16	2.12	0.69	-1.29

As shown in table 4.10, the mean for estimates of a_1 and a_2 are close to the true coefficient, but the estimates for a_3 and a_4 appear to be biased. The mean and median are close for each $\hat{a}_i, i = 1,...,4$ and the variability of each estimate is not large.

Samples of size 20 and 50 with combinations of different errors ($mult = 0.5, 0.8$) and correlations between all pairs of X_2, X_3 and X_4 ($\rho = 0.5, 0.8$) were considered as well. Descriptive statistics for each case are shown in appendix 4.2. The distributions of estimates are more skewed and the variability of estimates is larger than when the sample size is 100 with the corresponding errors and correlations.

4.3 Transformation of Coefficient Estimates

The simulations in the previous sections of Chapter 4 often show that the distributions of parameter estimates are skewed. The log transformation is often useful for transforming skewed distributions so that the transformed data are approximately normally distributed. Clarke (1980) suggested a test statistic based on the log transformed coefficient estimates and the statistic has an asymptotically standard normal distribution. We will use the same simulations discussed previously in this chapter and consider applying the log transformations to the coefficients.

To study the effect of the log transformation we consider only the simulated cases when all coefficient estimates are positive.

Apply the log transformation to parameter estimates for a_1, a_2 and a_3 for sample data with sample size of 20, small error ($mult = 0.15$) and no correlation between variables as shown in figure 4.1 and 4.2, and obtain the following histograms and normal QQplots.

Compared to the figure 4.2, figure 4.21 shows a less skewed distribution, however the transformed coefficients are not normally distributed.

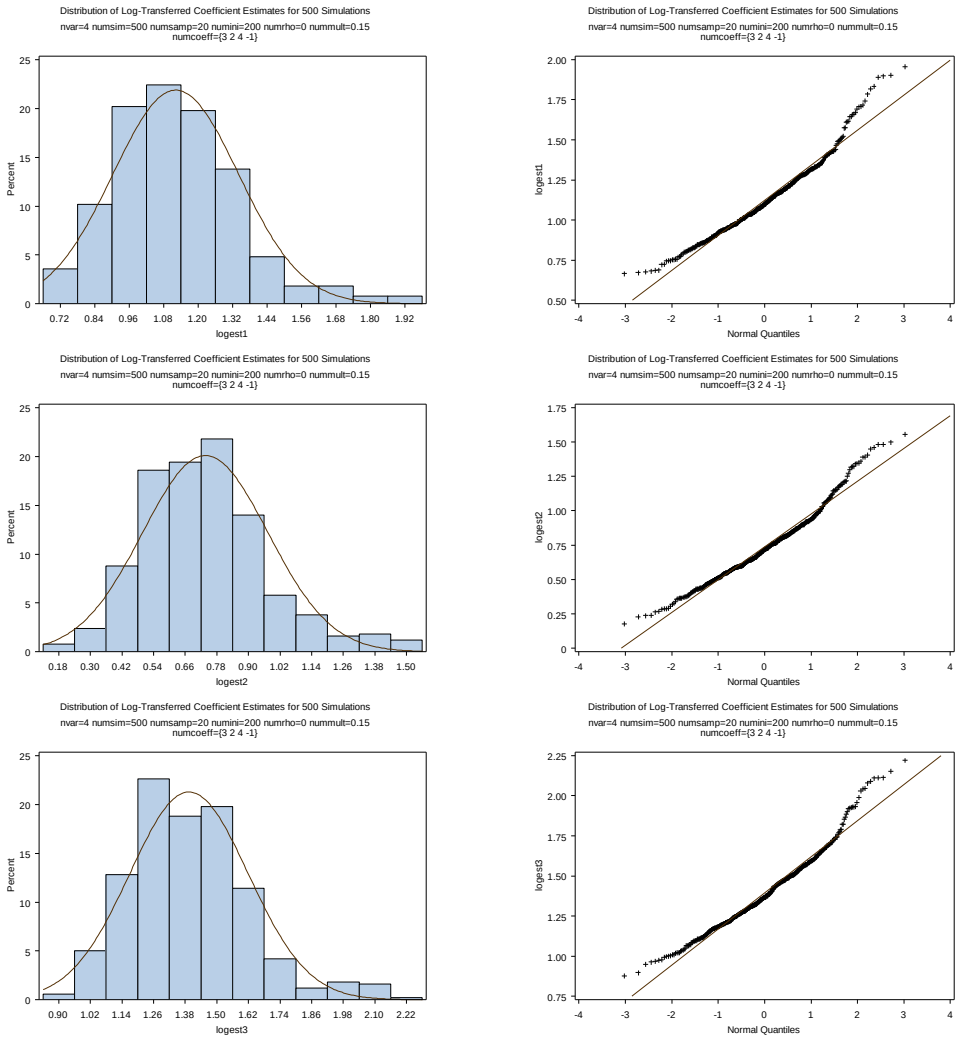

Figure 4.21 Histograms (*left*) and normal QQplots (*right*) of log transformed coefficient estimates for a_1, a_2 and a_3 for samples with $n = 20$, $\rho = 0$, $mult = 0.15$ and coefficients are {3, 2, 4, -1}

Apply the log transformation to estimates of a_1, a_2 and a_3 in sample data with sample size of 100, small error ($mult = 0.15$) and high correlation ($\rho = 0.8$) between variables shown in figure 4.13 and 4.14, and obtain the following histograms and normal QQplots.

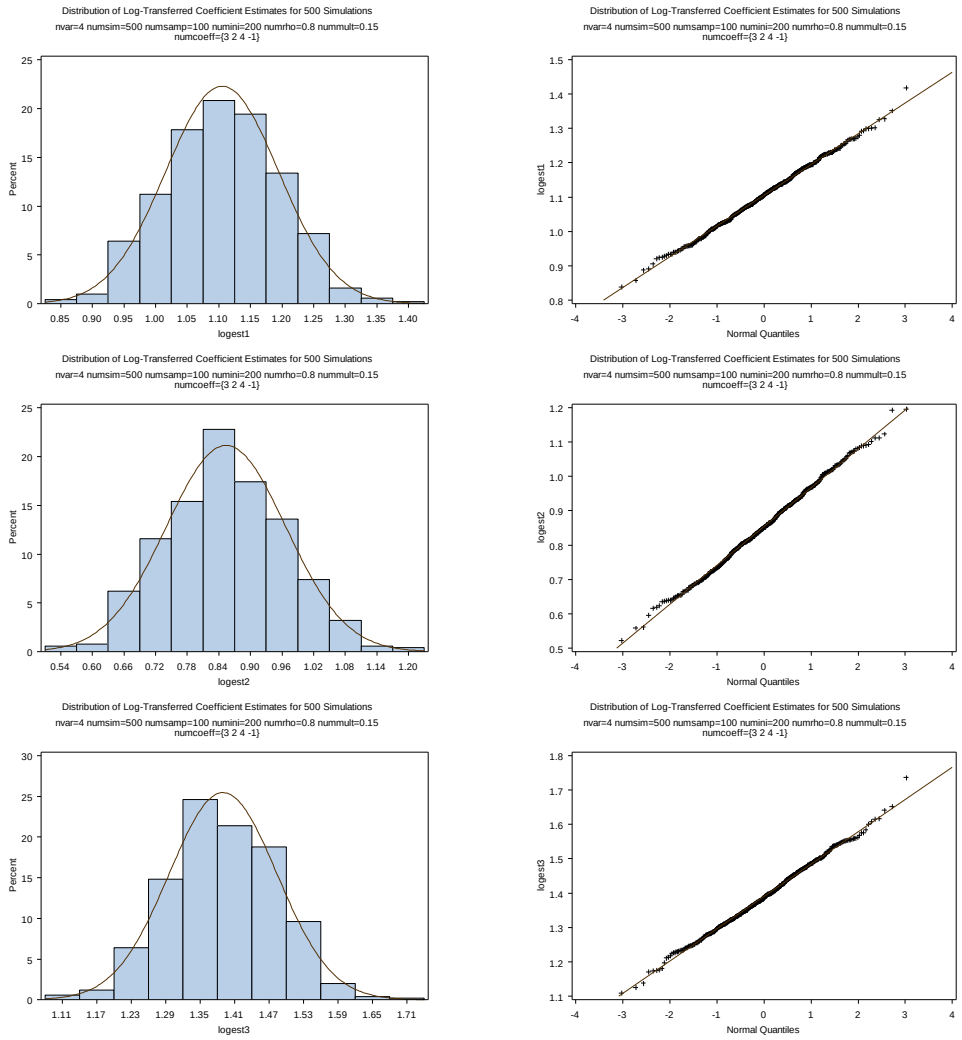

Figure 4.22 Histograms (*left*) and normal QQplots (*right*) of log transformed coefficient estimates for a_1, a_2 and a_3 for samples with $n = 100$, $\rho = 0.8$, $mult = 0.15$ and coefficients are {3, 2, 4, -1}

Figure 4.14 showed a reasonable normal distribution. After the log transformation as shown in figure 4.22, the distributions look even more symmetric and normal. Log transformation is applied to all samples discussed in Section 4.1 for all combination of sample sizes (20, 50 and 100), additive errors ($mult = 0.5, 0.8$) and correlations ($\rho = 0, 0.5, 0.8$) for which all coefficient estimates are positive. The results show that log transformation is helpful when additive errors are small.

However the distributions of log transformed coefficient estimates are still skewed and the log transformed coefficient estimates are not normally distributed. Log transformation is not helpful when additive errors are moderate or large. Further study will be needed to identify other transformations that might yield approximate normal distributions for coefficient estimates.

Chapter 5: INFERENCES

Since there are no closed forms for the parameter estimates, and the distribution of coefficient estimates do not appear to be normal in the many simulations considered in Chapter 4 and since exact and large sample distributions are not available we will consider using resampling methods to obtain inferences on the parameter coefficients. Bootstrapping methods (Efron and Tibshirani 1993) generally require no major assumptions other than simple random sampling and finite variance and have become commonly used for inferences when normal theory methods are not appropriate. Therefore a bootstrapping approach will be adopted to calculate confidence intervals for the coefficients in RMA regression.

5.1 Theory

5.1.1 Parameter Set Up

Previous simulations (Chapter 4) show that the distributions of coefficient estimates are reasonably well behaved if the sample size is large enough and the additive errors of each variable are relatively small. The bootstrap method should be appropriate in this case. However, the bootstrap standard error may not be consistent even for very smooth statistics when the population distribution has very heavy tails (Shao and Tu 1995, chap. 3). Thus bootstrapping methods might not be appropriate for the cases when additive errors are large. Several combinations of sample sizes and additive errors are used in assessing the effectiveness of the bootstrap method for constructing confidence intervals.

For each data set in the study, in order to obtain the confidence intervals for coefficients, the bootstrap method is applied to the original data. Then RMA

regression is performed for each bootstrapped data set and RMA estimates are obtained for coefficients. Since the coefficient estimates do not have normal distributions in general, according to Shao and Tu 1995, chap. 4, to perform a bootstrapping method on non-normal distributed parameters, at least 1000 re-sampling will be needed.

5.1.2 Six Bootstrapping Based Confidence Intervals

A SAS macro %JACKBOOT including %BOOT, %JACK and % BOOTCI ("Jackknife and Bootstrap" 2012) is used in all simulations. The %BOOTCI macro computes several varieties of confidence intervals that are suitable for sampling distributions that are not normal. The six different kinds of confidence intervals are obtained by the % BOOTCI macro are Normal, Percentile (PCTL), BC, BCa, Hybrid and Jackknife.

In the study, $100(1-\alpha)\%$ confidence intervals are considered for the RMA coefficient estimates $a_i, i = 1,..., p$.

(1) Normal bootstrap method

The Normal confidence interval is obtained by the %BOOT macro, which does elementary nonparametric bootstrap analyses for simple random samples and confidence intervals assuming a normal sampling distribution. The confidence interval is written as

$$\left(\hat{a}_i + \hat{\sigma} z_{(\alpha/2)}, \hat{a}_i + \hat{\sigma} z_{(1-\alpha/2)}\right)$$

where $\hat{\sigma}$ is the standard deviation of the bootstrap samples to estimate the standard error of $\hat{a}_i$ and $z_{(k)}$ is the $100k\%$ quantile of a standard normal distribution (Efron and Tibshirani 1993, chap. 13).

(2) Percentile (PCTL) bootstrap method

The percentile method simply uses the $\alpha/2$ and $1-\alpha/2$ percentiles of the bootstrap distribution to define the interval, written as

$$\left(H^{-1}{}_{(\alpha/2)}, H^{-1}{}_{(1-\alpha/2)}\right)$$

where H is the bootstrapping distribution and $H^{-1}_{(k)}$ is its $100k\%$ quantile.

This method performs well for quantiles and for statistics that are unbiased and have a symmetric sampling distribution (Efron and Tibshirani 1993, chap. 13).

(3) BC bootstrap method

The BC method corrects the percentile interval for median bias. The correction is performed by adjusting the percentile points to values other than $\alpha/2$ and $1-\alpha/2$. The confidence interval is written as

$$\left(H^{-1}{}_{(\alpha_1)}, H^{-1}{}_{(\alpha_2)}\right)$$

where H is the bootstrapping distribution and $H^{-1}_{(k)}$ is its $100k\%$ quantile.

The values of α_1, α_2 can be obtained by

$$\alpha_1 = \Phi\left(2\hat{z}_0 + z_{(\alpha/2)}\right) \text{ and } \alpha_2 = \Phi\left(2\hat{z}_0 + z_{(1-\alpha/2)}\right),$$

where Φ is the standard normal cumulative distribution function and $z_{(k)}$ is the $100k\%$ quantile of Φ. The $\hat{z}_0$ is the bias-correction (Efron 1987).

(4) BCa bootstrap method

The BCa method corrects the percentile interval for both bias and skewness which is related to the acceleration estimates. The acceleration can be estimated by the jackknife method which requires extra computation. The confidence interval can be written as

$$\left(H^{-1}{}_{(\alpha_1)}, H^{-1}{}_{(\alpha_2)}\right),$$

where H is the bootstrapping distribution and $H^{-1}_{(k)}$ is its $100k\%$ quantile.

The values of α_1, α_2 can be obtained by

$$\alpha_1 = \Phi\left(\hat{z}_0 + \frac{\hat{z}_0 + z_{(\alpha/2)}}{1-\hat{a}\left(\hat{z}_0 + z_{(\alpha/2)}\right)}\right)$$

$$\alpha_2 = \Phi\left(\hat{z}_0 + \frac{\hat{z}_0 + z_{(1-\alpha/2)}}{1-\hat{a}\left(\hat{z}_0 + z_{(1-\alpha/2)}\right)}\right),$$

where Φ is the standard normal cumulative distribution function and $z_{(k)}$ is the $100k\%$ quantile of Φ. $\hat{z}_0$ is the bias-correction and $\hat{a}(.)$ is the acceleration (Efron and Tibshirani 1993, chap. 14). It should be noted that when $\hat{a}(.)$ is zero, the BCa confidence interval becomes a BC confidence interval.

However, if the acceleration is not estimated accurately, the BCa interval could perform poorly. The length of the BCa interval is not monotonic with respect to the significance level (Hall 1992, pp. 134-135, 137). For large values of the acceleration and significance level, the BCa interval is excessively short.

(5) Hybrid bootstrap method

In the Hybrid method, the bootstrap distribution of $\hat{a}_i^* - \hat{a}_i$ is used to get approximate quantiles of the distribution of $\hat{a}_i - a_i$. This leads to the interval

$$\left(2\hat{a}_i - H^{-1}{}_{(1-\alpha/2)}, 2\hat{a}_i - H^{-1}{}_{(\alpha/2)}\right),$$

where H is the bootstrapping distribution and $H^{-1}_{(k)}$ is its $100k\%$ quantile. (Shao and Tu 1995).

The Hybrid method is the reverse of the percentile method. While the percentile method amplifies bias, the Hybrid method automatically adjusts for bias and skewness. The Hybrid method works well if the standard error of the statistic does not depend on any unknown parameters. Of all the methods in %BOOTCI, the Hybrid method seems to be the least likely to yield spectacularly wrong results, but often suffers from low coverage in relatively easy cases.

It is noted that the width of the confidence interval for the hybrid method is

$$H^{-1}{}_{(1-\alpha/2)} - H^{-1}{}_{(\alpha/2)},$$

which is the same as the confidence interval width (CIW) for the PCTL method. This result will be seen in the simulation that the confidence intervals provided by the Hybrid method are just a shift from that given by PCTL and the CIW are always the same.

(6) Jackknife method

The Jackknife method is not exactly a bootstrap method. It is included in the macro %BOOTCI as one of the approaches to calculate confidence intervals. It is obtained from jackknife samples provided by %JACK and does elementary

analyses for the samples obtained by deleting one observation at a time. The interval can be written as

$$\left(\hat{a}_i + \hat{s}_{Jack} z_{(\alpha/2)}, \hat{a}_i + \hat{s}_{Jack} z_{(1-\alpha/2)}\right)$$

where $\hat{s}_{Jack}$ is the estimate of the standard error of $\hat{a}_i$ obtained from the Jackknife samples and $z_{(k)}$ is the $100k\%$ quantile of a standard normal distribution (Shao and Tu 1995).

The PCTL, BC and BCa methods are closely related and they are equivariant under transformation of the parameters (Efron and Tibshirani 1993).

In terms of the accuracy of one-sided confidence intervals, the BCa method is better than the PCTL, BC, Hybrid and Normal method. However, since the use of the BCa depends on the estimate of acceleration, the other three bootstrap methods are still popular in use (Shao and Tu 1995).

5.1.3 Hit Rate and Hypothesis Test

It is of interest to determine how well the bootstrap methods work for obtaining confidence intervals for the coefficients in RMA regression.

Let γ_{ij} denote the true confidence coefficient for the confidence interval obtained for a_i by using the j^{th} bootstrap confidence interval method ($i = 1,..., p$, $j = 1,...,6$). An estimate of γ_{ij} for a specified model and assumptions can be obtained by simulating N data sets, obtaining the bootstrap confidence intervals by the various methods and determining the proportion of times each method produces a confidence interval that contains the true parameter value. Call this

proportion the hit rate and let hr_{ij} be the hit rate for the i^{th} coefficient for the j^{th} bootstrap method ($i = 1,..., p$, $j = 1,..., 6$). A test statistic (h_{ij}) and the p-value associated with hr_{ij} are obtained by performing the single-proportion z-test on the hr_{ij} for each confidence interval for nominal confidence coefficient of $\gamma_{ij} = 0.95$. The hypothesis tests that are proposed based on the hit rate are:

$$H_0 : \gamma_{ij} \geq 0.95 \text{ vs } H_a : \gamma_{ij} < 0.95,\ i = 1,..., p,\ j = 1,..., 6$$

The test statistic h_{ij} is defined as

$$h_{ij} = \frac{hr_{ij} - 0.95}{\sqrt{(0.95)(1-0.95)/N}}.$$

Then h_{ij} follows an approximately standard normal distribution and H_0 is rejected if $h_{ij} < z_{(\alpha)}$ where $z_{(k)}$ is the $100k\%$ quantile of a standard normal distribution. In general, the method that results in the fewest rejections of the null hypothesis is preferred.

Another way to compare the six methods would be sorting the $hr's$ from the largest to the smallest values and obtain the ranks ($r_{1j},..., r_{6j}$, $j = 1,..., p$) of $hr's$, where an average rank is taken if there is a tie. The averages of $r_{ij}'s$ across all simulations ($\bar{R}_{hi} = \sum_{j=1}^{p} r_{ij} / p, i = 1,..., 6$) are found and compared. Methods with smaller $\bar{R}_{hi}$ are preferred.

5.1.4 Confidence Interval Width

Alternatively, to evaluate the quality of the six different bootstrap confidence intervals, the confidence interval width (CIW) is calculated and compared. To compare the CIW for the six confidence interval methods for each data set from the simulation, the rank of CIW ($R_{1j},...,R_{6j},\ j=1,...,p$) sorted from the smallest to the largest is used, where an average rank is taken if there is a tie. The averages of R_{ij}'s across all simulations ($\bar{R}_{Wi}=\sum_{j=1}^{p}R_{ij}/p, i=1,...,6$) are found and compared. Methods with smaller $\bar{R}_{Wi}$ are preferred.

5.2 Simulation

As in the previous examples in Chapter 4, sample data sets with different combination of parameters, such as sample size, additive errors, and correlation between variables are generated either from an equation or from a multivariate normal distribution. Bootstrapping methods are applied to the sample data set and the six 95% confidence intervals provided by the SAS macro %BOOTCI are obtained. The number of simulations (N) is 500 and the bootstrapping number is 1000.

5.2.1 Data Generated by an Equation without Zero Coefficients

Sample data sets of sizes 20, 50 and 100 are generated. Two variables (X_2, X_3) are generated from $MN(\mathbf{0}, \mathbf{P})$ with correlations of 0, 0.5 and 0.8, and X_1 is obtained by the equation

$$3X_1+2X_2+1X_3=1,$$

with small, moderate and large additive errors ($mult = 0.15, 0.5, 0.8$).

There are 27*500 data sets originally generated from the equation. For each data set, 1000 bootstrap samples are obtained and the six confidence intervals, the hit rates with ranks and the CIW with ranks are obtained. In addition, for each confidence interval method, 27 hypothesis tests described in Section 5.1.3 are performed and 27 p -values are calculated. More details, such as the hit rate and its rank, the CIW and its rank of each coefficient and bootstrap are provided in Appendix 5.1.

Table 5.1 Overall comparison of six confidence interval methods with hit rates. The $\%p_i, i = 1,2,3$ is the percentage of the number of p -values that are greater than or equal to 0.05 to the total number of p -values calculated (27) associated with $\hat{a}_i$.

	$\%p_1$	$\%p_2$	$\%p_3$
Normal	100.0%	77.8%	81.5%
BC	48.1%	33.3%	0.0%
BCA	44.4%	33.3%	0.0%
Hybrid	11.1%	14.8%	77.8%
PCTL	77.8%	74.1%	33.3%
Jackknife	14.8%	18.5%	44.4%

As table 5.1 shows, the Normal method has the best hit rate among all methods. As shown in table 5.2, the hit rates for each a_i do not have a clear trend except that the Normal method has the highest hit rates. The average ranks of hit rates analyzed over a_1, a_2, a_3 are different as the sample size changes. In general, more Normal confidence intervals contain the true coefficients and the average rank of the hit rates is smaller compared to other methods. However, the widths of the confidence intervals need to be considered in determining which method is better.

Table 5.2 Hit rate analysis by sample size. The $\%p_i, i=1,2,3$ is the percentage of the number of p-values that are greater than or equal to 0.05 to the total number of p-values calculated (9) associated with $\hat{a}_i$.

$n=20$	$\%p_1$	$\%p_2$	$\%p_3$	$\bar{R}_{hi}$
Normal	100.0%	77.8%	100.0%	4.00
BC	0.0%	0.0%	0.0%	14.00
BCA	0.0%	0.0%	0.0%	12.56
Hybrid	0.0%	0.0%	77.8%	11.67
PCTL	66.7%	66.7%	66.7%	8.11
Jackknife	11.1%	11.1%	33.3%	12.11
$n=50$	$\%p_1$	$\%p_2$	$\%p_3$	$\bar{R}_{hi}$
Normal	100.0%	77.8%	66.7%	6.33
BC	100.0%	66.7%	0.0%	12.00
BCA	100.0%	66.7%	0.0%	11.11
Hybrid	33.3%	33.3%	66.7%	12.44
PCTL	100.0%	77.8%	0.0%	6.89
Jackknife	33.3%	33.3%	44.4%	12.33
$n=100$	$\%p_1$	$\%p_2$	$\%p_3$	$\bar{R}_{hi}$
Normal	100.0%	77.8%	66.7%	6.00
BC	100.0%	66.7%	0.0%	11.00
BCA	100.0%	66.7%	0.0%	11.11
Hybrid	33.3%	33.3%	66.7%	11.78
PCTL	100.0%	77.8%	0.0%	8.22
Jackknife	33.3%	33.3%	44.4%	11.89

Table 5.3 Average CIW and CIW rank analysis by sample size

$n=20$	CIW for a_1	CIW for a_2	CIW for a_3	$\bar{R}_{Wi}$
Normal	1138.98	964.99	898.08	14.44
BC	2358.71	1892.98	1538.24	14.56
BCA	2106.06	1795.73	1500.77	12.78
Hybrid	35.49	30.26	26.43	5.11
PCTL	35.49	30.26	26.43	5.11
Jackknife	421.25	378.06	288.32	8.00
$n=50$	CIW for a_1	CIW for a_2	CIW for a_3	$\bar{R}_{Wi}$
Normal	492.46	476.23	416.23	14.78
BC	554.07	535.56	456.73	11.78
BCA	535.97	530.18	465.15	12.22
Hybrid	22.46	20.11	17.17	4.56
PCTL	22.46	20.11	17.17	4.56
Jackknife	364.20	307.27	258.87	12.11
$n=100$	CIW for a_1	CIW for a_2	CIW for a_3	$\bar{R}_{Wi}$
Normal	309.42	310.83	254.80	14.78
BC	154.07	219.07	197.31	12.44
BCA	196.90	248.00	214.01	13.00
Hybrid	11.18	10.19	9.11	4.89
PCTL	11.18	10.19	9.11	4.89
Jackknife	48.75	46.27	35.37	10.00

As shown in table 5.3, when looking at a certain method, the CIW for each coefficient decreases as the sample size increases. The method Normal, BC and BCA have larger CIW compared to other methods. The Hybrid and PCTL methods have the same and smallest CIW among all methods. The Jackknife method also provides smaller CIWs. Therefore, due to its very wide confidence interval the Normal method does not provide the best confidence intervals. Combining results from hit rates and CIW rank analysis, the PCTL actually provides reasonable results for bootstrapped confidence intervals.

Table 5.4 Hit rate analysis by additive error multiplier. The $\%p_i, i=1,2,3$ is the percentage of the number of p -values that are greater than or equal to 0.05 to the total number of p -values calculated (9) associated with $\hat{a}_i$.

$mult=0.15$	$\%p_1$	$\%p_2$	$\%p_3$	$\bar{R}_{hi}$
Normal	100.0%	33.3%	44.4%	6.56
BC	44.4%	22.2%	0.0%	10.67
BCA	33.3%	22.2%	0.0%	10.44
Hybrid	33.3%	11.1%	55.6%	11.89
PCTL	33.3%	22.2%	0.0%	11.89
Jackknife	44.4%	33.3%	44.4%	8.56
$mult=0.5$	$\%p_1$	$\%p_2$	$\%p_3$	$\bar{R}_{hi}$
Normal	100.0%	100.0%	100.0%	5.11
BC	33.3%	22.2%	0.0%	11.89
BCA	33.3%	22.2%	0.0%	11.22
Hybrid	0.0%	22.2%	77.8%	12.89
PCTL	100.0%	100.0%	66.7%	6.44
Jackknife	0.0%	11.1%	33.3%	13.78
$mult=0.8$	$\%p_1$	$\%p_2$	$\%p_3$	$\bar{R}_{hi}$
Normal	100.0%	100.0%	100.0%	4.67
BC	33.3%	22.2%	0.0%	14.44
BCA	33.3%	22.2%	0.0%	13.11
Hybrid	0.0%	22.2%	77.8%	11.11
PCTL	100.0%	100.0%	66.7%	4.89
Jackknife	0.0%	11.1%	33.3%	14.00

As shown in table 5.4, the hit rates for each a_i do not have a clear trend except that Normal method has the highest hit rates. The average ranks of hit rates analyzed over a_1, a_2, a_3 are different as the error magnitude changes. More Normal confidence intervals contain the true coefficients and the average rank of the hit rates is the smallest compared to other methods. However, the widths of the confidence intervals need to be considered as well to conclude which method is better.

Table 5.5 Average CIW and CIW rank analysis by additive error multiplier

$mult = 0.15$	CIW for a_1	CIW for a_2	CIW for a_3	$\bar{R}_{Wi}$
Normal	2.77	2.61	1.81	14.00
BC	1.22	0.97	0.67	9.44
BCA	1.46	1.30	0.81	9.67
Hybrid	1.21	0.95	0.67	7.11
PCTL	1.21	0.95	0.67	7.11
Jackknife	1.17	0.90	0.62	12.67
$mult = 0.5$	CIW for a_1	CIW for a_2	CIW for a_3	$\bar{R}_{Wi}$
Normal	454.48	389.13	305.84	15.67
BC	505.14	480.92	415.93	13.78
BCA	359.84	547.78	425.50	13.56
Hybrid	24.36	20.72	17.09	4.44
PCTL	24.36	20.72	17.09	4.44
Jackknife	286.30	279.68	176.31	8.11
$mult = 0.8$	CIW for a_1	CIW for a_2	CIW for a_3	$\bar{R}_{Wi}$
Normal	1483.61	1360.32	1261.47	14.33
BC	2560.50	2165.72	1775.68	15.56
BCA	2477.62	2024.83	1753.62	14.78
Hybrid	43.55	38.89	34.95	3.00
PCTL	43.55	38.89	34.95	3.00
Jackknife	546.73	451.02	405.62	9.33

As shown in table 5.5, when looking at a certain method, the CIW for each coefficient estimate increases as the errors increase. When the error is small ($mult = 0.15$), all methods have reasonably small CIWs. However when the error term becomes larger, the CIWs become very wide compared to the magnitude of original variables. The Normal, BC and BCA methods have larger CIW compared to other methods. The Hybrid and PCTL methods have the same and smallest CIW among all methods. Therefore, due to its very wide confidence interval the Normal method does not provide the best confidence intervals. Combining results from hit rates and CIW rank analysis, the PCTL actually provides reasonable results for bootstrapped confidence intervals.

Table 5.6 Hit rate analysis by correlation between X_2 and X_3. The $\%p_i, i=1,2,3$ is the percentage of the number of p-values that are greater than or equal to 0.05 to the total number of p-values calculated (9) associated with $\hat{a}_i$.

$\rho=0$	$\%p_1$	$\%p_2$	$\%p_3$	$\overline{R}_{hi}$
Normal	100.0%	100.0%	88.9%	6.22
BC	44.4%	55.6%	0.0%	12.22
BCA	44.4%	55.6%	0.0%	11.56
Hybrid	11.1%	22.2%	77.8%	12.89
PCTL	77.8%	88.9%	33.3%	7.67
Jackknife	22.2%	55.6%	66.7%	10.44
$\rho=0.5$	$\%p_1$	$\%p_2$	$\%p_3$	$\overline{R}_{hi}$
Normal	100.0%	66.7%	77.8%	5.22
BC	44.4%	11.1%	0.0%	12.44
BCA	44.4%	11.1%	0.0%	11.00
Hybrid	11.1%	22.2%	66.7%	12.22
PCTL	77.8%	66.7%	33.3%	8.22
Jackknife	11.1%	0.0%	11.1%	12.11
$\rho=0.8$	$\%p_1$	$\%p_2$	$\%p_3$	$\overline{R}_{hi}$
Normal	100.0%	66.7%	77.8%	4.89
BC	44.4%	11.1%	0.0%	12.33
BCA	44.4%	11.1%	0.0%	12.22
Hybrid	11.1%	22.2%	66.7%	10.78
PCTL	77.8%	66.7%	33.3%	7.33
Jackknife	11.1%	0.0%	11.1%	13.78

As shown in table 5.6, the Normal and PCTL methods have higher hit rates for each a_i. The average ranks of hit rates analyzed over a_1, a_2, a_3 are different as the correlation changes. In general, more Normal and PCTL confidence intervals contain the true coefficients and the average ranks of the hit rates are smaller compared to other methods. However, the widths of the confidence intervals need to be considered as well to conclude which method is better.

Table 5.7 Average CIW and CIW rank analysis by correlation between X_2 and X_3

$\rho = 0$	CIW for a_1	CIW for a_2	CIW for a_3	$\bar{R}_{Wi}$
Normal	813.78	682.65	664.61	15.11
BC	1088.23	773.78	429.31	13.56
BCA	985.85	645.19	377.10	11.00
Hybrid	22.60	18.08	15.18	4.67
PCTL	22.60	18.08	15.18	4.67
Jackknife	375.86	299.52	217.28	11.00
$\rho = 0.5$	CIW for a_1	CIW for a_2	CIW for a_3	$\bar{R}_{Wi}$
Normal	549.48	436.33	378.93	14.44
BC	1412.43	1103.18	1039.31	12.89
BCA	1346.13	1021.45	998.85	13.00
Hybrid	22.48	18.70	16.05	5.33
PCTL	22.48	18.70	16.05	5.33
Jackknife	206.61	173.38	139.32	9.00
$\rho = 0.8$	CIW for a_1	CIW for a_2	CIW for a_3	$\bar{R}_{Wi}$
Normal	577.59	633.07	525.57	14.44
BC	566.20	770.65	723.65	12.33
BCA	506.95	907.27	803.97	14.00
Hybrid	24.04	23.77	21.48	4.56
PCTL	24.04	23.77	21.48	4.56
Jackknife	251.73	258.69	225.96	10.11

As shown in table 5.7, when looking at a certain method, the CIW for each coefficient estimate does not have clear trends as the correlation changes. The Normal, BC and BCA methods have larger CIWs compared to other methods. The Hybrid and PCTL methods have the same and smallest CIW among all methods. The Jackknife method also provides smaller CIWs. Therefore, due to its very wide confidence interval the Normal method does not provide the best confidence intervals. Combining results from hit rates and CIW rank analysis, the PCTL actually provides reasonable results for bootstrapped confidence intervals.

Hence, considering comprehensive results about hit rates and CIWs overall, by sample size, by error magnitudes and by correlation between X_2 and X_3, the Normal and PCTL methods tend to provide better hit rates and the PCTL and Hybrid methods tend to provide smaller CIWs. Which method is more appropriate for the model is a judgment call; however, the PCTL method is more likely to provide a good confidence interval for the coefficients according to the simulation results when data are generated from an equation.

5.2.2 Data Generated by an Equation with Zero Coefficients

Sample data sets of sizes 20, 50 and 100 are generated. Two variables (X_2, X_3) are generated from $MN(\mathbf{0}, \mathbf{P})$ with correlations of 0, 0.5 and 0.8, X_1 is obtained by the equation

$$3X_1 + 2X_2 + 0X_3 = 1,$$

with small, moderate and large additive errors ($mult = 0.15, 0.5, 0.8$).

There are 27*500 data sets originally generated from the equation. For each data set, 1000 bootstrap samples are obtained and the six bootstrap confidence intervals, the hit rates with ranks and the CIW with ranks are obtained. In addition, for each confidence interval method, 27 hypothesis tests described in Section 5.1.3 are performed and 27 p -values are calculated.

Table 5.8 Overall comparison of six confidence interval methods with hit rates. The $\%p_i, i=1,2,3$ is the percentage of the number of p-values that are greater than or equal to 0.05 to the total number of p-values calculated (27) associated with $\hat{a}_i$.

	$\%p_1$	$\%p_2$	$\%p_3$
Normal	100.0%	81.5%	55.6%
BC	66.7%	37.0%	7.4%
BCA	55.6%	33.3%	11.1%
Hybrid	0.0%	14.8%	22.2%
PCTL	100.0%	92.6%	74.1%
Jackknife	0.0%	0.0%	0.0%

As table 5.8 shows, the PCTL method has the best hit rate among all methods.

Table 5.9 Hit rate analysis by sample size. The $\%p_i, i=1,2,3$ is the percentage of the number of p-values that are greater than or equal to 0.05 to the total number of p-values calculated (9) associated with $\hat{a}_i$.

$n=20$	$\%p_1$	$\%p_2$	$\%p_3$	$\bar{R}_{hi}$
Normal	100.0%	88.9%	66.7%	5.89
BC	33.3%	11.1%	0.0%	13.11
BCA	22.2%	22.2%	0.0%	12.33
Hybrid	0.0%	0.0%	22.2%	12.33
PCTL	100.0%	100.0%	66.7%	3.78
Jackknife	0.0%	0.0%	0.0%	15.44
$n=50$	$\%p_1$	$\%p_2$	$\%p_3$	$\bar{R}_{hi}$
Normal	100.0%	77.8%	33.3%	6.33
BC	100.0%	55.6%	11.1%	9.89
BCA	100.0%	44.4%	11.1%	10.78
Hybrid	0.0%	44.4%	11.1%	14.00
PCTL	100.0%	77.8%	55.6%	4.00
Jackknife	0.0%	0.0%	0.0%	17.44
$n=100$	$\%p_1$	$\%p_2$	$\%p_3$	$\bar{R}_{hi}$
Normal	100.0%	77.8%	33.3%	7.78
BC	100.0%	55.6%	11.1%	9.11
BCA	100.0%	44.4%	11.1%	9.89
Hybrid	0.0%	44.4%	11.1%	14.22
PCTL	100.0%	77.8%	55.6%	4.11
Jackknife	0.0%	0.0%	0.0%	17.11

As shown in table 5.9, the hit rates for each a_i improve as the sample size increases from 20 to 50. The average ranks of hit rates analyzed over a_1, a_2, a_3 are slightly different as the sample size changes, and the Jackknife method seems to provide the worst hit rates while PCTL is the best one in general. However, the widths of the confidence intervals need to be considered as well to conclude which method is better.

Table 5.10 Average CIW and CIW rank analysis by sample size

$n=20$	CIW for a_1	CIW for a_2	CIW for a_3	$\bar{R}_{Wi}$
Normal	1639.91	1555.65	1643.45	14.44
BC	3427.76	3060.18	2134.96	14.78
BCA	3684.86	2780.82	2526.80	15.44
Hybrid	36.56	32.07	25.30	3.11
PCTL	36.59	32.07	25.30	3.22
Jackknife	513.19	418.83	328.40	9.11
$n=50$	CIW for a_1	CIW for a_2	CIW for a_3	$\bar{R}_{Wi}$
Normal	1058.78	968.37	742.64	13.78
BC	4151.96	3974.84	3277.74	12.22
BCA	4085.69	3880.60	3339.94	14.78
Hybrid	26.16	24.79	19.26	3.33
PCTL	26.16	24.79	19.26	3.33
Jackknife	403.60	395.65	294.56	12.56
$n=100$	CIW for a_1	CIW for a_2	CIW for a_3	$\bar{R}_{Wi}$
Normal	274.06	257.84	202.31	16.22
BC	166.46	158.47	142.14	9.67
BCA	196.19	230.41	216.43	13.89
Hybrid	12.81	12.63	10.96	5.11
PCTL	12.81	12.63	10.96	5.11
Jackknife	63.95	61.16	68.02	10.00

As shown in table 5.10, as discussed in the previous section, the Normal, BC and BCA methods have larger CIWs compared to other methods. The Hybrid and PCTL methods have the same and smallest CIW among all methods. Therefore,

combining results from hit rates and CIW rank analysis, the PCTL method provides reasonable results for bootstrapped confidence intervals.

Table 5.11 Hit rate analysis by additive error multiplier. The $\%p_i, i=1,2,3$ is the percentage of the number of p -values that are greater than or equal to 0.05 to the total number of p -values calculated (9) associated with $\hat{a}_i$.

$mult = 0.15$	$\%p_1$	$\%p_2$	$\%p_3$	$\bar{R}_{hi}$
Normal	100.0%	44.4%	0.0%	9.44
BC	100.0%	55.6%	11.1%	7.67
BCA	77.8%	33.3%	22.2%	9.11
Hybrid	0.0%	0.0%	0.0%	16.00
PCTL	100.0%	88.9%	44.4%	3.33
Jackknife	0.0%	0.0%	0.0%	17.00
$mult = 0.5$	$\%p_1$	$\%p_2$	$\%p_3$	$\bar{R}_{hi}$
Normal	100.0%	100.0%	100.0%	5.33
BC	33.3%	22.2%	0.0%	11.11
BCA	33.3%	33.3%	0.0%	10.89
Hybrid	0.0%	22.2%	44.4%	13.22
PCTL	100.0%	100.0%	100.0%	4.78
Jackknife	0.0%	0.0%	0.0%	17.11
$mult = 0.8$	$\%p_1$	$\%p_2$	$\%p_3$	$\bar{R}_{hi}$
Normal	100.0%	100.0%	100.0%	5.22
BC	33.3%	22.2%	0.0%	13.33
BCA	33.3%	33.3%	0.0%	13.00
Hybrid	0.0%	22.2%	44.4%	11.33
PCTL	100.0%	100.0%	100.0%	3.78
Jackknife	0.0%	0.0%	0.0%	15.89

As shown in table 5.11, the hit rates for each a_i do not have a clear trend as the error multiplier changes except that the PCTL and Normal methods have the highest hit rates in most of the cases. The average ranks of hit rates analyzed over a_1, a_2, a_3 are different as the error magnitude changes. More Normal and PCTL confidence intervals contain the true coefficients and the average ranks of the hit rates for these methods are the smallest compared to other methods. However, the

widths of the confidence intervals need to be considered as well to conclude which method is better.

Table 5.12 Average CIW and CIW rank analysis by additive error multiplier

$mult = 0.15$	CIW for a_1	CIW for a_2	CIW for a_3	$\bar{R}_{Wi}$
Normal	5.13	4.61	3.04	14.56
BC	3.16	2.62	2.32	8.11
BCA	4.15	3.75	3.03	13.22
Hybrid	1.67	1.64	1.43	4.56
PCTL	1.67	1.64	1.43	4.56
Jackknife	1.91	2.23	2.75	15.00
$mult = 0.5$	CIW for a_1	CIW for a_2	CIW for a_3	$\bar{R}_{Wi}$
Normal	1335.23	1308.56	1377.93	16.78
BC	782.93	879.73	624.55	13.22
BCA	770.68	767.91	708.81	15.00
Hybrid	28.13	25.62	18.72	3.67
PCTL	28.17	25.62	18.72	3.78
Jackknife	199.51	151.44	124.27	7.67
$mult = 0.8$	CIW for a_1	CIW for a_2	CIW for a_3	$\bar{R}_{Wi}$
Normal	1632.39	1468.69	1207.44	13.11
BC	6960.08	6311.15	4927.97	15.33
BCA	7191.91	6120.17	5371.33	15.89
Hybrid	45.73	42.22	35.37	3.33
PCTL	45.73	42.22	35.37	3.33
Jackknife	779.30	721.97	563.97	9.00

As shown in table 5.12, and as discussed in the previous section, the CIW for each coefficient obviously increases as the errors increase. When the error is small ($mult = 0.15$), all methods have reasonably small CIWs. However when the error terms become larger, the CIWs become very large compared to the magnitude of the original coefficients. The Normal, BC and BCA methods have larger CIWs compared to other methods. The Hybrid and PCTL methods have the same and smallest CIW among all methods. The Jackknife method also provides smaller CIWs. Combining results from hit rates and CIW rank analysis, the PCTL actually provides reasonable results for bootstrapped confidence intervals.

Table 5.13 Hit rate analysis by correlation between X_2 and X_3. The $\%p_i, i=1,2,3$ is the percentage of the number of p-values that are greater than or equal to 0.05 to the total number of p-values calculated (9) associated with $\hat{a}_i$.

$\rho = 0$	$\%p_1$	$\%p_2$	$\%p_3$	$\overline{R}_{hi}$
Normal	100.0%	100.0%	55.6%	6.67
BC	66.7%	66.7%	11.1%	10.22
BCA	44.4%	66.7%	22.2%	10.67
Hybrid	0.0%	11.1%	11.1%	14.33
PCTL	100.0%	100.0%	88.9%	3.67
Jackknife	0.0%	0.0%	0.0%	16.78
$\rho = 0.5$	$\%p_1$	$\%p_2$	$\%p_3$	$\overline{R}_{hi}$
Normal	100.0%	66.7%	55.6%	6.78
BC	66.7%	11.1%	0.0%	10.44
BCA	55.6%	11.1%	0.0%	11.00
Hybrid	0.0%	22.2%	33.3%	13.22
PCTL	100.0%	77.8%	66.7%	4.00
Jackknife	0.0%	0.0%	0.0%	17.11
$\rho = 0.8$	$\%p_1$	$\%p_2$	$\%p_3$	$\overline{R}_{hi}$
Normal	100.0%	66.7%	55.6%	6.56
BC	66.7%	11.1%	0.0%	11.44
BCA	55.6%	11.1%	0.0%	11.33
Hybrid	0.0%	22.2%	33.3%	13.00
PCTL	100.0%	77.8%	66.7%	4.22
Jackknife	0.0%	0.0%	0.0%	16.11

As shown in table 5.13, the Normal and PCTL methods have higher hit rates for each a_i across all values of ρ. The average ranks of hit rates analyzed over a_1, a_2, a_3 are different as the correlation changes. In general, more Normal and PCTL confidence intervals contain the true coefficients and the average ranks of the hit rates are smaller compared to other methods. However, the widths of the confidence intervals need to be considered as well to conclude which method is better.

Table 5.14 Average CIW and CIW rank analysis by correlation between X_2 and X_3

$\rho = 0$	CIW for a_1	CIW for a_2	CIW for a_3	$\bar{R}_{Wi}$
Normal	1064.71	919.29	731.03	15.11
BC	4076.51	3856.68	3082.62	11.56
BCA	4315.83	3815.44	3578.39	14.33
Hybrid	24.26	19.68	15.33	4.22
PCTL	24.26	19.68	15.33	4.22
Jackknife	402.46	307.64	224.07	10.56
$\rho = 0.5$	CIW for a_1	CIW for a_2	CIW for a_3	$\bar{R}_{Wi}$
Normal	839.30	681.73	503.63	14.56
BC	2817.26	2262.34	1643.96	12.33
BCA	2714.54	2211.75	1735.90	14.89
Hybrid	24.93	22.38	17.28	3.89
PCTL	24.93	22.38	17.28	3.89
Jackknife	236.38	219.40	161.71	10.44
$\rho = 0.8$	CIW for a_1	CIW for a_2	CIW for a_3	$\bar{R}_{Wi}$
Normal	1068.75	1180.84	1353.74	14.78
BC	852.40	1074.47	828.27	12.78
BCA	936.36	864.65	768.89	14.89
Hybrid	26.33	27.42	22.91	3.44
PCTL	26.37	27.42	22.91	3.56
Jackknife	341.89	348.60	305.20	10.67

As shown in table 5.14, and as discussed in the previous section, the CIW does not have a clear trend of change as the correlation increases. The Normal, BC and BCA methods have larger CIWs compared to other methods. The Hybrid and PCTL methods have the same and smallest CIW among all methods. Combining results from hit rates and CIW rank analysis, the PCTL method actually provides reasonable results for bootstrapped confidence intervals.

Hence, considering comprehensive results about hit rates and CIWs overall, by sample size, by error magnitudes and by correlation between X_2 and X_3, the Normal and PCTL methods tend to provide better hit rates and the PCTL and Hybrid methods tend to provide smaller CIWs. Which method is more appropriate

for the model is a judgment call; however, PCTL is more likely to provide a good confidence interval for each coefficient estimate according to the simulation results when data are generated from an equation with one of the coefficients being zero.

5.2.3 Data Generated without an Equation

Sample data sets of sizes 20, 50 and 100 are generated. All three variables (X_1, X_2, X_3) are generated from $MN(\mathbf{0}, \mathbf{P})$ with correlations of 0, 0.5 and 0.8, with small, moderate and large additive errors ($mult = 0.15, 0.5, 0.8$).

When data are generated from the multivariate normal distribution without an equation, the estimates tend to be very close to zero and CIWs for each coefficient estimate are similar regardless of the different sample sizes, error terms and correlations. Not all results will be shown in this section since they are very similar. The following example will show $\hat{a}_1, \hat{a}_2$ and $\hat{a}_3$ are all close to zero and each method provides similar CIWs for all coefficient estimates.

Table 5.15 Mean estimate and CIW for estimating a_1, a_2 and a_3 with $n = 100$, $\rho = 0$ and $mult = 0.15$

		a_1	a_2	a_3	a_1	a_2	a_3
$n / \rho / mult$	Method	Mean	Mean	Mean	CIW	CIW	CIW
100/0/0.15	Normal	-0.057	-0.069	0.099	11.177	12.067	8.769
100/0/0.15	BC	-0.057	-0.069	0.099	47.047	48.811	35.692
100/0/0.15	BCA	-0.057	-0.069	0.099	47.503	49.04	35.73
100/0/0.15	Hybrid	-0.057	-0.069	0.099	0.752	0.78	0.559
100/0/0.15	PCTL	-0.057	-0.069	0.099	0.752	0.78	0.559
100/0/0.15	Jackknife	-0.057	-0.069	0.099	1.634	1.676	1.187

Since the data are generated without an equation, the hit rates and hypothesis tests cannot be discussed.

Therefore, as shown in previous sections, when the errors are small, all confidence intervals gave similar results, and the confidence interval widths (CIW) are relatively small.

Again, there is no absolute rule for which method is superior based on theory and the simulation results. However, in the simulations, the PCTL, Hybrid and Jackknife methods tend to give narrower confidence intervals compared to other methods. The confidence intervals provided by the Hybrid method are just a shift from that given by PCTL and the CIW are always the same (there might be rounding errors to the 4^{th} decimal). In general the simulations in this chapter suggest that the PCTL bootstrap confidence interval performs better over a range of models and model assumptions. However the simulations considered here are limited in scope and further research is needed.

Chapter 6: RESIDUALS

Traditionally, the residuals and residual plots are used as diagnostic tools to assess the least squares fit of a linear model. As in OLS, diagnostics for RMA regression can be obtained by calculating the predicted values and residuals for each variable, and using residual plots to assess the fit of the model.

6.1 Theory

6.1.1 Predicted Value and Residual

To obtain the predicted value for the i^{th} observation on the j^{th} variable, $\hat{X}_{ij}$, solve for the j^{th} variable in the estimated RMA equation and evaluate at the values of the i^{th} observation for the other $p-1$ variables. So

$$\hat{X}_{ij} = \frac{1}{\hat{a}_j}\left(1 - \sum_{\substack{k=1 \\ k\neq j}}^{p} \hat{a}_k X_{ik}\right).$$

Note that

$$\begin{aligned}
\bar{\hat{X}}_j &= \frac{1}{n}\sum_{i=1}^{n}\hat{X}_{ij} = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{\hat{a}_j}\left(1 - \sum_{k\neq j}^{p}\hat{a}_k X_{ik}\right) \\
&= \frac{1}{\hat{a}_j}\left(1 - \frac{1}{n}\sum_{i=1}^{n}\sum_{k\neq j}^{p}\hat{a}_k X_{ik}\right) = \frac{1}{\hat{a}_j}\left(1 - \sum_{k\neq j}^{p}\hat{a}_k \bar{X}_k\right) \\
&= \frac{1}{\hat{a}_j}\left(1 - \sum_{k=1}^{p}\hat{a}_k \bar{X}_k + \hat{a}_j \bar{X}_j\right) = \frac{1}{\hat{a}_j}\left(1 - \sum_{k=1}^{p}\hat{a}_k \bar{X}_k\right) + \bar{X}_j \\
&= \bar{X}_j,\ j = 1,\ldots, p
\end{aligned}$$

since $\frac{1}{n}\sum_{i=1}^{n}\sum_{k=1}^{p}\hat{a}_k X_{ik} = 1$ was proved in Chapter 3 (eq. 3.2).

Therefore $\bar{\hat{X}}_j = \bar{X}_j,\ j = 1,\ldots, p$.

The residual associated with X_{ij} is the difference between the original value and the predicted value. Then the residual for X_{ij} is

$$\begin{aligned}\hat{e}_{ij} &= X_{ij} - \hat{X}_{ij} = X_{ij} - \frac{1}{\hat{a}_j}\left(1 - \sum_{\substack{k=1 \\ k\neq j}}^{p} \hat{a}_k X_{ik}\right) \\ &= \frac{1}{\hat{a}_j}\left(\sum_{k=1}^{p} \hat{a}_k X_{ik} - 1\right), \forall i = 1,...,n,\ j = 1,...,p\end{aligned} \tag{6.1}$$

Since, $\hat{e}_{ij} = \frac{1}{\hat{a}_j}\left(\sum_{k=1}^{p} \hat{a}_k X_{ik} - 1\right), \forall i = 1,...,n,\ j = 1,...,p$ by (6.1), we have

$$\hat{a}_j \hat{e}_{ij} = \hat{a}_k \hat{e}_{ik},\ for\ \forall j,k = 1,...,p, i = 1,...,n.$$

Therefore it is equivalent to consider the residual of any variable in the model. By definition

$$\hat{e}_{ij} = X_{ij} - \hat{X}_{ij} = \frac{1}{\hat{a}_j}\left(\sum_{k=1}^{p} \hat{a}_k X_{ik} - 1\right), \forall i = 1,...,n,\ j = 1,...,p.$$

Then the average of residuals for a variable is zero,

$$\frac{1}{n}\sum_{i=1}^{n} \hat{e}_{ij} = \frac{1}{n}\frac{1}{\hat{a}_j}\sum_{i=1}^{n}\left(\sum_{k=1}^{p} \hat{a}_k X_{ik} - 1\right) = \frac{1}{\hat{a}_j}\left(\frac{1}{n}\sum_{i=1}^{n}\sum_{k=1}^{p} \hat{a}_k X_{ik} - 1\right) = 0, \forall j = 1,...,p,$$

since $\frac{1}{n}\sum_{i=1}^{n}\sum_{k=1}^{p} \hat{a}_k X_{ik} = 1$ was proved in Chapter 3 (eq. 3.2).

6.1.2 Some Results for Predicted Values and Residuals

As in OLS regression, certain trends and patterns in plots of residuals versus predicted values can be used to detect violations of regression assumptions. In OLS, the interpretation of residual plots is aided by the fact that residuals are uncorrelated with the original observations and with the predicted values.

Unfortunately a similar result does not hold for RMA regression. Sample covariances and correlations between variables, predicted variable values and residuals are obtained here for RMA regression. Define the vector of values for the j^{th} variable $\mathbf{X}_j$, its residual $\hat{\mathbf{e}}_j$ and the corresponding fitted value $\hat{\mathbf{X}}_j$ as the following respectively.

$$\mathbf{X}_j = \left(X_{1j}, ..., X_{nj} \right)', \; j = 1, ..., p$$

$$\hat{\mathbf{e}}_j = \left(\hat{e}_{1j}, ..., \hat{e}_{nj} \right)', \; j = 1, ..., p$$

$$\hat{\mathbf{X}}_j = \left(\hat{X}_{1j}, ..., \hat{X}_{nj} \right)', \; j = 1, ..., p .$$

(1) Sample correlation between a variable and its predicted value

$$corr\left(\mathbf{X}_j, \hat{\mathbf{X}}_j \right) = \frac{\sum_{i=1}^{n} \left(X_{ij} - \bar{X}_j \right) \left(\hat{X}_{ij} - \bar{\hat{X}}_j \right)}{\sqrt{\sum_{i=1}^{n} \left(X_{ij} - \bar{X}_j \right)^2 \sum_{i=1}^{n} \left(\hat{X}_{ij} - \bar{\hat{X}}_j \right)^2}} = \frac{cov\left(\mathbf{X}_j, \hat{\mathbf{X}}_j \right)}{\sqrt{s^2_{\mathbf{X}_j} s^2_{\hat{\mathbf{X}}_j}}}$$

where

$$cov\left(\mathbf{X}_j, \hat{\mathbf{X}}_j \right) = \frac{1}{n-1} \sum_{i=1}^{n} \left(X_{ij} - \bar{X}_j \right) \left(\hat{X}_{ij} - \bar{\hat{X}}_j \right)$$

$$s^2_{\mathbf{X}_j} = \frac{1}{n-1} \sum_{i=1}^{n} \left(X_{ij} - \bar{X}_j \right)^2 \text{ and } s^2_{\hat{\mathbf{X}}_j} = \frac{1}{n-1} \sum_{i=1}^{n} \left(\hat{X}_{ij} - \bar{\hat{X}}_j \right)^2 .$$

Now,

$$cov\left(\mathbf{X}_j, \hat{\mathbf{X}}_j \right) = cov\left(\mathbf{X}_j, \frac{1}{\hat{a}_j} \left(1 - \sum_{k \neq j}^{p} \hat{a}_k \mathbf{X}_k \right) \right) = -\frac{1}{\hat{a}_j} \sum_{k \neq j}^{p} \hat{a}_k cov\left(\mathbf{X}_j, \mathbf{X}_k \right)$$

and

$$s^2_{\hat{\mathbf{X}}_j} = Var\left(\hat{\mathbf{X}}_j\right) = Var\left(\frac{1}{\hat{a}_j}\left(1-\sum_{k\neq j}^{p}\hat{a}_k\mathbf{X}_k\right)\right)$$

$$= \frac{1}{\hat{a}_j^2}\left(\sum_{k\neq j}^{p}\hat{a}_k^2 Var\left(\mathbf{X}_k\right) + 2\sum_{\substack{k<m\\k\neq j\\m\neq j}}^{p}\hat{a}_k\hat{a}_m cov\left(\mathbf{X}_k,\mathbf{X}_m\right)\right)$$

(2) Sample correlation between a variable and another variable's predicted value.

$$corr\left(\mathbf{X}_j,\hat{\mathbf{X}}_i\right) = \frac{\sum_{k=1}^{n}\left(X_{kj}-\bar{X}_i\right)\left(\hat{X}_{kj}-\bar{\hat{X}}_i\right)}{\sqrt{\sum_{k=1}^{n}\left(X_{kj}-\bar{X}_i\right)^2\sum_{k=1}^{n}\left(\hat{X}_{kj}-\bar{\hat{X}}_i\right)^2}} = \frac{cov\left(\mathbf{X}_j,\hat{\mathbf{X}}_i\right)}{\sqrt{s^2_{\mathbf{X}_j}s^2_{\hat{\mathbf{X}}_i}}}$$

where

$$cov\left(\mathbf{X}_j,\hat{\mathbf{X}}_i\right) = cov\left(\mathbf{X}_j,\frac{1}{\hat{a}_i}\left(1-\sum_{k\neq i}^{n}\hat{a}_k\mathbf{X}_k\right)\right) = -\frac{1}{\hat{a}_i}\sum_{k\neq i}^{p}\hat{a}_k cov\left(\mathbf{X}_j,\mathbf{X}_k\right).$$

(3) Sample correlation between a variable and its residual.

$$corr\left(\mathbf{X}_j,\hat{\mathbf{e}}_j\right) = corr\left(\mathbf{X}_j,\mathbf{X}_j-\hat{\mathbf{X}}_j\right)$$

$$= \frac{cov\left(\mathbf{X}_j,\mathbf{X}_j-\hat{\mathbf{X}}_j\right)}{\sqrt{s^2_{\mathbf{X}_j}s^2_{\mathbf{X}_j-\hat{\mathbf{X}}_j}}} = \frac{Var\left(\mathbf{X}_j\right)-cov\left(\mathbf{X}_j,\hat{\mathbf{X}}_j\right)}{\sqrt{s^2_{\mathbf{X}_j}s^2_{\mathbf{X}_j-\hat{\mathbf{X}}_j}}}$$

where $s^2_{\mathbf{X}_j-\hat{\mathbf{X}}_j} = Var\left(\mathbf{X}_j\right)+Var\left(\hat{\mathbf{X}}_j\right)-2cov\left(\mathbf{X}_j,\hat{\mathbf{X}}_j\right)$.

(4) Sample correlation between a variable and another variable's residual.

$$corr\left(\mathbf{X}_j,\hat{\mathbf{e}}_i\right) = corr\left(\mathbf{X}_j,\mathbf{X}_i-\hat{\mathbf{X}}_i\right)$$

$$= \frac{cov\left(\mathbf{X}_j,\mathbf{X}_i-\hat{\mathbf{X}}_i\right)}{\sqrt{s^2_{\mathbf{X}_j}s^2_{\mathbf{X}_i-\hat{\mathbf{X}}_j}}} = \frac{cov\left(\mathbf{X}_j,\mathbf{X}_i\right)-cov\left(\mathbf{X}_j,\hat{\mathbf{X}}_i\right)}{\sqrt{s^2_{\mathbf{X}_j}s^2_{\mathbf{X}_i-\hat{\mathbf{X}}_i}}}$$

where $s^2_{\mathbf{X}_i-\hat{\mathbf{X}}_i} = Var\left(\mathbf{X}_i\right)+Var\left(\hat{\mathbf{X}}_i\right)-2cov\left(\mathbf{X}_i,\hat{\mathbf{X}}_i\right)$.

(5) Sample correlation between the predicted value of a variable and its residual.

$$corr\left(\hat{\mathbf{X}}_j,\hat{\mathbf{e}}_j\right)=corr\left(\hat{\mathbf{X}}_j,\mathbf{X}_j-\hat{\mathbf{X}}_j\right)$$
$$=\frac{cov\left(\hat{\mathbf{X}}_j,\mathbf{X}_j-\hat{\mathbf{X}}_j\right)}{\sqrt{s^2_{\hat{\mathbf{X}}_j}s^2_{\mathbf{X}_j-\hat{\mathbf{X}}_j}}}=\frac{cov\left(\mathbf{X}_j,\hat{\mathbf{X}}_j\right)-Var\left(\hat{\mathbf{X}}_j\right)}{\sqrt{s^2_{\hat{\mathbf{X}}_j}s^2_{\mathbf{X}_j-\hat{\mathbf{X}}_j}}}$$

The following relationship of correlations is concluded from the fact that $a_j\hat{e}_{ij}=a_k\hat{e}_{ik}$ for $\forall j,k=1,...,p,i=1,...,n$.

$$corr\left(\hat{\mathbf{X}}_i,\hat{\mathbf{e}}_j\right)=corr\left(\hat{\mathbf{X}}_i,\hat{\mathbf{e}}_k\right),\ for\ \forall i,\ j,k=1,...,p$$
$$corr\left(\mathbf{X}_i,\hat{\mathbf{e}}_j\right)=corr\left(\mathbf{X}_i,\hat{\mathbf{e}}_k\right),\ for\ \forall i,\ j,k=1,...,p$$

6.2 Residual Plots

6.2.1 Residuals are Correlated with Original or Predicted Variables

When residual plots are obtained in OLS regression, the uncorrelated relationship between residuals and original or predicted variables is useful in interpreting the plots. However by the results shown in section 6.1, the residuals are in general correlated with the original values and the predicted values of the variables. Figure 6.1 shows residual plots for a random sample of size 100 obtained from 2 uncorrelated variables $\left(X_2,X_3\right)$ that were generated from $MN\left(\mathbf{0},\mathbf{I}\right)$ and X_1 was obtained from the equation

$$3X_1+2X_2+X_3=1,$$

with small additive error ($mult=0.15$).

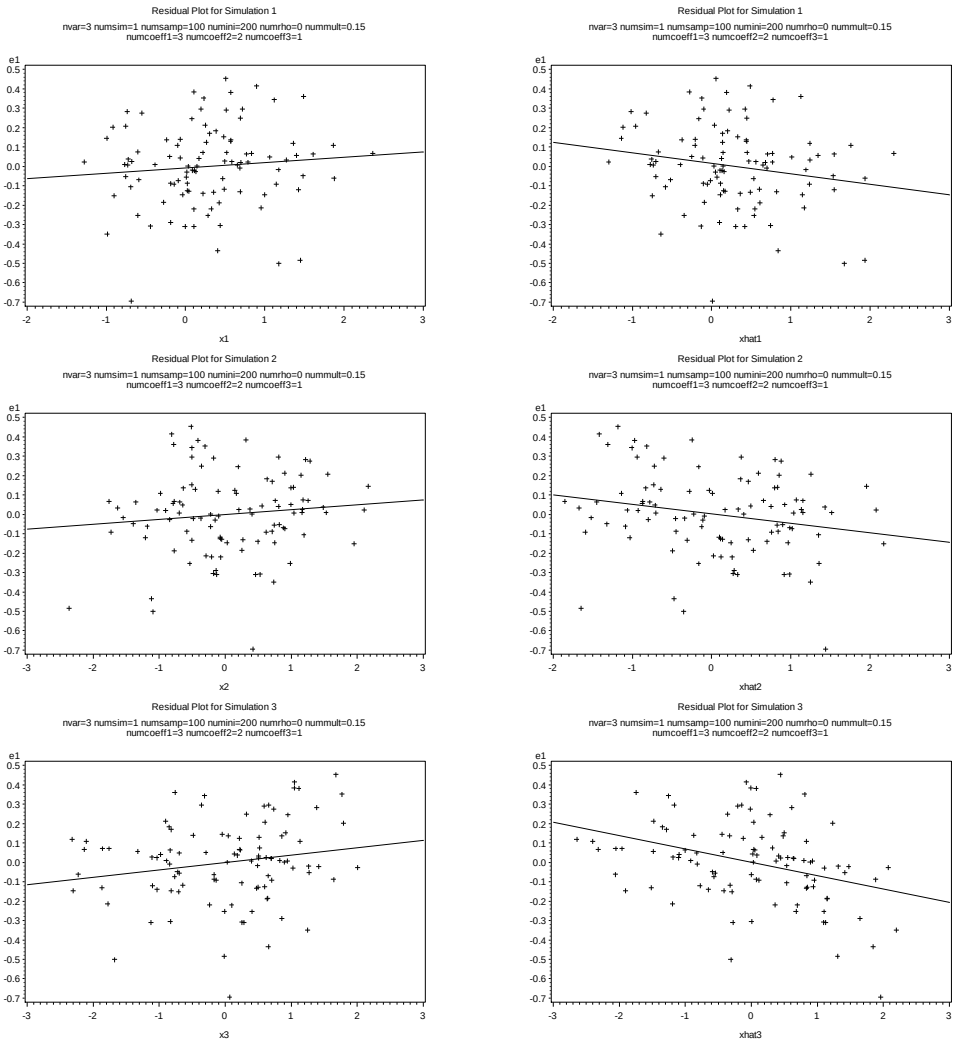

Figure 6.1 Residual plots of $\hat{e}_1$ versus X_i (*left*) and $\hat{X}_i, i = 1,2,3$ (*right*) for a simulated random sample with $n = 100, \rho = 0, mult = 0.15$ and $coeff = \{3, 2, 1\}$ with OLS regression line.

As shown in the figure 6.1, there are some trends in the residual plots and the residuals are correlated with the original variables and the predicted values of variables.

6.2.2 De-trending is Helpful

The plots of residuals versus original or predicted values in figure 6.1 would sometimes show a linear trend. Although some residual plots do not show obvious linear relationships in many of the examples, we consider obtaining de-trended plots.

To perform the de-trending on the RMA residual plots, use OLS regression to regress $\hat{e}_1$ on X_1, X_2 and X_3 (or on $\hat{X}_1, \hat{X}_2$ and $\hat{X}_3$) and obtain the new OLS residuals. The plots of the new OLS residuals versus the original X_1, X_2, X_3 (or on $\hat{X}_1, \hat{X}_2$ and $\hat{X}_3$) are examined. Since OLS regression is used, the new OLS residuals are uncorrelated with the original values or predicted values and the new OLS residual plots will show no linear trends.

As shown in figure 6.2, after the de-trending procedure, residual plots show no linear trends.

Residual plots are also helpful in determining if the original data follow a relationship that is not a linear relationship. Consider a random sample of size 100 obtained from 2 uncorrelated variables $\left(X_2, X_3\right)$ that were generated from $MN\left(\mathbf{0}, \mathbf{I}\right)$ and X_1 that was obtained from the equation

$$3X_1 + 2X_2^2 + X_3 = 1,$$

with small additive error ($mult = 0.15$).

The residual plots are shown in figure 6.3 and the de-trended residual plots are in figure 6.4.

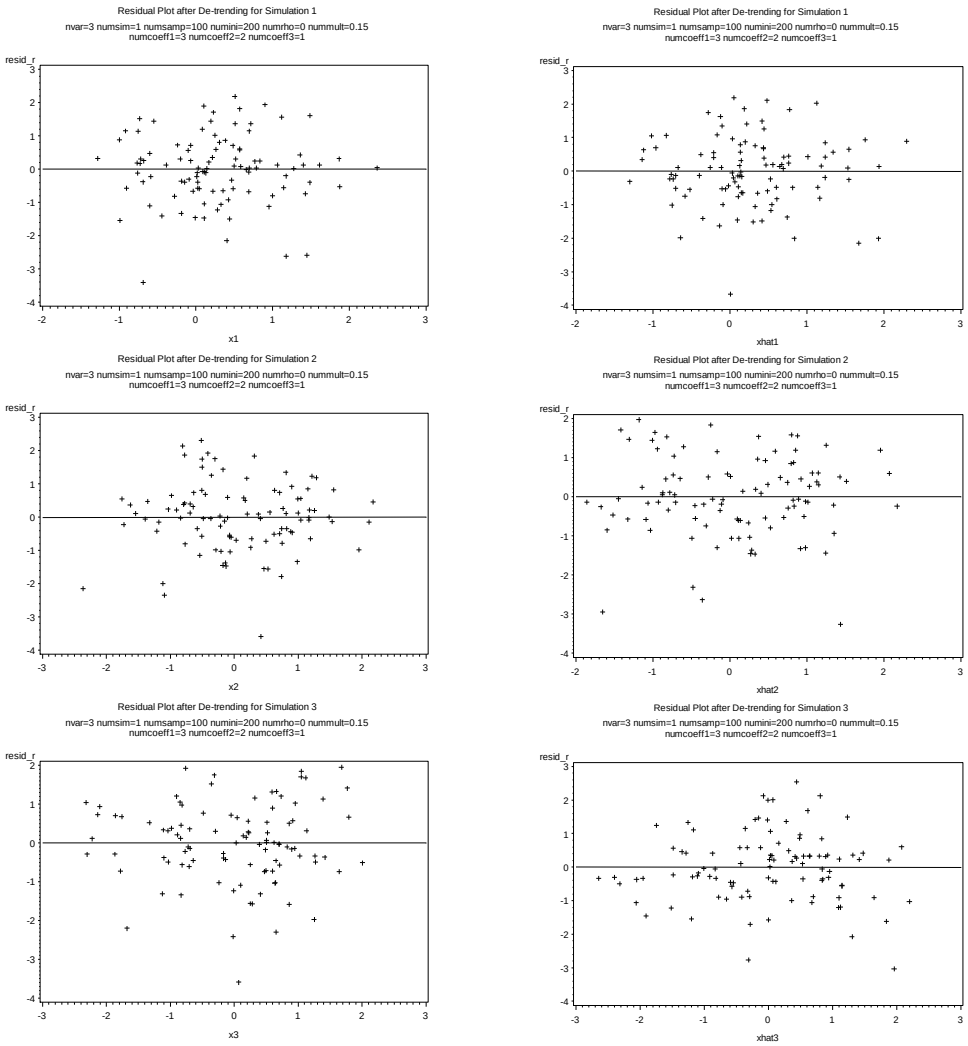

Figure 6.2 Residual plots of new residuals from regressing $\hat{e}_1$ on X_i by OLS versus the original variable X_i (*left*) and $\hat{X}_i, i = 1,2,3$ (*right*) for a simulated random sample with $n = 100, \rho = 0, mult = 0.15$ and $coeff = \{3, 2, 1\}$.

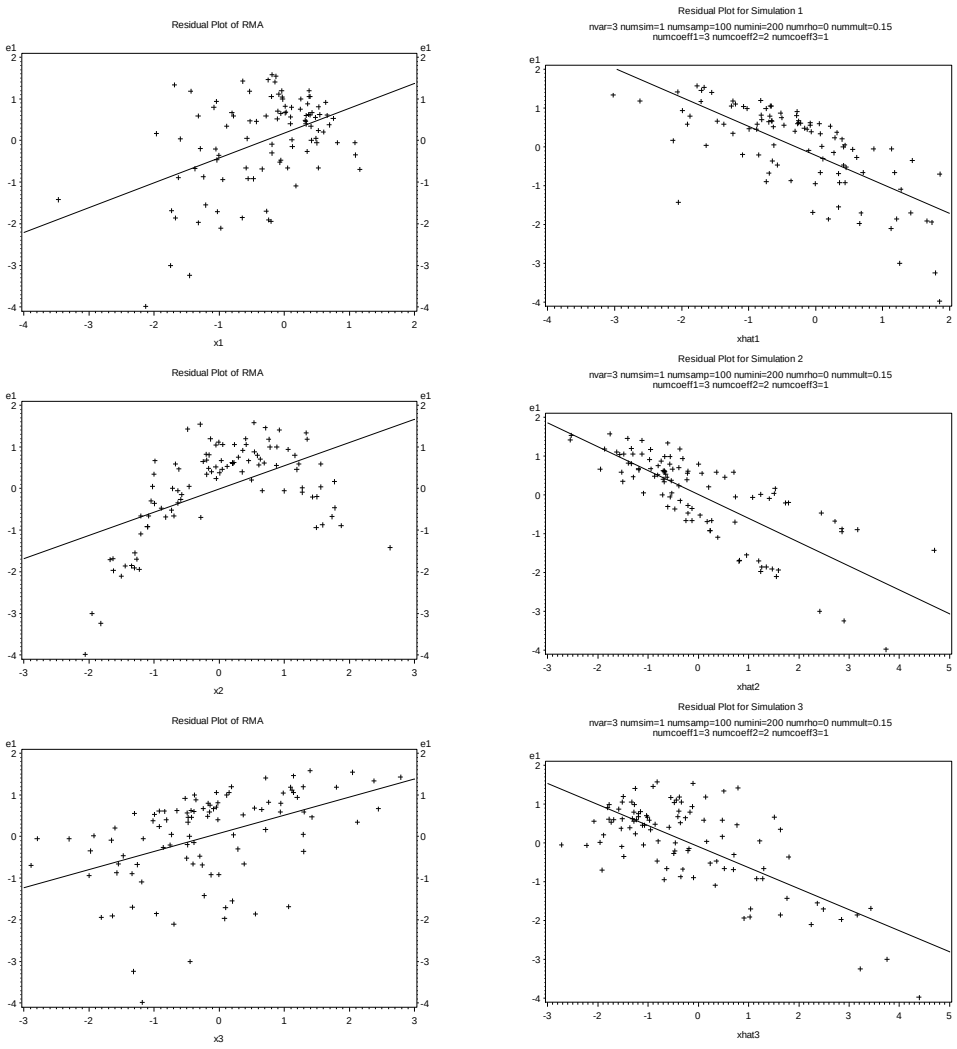

Figure 6.3 Residual Plots of $\hat{e}_1$ versus X_i (*left*) and $\hat{X}_i, i = 1,2,3$ (*right*) for a simulated random sample with $n = 100, \rho = 0, mult = 0.15$ with quadratic term in X_2

As shown in figure 6.3, the residual plots associated with X_2 and $\hat{X}_2$ show quadratic trends. Similarly, OLS regression was used to regress $\hat{e}_1$ on X_1, X_2 and X_3 (or on $\hat{X}_1, \hat{X}_2$ and $\hat{X}_3$) to obtain the new OLS residuals. The plots of the new OLS residuals versus the original X_1, X_2, X_3 and predicted $\hat{X}_1, \hat{X}_2$ and $\hat{X}_3$ are examined.

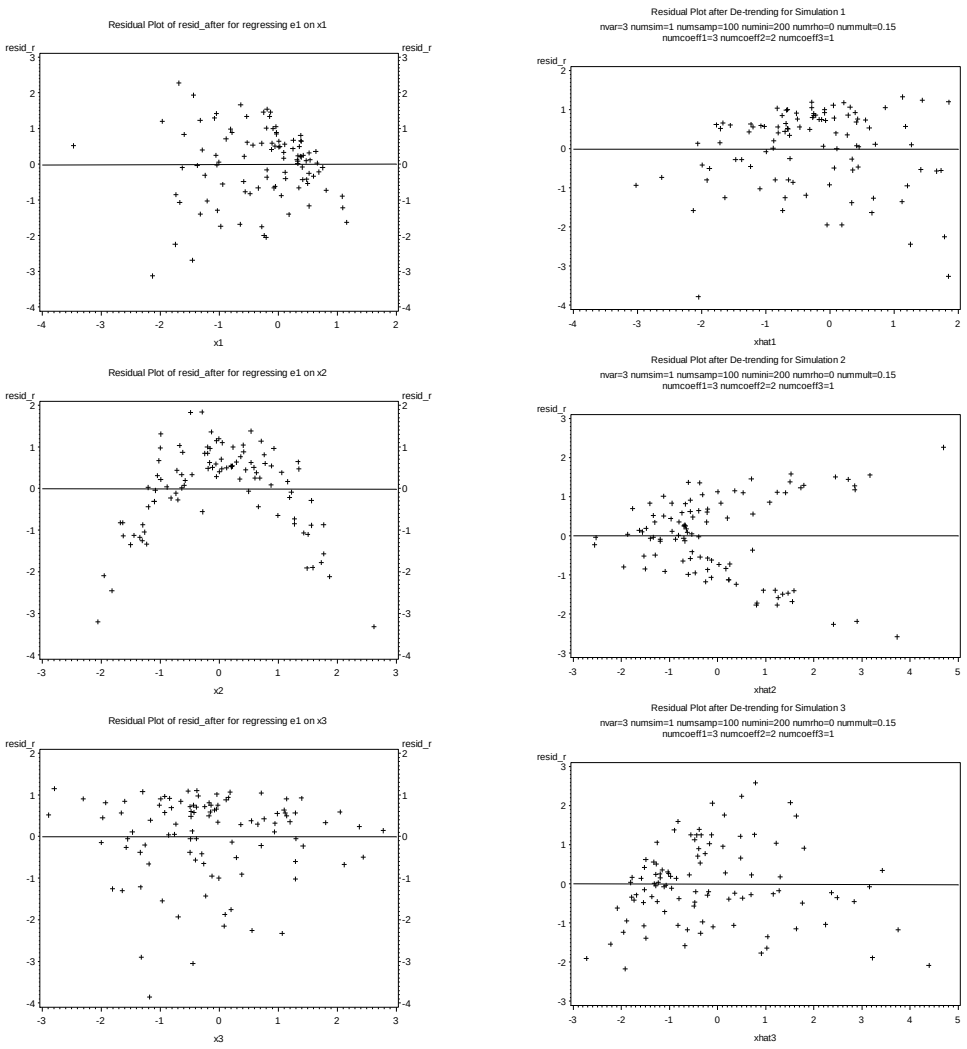

Figure 6.4 Residual plots of new residuals from regressing $\hat{e}_1$ on X_i by OLS versus the original variable X_i (*left*) and $\hat{X}_i, i = 1,2,3$ (*right*) for a simulated random sample with $n = 100, \rho = 0, mult = 0.15$ with quadratic term in X_2

As shown in figure 6.4, the left column plots and the right column plots the residual plots after de-trending. It is clear that the residual plots of $\hat{e}_1$ versus X_2 and $\hat{X}_2$ and the new residual associated with $\hat{e}_1$ versus X_2 and $\hat{X}_2$ have an obvious quadratic trend that can be seen in both the residual plots and the de-trended residual plots.

Chapter 7: DIAGNOSTICS FOR OUTLIERS AND INFLUENTIAL POINTS

The residual plots can be used to investigate the presence of outliers in the original data.

7.1 Outliers Detection

Examining the residual plots or the scatter plots of the original data could give some indication of the presence of outliers.

Consider a random sample of size 50 with 2 uncorrelated variables (X_2, X_3) that were generated from $MN(\mathbf{0}, \mathbf{I})$ and X_1 that was obtained from the equation

$$3X_1 + 2X_2 + X_3 = 1,$$

with small additive error ($mult = 0.15$). The last observation (50^{th}) was generated as an outlier on purpose with a shift of 5, 10 and 8 in the values of X_1, X_2 and X_3 respectively. The scatter plots of original data and residual plots of residuals versus each variable are shown in figure 7.1.

In figure 7.1, the left column plots from top to bottom are the scatter plots of X_1 *vs* X_2, X_1 *vs* X_3 and X_2 *vs* X_3; the right column plots from top to bottom are the residual plots of $\hat{e}_1$ *vs* $X_i, i = 1,2,3$.

Figure 7.1 shows that both scatter plots and residuals plots give clear indications of the outlier in the sample simulation.

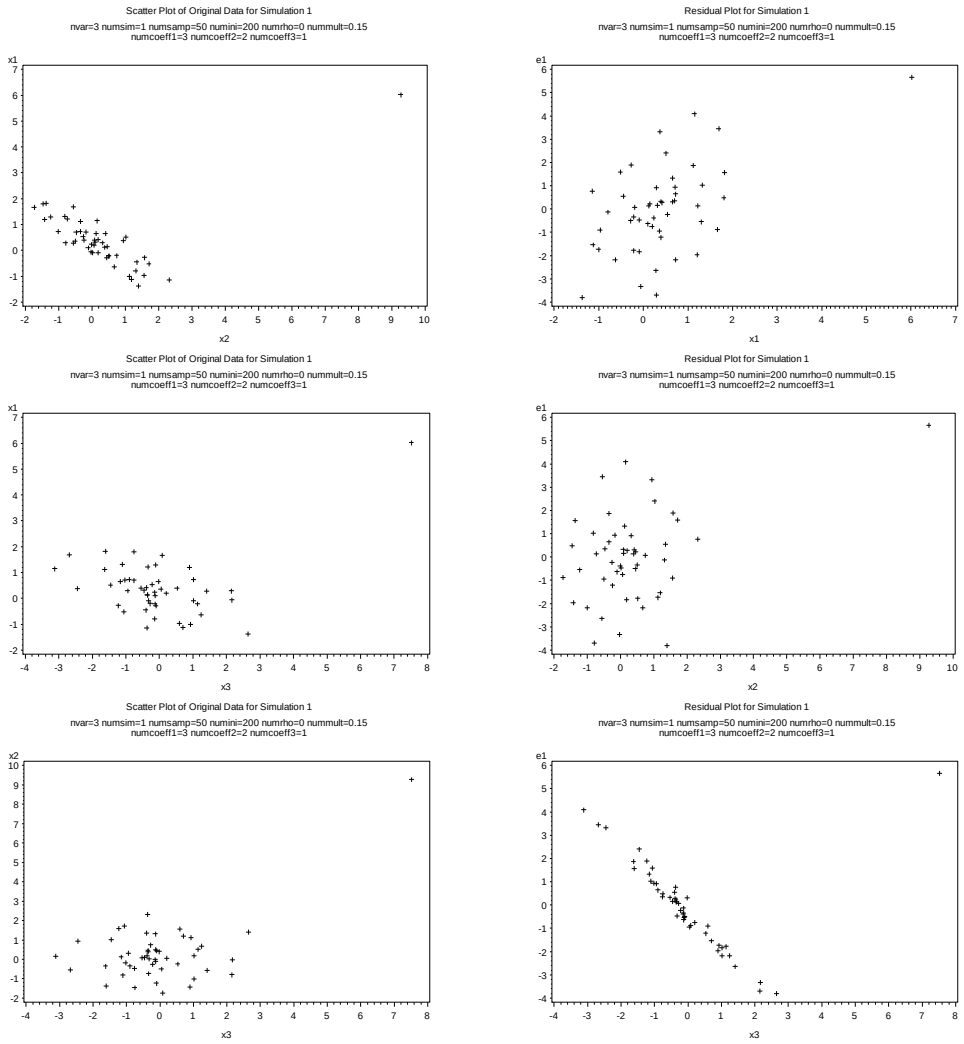

Figure 7.1 Scatter plots of original data (*left*) and residual plots of residuals versus X_1, X_2, X_3 (*right*) for the simulated random sample with $n = 50$, $\rho = 0$, $mult = 0.15$ and $coeff = \{3, 2, 1\}$

In OLS, cross-validation is often used for assessing the goodness of fit of a model. One of the cross-validation approaches is leave-one-out, which involves using a single observation from the original sample as the validation data and the remaining observations as the training data. Some statistics that are generated with the leave-one-out approach are deleted residuals, Cook's distance, DFFITS and DFBETAS (Kutner, Nachtsheim, Neter and Li 2004) in OLS regression. Similarly in RMA, one observation of the data (sample size is n) can be omitted

each time. Then RMA is performed for the data with the $n-1$ observations to obtain the deleted estimates of coefficients, deleted fitted values and deleted residuals.

Let $\hat{X}_{i(i)j}$ be the fitted value of X_{ij} obtained from the RMA equation with the i^{th} observation deleted. We have

$$\hat{X}_{i(i)j} = \frac{1}{\hat{a}_{j(i)}}\left(1 - \sum_{k \neq j}^{p} \hat{a}_{k(i)} X_{ik}\right),$$

where $\hat{a}_{j(i)}$ is the deleted coefficient estimate for the j^{th} variable with the i^{th} case removed.

The deleted residual associated with X_{ij} is given by

$$d_{i(i)j} = X_{ij} - \hat{X}_{i(i)j}, \; i = 1, ..., n, \; j = 1, ..., p \; .$$

7.2 Influential Observation Detection

The leave-one-out approach can be used for diagnostics to study the influence of a case in the following ways.

7.2.1 Influence on Coefficients

(1) Influence on individual coefficients

Define the "distance" between the coefficient for the i^{th} variable (a_i) based on all the data and the deleted coefficient ($\hat{a}_{j(i)}$) for the variable X_j when the i^{th} observation is deleted as

$$DC_{j(i)} = \hat{a}_j - \hat{a}_{j(i)},\ j = 1,..., p, i = 1,..., n.$$

This distance indicates what impact the deleted observation has on the coefficient estimate associated with each variable before it was deleted.

The same sample data were generated as described in Section 7.1 and the leave-one-out approach was applied on this data set. Figure 7.2 shows plots of $DC_{j(i)}$ plotted against observation number for each variable.

The plots of $DC_{j(i)}$ versus the observation number for each variable do not show that observation 50 is influential when assessing the coefficients. Observation 34 appears to have the most influence on the coefficients. Further investigation of observation 34 will be given at the end of Section 7.2.3.

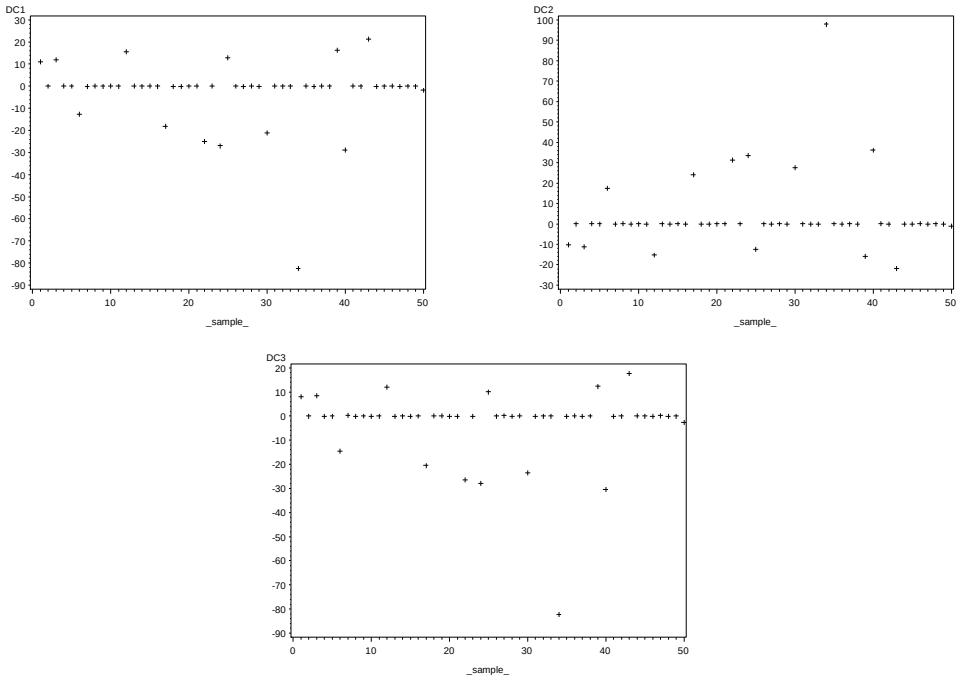

Figure 7.2 Plot of $DC_{j(i)}$ versus observation number for each variable

(2) Influence on all coefficients at the same time

In addition to what was described in (1), let

$$\hat{\mathbf{a}}_{(i)} = \left(\hat{a}_{1(i)},...,\hat{a}_{p(i)}\right)' \text{ and } \hat{\mathbf{a}} = \left(\hat{a}_1,...,\hat{a}_p\right)'$$

and define the squared Euclidean distance between $\hat{\mathbf{a}}_{(i)}$ and $\hat{\mathbf{a}}$ as

$$EC_{(i)} = \left(\hat{\mathbf{a}} - \hat{\mathbf{a}}_{(i)}\right)'\left(\hat{\mathbf{a}} - \hat{\mathbf{a}}_{(i)}\right).$$

$EC_{(i)}$ measures the overall influence on all coefficients at the same time when the i^{th} observation is deleted.

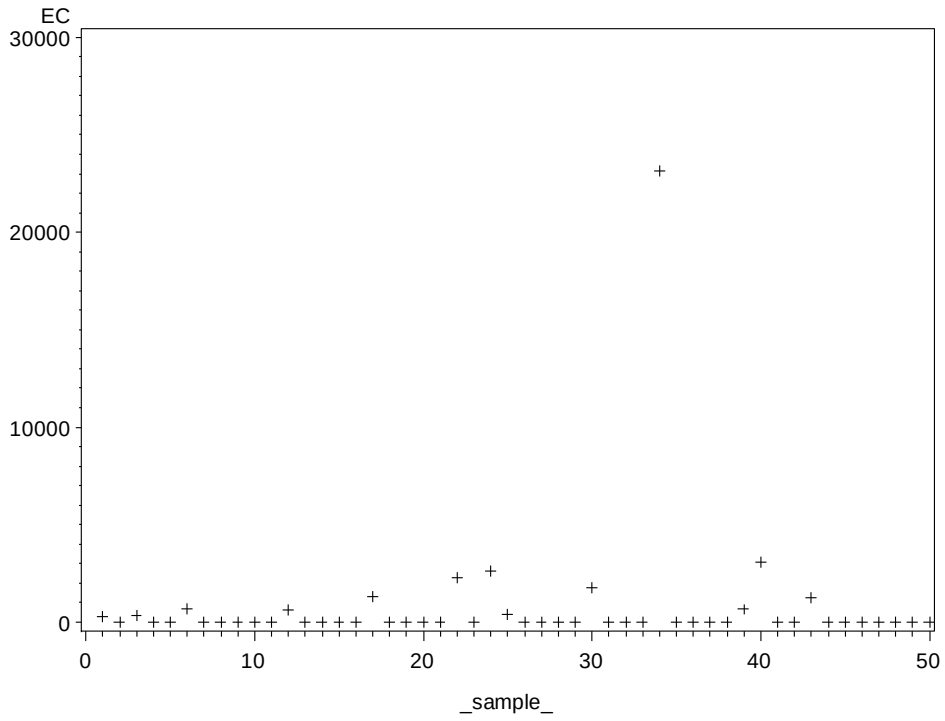

Figure 7.3 Plot of $EC_{(i)}$ versus observation number for each variable

As shown in figure 7.3, $EC_{(i)}$ provides a clear indication that observation 34 is the most influential on the coefficients. Observation 50 does not appear to influence the values of the coefficients.

7.2.2 Influence on Fitted Values

(1) Influence on individual fitted values

Define the "distance" between the fitted value based on all the data and the deleted fitted value for X_{ij} when the i^{th} observation is deleted as

$$DF_{i(i)j} = \hat{X}_{ij} - \hat{X}_{i(i)j}$$
$$= X_{ij} - \hat{X}_{i(i)j} - \left(X_{ij} - \hat{X}_{ij} \right) = d_{i(i)j} - \hat{e}_{ij},\ j = 1,..., p,\ i = 1,..., n.$$

This distance indicates what impact the deleted observation had on the fitted value associated with each case before it was deleted. This difference is the same as the difference between the deleted residual and the residual.

Consider the example described previously in Section 7.2.1 and obtain the following plots of $DF_{i(i)j}$ versus observation number for each variable.

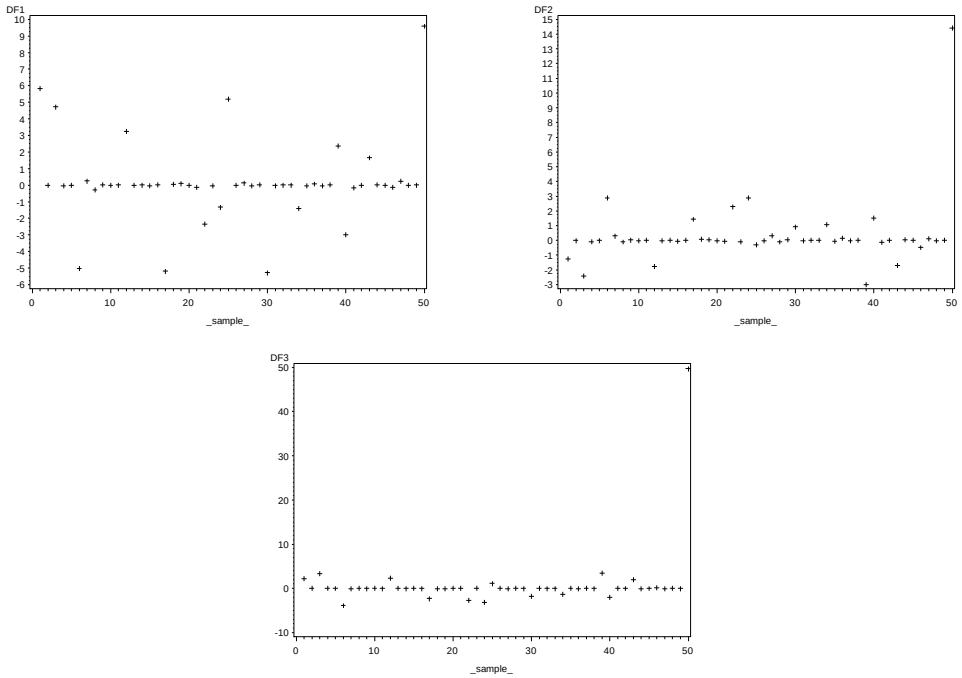

Figure 7.4 Plot of $DF_{i(i)j}$ versus observation number for each variable

The plots of $DF_{i(i)j}$ versus the observation number for each variable in figure 7.4 indicate that observation 50 is influential. However the $DF_{i(i)1}$ was not as useful as $DF_{i(i)2}, DF_{i(i)3}$ to identify the influential case.

(2) Influence on all fitted values for each variable

Let the fitted values for the n observations for the j^{th} variable and the deleted fitted values for the n observations for the j^{th} variable be

$$\hat{\mathbf{X}}_j = \left(\hat{X}_{1j},...,\hat{X}_{nj}\right)' \text{ and } \hat{\mathbf{X}}_{(i)j} = \left(\hat{X}_{1(i)j},...,\hat{X}_{n(i)j}\right)'.$$

Define the squared Euclidean distance between all fitted values and the deleted fitted values for each variable by

$$EF_{(i)j} = \left(\hat{\mathbf{X}}_j - \hat{\mathbf{X}}_{(i)j}\right)'\left(\hat{\mathbf{X}}_j - \hat{\mathbf{X}}_{(i)j}\right).$$

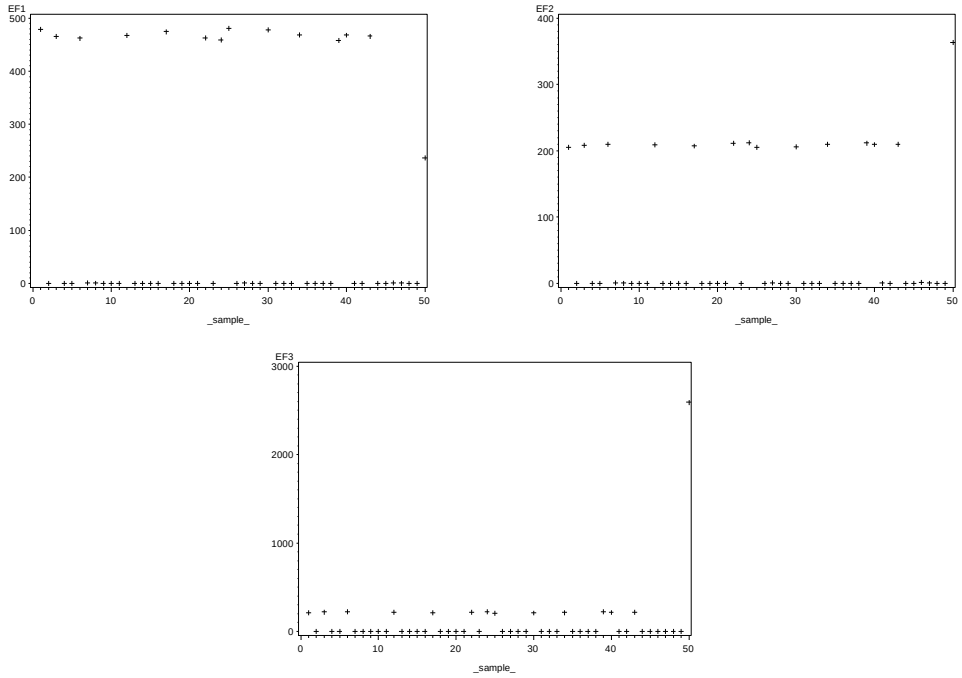

Figure 7.5 Plot of $EF_{(i)j}$ versus observation number for each variable

As shown in figure 7.5, the $EF_{(i)j}$ provides a much more clear indication that the 50th observation is an influential case than the individual $DF_{i(i)j}$.

7.2.3 Influence on Objective Function Values

Recall that the criterion to be minimized to find the RMA fit is written as L_G^p. Let the objective function value for the model fitted with all n observations and the one with the i^{th} case deleted be L and $L_{(i)}$ respectively. Define

$$DL_{(i)} = L - L_{(i)}, i = 1,...,n.$$

The same sample data were generated as described in the Section 7.1. The leave-one out approach was applied to the data. The following plot shows how the $DL_{(i)}$ could help identify the outliers or influential points.

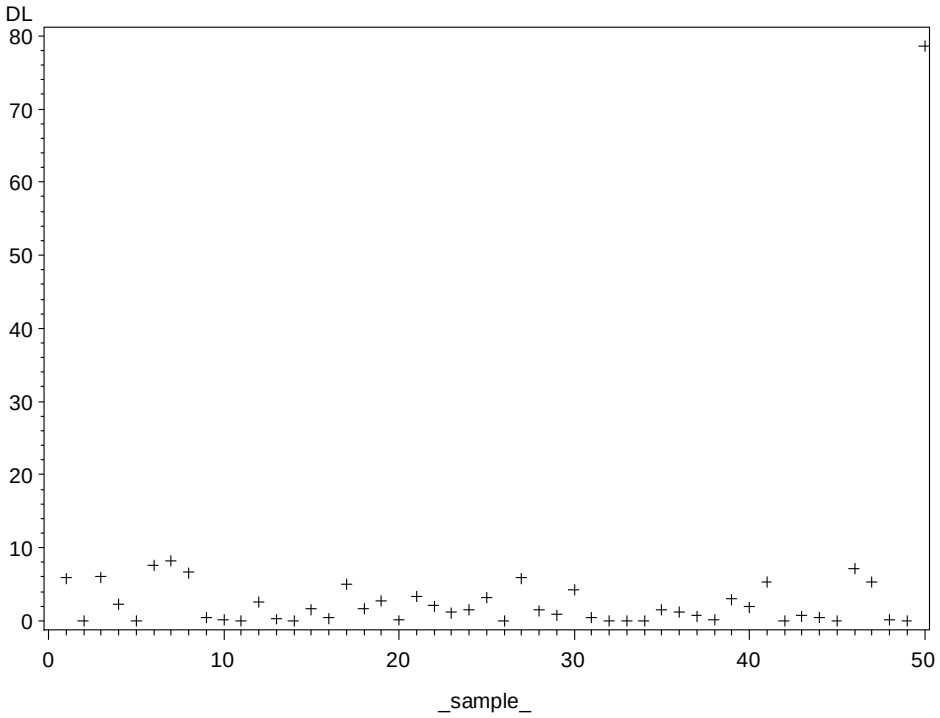

Figure 7.6 Plot of $DL_{(i)}$ versus the observation number

Figure 7.6 shows the deleted objective function values versus the observation numbers. As shown in the plot, the "distances" between the objective function values for the model with all observations included and the model with one observation omitted each time show the obvious influence of the last observation on the objective function.

7.2.4 Observation 34

Plots of $DC_{j(i)}$ and $EC_{(i)}$ vs sample number in figure 7.2 and 7.3 show observation 34 is the most influential; however observation 34 is not an outlier according to the data generation process.

Table 7.1 Observation 34, 49 and 50 of the original data in simulation 1

Obs #	X_1	X_2	X_3
34	-0.1945	0.7433	-0.2791
49	0.7022	-0.4693	-0.7596
50	6.0217	9.2746	7.5180

Note: observation 34 and 49 are not generated as an outlier and X_1, X_2 and X_3 of observation 34 and 49 are relatively close to each other.

Table 7.2 The deleted coefficient estimates and objective function value in simulation1 with observation 34, 49 and 50 deleted

Obs #	$\hat{a}_{1(obs\#)}$	$\hat{a}_{2(obs\#)}$	$\hat{a}_{3(obs\#)}$	$L_{(obs\#)}$
34 deleted	83.8351	-96.5989	80.7278	81.0169
49 deleted	1.3189	1.2758	-1.5069	81.0077
50 deleted	3.1114	2.3435	1.0604	2.4772
None deleted	1.3051	1.2659	-1.4916	81.0661

Observation 49 is representative of the majority of data points in this sample data set. Most of the observations have deleted coefficient estimates and objective function values very close to what observation 49 has. Compared to observation 49, observation 34 has deleted objective function 81.0169 which is very close to 81.0077, when observation 34 is deleted the estimates of coefficients are

completely different from those when observation 49 is deleted and from the coefficients using all the data. This happens because the optimization process found a local minimum and it failed to find the correct RMA estimates. Even when observation 34 is deleted the objective function value is reasonable and a set of coefficient estimates that minimize the objective function locally is obtained. These estimates are very large in magnitude and therefore the $DC_{j(34)}$ and $EC_{(34)}$ calculated above based on deleted estimates are much larger than others.

As long as the optimization approach based on initial guesses is used, there is always a chance that the optimization will fail to find the appropriate minimum. Further study of alternative optimization methods or other approaches for finding RMA coefficient estimates needs to be considered. Care must also be taken in evaluating the influence measures considered here due to the possibility that an influence diagnostic might be affected by the inability of the optimization method to find the global minimum of the objective function.

Chapter 8: SUBSET SELECTION

When there are p variables in a data set, it is often of interest to obtain a model that contains a subset of the most important variables.

8.1 Theory

A forward selection approach based on RMA is proposed as a method for model selection. Suppose there are p variables in total. The proposed forward selection method proceeds according to the following steps.

(1) Select the first two variables to enter the model as the ones that have the highest Pearson's correlation coefficient in absolute value, say X_1 and X_2.

(2) For the remaining $(p-2)$ variables, one is added (X_{3j}, $j=1,...,(p-2)$) to the model each time and RMA is performed on X_1, X_2 and X_{3j}. Residuals for each RMA are calculated as discussed in Chapter 6. For each model with three variables the sums of squared residuals can be obtained for each variable in the model, say $\sum_{i=1}^{n}\hat{e}_{ik}^2$, $k=1,2,3j$. Since the residuals for one variable are just a multiple of that of another variable in the same model (discussed in Chapter 6 Section 6.1), and since we want to be able to compare the sums of squares of residuals for each model with three variables and select the one with the smallest sum of squared residuals, we will use $\sum_{i=1}^{n}\hat{e}_{i1}^2$ for X_1 for the $(p-2)$ RMAs to compare models. The variable with the smallest $\sum_{i=1}^{n}\hat{e}_{i1}^2$, say X_3, will be entered into the model at this step.

(3) Select the m^{th} variable by adding X_{mj}, $j=1,...,(p-m+1)$ to the model with $X_1, X_2, ..., X_{m-1}$ from the previous step, perform RMA on $X_1, X_2, ..., X_{m-1}$ and X_{mj}, $j=1,...,(p-m+1)$, calculate the $\sum_{i=1}^{n}\hat{e}_{i1}^2$ for X_1, compare the $\sum_{i=1}^{n}\hat{e}_{i1}^2$ for the $(p-m+1)$ RMAs, and enter the variable (X_m) with the smallest $\sum_{i=1}^{n}\hat{e}_{i1}^2$.

(4) The stopping criterion is a judgment call. Unlike OLS regression, $\sum_{i=1}^{n}\hat{e}_{i1}^2$ is not decreasing all the time as more variables are added to the model. However, a variable is worthy of consideration to be entered if the $\sum_{i=1}^{n}\hat{e}_{i1}^2$ decreases when this variable is added in the model. When the $\sum_{i=1}^{n}\hat{e}_{i1}^2$ increases it is possible that the variable being added to the model should not be selected. The plot of $\sum_{i=1}^{n}\hat{e}_{i1}^2$ versus the number of variables selected will be helpful to decide when to stop entering variables.

8.2 Simulation

In this chapter, simulation results will be shown in order to discuss the forward selection approach. Simulated data sets with 4 to 6 variables are generated with different models and small errors ($mult = 0.15$). The following types of data sets are considered for the forward selection method.

8.2.1 Data generated without an equation

(1) Data are generated with p variables from the multivariate normal distribution $MN(\mathbf{0}, \mathbf{P})$ with additive errors, where $\mathbf{P}$ is its covariance matrix with 0 correlations.

A random sample of size 50 with 4 uncorrelated variables ($X_1, ..., X_4$) that are obtained from $MN(\mathbf{0}, \mathbf{I})$ is generated with small additive error ($mult = 0.15$) for each variable.

Table 8.1 Table of variables entered in the model with the sum of squared residuals in 10 simulations

	Number of Variables in Model (sum of squared residuals)		
	2	3	4
Sim=1	1, 2 (72.86)	3 (100.21)	4 (123.25)
Sim=2	2, 4 (57.13)	3 (82.24)	1 (108.61)
Sim=3	2, 4 (85.16)	3 (124.57)	1 (163.93)
Sim=4	3, 4 (57.57)	1 (88.46)	2 (123.74)
Sim=5	1, 2 (70.49)	3 (71.18)	4 (91.27)
Sim=6	3, 4 (66.30)	1 (89.07)	2 (130.43)
Sim=7	1, 2 (82.57)	3 (108.49)	4 (140.71)
Sim=8	1, 3 (69.55)	4 (88.26)	2 (109.02)
Sim=9	1, 3 (64.48)	2 (80.41)	4 (106.84)
Sim=10	1, 3 (86.40)	2 (113.89)	4 (160.90)

Table 8.1 shows the results of performing forward selection for 10 simulated data sets. For each simulation the three columns give the variables entered into the model with the corresponding $\sum_{i=1}^{n} \hat{e}_{i1}^2$.

Since all 4 variables are generated from a multivariate normal distribution with zero correlations and no specific equations, as shown in table 8.1, the variables entered in the model in no certain order. The sum of squared residuals increase at each successive step in all 10 simulations.

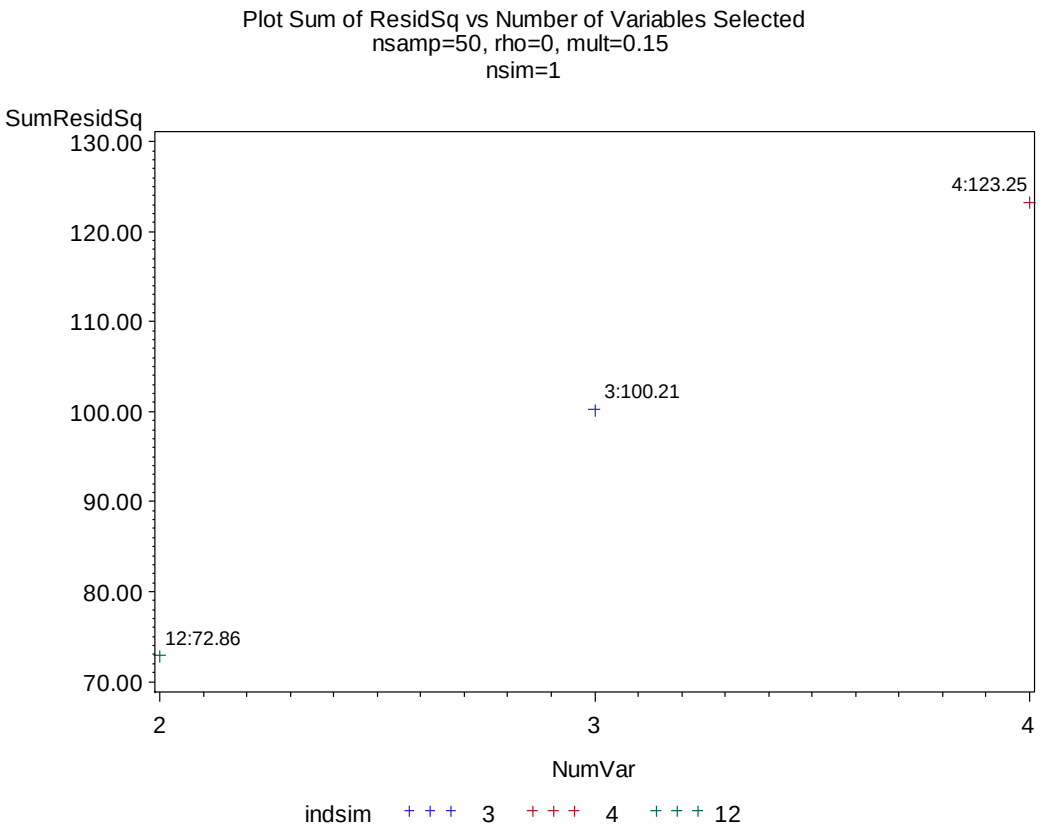

Figure 8.1 Sample plot of $\sum_{i=1}^{n} \hat{e}_{i1}^2$ versus the number of variables selected in Simulation 1 for sample of $n = 50$, $\rho = 0$, $mult = 0.15$ and no specific coefficients.

Figure 8.1 shows the plot of the sum of squared residuals versus number of variables in the RMA model for the first simulation.

(2) Data are generated from

$$MN\left(\mathbf{0},\begin{pmatrix}\mathbf{P} & \mathbf{0}\\ \mathbf{0} & \mathbf{I}\end{pmatrix}\right)$$

where $\mathbf{P}$ is $k\times k$ with all correlation $\rho=0.8$ and $\mathbf{I}$ is $(p-k)\times(p-k)$.

In the following example, among the 6 variables, $X_1,...,X_4$ are from $MN(\mathbf{0},\mathbf{P})$ with $\rho=0.8$ and X_5, X_6 are from $MN(\mathbf{0},\mathbf{I})$ with $X_1,...,X_4$ independent of X_5 and X_6. The first 4 variable are highly correlated with a correlation 0.8, and the other 2 are uncorrelated with any other variables. To fit a linear plane to the 6 variables, $X_1,...,X_4$ will be expected to be entered in the model.

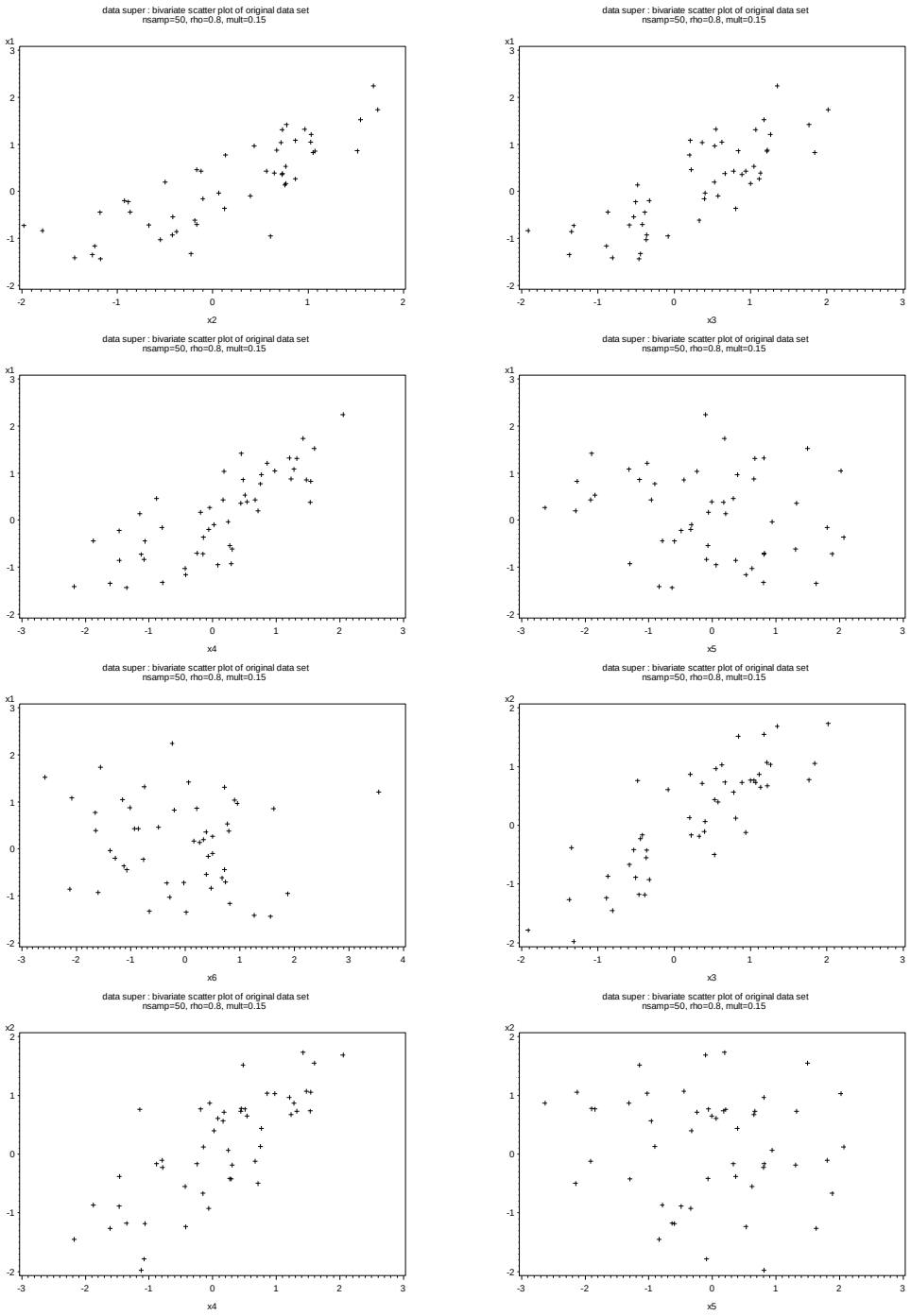

Figure 8.2 Scatter plots of the original data in Simulation 1 for sample of $n = 50$, $\rho = 0.8$ between $X_1, ..., X_4$ and $\rho = 0$ between X_5, X_6 which are independent from $X_1, ..., X_4$, $mult = 0.15$ and no specific coefficients.

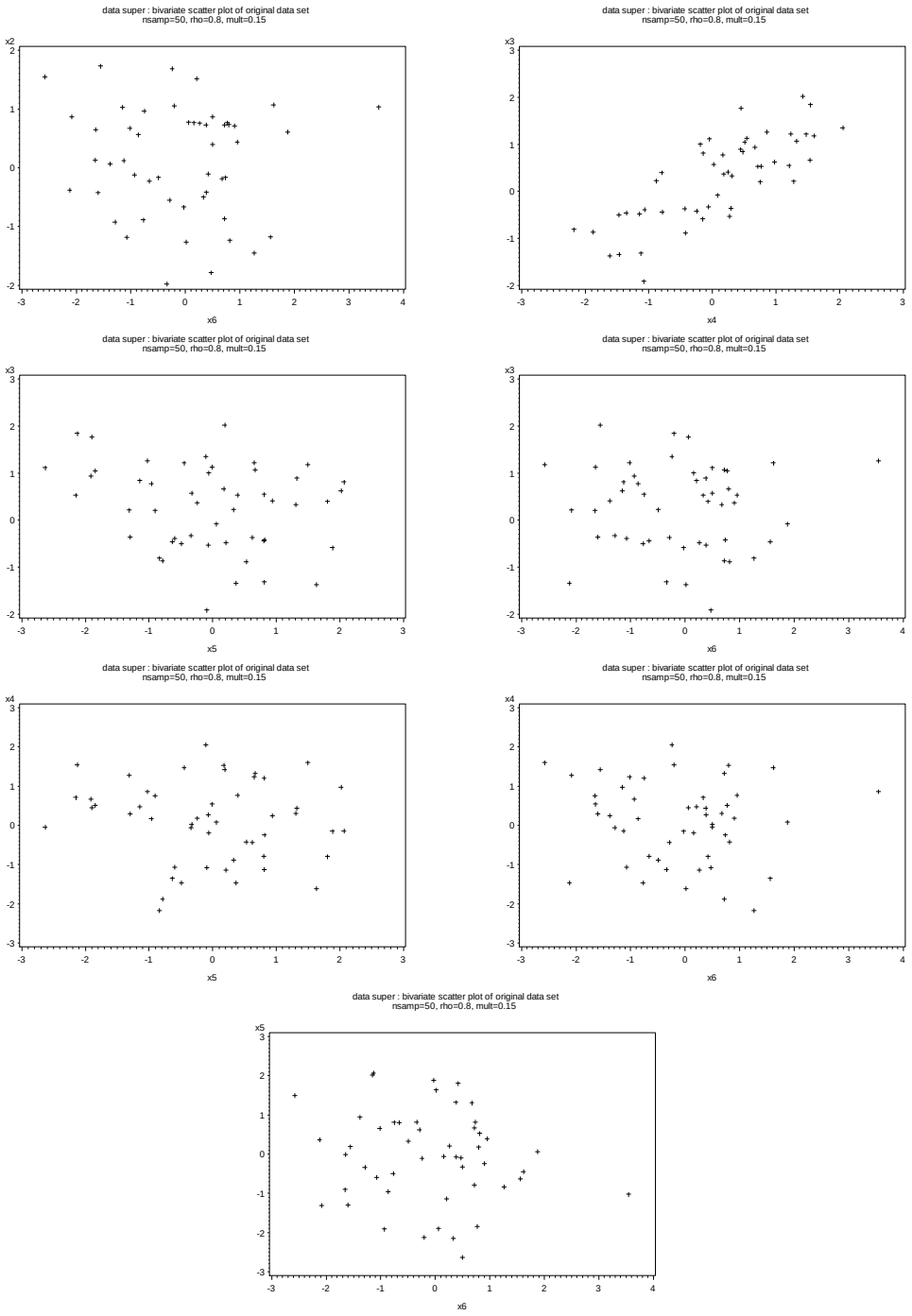

Figure 8.2 (*continued*)

As shown in figure 8.2, there are strong linear relationships between $X_1,...,X_4$ but not in the others.

Table 8.2 Table of variables entered in the model with the sum of squared residuals in 10 simulations

	Number of Variables in Model (sum of squared residuals)				
	2	3	4	5	6
Sim=1	2, 3 (17.26)	1 (10.42)	6 (13.75)	5 (14.75)	4 (25.35)
Sim=2	3, 4 (13.73)	1 (10.83)	2 (11.72)	6 (11.75)	5 (12.15)
Sim=3	2, 3 (18.12)	1 (13.90)	4 (12.37)	6 (13.88)	5 (15.94)
Sim=4	1, 4 (18.80)	2 (13.88)	6 (17.85)	5 (22.66)	3 (35.69)
Sim=5	1, 2 (15.97)	3 (11.67)	4 (11.46)	5 (12.80)	6 (26.89)
Sim=6	1, 2 (23.23)	3 (17.03)	6 (20.30)	5 (20.65)	4 (18.84)
Sim=7	3, 4 (15.81)	1 (12.55)	6 (15.22)	5 (15.95)	2 (48.13)
Sim=8	1, 4 (11.73)	2 (10.32)	3 (9.45)	6 (9.68)	5 (11.17)
Sim=9	2, 3 (14.65)	6 (20.54)	5 (24.24)	1 (59.05)	4 (124.10)
Sim=10	1, 4 (13.62)	6 (18.70)	5 (23.53)	3 (52.42)	2 (95.40)

Table 8.2 shows the results of performing forward selection for 10 simulated data sets. For each simulation the five columns give the variables entered at each step with the corresponding sum of squared residuals.

As seen in table 8.2, in 8 out of the 10 simulations, variables X_5, X_6 are entered later than the 3rd step. Since X_5, X_6 are generated from a multivariate normal distribution with no correlations and no specific equations, these two variables are entered late into the model. The first four variables ($X_1,..., X_4$) are entered early into the model but with no certain order.

As shown in figure 8.3, there is no certain order in which variables are entered into the equations, since there are no specific coefficients assigned to variables when the data are generated. Except for the last two simulations when X_6 is entered at the second step, the sum of squared residuals decreased from step 1 to step 2. A further decrease in the sum of squared residuals is seen at step 3 for three of the four simulations if one of the first four variables is entered.

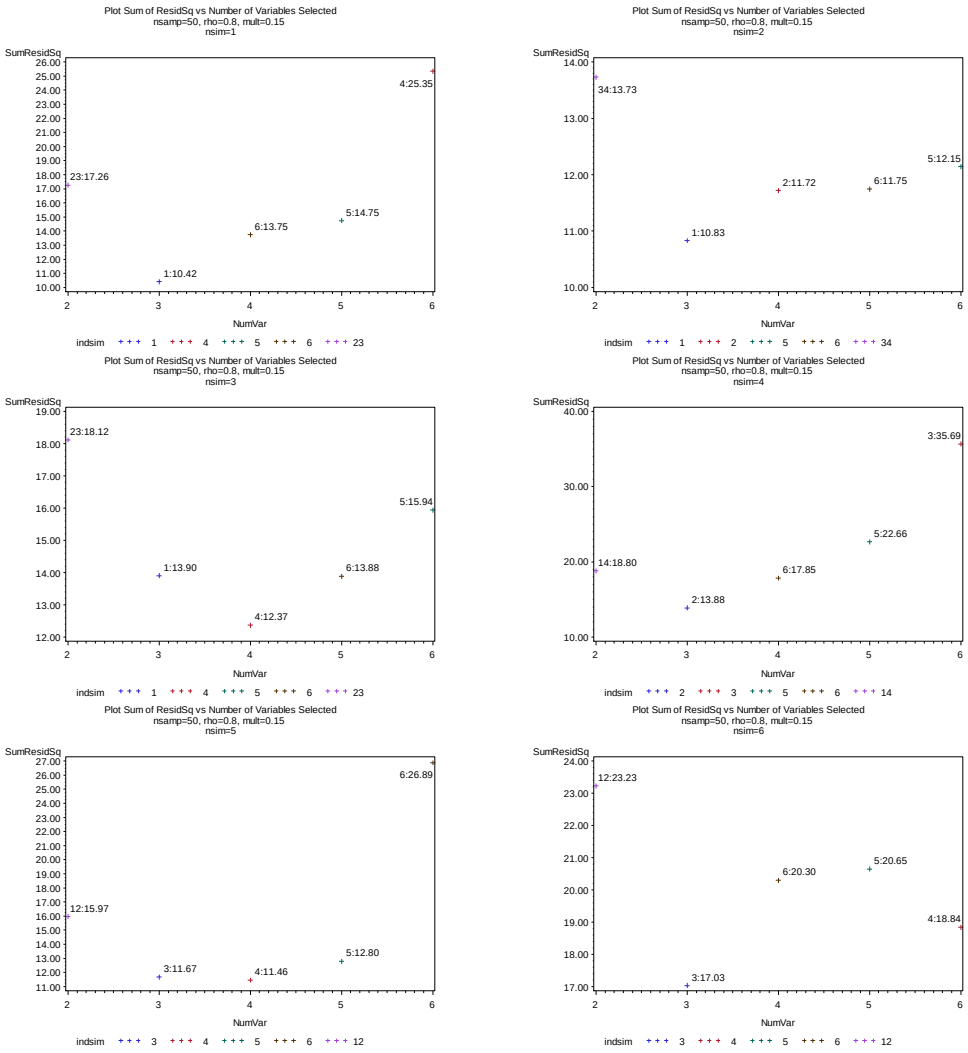

Figure 8.3 Plots of $\sum_{i=1}^{n} \hat{e}_{i1}^2$ versus the number of variables selected for 10 simulations of samples of $n = 50$, $\rho = 0.8$ between $X_1, ..., X_4$ and $\rho = 0$ between X_5, X_6 which are independent from $X_1, ..., X_4$, $mult = 0.15$ and no specific coefficients

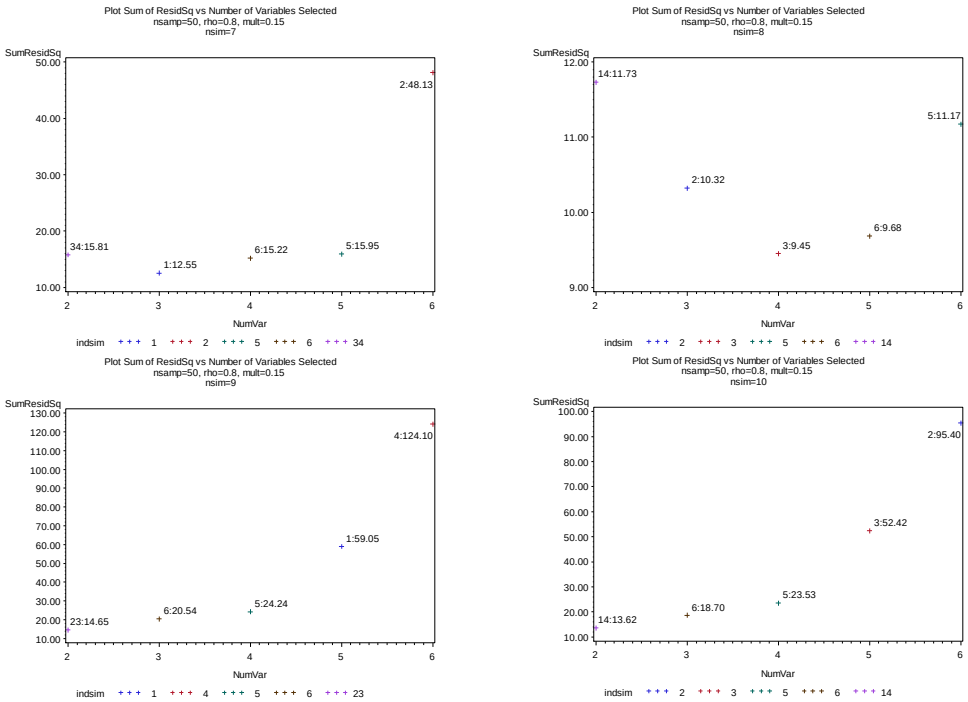

Figure 8.3 (*continued*)

8.2.2 Data generated with an equation.

(1) Data are generated with non-zero coefficients with low / high correlation between each variable.

The following example will show some results when variables are related by an equation. Sample data of 50 cases are generated with 4 variables. The 3 variables X_2, X_3, X_4 are obtained from $MN(\mathbf{0}, \mathbf{I})$ and X_1 is obtained by the equation

$$3X_1 + 2X_2 + X_3 + 4X_4 = 1, \tag{8.1}$$

with small additive error ($mult = 0.15$).

As shown in figure 8.4, there are strong linear relationships in plots of X_1 *vs* X_j, $j = 2,3,4$ but not in the other plots.

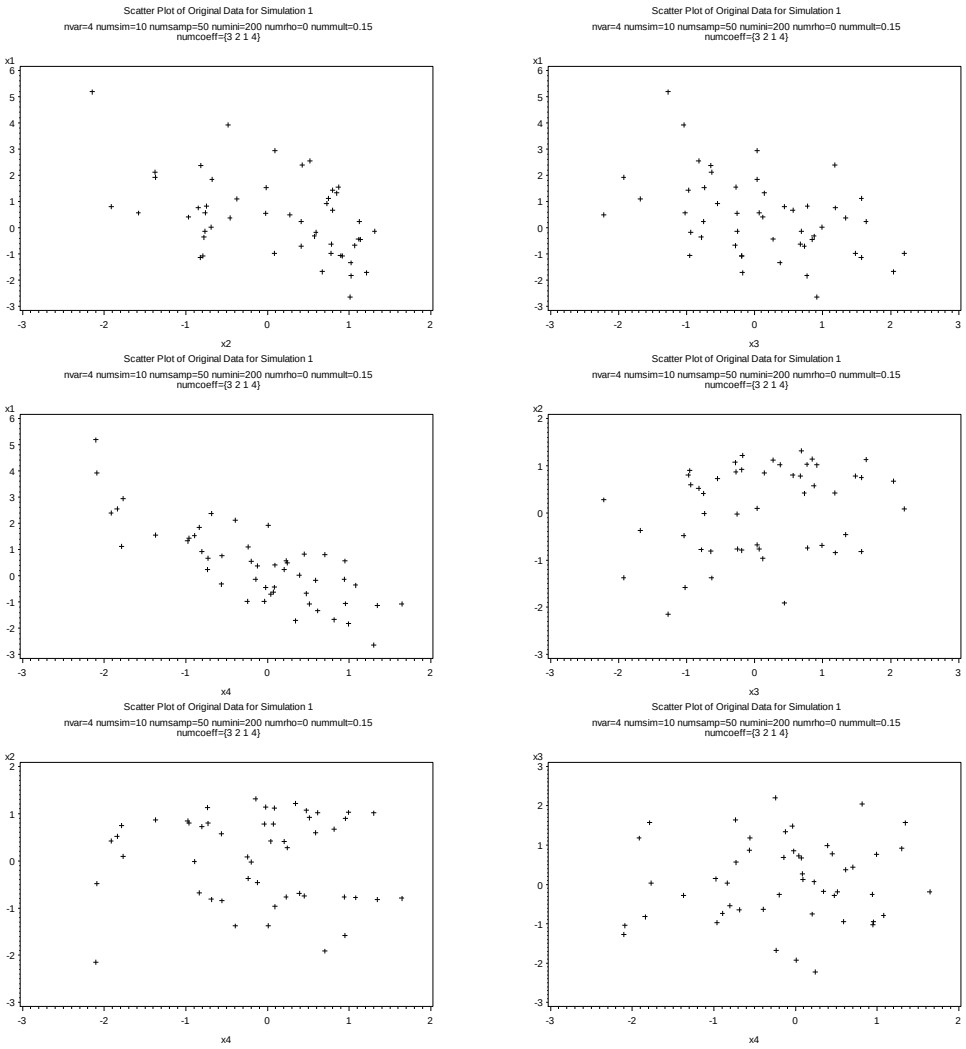

Figure 8.4 Scatter plots of the original data in Simulation 1 for sample of $n = 50$, $\rho = 0$ between $X_2, ..., X_4$, $mult = 0.15$ and $coeff = \{3, 2, 1, 4\}$

Table 8.3 Table of variables entered in the model with the sum of squared residuals in 10 simulations

	Number of Variables in Model (sum of squared residuals)		
	2	3	4
Sim=1	1, 4 (44.53)	2 (12.37)	3 (5.46)
Sim=2	1, 4 (37.44)	2 (8.17)	3 (4.24)
Sim=3	1, 4 (32.59)	2 (10.29)	3 (4.19)
Sim=4	1, 4 (38.30)	2 (9.91)	3 (3.94)
Sim=5	1, 4 (32.83)	2 (10.89)	3 (3.62)
Sim=6	1, 4 (24.11)	2 (8.09)	3 (3.91)
Sim=7	1, 4 (25.34)	2 (8.81)	3 (3.32)
Sim=8	1, 4 (28.44)	2 (6.75)	3 (2.96)
Sim=9	1, 4 (19.99)	2 (7.80)	3 (2.96)
Sim=10	1, 4 (22.04)	2 (6.85)	3 (4.32)

As shown in table 8.3, all simulations enter X_1 and X_4 first and then enter X_2. The variable X_3 is the last one to enter into the model. In fact X_4 has the largest coefficient in the equation and X_1 is highly related to all other variable; it is expected to enter these two variables first. When there are not many variables and the additive errors are small, the order of entering variables into the model would depend on the magnitude of coefficients.

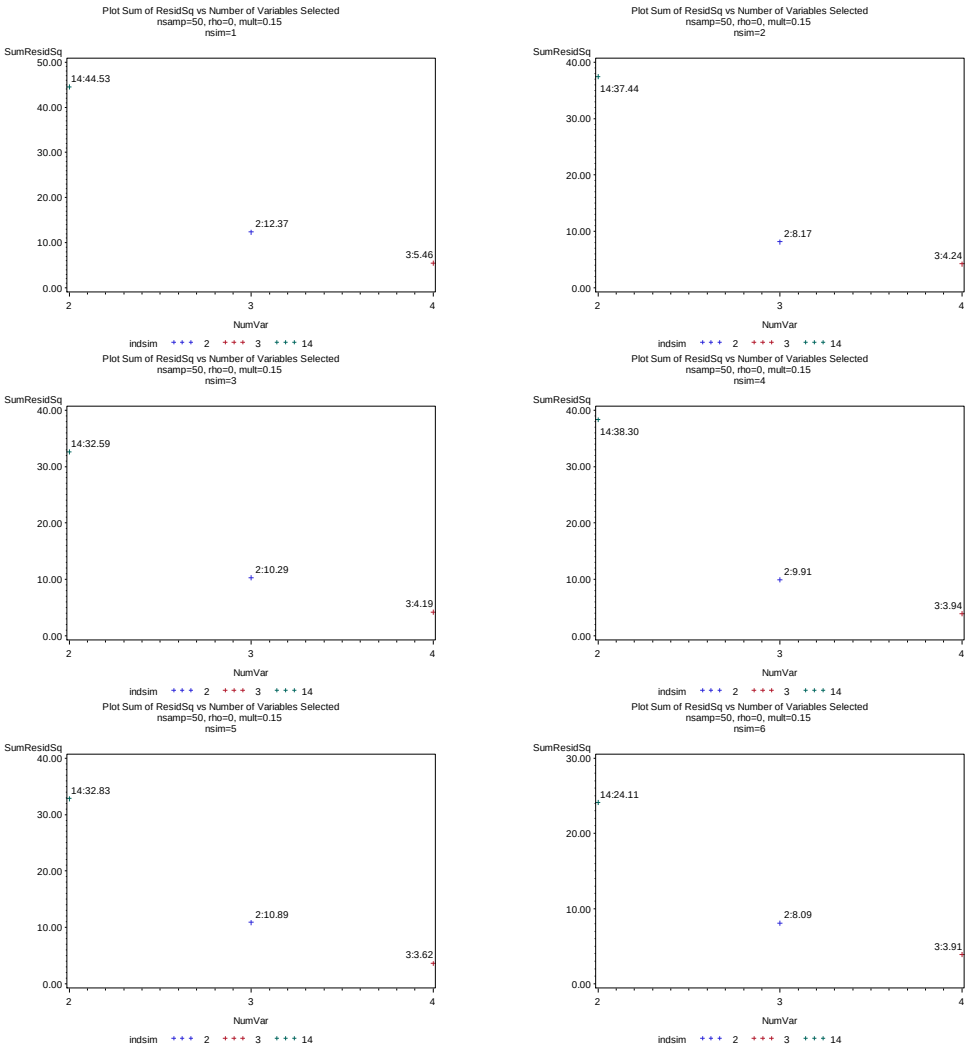

Figure 8.5 Plots of $\sum_{i=1}^{n} \hat{e}_{i1}^2$ versus the number of variables selected for sample of $n = 50$, $\rho = 0$ between $X_2, ..., X_4$, $mult = 0.15$ and $coeff = \{3, 2, 1, 4\}$ for ten simulations

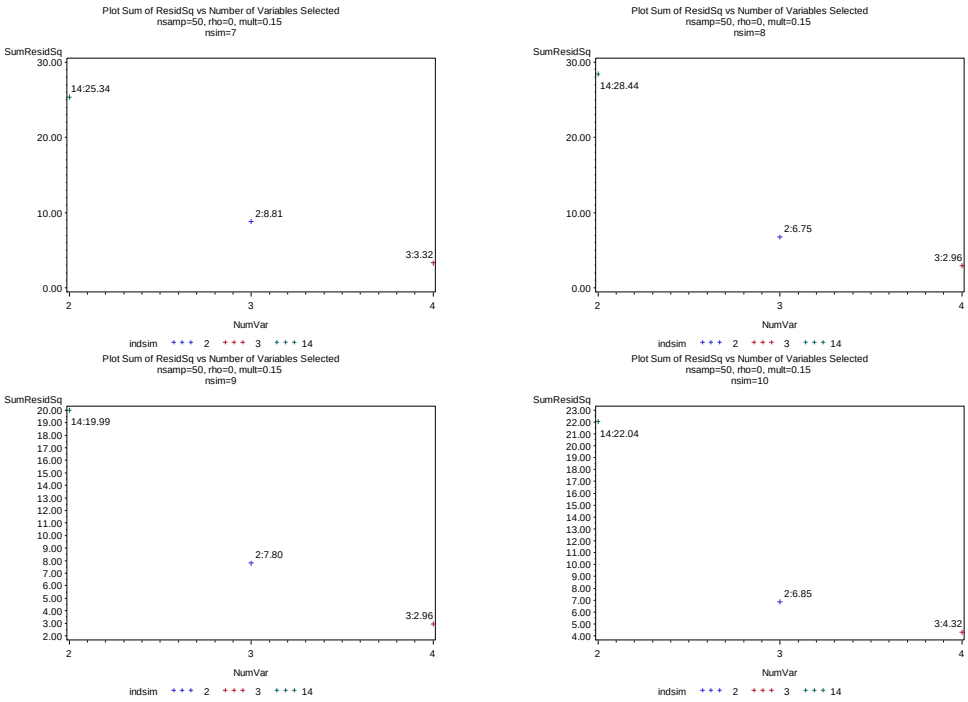

Figure 8.5 (*continued*)

In figure 8.5, $\sum_{i=1}^{n} \hat{e}_{i1}^2$ decreases as each variable is entered into the model. It is not guaranteed that $\sum_{i=1}^{n} \hat{e}_{i1}^2$ gets smaller when a variable is entered, but a decrease is one of the indications to include that variable in the model.

The following example will show some results when variables have high correlation ($\rho = 0.8$) between each other. Sample data are generated as described in the previous example (eq. 8.1) with $n = 50$, $\rho = 0.8$ between $X_2, ..., X_4$, $mult = 0.15$ and $coeff = \{3, 2, 1, 4\}$.

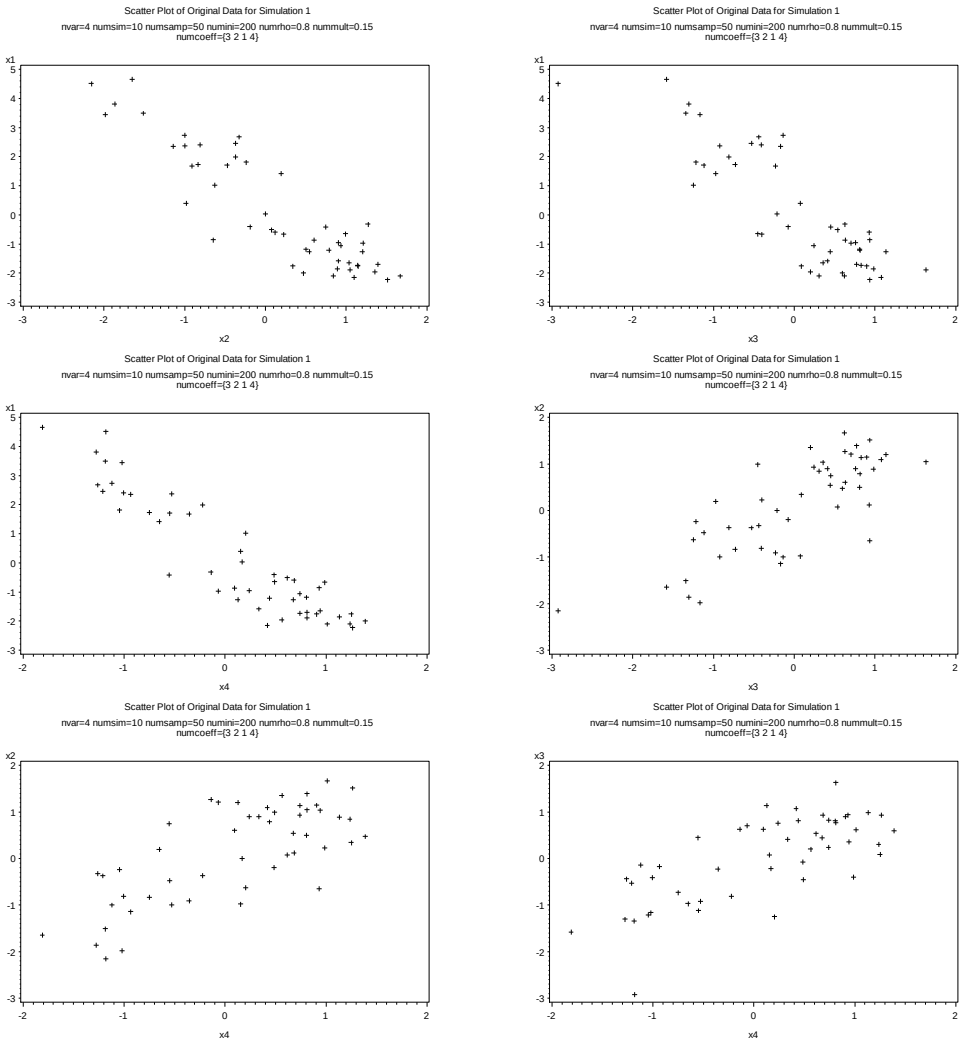

Figure 8.6 Scatter plots of the original data in Simulation 1 for sample of $n = 50$, $\rho = 0.8$ between $X_2, ..., X_4$, $mult = 0.15$ and $coeff = \{3, 2, 1, 4\}$

As shown in figure 8.6, there are strong linear relationships between all pairs of variables.

As shown in table 8.4, the results are very similar to those in table 8.3. The variable with the largest coefficient enters first and that with the smallest coefficient is entered last. Again the sum of squared residuals decreases as each variable is entered.

Table 8.4 Table of variables entered in the model with the sum of squared residuals in 10 simulations

	Number of Variables in Model (sum of squared residuals)		
	2	3	4
Sim=1	1, 4 (51.50)	2 (7.56)	3 (5.11)
Sim=2	1, 4 (76.13)	2 (7.92)	3 (4.24)
Sim=3	1, 4 (34.66)	2 (6.13)	3 (3.78)
Sim=4	1, 4 (54.92)	2 (6.64)	3 (3.72)
Sim=5	1, 4 (38.88)	2 (5.56)	3 (4.20)
Sim=6	1, 4 (42.22)	2 (6.94)	3 (3.45)
Sim=7	1, 4 (29.66)	2 (6.49)	3 (2.89)
Sim=8	1, 4 (41.10)	2 (4.06)	3 (3.38)
Sim=9	1, 4 (50.96)	2 (4.75)	3 (2.92)
Sim=10	1, 4 (44.11)	2 (5.72)	3 (4.41)

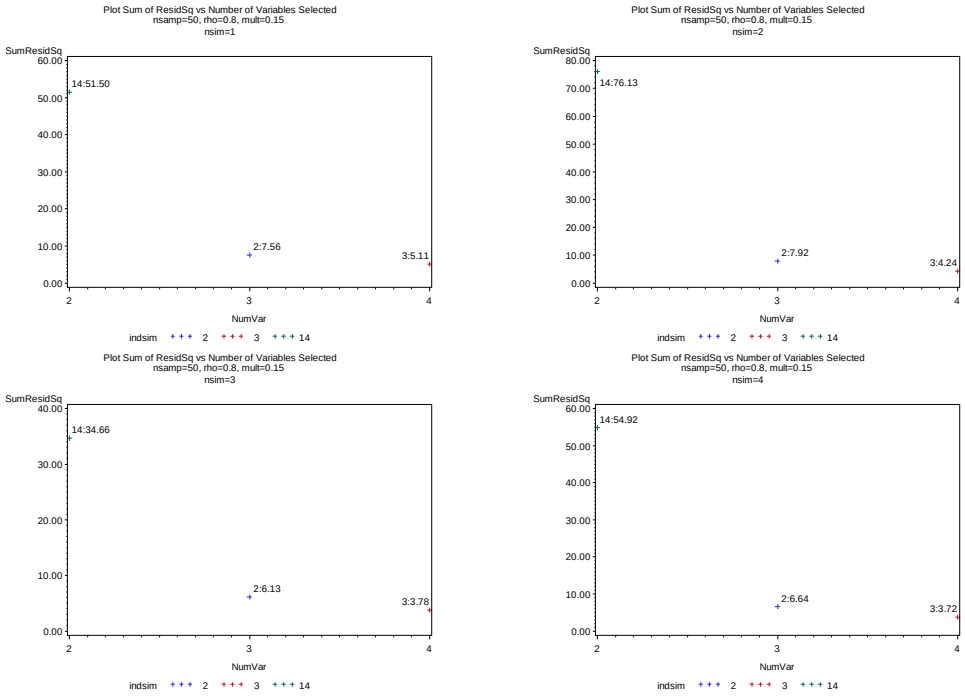

Figure 8.7 Plots of $\sum_{i=1}^{n} \hat{e}_{i1}^2$ versus the number of variables selected for sample of $n = 50$, $\rho = 0.8$ between $X_2, ..., X_4$, $mult = 0.15$ and $coeff = \{3, 2, 1, 4\}$ for ten simulations

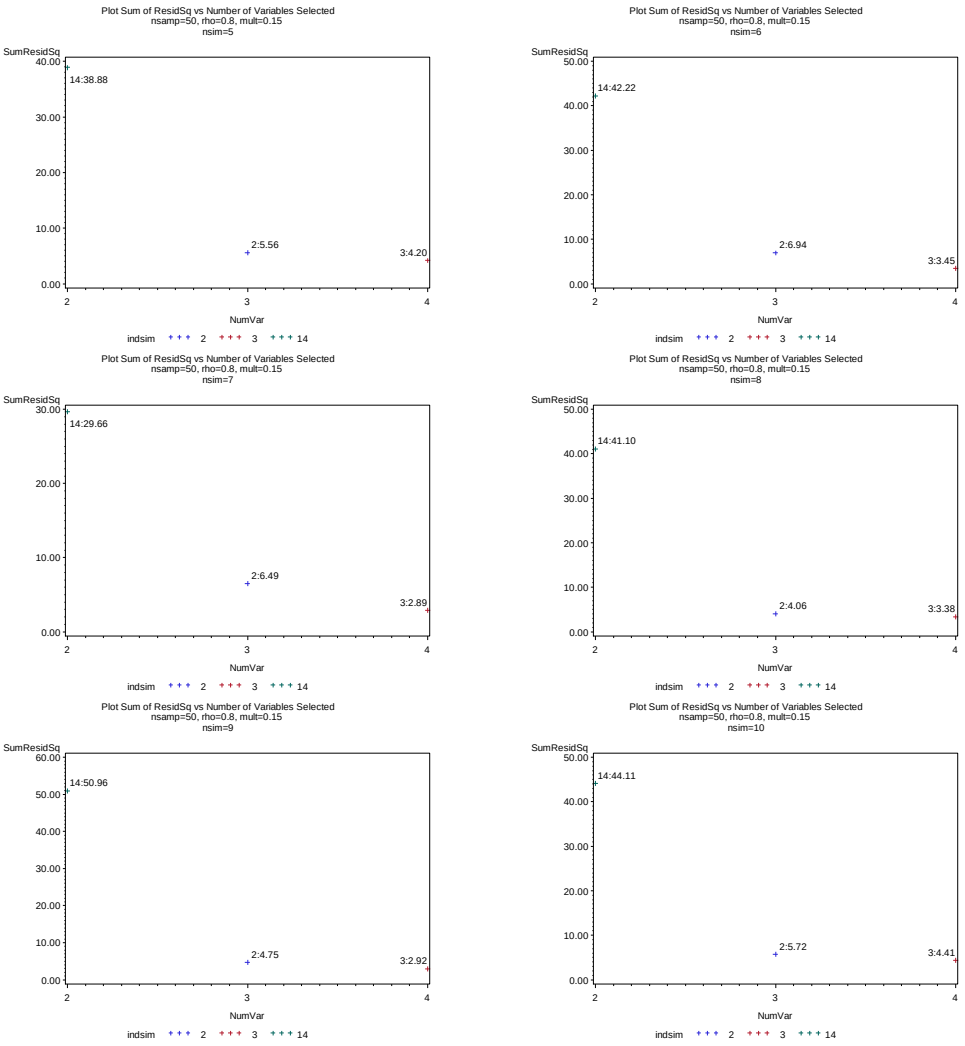

Figure 8.7 (*continued*)

All plots of $\sum_{i=1}^{n} \hat{e}_{i1}^2$ in this case show decreasing trends which suggest considering to enter all variables into the model.

(2) Data are generated so that one or more of the variables have zero coefficients with low / high correlation between each variable.

The following example will show some results when variables have no correlation between each other and one coefficient is zero. Sample data of 50 cases are generated with 4 variables. The 3 variables X_2, X_3, X_4 are obtained from $MN(\mathbf{0}, \mathbf{I})$ and X_1 is obtained by the equation

$$3X_1 + 2X_2 + 0X_3 + 4X_4 = 1, \tag{8.1}$$

with small additive error ($mult = 0.15$).

In theory, the variable with a zero coefficient should not be in the true model. Therefore the variable X_3 is expected to be entered late into the model.

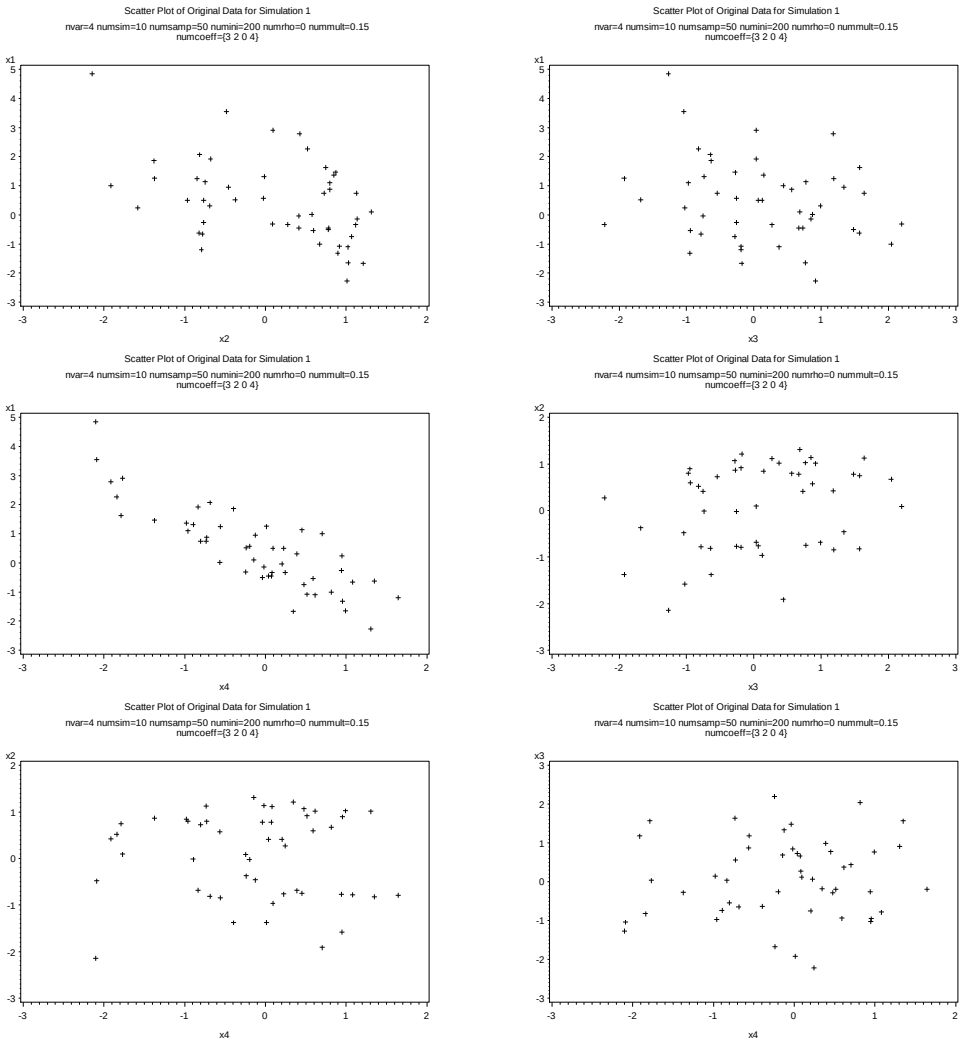

Figure 8.8 Scatter plots of the original data in Simulation 1 for sample of $n = 50$, $\rho = 0$ between $X_2, ..., X_4$, $mult = 0.15$ and $coeff = \{3, 2, 0, 4\}$

As shown in figure 8.8, there is a strong linear relationship in the plot of X_1 *vs* X_4

but not in the others.

Table 8.5 Table of variables entered in the model with the sum of squared residuals in 10 simulations

	Number of Variables in Model (sum of squared residuals)		
	2	3	4
Sim=1	1, 4 (42.61)	2 (5.27)	3 (6.45)
Sim=2	1, 4 (39.29)	2 (4.36)	3 (4.93)
Sim=3	1, 4 (44.19)	2 (4.04)	3 (5.10)
Sim=4	1, 4 (46.63)	2 (3.67)	3 (4.31)
Sim=5	1, 4 (42.12)	2 (3.74)	3 (4.77)
Sim=6	1, 4 (38.84)	2 (3.44)	3 (4.50)
Sim=7	1, 4 (30.78)	2 (3.46)	3 (4.33)
Sim=8	1, 4 (40.93)	2 (2.70)	3 (3.28)
Sim=9	1, 4 (48.54)	2 (2.59)	3 (3.20)
Sim=10	1, 4 (90.07)	2 (3.51)	3 (4.25)

As shown in table 8.5, all simulations suggest entering X_1, X_4 first and then enter X_2 next. The variable X_3 is always the last to be entered into the model. The sum of squared residuals $\sum_{i=1}^{n} \hat{e}_{i1}^2$ decreases as X_1, X_4, X_2 are entered and increases when X_3, whose true coefficient is zero, is entered into the model. It is suggesting to considering dropping X_3 from the model and this is the correct decision.

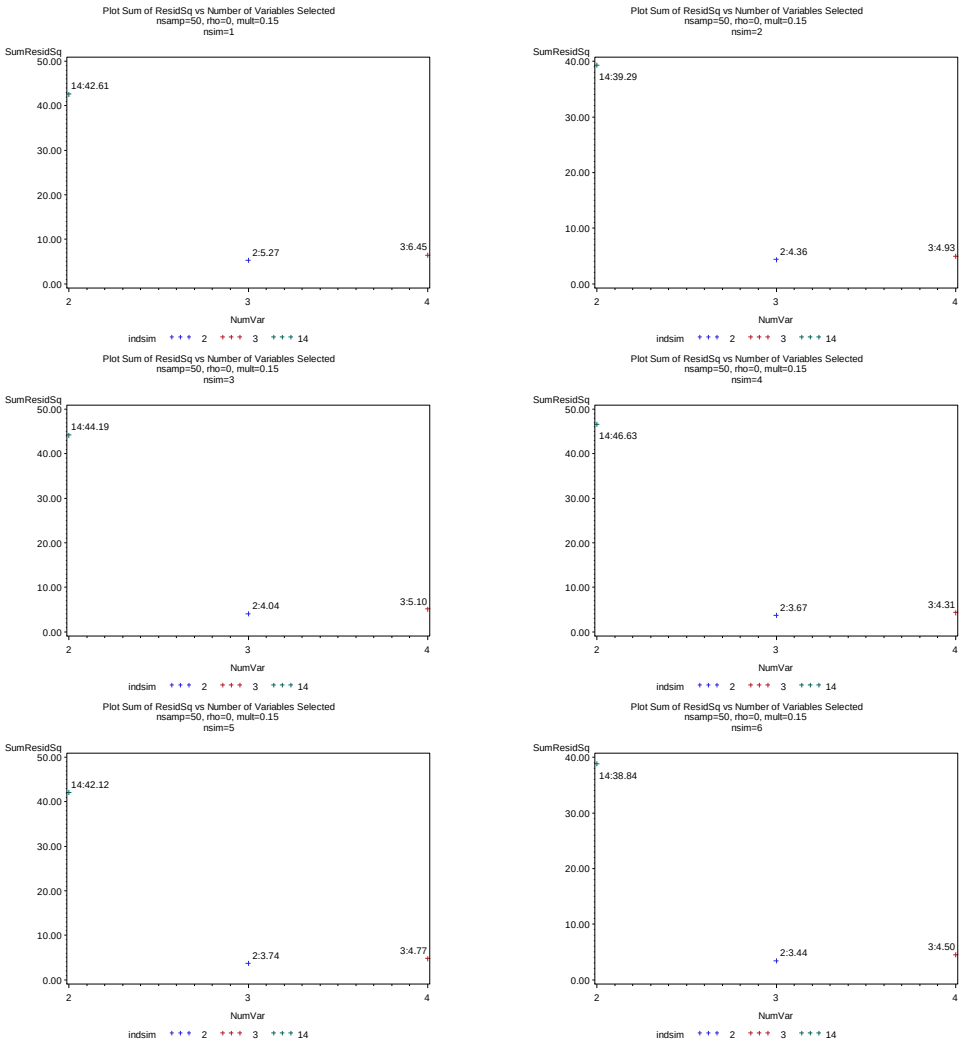

Figure 8.9 Plots of $\sum_{i=1}^{n} \hat{e}_{i1}^2$ versus the number of variables selected for sample of $n = 50$, $\rho = 0$ between $X_2, ..., X_4$, $mult = 0.15$ and $coeff = \{3, 2, 0, 4\}$ for ten simulations

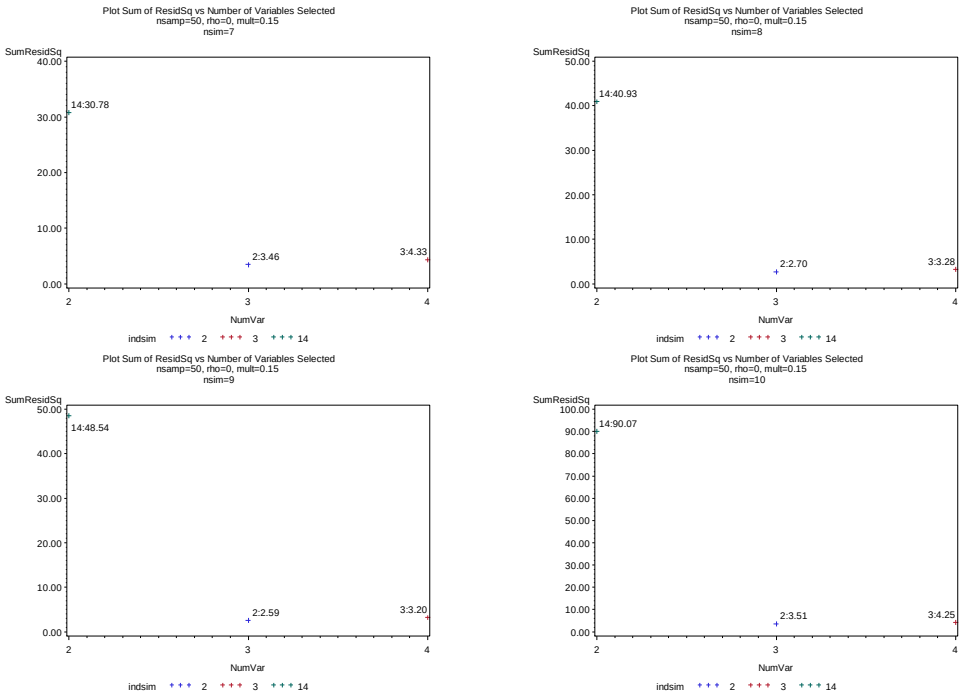

Figure 8.9 (*continued*)

Figure 8.9 also suggests not to include X_3 in the model.

The following example will show some results when variables have high correlation ($\rho = 0.8$) between each other and one coefficient is zero. Sample data are generated as described in the previous example (eq. 8.1) with $n = 50$, $\rho = 0.8$ between $X_2, ..., X_4$, $mult = 0.15$ and $coeff = \{3, 2, 0, 4\}$.

As shown in figure 8.10, there are strong linear relationships in all plots.

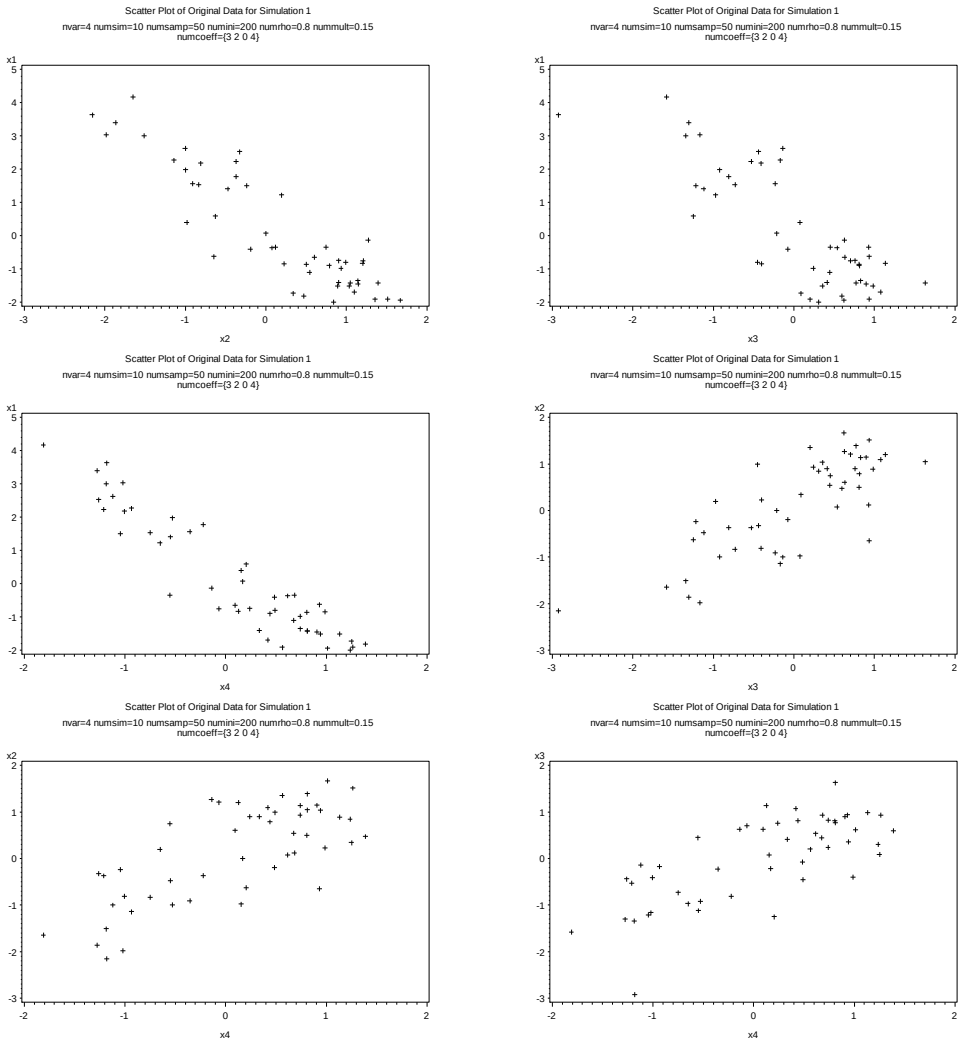

Figure 8.10 Scatter plots of the original data in Simulation 1 for sample of $n = 50$, $\rho = 0.8$ between $X_2, ..., X_4$, $mult = 0.15$ and $coeff = \{3, 2, 0, 4\}$

Table 8.6 shows a very similar result as in table 8.5, except that in only 8 out of the 10 simulations the $\sum_{i=1}^{n} \hat{e}_{i1}^2$ increases when X_3 is entered. However a large majority of the simulations suggest to leave X_3 out of the model.

Table 8.6 Table of variables entered in the model with the sum of squared residuals in 10 simulations

	Number of Variables in Model (sum of squared residuals)		
	2	3	4
Sim=1	1, 4 (39.00)	2 (5.05)	3 (5.27)
Sim=2	1, 4 (59.74)	2 (4.41)	3 (4.80)
Sim=3	1, 4 (26.66)	2 (3.90)	3 (4.02)
Sim=4	1, 4 (40.35)	2 (3.50)	3 (3.78)
Sim=5	1, 4 (34.76)	2 (4.35)	3 (7.05)
Sim=6	1, 4 (34.55)	2 (3.43)	3 (3.28)
Sim=7	1, 4 (25.99)	2 (3.34)	3 (2.99)
Sim=8	1, 4 (33.44)	2 (3.12)	3 (3.95)
Sim=9	1, 4 (39.07)	2 (2.58)	3 (2.84)
Sim=10	1, 4 (38.96)	2 (3.65)	3 (5.36)

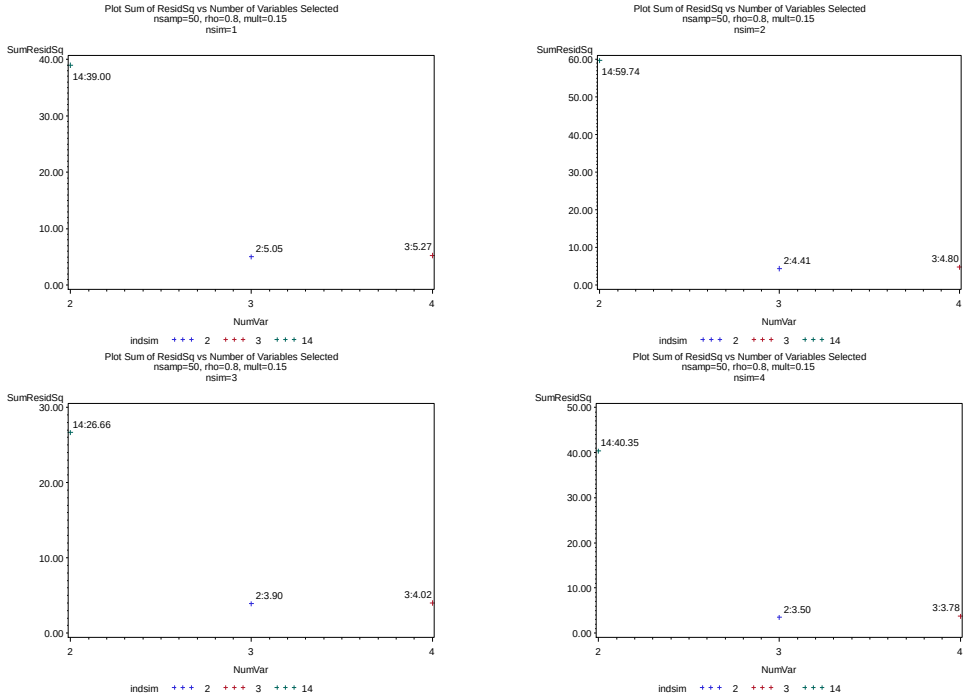

Figure 8.11 Plots of $\sum_{i=1}^{n} \hat{e}_{i1}^2$ versus the number of variables selected for sample of $n = 50$, $\rho = 0.8$ between $X_2, ..., X_4$, $mult = 0.15$ and $coeff = \{3, 2, 0, 4\}$ for ten simulations

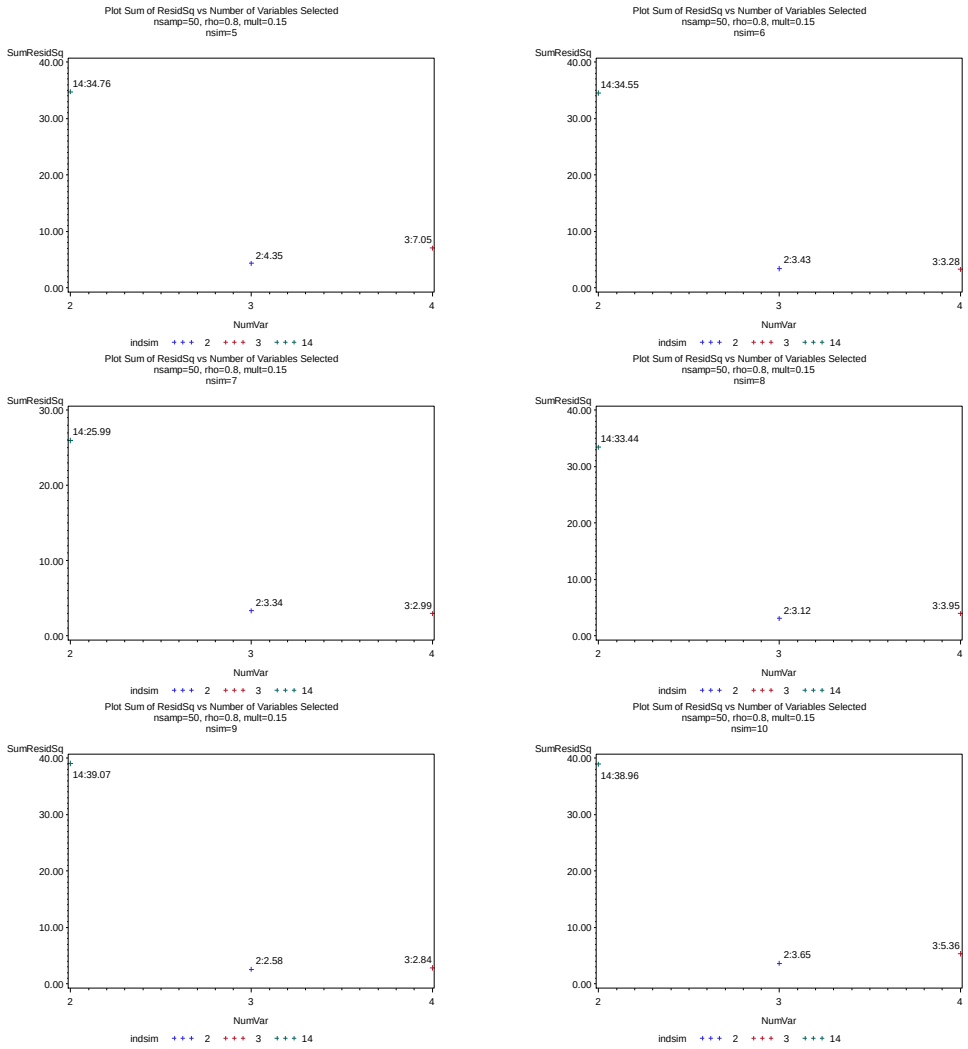

Figure 8.11 (continued)

Figure 8.11 also suggests that in most of the simulations, the model should only include X_1, X_2, X_4.

The following example will show some results when variables have no correlation between each other and one coefficient is zero. Sample data of 50 cases are generated with 6 variables. The 5 variables $X_2,...,X_6$ are obtained from $MN(\mathbf{0},\mathbf{I})$ and X_1 is obtained by the equation

$$2X_1+4X_2+X_3+0X_4+2X_5+0X_6=1,$$

with small additive error ($mult=0.15$).

In theory, the variables with a zero coefficient should not be in the model. Therefore the variables X_4, X_6 are expected to be entered late into the model. As shown in figure 8.12, there is a strong linear relationship in the plot of X_1 *vs* X_2 and X_1 *vs* X_5 but not in the others.

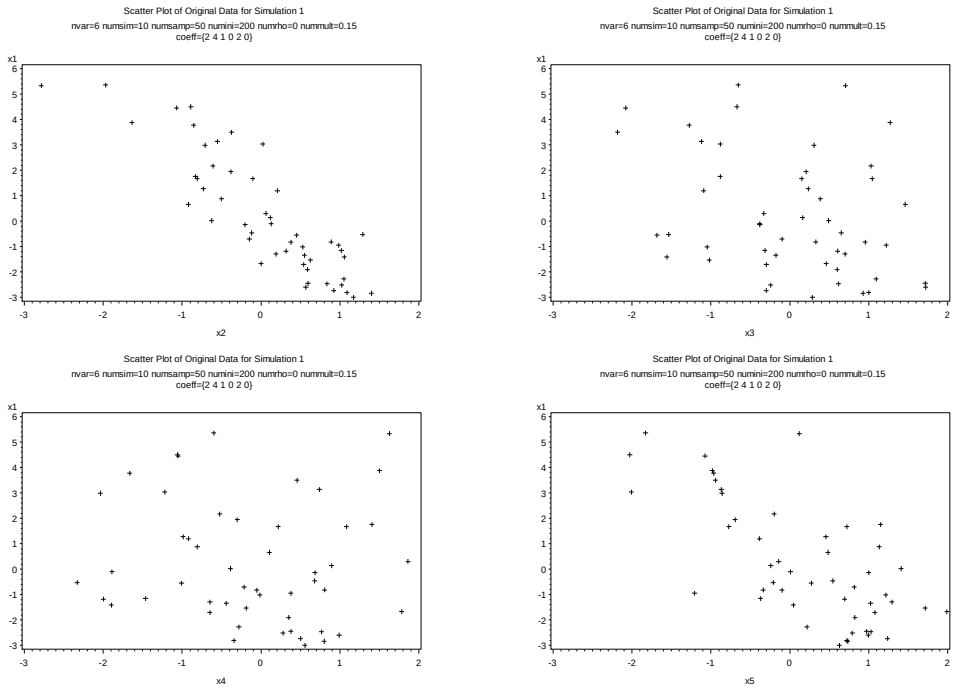

Figure 8.12 Scatter plots of the original data in Simulation 1 for sample of $n=50$, $\rho=0$ between $X_2,...,X_6$, $mult=0.15$ and $coeff=\{2,4,1,0,2,0\}$

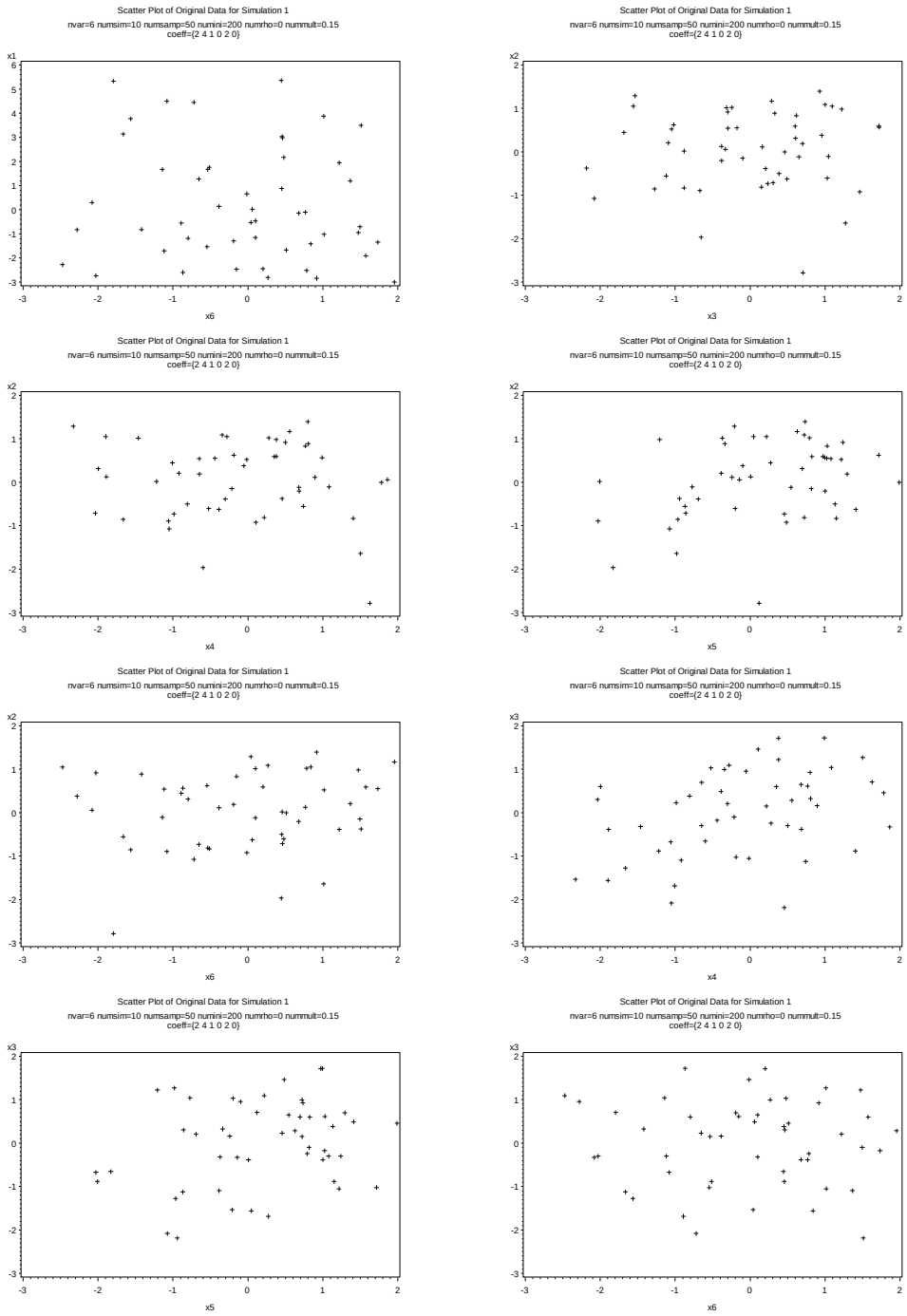

Figure 8.12 (*continued*)

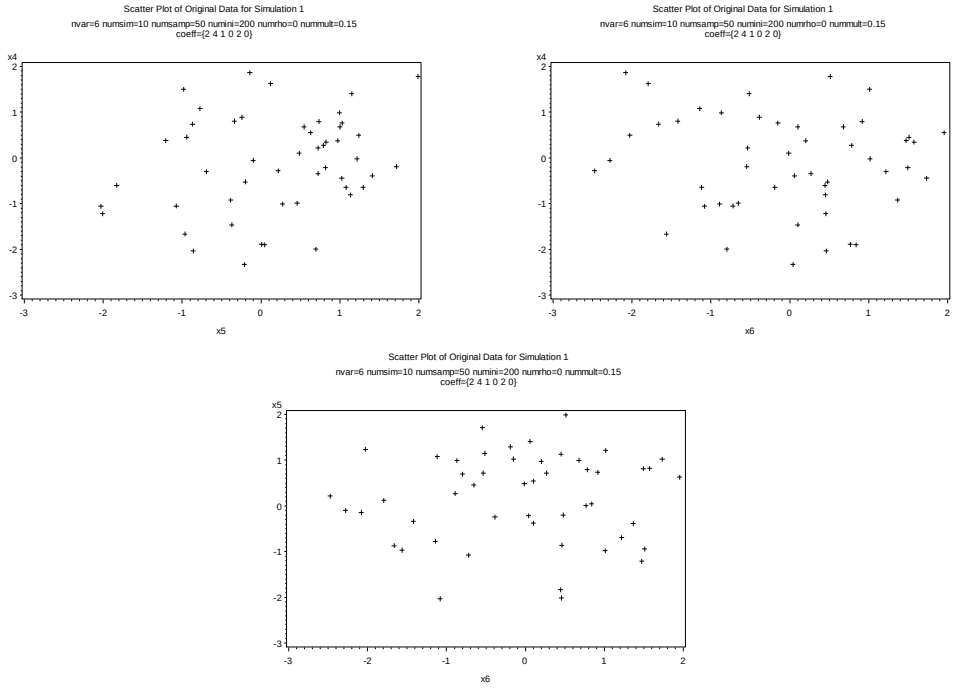

Figure 8.12 (*continued*)

As shown in table 8.7, all simulations suggest entering X_1, X_2 first and then all but one enter X_3 next. In 9 out of the 10 simulations, the variable X_5 is always the second last to be entered into the model. The sum of squared residuals

Table 8.7 Table of variables entered in the model with the sum of squared residuals in 10 simulations

	Number of Variables in Model (sum of squared residuals)				
	2	3	4	5	6
Sim=1	1, 2 (89.51)	3 (50.16)	4 (54.21)	5 (9.93)	6 (12.04)
Sim=2	1, 2 (146.71)	3 (94.00)	4 (113.71)	5 (6.98)	6 (6.40)
Sim=3	1, 2 (178.95)	6 (83.09)	5 (14.31)	3 (7.00)	4 (8.66)
Sim=4	1, 2 (238.49)	3 (85.96)	6 (101.86)	5 (8.02)	4 (9.37)
Sim=5	1, 2 (84.66)	3 (42.98)	4 (57.98)	5 (7.03)	6 (7.89)
Sim=6	1, 2 (113.30)	3 (59.60)	6 (70.56)	5 (7.42)	4 (8.10)
Sim=7	1, 2 (182.53)	3 (85.47)	6 (96.51)	5 (8.30)	4 (9.04)
Sim=8	1, 2 (91.67)	3 (49.69)	4 (52.21)	5 (4.38)	6 (5.09)
Sim=9	1, 2 (152.58)	3 (56.40)	4 (70.95)	5 (4.33)	6 (5.19)
Sim=10	1, 2 (76.07)	3 (65.50)	6 (80.33)	5 (5.12)	4 (6.10)

$\sum_{i=1}^{n} \hat{e}_{i1}^2$ decreases as X_1, X_2, X_3 are entered. In all simulations, the last variable entered into the model is always X_4 (5 times) or X_6 (5 times), which have zero coefficients. In the 3rd simulation, X_6 is the 3rd variable to enter and X_5, X_3, X_4 are the last variables to enter into the model. This is the only simulation that when X_6 is entered $\sum_{i=1}^{n} \hat{e}_{i1}^2$ decreases. In all other simulations, when X_4 and X_6 are entered into the model $\sum_{i=1}^{n} \hat{e}_{i1}^2$ increases, which suggests not to include these variables in the model. X_5 always enters into the model late, however when X_5 is entered $\sum_{i=1}^{n} \hat{e}_{i1}^2$ always decreases substantially, which suggests to include X_5 in the model.

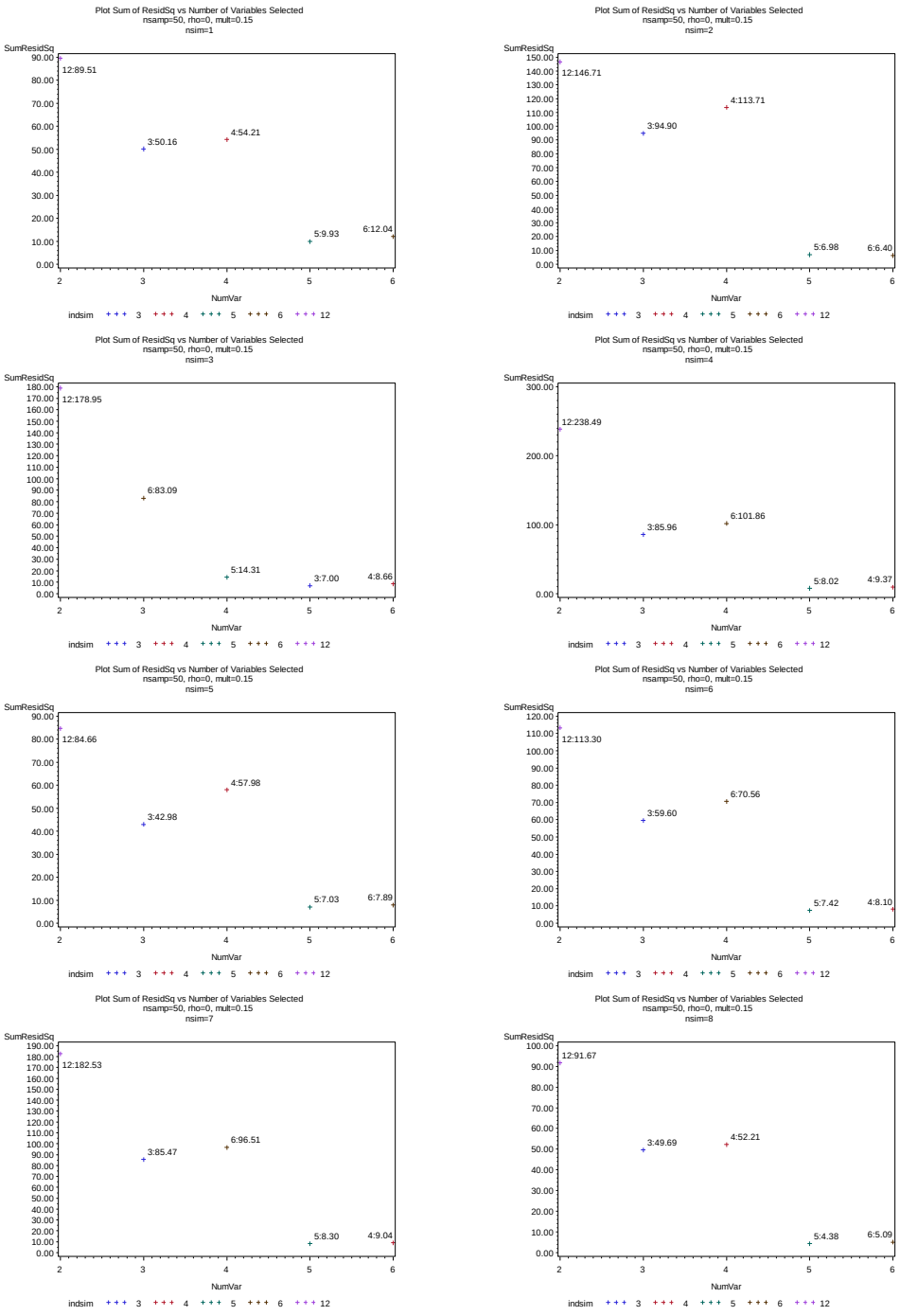

Figure 8.13 Plots of $\sum_{i=1}^{n} \hat{e}_{i1}^2$ versus the number of variables selected for sample of $n = 50$, $\rho = 0$ between $X_2, ..., X_6$, $mult = 0.15$ and $coeff = \{2,4,1,0,2,0\}$ for ten simulations

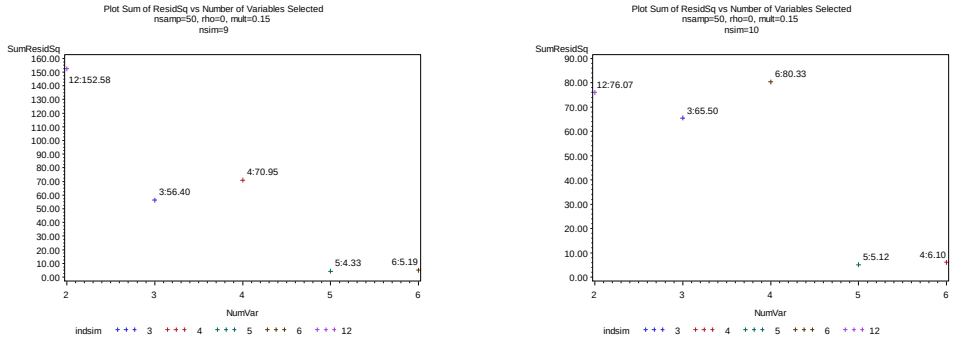

Figure 8.13 (*continued*)

The forward selection method for entering variables based on the sum of squared residuals obtained by RMA appears to be a useful method to advise on selecting appropriate variables to include in the linear model. However, just as in the use of forward selection in OLS regression, there is no guarantee that selection of the true model will be obtained.

The simulations suggest including a variable at a step when there is a decrease in the sum of squared residuals. When the sum of squared residuals increases from one step to the next, the subsequent variables would generally not be included in the model unless there is another substantial decreasing in the sum of squared residuals at a later step.

Further research is needed in this area.

Chapter 9: IRIS VIRGINICA DATA APPLICATION

9.1 Iris Virginica Data

The iris data (appendix 9.1) is a multivariate data set introduced by Sir Ronald Aylmer Fisher (1936) as an example for discriminant analysis ("Iris flower data"). Edgar Anderson collected the data on iris flowers of three related species. Four variables – the sepal length (X_1), sepal width (X_2), petal length (X_3) and petal width (X_4) were measured in centimeters for each sample.

50 samples of Iris Virginica (one of the three species of iris flowers) data will be considered in this chapter.

Table 9.1 Descriptive statistics of each variable

Statistics	X_1	X_2	X_3	X_4
Mean	6.5880	2.9740	5.5520	2.0260
StdDev	0.6359	0.3225	0.5519	0.2747
Median	6.5000	3.0000	5.5500	2.0000

In table 9.1, the descriptive statistics of each variable are obtained. The mean and median of each variable are close and X_1, X_3 have higher standard deviation than X_2, X_4.

Table 9.2 Pearson correlation coefficients between each variable

	X_1	X_2	X_3	X_4
X_1	1.0000	0.4572	0.8642	0.2811
X_2	0.4572	1.0000	0.4010	0.5378
X_3	0.8642	0.4010	1.0000	0.3221
X_4	0.2811	0.5377	0.3221	1.0000

As table 9.2 shows, variable X_1 and X_3 have the highest correlation (0.8642) among all pairs of variables.

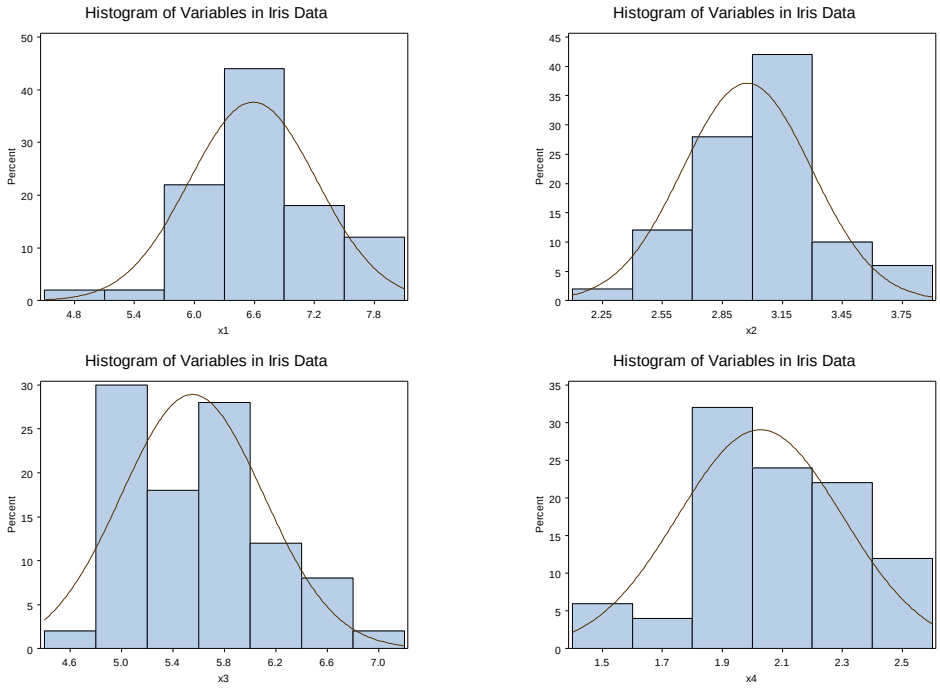

Figure 9.1 Histograms of each variable of original Iris Virginica data

Histograms of X_1 and X_2 are more symmetric than histograms of X_3 and X_4 as shows in figure 9.1.

Scatter plots of the original data are obtained to start understanding the data and correlations between variables in figure 9.2.

As figure 9.2 indicates, there might be some linear relationship between X_1 & X_2, and X_2 & X_3. X_1 and X_3 are clearly linear correlated.

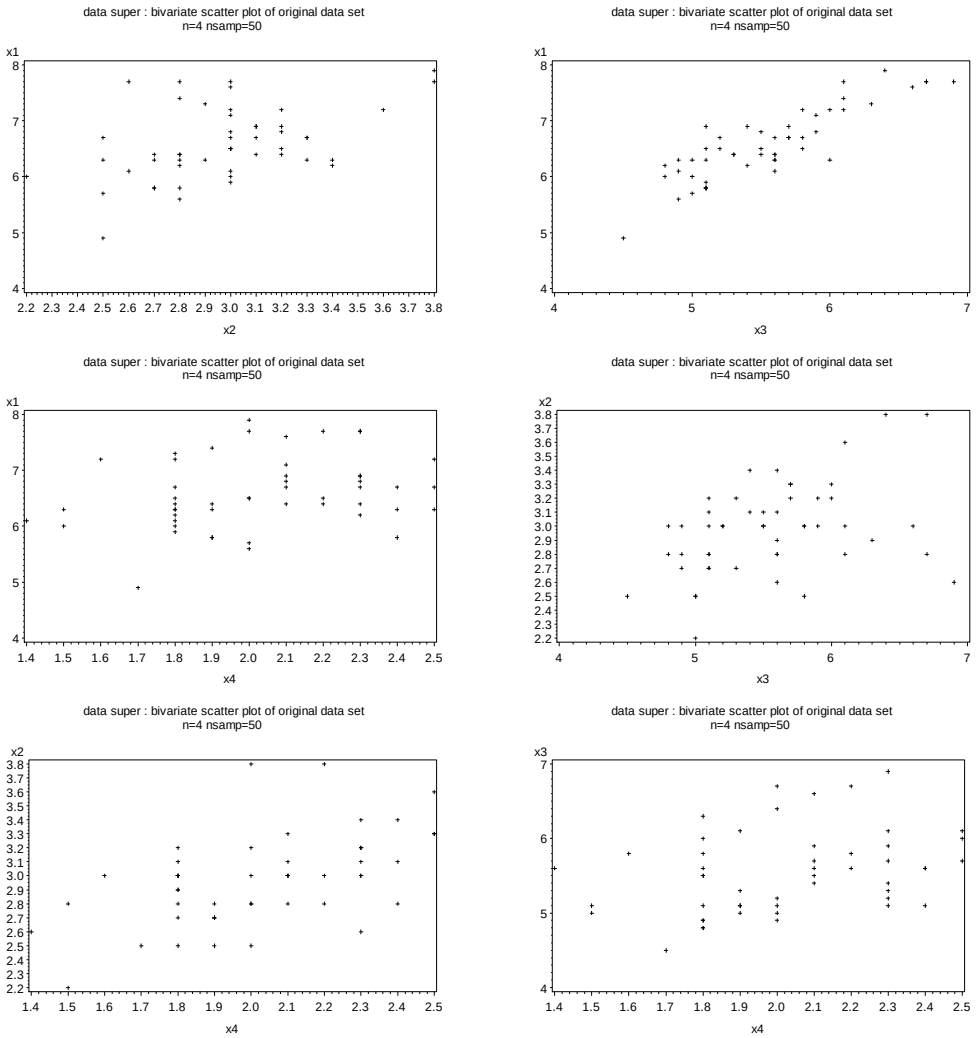

Figure 9.2 Scatter plots of original Iris Virginica data

If we would like to investigate the relationship among these four variables and if none of them are viewed as the dependent variable, and since there might be measurement error related to each of the variables, RMA regression would be appropriate to use.

9.2 Coefficient Estimates

Table 9.3 The coefficient estimates in canonical form with R-squared for the four OLS regression

X_1	X_2	X_3	X_4	R-squared
1.4288	-0.4720	-1.3510	0.2425	0.7652
-0.2627	1.1986	0.0833	-0.6397	0.3943
-1.5977	0.1769	2.1504	-0.4637	0.7551
0.1851	-0.8769	-0.2992	1.9990	0.3136

These OLS estimates in table 9.3 with the largest R-squared (1.4288, -0.4720, -1.3510, 0.2425) will be used to generate 200 initial values to perform the non-linear optimization. The RMA coefficient estimates obtained are shown in table 9.4.

Table 9.4 RMA coefficient estimates for each variable and objective function

$\hat{a}_1$	$\hat{a}_2$	$\hat{a}_3$	$\hat{a}_4$	Min Obj Func
-9.5109	10.2131	10.0551	-11.1262	3.4395

9.3 Inferences

As discussed in Chapter 5, the bootstrap approach is adopted to calculate confidence intervals for the coefficients in RMA regression.

The Normal, Percentile (PCTL), BC, BCa, Hybrid bootstrap and Jackknife confidence intervals are shown in table 9.5.

Table 9.5 Lower and upper 95% confidence limits for each coefficient for the six bootstrap confidence intervals

	a_1		a_2		a_3		a_4	
Method	Lower	Upper	Lower	Upper	Lower	Upper	Lower	Upper
Normal	-5828.60	4328.16	-4900.53	6122.44	-4667.56	6281.41	-6600.77	5249.45
PCTL	-24.78	10344.16	-10589.75	46.68	-10936.56	25.69	-50.79	11512.71
BC	-794.45	-4.85	4.09	23899.61	4.57	876.15	-22722.95	-4.14
BCa	-794.45	-5.02	4.18	23899.61	4.59	876.15	-22722.95	-4.23
Hybrid	-10363.19	5.76	-26.26	10610.18	-5.58	10956.67	-11534.97	28.53
Jackknife	-421.77	75.60	-79.07	455.36	-79.63	437.78	-494.60	88.47

As seen in table 9.5, the BC and BCa methods provide almost the same confidence intervals for each variable. Except for the Jackknife method, the confidence intervals' widths are very large. Confidence interval widths (CIW) are calculated for each method in table 9.6.

Table 9.6 Confidence interval width (CIW) for each coefficient for the six bootstrap confidence intervals

Method	CIW for a_1	CIW for a_2	CIW for a_3	CIW for a_4
Normal	10156.76	11022.97	10948.97	11850.22
PCTL	10368.94	10636.43	10962.25	11563.50
BC	789.60	23895.52	871.58	22718.81
BCa	789.43	23895.43	871.56	22718.72
Hybrid	10368.95	10636.44	10962.25	11563.50
Jackknife	497.37	534.43	517.41	583.07

As discussed in Chapter 5, the PCTL and Hybrid bootstrap methods always provide confidence intervals with the same CIW. In general the CIW are large for each coefficient of each method. The BC and BCa methods have relatively smaller CIW for a_1 and a_3, and Jackknife has the smallest CIW compared to all other bootstrap methods.

Since the true coefficients are unknown, it is inappropriate to consider hit rates and hypothesis tests as described in Chapter 5. Inferences shown in table 9.5 and table 9.6 suggest that Jackknife method provides more reasonable results than other methods. It is interesting to note that if the Jackknife confidence intervals are used, that zero is contained in each of the four confidence intervals, and one could not reject the hypothesis that each coefficient is zero. Since the confidence intervals for each coefficient are individual confidence intervals and since association between parameter estimates is not taken into account, the true coefficient values might not all be zero. In OLS regression, it can occur that all

the individual t-tests that a coefficient is zero do not reject the null hypothesis, even though the overall test for model significance is significant. This occurs because of multicolinearity. Whether multicolinearity creates a similar problem with RMA estimates needs further investigation.

9.4 Residual Plots and Influential Diagnostics

Fitted values $\hat{X}_{ij}, i = 1,...,50,\ j = 1,...,4$ and residuals $\hat{e}_{ij}, j = 1,...,4$ can be calculated and the residual plots of $\hat{e}_{ij}$ *vs* X_{ij} and $\hat{e}_{ij}$ *vs* $\hat{X}_{ij}, i = 1,...,50,\ j = 1,...,4$ are obtained as shown in figure 9.3. The residuals $\hat{e}_{ij}, i = 1,...,50,\ j = 1,...,4$ are correlated with both the variables X_{ij} and the fitted values $\hat{X}_{ij},\ j = 1,...,4$ as discussed in Chapter 6. Then to obtain the de-trended plots, use OLS regression to regress $\hat{e}_1$ on $X_1,...,X_4$ and $\hat{X}_1,...,\hat{X}_4$, and examine the plots of new OLS residuals versus the original variables and the fitted values as in figure 9.4. The de-trended plots do not show clear linear relationships any more.

There do not appear to be obvious outliers in the residual plots.

A leave-one-out approach described in Chapter 7 is performed to look for influential observations. Plots of $DC_{j(i)}, EC_{(i)}, DF_{i(i)j}, EF_{(i)j}$ and DL versus the observation number are obtained. All estimates and the corresponding minimum objective function values of each data set with the i^{th} observation deleted are provided in appendix 9.2.

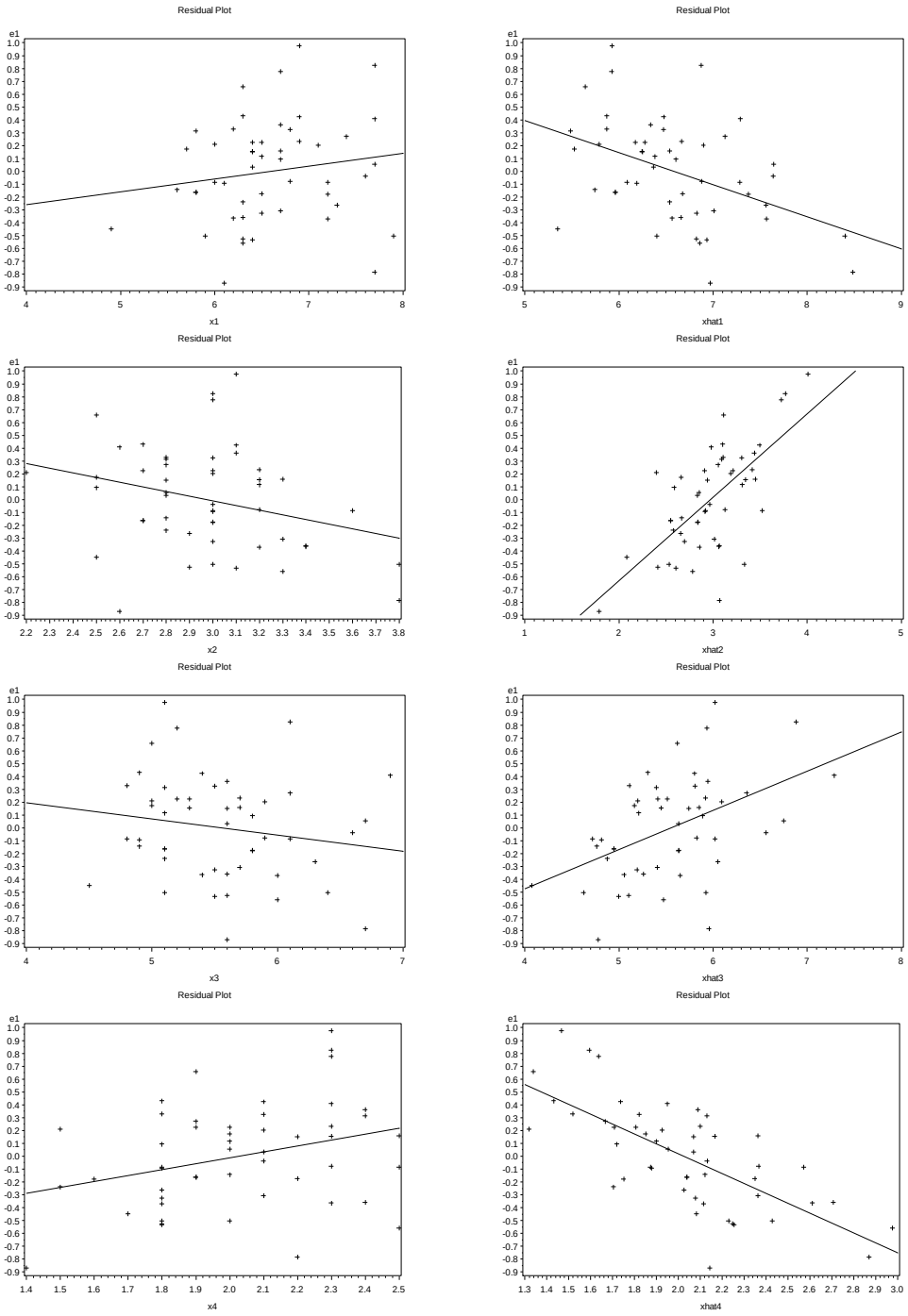

Figure 9.3 Residual plots of $\hat{e}_{i1}$ versus X_{ij} (*left*) and $\hat{X}_{ij}, i = 1,...,4$ (*right*) for the Iris Virginica data set with OLS regression line

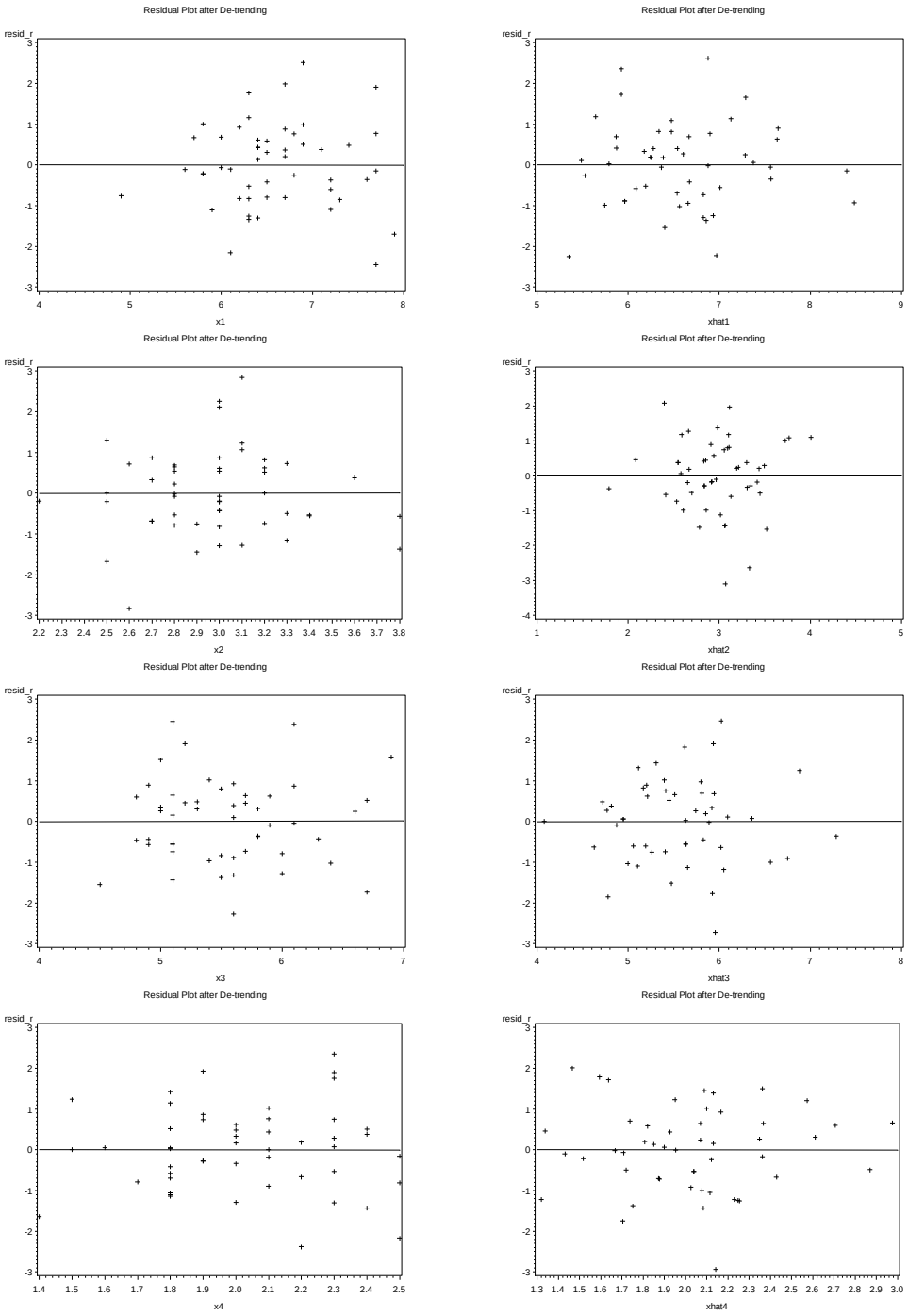

Figure 9.4 Residual plots of new residuals from regressing $\hat{e}_{i1}$ on X_{ij} using OLS versus the original variable X_{ij} (*left*) and $\hat{X}_{ij}$, $j = 1,...,4$ (*right*)

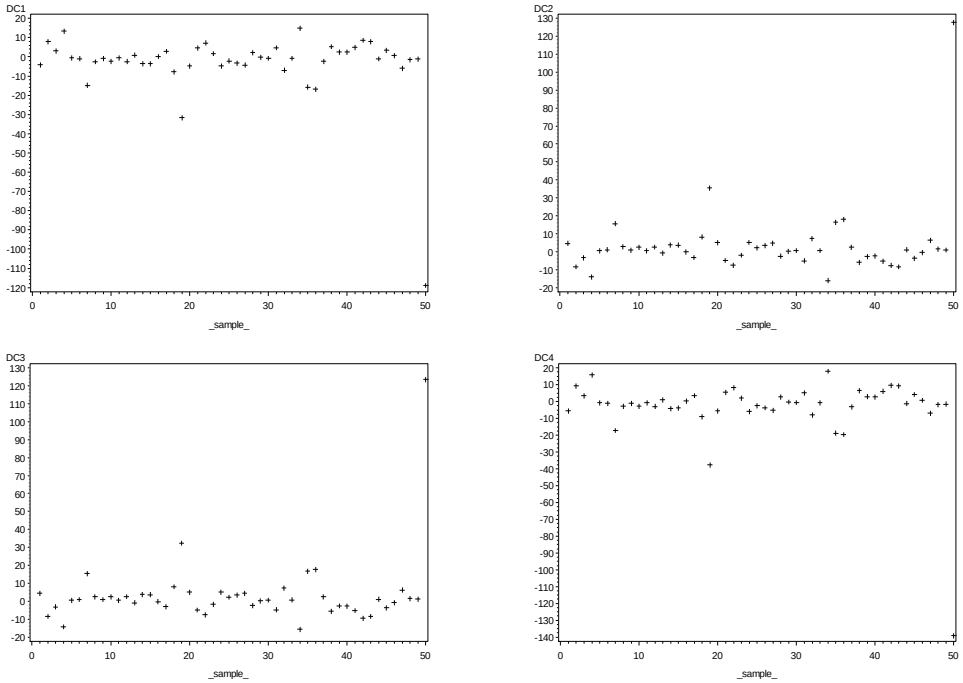

Figure 9.5 Plots of $DC_{j(i)}$ versus observation number for each Iris Virigina variable

Looking at figure 9.5 and 9.6, the plots of $DC_{j(i)}$ and $EC_{(i)}$ versus the observation number for each variable show that observation 50 is influential when assessing the coefficients. Some investigation of observation 50 will be given later in this chapter.

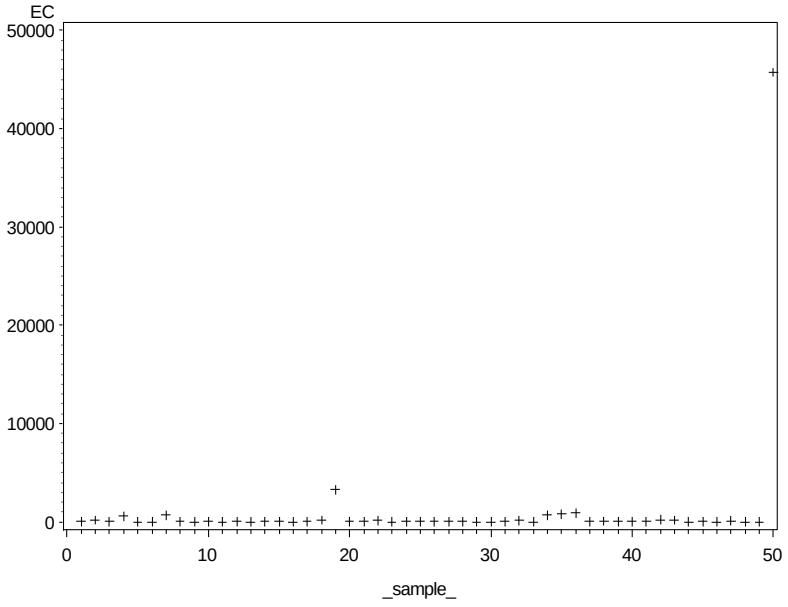

Figure 9.6 Plot of $EC_{(i)}$ versus observation number

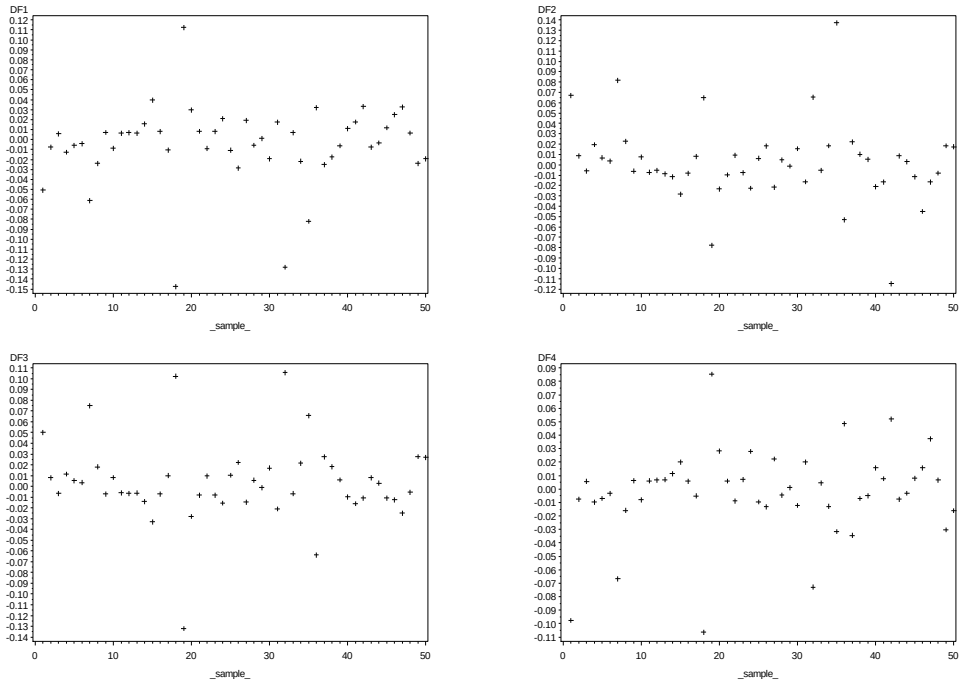

Figure 9.7 Plots of $DF_{i(i)j}$ versus observation number for each variable

In figure 9.7, observation 50 is not identified as an influential point when assessing the influence on fitted values. There might be some deviations of observation 18, 19, 22 and 36 from others, but the scale of the plots is relatively small, and these observations are not obvious influential points.

Looking at the influence on all fitted values for each variable as in figure 9.8, observation 50 is not influential. Again there might be some deviations of observation 18, 19 and 36 from others, but when compared to the scale of the plots, the influence is not obvious.

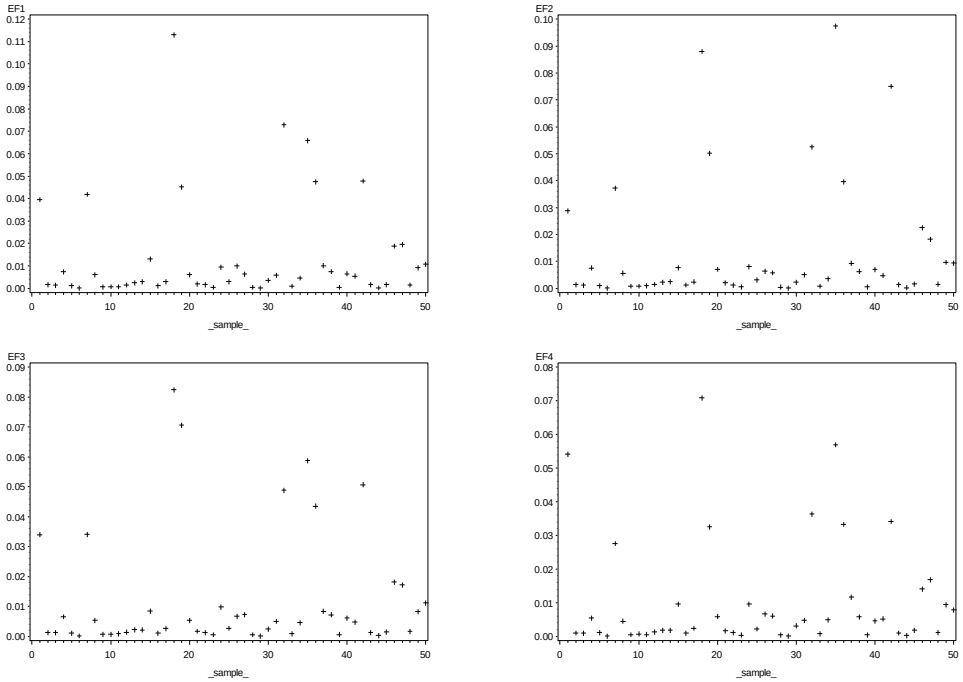

Figure 9.8 Plots of $EF_{(i)j}$ versus observation number for each variable

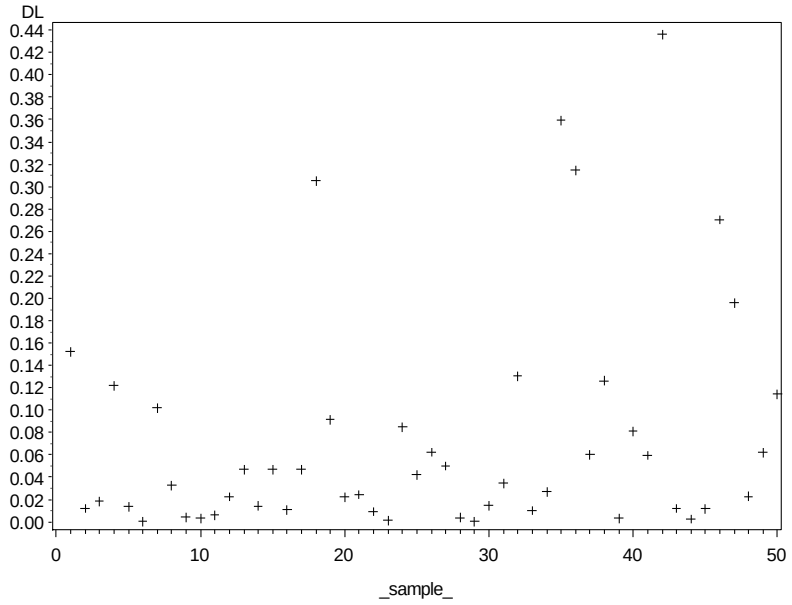

Figure 9.9 Plot of $DL_{(i)}$ versus the observation number

Figure 9.9 shows the influence on objective function values and observation 36 and 43 have a DL value somewhat larger than the others.

Table 9.7 Observation 5, 18, 19, 22, 36, 43 and 50 of the original data

Obs#	X_1	X_2	X_3	X_4
5	6.5	3.0	5.8	2.2
18	7.7	3.8	6.7	2.2
19	7.7	2.6	6.9	2.3
22	5.6	2.8	4.9	2.0
36	7.7	3.0	6.1	2.3
43	5.8	2.7	5.1	1.9
50	5.9	3.0	5.1	1.8

Observation 5 does not appear to be influential in any of the plots above. Table 9.7 includes observation 5 to be compared with other possible influential points. As shown in table 9.7, X_1 for observations 18, 19 and 36 is 7.7, which is relatively larger than the X_1 value for observation 5. X_3 for observation 18 and 19 is a little larger than the others.

Table 9.8 The deleted coefficient estimates with observation 5, 18, 19, 22, 36, 43 and 50 of the original data deleted

Obs# deleted	$\hat{a}_{1(obs\#)}$	$\hat{a}_{2(obs\#)}$	$\hat{a}_{3(obs\#)}$	$\hat{a}_{4(obs\#)}$	$L_{(obs\#)}$
5	-9.0470	9.6430	9.5730	-10.4610	3.4258
18	-1.7400	2.0390	1.9210	-2.0870	3.1341
19	22.2120	-25.2530	-22.3210	26.6160	3.3477
22	-16.7280	17.8330	17.5590	-19.3840	3.4301
36	7.4230	-7.7450	-7.5350	8.4370	3.1250
43	-17.5040	18.6070	18.4160	-20.3400	3.4273
50	109.3570	-117.5350	-113.5570	128.0530	3.3253

As in table 9.7, the RMA estimates when observation 5 is deleted are -9.0470, 9.6430, 9.5730 and -10.4610 which are very close to those RMA estimates of original data in table 9.4. And the minimum objective function value is 3.4258 when observation 5 is deleted. We can see that when other observations shown in table 9.8 are deleted, the RMA estimates are different from those when observation 5 is deleted except for observation 36. The RMA estimates are very far away from others when observation 50 is deleted. However, all observations in table 9.8 have minimum objective function values close to what is in table 9.4 (3.4395).

Combining results from table 9.7 and table 9.8, we can see observations 18, 19, 22, 36, 43 appear to be slightly influential because of the RMA estimates are a little further away from those in table 9.4. Observation 50 does not look like an outlier in relation to others in table 9.7, however the RMA estimates when it is deleted are wildly different from the others. That is why observation 50 only appears to be influential when assessing the influence on coefficients in figure 9.5 and 9.6 but not in other plots.

As discussed in Chapter 7, this happens due to the possibility that an influence diagnostic is affected by the inability of the optimization method to find the global minimum of the objective function.

9.5 Subset Selection

The forward subset selection approach proposed in Chapter 8 is applied to the Iris Virginica data.

Table 9.9 Table of variables entered in the model with the sum of squared residuals

	Number of Variables in Model		
	2	3	4
Variables in Model	1, 3	2	4
$\sum_{i=1}^{n} \hat{e}_{i1}^2$	5.38	6.13	7.93

Table 9.9 and figure 9.10 show that X_1 and X_3 have the highest correlation and are entered into the model at the first step, then X_2 and X_4 are entered subsequently. $\sum_{i=1}^{n} \hat{e}_{i1}^2$ increases when each variable is entered into the model. As discussed in Chapter 8, it is suggested to keep only X_1 (sepal length) and X_3 (petal length) in the model.

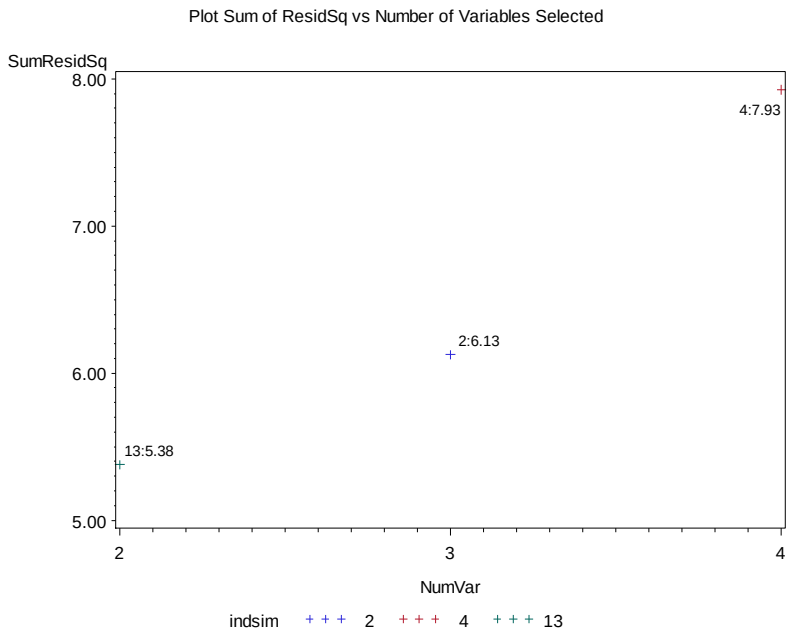

Figure 9.10 Plot of $\sum_{i=1}^{n} \hat{e}_{i1}^2$ versus the number of variables selected

Chapter 10: CONCLUSIONS AND FURTHER STUDIES

10.1 Conclusions

This dissertation has developed methods for RMA regression with more than two variables that parallel the methods used in OLS regression.

Specifically the previous chapters have discussed the following for RMA regression: obtaining parameter estimates by using a minimum objective function approach, the distribution of coefficient estimates, confidence intervals for coefficients obtained by bootstrapping methods and how good those confidence intervals perform, residual analysis and influential points detection and a forward subset selection approach by assessing the sum of squared residuals.

It is shown that RMA will provide reasonable coefficient estimates when the data have a linear relationship between the variables with relatively large sample size (more than 50 or 100 preferably), and the variables do not have large measurement errors. According to simulation studies, the distribution of coefficient estimates is reasonably normal when sample size is 100 and error is small ($mult = 0.15$). Six different confidence intervals were calculated by bootstrapping or jackknifing the original data and obtaining RMA estimates for each resampled data set. There is no rule for which method is superior; however the simulations show that the PCTL method generally does a good job of obtaining confidence intervals for coefficients when considering both how many confidence intervals contain the true coefficient values and how wide the confidence intervals are. Similarly as in OLS regression, residuals are used to obtain residual plots of residuals versus original variable values or fitted variable

values to assess model assumptions and to detect outliers. One at a time deletion of observations is used to develop diagnostics to identify points that influence the coefficient estimates, fitted values and objective function values. A forward subset selection method is proposed by using the sum of squared residuals of a certain variable entered in the model at the first step. Generally, the method suggests to select variables companioned by a decrease in the sum of squared residuals and to drop those variables when the sum of squared residuals increases when the variables are entered in the model. These methods should be useful to data analysts who want to investigate the relationship among multiple variables when no variable is identified as the dependent variable and where there are errors in measuring each variable.

10.2 Further Study

The research presented here is a first step at providing tools to analyze relationships among a number of variables using RMA regression. The goal was to develop methods similar to those used in OLS regression. There are many areas that would benefit from further study.

(1) Finding the global minimum of the objective function

The need for good optimization was mentioned in Chapters 5, 7 and 9. Sometimes, due to the failure of finding the global minimum by the nonlinear optimization method used and its dependency on initial values, correct coefficient estimates cannot be found even though the objective function value that is found is very close to the minimum. Obtaining what appear to be wild estimates of the RMA coefficients can have a serious effect on finding bootstrap confidence

intervals (very wide intervals) and in influence diagnostics (incorrect points identified).

As discussed in Chapter 3, there is an exact solution in the case of 3 variables. When the first derivative of the objective function is taken and set equal to zero a system of p equations of 2^{nd} degree in p unknowns is obtained. Trying to solve the system of those equations might be a way to obtain a closed form for coefficient estimates instead of using an optimization algorithm.

(2) Transformation

A log transformation of the coefficients was mentioned in Chapter 4. However it was only applied on those cases with all positive estimated coefficients. How to deal with coefficient estimates that might be negative or the case when the true coefficient is zero will need further study.

(3) A measure of goodness of fit of a model

As discussed in Chapter 8, the objective function is not always decreasing as more variables are included in the model. It is not clear how to derive an analogue of the OLS R^2 for the RMA regression to evaluate the overall goodness of fit of a model.

(4) Subset selection

As in OLS regression forward and backward subset selection, the forward subset selection approach based on the sum of squared residuals proposed in Chapter 8 does not have a general stopping rule. When there are more variables ($p > 4$), results could be affected by such things as high correlation between variables, or relatively large measurement errors in the variables. The selection process

requires making a judgment call at this time. A general rule of thumb is needed to select the best model and interpret the selection appropriately.

REFERENCES

Barker, F., Soh, Y. and Evans, R. 1988. Properties of the geometric mean functional relationship. *Biometrics* 44:279-281.

Cheng, C.-L. and Van Ness, J. 1999. *Statistical Regression with Measurement Error*. New York: Oxford University Press Inc.

Clarke, M. 1980. The reduced major axis of a bivariate sample. *Biometrika* 67:441-446.

Draper, N. and Yang, Y. 1997. Generalization of the geometric mean functional relationship. *Computational Statistics and Data Analysis* 23:355-372.

Efron, Bradley. 1987 . Better Bootstrap Confidence Intervals. *Journal of the American Statistical Association* 82:171-185.

Efron, B. and Tibshirani, R. 1986. Bootstrap methods for standard errors, confidence intervals, and other measures of statistical accuracy. *Statistical Science* 1:54-75.

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

Feigelson, E. 1992. "Censoring in astronomical data due to nondetections (with discussion) ". In *E. D. Feigelson, & C. J. Babu, Statistical Challenges in Modern Astronomy,* p. 221. New York: Springer-Verlag.

Finney, D. 1938. The distribution of the ratio of estimates of the two variances in a sample from a bivariate normal population. *Biometrika* 30:190-192.

Goodman, T. and Tofallis, C. 2003. Neutral data fitting in two and three dimensions. *Bussiness School Working Papers, University of Hertfordshire, UHBS* 2003:9. http://hdl.handle.net/2299/1409

Hall, P. 1992. *The Bootstrap and Edgeworth Expansion*. New York: Springer-Verlag.

Imbrie, J. 1956. Biometrical methods in the study of invertebrate fossils. *Bulletin of the American Museum of Natural* 198:211-252.

"Iris Flower Data Set," *Wikipedia*, last modified on 18 March 2012. http://en.wikipedia.org/wiki/Iris_flower_data_set

Jolicoeu, P. and Mosimann, D. 1968. Intervalles de confiance pour la pente de l'axe majeur d'une distribution normale bidimensionnelle. *Biométrie-Praximétrie* 9:121-140.

Jolicoeur, P. 1975. Linear regressions in fishery research: some comments. *Journal of the Fisheries Research Board of Canada* 32:1491-1494.

Jolicoeur, P. 1990. Bivariate alometry: interval estimation of the slopes of the ordinary and standardized normal major axes and structural relationship. *Journal of Theoretical Biology* 144:275-285.

Kermack, K. and Haldane, J. 1950. Organic correlation and allometry. *Biometrika* 37:30-41.

Kruskal, W. 1953. On the uniqueness of the line of organic correlation. *Biometrics* 9:47-58.

Kuhry, B. and Marcus, L. 1977. Bivariate linear models in biometry. *Systematic Zoology* 26:201-209.

Kutner, M., Nachtsheim, C., Neter, J. and Li, W. 2004. *Applied Linear Statistical Models.* McGraw-Hill/Irwin.

Lindley, D. 1947. Regression lines and the linear functional relationship. *Journal of the Royal Statistical Society Suppl*. 9:218-244.

McArdle, B. 1988. The structural relationship: regression in biology. *Canadian Journal of Zoology* 66:2329-2339.

McArdle, B. 2003. Lines, models, and errors: regression in the field. *Limnology and Oceanography* 48:1363-1366.

Plotnick, R. 1989. Application of bootstrap methods to reduced major axis line fitting. *Systematic Zoology* 38:144-153.

Rayner, J. 1985. Linear relations in biomechanics: The statistics of scaling functions. *Journal of Zoology, London(A)* 206:415-439.

Richer, W. 1975. A note concerning professor Jolicoeur's comments. *Journal of Fisheries Research Board of Canada* 32:1494-1498.

Richer, W. 1984. Computation and uses of central trend lines. *Canadian Journal of Zoology* 62:1897-1905.

Ricker, W. 1973. Linear regressions in fishery research. *Journal of Fisheries Research Board of Canada* 30:409-434.

Samuelson, P. A. 1942. A note on alternative regressions. *Econometrica* 10:80-83.

SAS Institute. "Jackknife and Bootstrap Analyses," accessed April 1, 2012. http://support.sas.com/kb/24/982.html

Shao, J. and Tu, D. 1995. *The Jackknife and Bootstrap*. New York: Springer-Verlag.

Smith, R. 2009. Use and misuse of the reduced major axis for line-fitting. *American Journal of Physical Anthropology* 140:476-486.

Sprent, P. 1966. A generalized least squares approach to linear functional relationships. *Journal of the Royal Statistical Society (B)* 28:278-297.

Sprent, P. and Dolby, G. 1980. The geometric mean functional relationship. *Biometrics* 36:547-550.

Stuart, A. and Ord, J. 1994. *Kendall's Advanced Theory of Statistics.* London: Edward Arnold.

Teissier, G. 1948. La relation d'allometrie: sa signification statistique et biologique. *Biometrica* 4:14.

Tofallis, C. 2002. "Model fitting for multiple variables by minimising the geometric mean deviation". In *S. V. Huffel, & P. Lemmerling, Total Least Squares and Errors-in-Variables Modeling,* p. 261-267. Netherlands: Kluwer Academic Publishers.

Ward, J., MacDonald, B., Thompson, R. and Beninger, P. 1993. "Mechanisms of suspension feeding in bivalves: Resolution of current controversies by means of endoscopy". In *Limnology and Oceanography,* p. 265-272. American Society of Limnology and Oceanography.

Warton, D., Wright, I., Falster, D. and Westoby, M. 2006. Bivariate line-fitting methods for allometry. *Biological Reviews* 81:259-291.

APPENDIX 4.1 DESCRIPTIVE STATISTICS OF COEFFICIENT ESTIMATES WITH TRUE COEFFICIENT {3 2 4 -1}

$n / \rho / mult$	Statistics	$\hat{a}_1$	$\hat{a}_2$	$\hat{a}_3$	$\hat{a}_4$
20/0/0.15	Mean	3.15	2.15	4.14	-1.19
20/0/0.15	StdDev	0.77	0.57	1.03	0.37
20/0/0.15	Median	3.00	2.05	3.92	-1.14
20/0/0.15	Min	1.95	1.19	2.40	-3.36
20/0/0.15	Max	7.07	4.74	9.21	-0.50
20/0/0.15	Q1	2.64	1.79	3.46	-1.37
20/0/0.15	Q3	3.49	2.38	4.58	-0.93
20/0/0.5	Mean	-113.56	-70.73	-118.33	-78.19
20/0/0.5	StdDev	3,646.21	2,403.67	4,284.40	1,967.39
20/0/0.5	Median	2.72	2.09	3.32	-1.46
20/0/0.5	Min	-80,225.26	-52,331.04	-93,481.63	-43,072.28
20/0/0.5	Max	12,848.96	10,751.78	19,715.02	8,136.12
20/0/0.5	Q1	1.81	1.42	2.21	-2.93
20/0/0.5	Q3	4.97	3.95	6.13	-0.92
20/0/0.8	Mean	-1,329.30	-785.36	-1,353.19	814.82
20/0/0.8	StdDev	28,101.52	16,932.61	27,675.19	16,207.25
20/0/0.8	Median	1.98	1.63	2.27	-1.15
20/0/0.8	Min	-625,009.54	-376,205.83	-613,175.49	-10,112.57
20/0/0.8	Max	14,976.41	15,565.77	13,020.65	357,787.80
20/0/0.8	Q1	1.04	0.75	1.17	-3.01
20/0/0.8	Q3	4.08	3.63	4.61	1.25
20/0.5/0.15	Mean	3.16	2.25	4.22	-1.27
20/0.5/0.15	StdDev	0.82	0.66	1.15	0.47
20/0.5/0.15	Median	3.02	2.11	4.01	-1.17
20/0.5/0.15	Min	1.70	1.02	2.45	-6.78
20/0.5/0.15	Max	9.91	8.49	14.70	-0.57
20/0.5/0.15	Q1	2.66	1.82	3.49	-1.46
20/0.5/0.15	Q3	3.46	2.52	4.63	-0.98
20/0.5/0.5	Mean	155.31	100.11	101.86	29.13
20/0.5/0.5	StdDev	2,881.67	1,464.61	1,674.21	955.75
20/0.5/0.5	Median	2.70	2.37	3.27	-1.58
20/0.5/0.5	Min	-2,763.12	-1,323.99	-6,326.50	-2,254.51
20/0.5/0.5	Max	64,239.31	32,381.44	36,429.48	20,758.86
20/0.5/0.5	Q1	1.79	1.57	2.24	-3.07
20/0.5/0.5	Q3	4.52	4.25	5.57	-0.90
20/0.5/0.8	Mean	-444.84	-596.44	237.51	113.27
20/0.5/0.8	StdDev	11,885.57	16,015.00	22,464.54	9,321.96
20/0.5/0.8	Median	1.82	1.79	2.20	-1.09
20/0.5/0.8	Min	-151,208.85	-291,625.96	-338,980.64	-107,005.13
20/0.5/0.8	Max	119,923.47	108,518.18	312,928.45	120,380.72
20/0.5/0.8	Q1	0.99	0.69	1.13	-2.65
20/0.5/0.8	Q3	3.95	3.60	4.85	1.66
20/0.8/0.15	Mean	3.19	2.49	4.28	-1.49
20/0.8/0.15	StdDev	0.92	0.90	1.33	0.76
20/0.8/0.15	Median	3.03	2.33	4.00	-1.35
20/0.8/0.15	Min	1.65	0.88	2.31	-12.26
20/0.8/0.15	Max	12.87	14.06	19.54	1.58
20/0.8/0.15	Q1	2.67	1.95	3.48	-1.77
20/0.8/0.15	Q3	3.47	2.81	4.69	-1.10
20/0.8/0.5	Mean	1,137.67	-2,638.63	5,763.93	-2,265.74
20/0.8/0.5	StdDev	40,746.36	33,195.08	110,401.20	44,223.96
20/0.8/0.5	Median	2.58	2.36	3.06	-1.46
20/0.8/0.5	Min	-356,743.59	-607,260.70	-284,020.66	-942,749.15
20/0.8/0.5	Max	811,256.47	6,369.83	2,385,648.96	160,648.67
20/0.8/0.5	Q1	1.72	1.38	2.03	-2.87
20/0.8/0.5	Q3	4.73	4.22	5.72	1.60
20/0.8/0.8	Mean	-925.85	-11,164.62	13,660.08	3,555.03
20/0.8/0.8	StdDev	96,206.31	201,720.12	186,962.61	123,710.24
20/0.8/0.8	Median	1.86	1.77	2.02	0.94
20/0.8/0.8	Min	-1,595,737.83	-4,208,373.22	-1,027,902.67	-1,164,596.23
20/0.8/0.8	Max	1,010,824.62	424,621.96	2,690,817.15	2,247,968.31
20/0.8/0.8	Q1	1.02	-1.36	1.01	-2.15
20/0.8/0.8	Q3	3.82	3.73	4.41	2.80
50/0/0.15	Mean	3.05	2.08	4.02	-1.16
50/0/0.15	StdDev	0.37	0.28	0.51	0.19
50/0/0.15	Median	3.01	2.04	3.95	-1.14

$n / \rho / mult$	Statistics	$\hat{a}_1$	$\hat{a}_2$	$\hat{a}_3$	$\hat{a}_4$
50/0/0.15	Min	2.17	1.43	2.90	-1.89
50/0/0.15	Max	4.46	3.14	5.82	-0.71
50/0/0.15	Q1	2.79	1.88	3.67	-1.27
50/0/0.15	Q3	3.30	2.26	4.34	-1.02
50/0/0.5	Mean	19.62	16.09	24.21	-11.99
50/0/0.5	StdDev	177.09	146.40	217.78	107.88
50/0/0.5	Median	3.00	2.35	3.69	-1.78
50/0/0.5	Min	1.18	0.99	1.48	-1,420.04
50/0/0.5	Max	2,333.00	2,014.09	2,772.29	2.94
50/0/0.5	Q1	2.33	1.79	2.86	-2.54
50/0/0.5	Q3	4.19	3.35	5.19	-1.35
50/0/0.8	Mean	42.59	34.21	50.64	-25.72
50/0/0.8	StdDev	616.80	538.31	779.17	499.37
50/0/0.8	Median	2.86	2.49	3.30	-1.98
50/0/0.8	Min	-4,780.40	-4,242.64	-6,828.70	-3,429.94
50/0/0.8	Max	4,297.56	4,350.59	5,966.46	4,412.36
50/0/0.8	Q1	1.92	1.68	2.25	-3.43
50/0/0.8	Q3	4.86	4.29	5.73	-1.24
50/0.5/0.15	Mean	3.05	2.19	4.05	-1.21
50/0.5/0.15	StdDev	0.39	0.32	0.53	0.21
50/0.5/0.15	Median	3.02	2.15	4.02	-1.18
50/0.5/0.15	Min	2.20	1.47	2.80	-2.22
50/0.5/0.15	Max	4.60	3.34	6.10	-0.71
50/0.5/0.15	Q1	2.78	1.95	3.67	-1.32
50/0.5/0.15	Q3	3.28	2.38	4.37	-1.07
50/0.5/0.5	Mean	42.08	38.52	49.94	-16.95
50/0.5/0.5	StdDev	317.67	300.24	370.58	223.49
50/0.5/0.5	Median	3.04	2.88	3.92	-1.99
50/0.5/0.5	Min	1.23	1.00	1.56	-3,458.88
50/0.5/0.5	Max	3,883.66	4,249.69	4,625.49	1,435.30
50/0.5/0.5	Q1	2.33	2.08	2.83	-2.91
50/0.5/0.5	Q3	4.45	4.20	5.42	-1.45
50/0.5/0.8	Mean	1,184.91	1,691.47	584.23	1,334.89
50/0.5/0.8	StdDev	31,255.00	24,738.64	36,670.19	20,854.18
50/0.5/0.8	Median	2.89	2.76	3.43	-1.99
50/0.5/0.8	Min	-207,728.07	-5,712.01	-499,894.83	-4,801.51
50/0.5/0.8	Max	664,714.94	489,650.83	646,312.37	381,043.11
50/0.5/0.8	Q1	1.92	1.71	2.23	-3.78
50/0.5/0.8	Q3	5.42	5.27	6.23	-1.00
50/0.8/0.15	Mean	3.06	2.40	4.07	-1.40
50/0.8/0.15	StdDev	0.41	0.40	0.58	0.29
50/0.8/0.15	Median	3.02	2.35	4.02	-1.36
50/0.8/0.15	Min	2.16	1.51	2.71	-2.70
50/0.8/0.15	Max	4.64	3.97	6.25	-0.74
50/0.8/0.15	Q1	2.78	2.11	3.66	-1.57
50/0.8/0.15	Q3	3.29	2.64	4.38	-1.22
50/0.8/0.5	Mean	181.53	123.48	162.56	74.19
50/0.8/0.5	StdDev	3,737.01	2,411.47	3,225.51	1,919.15
50/0.8/0.5	Median	3.08	2.96	3.73	-2.13
50/0.8/0.5	Min	-404.15	-463.97	-466.54	-1,387.80
50/0.8/0.5	Max	83,540.56	53,860.21	72,075.75	42,844.44
50/0.8/0.5	Q1	2.30	2.08	2.63	-3.18
50/0.8/0.5	Q3	4.31	4.59	5.28	-1.29
50/0.8/0.8	Mean	967.38	2,476.37	-438.52	2,393.61
50/0.8/0.8	StdDev	148,225.18	100,962.91	115,197.35	83,916.69
50/0.8/0.8	Median	2.78	2.60	2.92	-1.11
50/0.8/0.8	Min	-2,283,372.26	-1,380,854.02	-1,508,492.90	-1,091,960.26
50/0.8/0.8	Max	2,326,336.30	1,674,161.09	1,599,790.90	1,370,913.98
50/0.8/0.8	Q1	1.79	1.60	2.00	-2.59
50/0.8/0.8	Q3	5.26	4.70	5.58	2.64
100/0/0.15	Mean	3.03	2.06	4.00	-1.14
100/0/0.15	StdDev	0.27	0.20	0.37	0.13
100/0/0.15	Median	3.02	2.05	3.97	-1.13
100/0/0.15	Min	2.32	1.57	3.03	-1.58
100/0/0.15	Max	4.01	2.69	5.21	-0.83

$n / \rho / mult$	Statistics	$\hat{a}_1$	$\hat{a}_2$	$\hat{a}_3$	$\hat{a}_4$
100/0/0.15	Q1	2.84	1.92	3.73	-1.22
100/0/0.15	Q3	3.21	2.20	4.25	-1.05
100/0/0.5	Mean	3.43	2.66	4.20	-2.02
100/0/0.5	StdDev	1.47	1.16	1.81	0.91
100/0/0.5	Median	3.02	2.38	3.71	-1.78
100/0/0.5	Min	1.47	1.24	1.75	-8.54
100/0/0.5	Max	14.48	11.59	18.65	-0.82
100/0/0.5	Q1	2.50	1.91	3.03	-2.33
100/0/0.5	Q3	3.94	3.06	4.82	-1.43
100/0/0.8	Mean	14.07	10.18	13.08	-2.83
100/0/0.8	StdDev	526.41	447.53	638.03	369.65
100/0/0.8	Median	3.04	2.62	3.46	-2.15
100/0/0.8	Min	-8,253.13	-6,559.19	-10,123.88	-3,657.34
100/0/0.8	Max	5,120.79	4,228.19	5,831.05	5,641.82
100/0/0.8	Q1	2.23	1.85	2.53	-3.44
100/0/0.8	Q3	4.90	4.05	5.58	-1.51
100/0.5/0.15	Mean	3.03	2.16	4.02	-1.20
100/0.5/0.15	StdDev	0.26	0.22	0.36	0.15
100/0.5/0.15	Median	3.03	2.15	3.99	-1.19
100/0.5/0.15	Min	2.34	1.62	3.11	-1.84
100/0.5/0.15	Max	4.03	2.94	5.48	-0.83
100/0.5/0.15	Q1	2.84	2.00	3.76	-1.29
100/0.5/0.15	Q3	3.21	2.31	4.26	-1.10
100/0.5/0.5	Mean	4.50	4.19	5.87	-3.14
100/0.5/0.5	StdDev	24.43	22.85	34.41	18.65
100/0.5/0.5	Median	3.12	2.84	3.85	-2.05
100/0.5/0.5	Min	1.50	1.29	1.87	-418.72
100/0.5/0.5	Max	548.86	513.27	772.61	-0.93
100/0.5/0.5	Q1	2.49	2.27	3.10	-2.65
100/0.5/0.5	Q3	3.98	3.65	5.05	-1.64
100/0.5/0.8	Mean	-23.51	45.66	32.95	-73.81
100/0.5/0.8	StdDev	1,176.11	935.54	1,069.16	1,193.49
100/0.5/0.8	Median	3.08	3.02	3.62	-2.37
100/0.5/0.8	Min	-22,304.46	-11,826.17	-16,476.66	-23,423.02
100/0.5/0.8	Max	7,611.95	14,471.40	13,066.52	4,987.73
100/0.5/0.8	Q1	2.18	2.14	2.61	-3.78
100/0.5/0.8	Q3	4.82	4.87	5.73	-1.63
100/0.8/0.15	Mean	3.03	2.36	4.03	-1.39
100/0.8/0.15	StdDev	0.27	0.27	0.38	0.19
100/0.8/0.15	Median	3.02	2.34	4.00	-1.38
100/0.8/0.15	Min	2.31	1.69	3.03	-2.39
100/0.8/0.15	Max	4.13	3.31	5.68	-0.93
100/0.8/0.15	Q1	2.84	2.18	3.76	-1.51
100/0.8/0.15	Q3	3.22	2.54	4.29	-1.26
100/0.8/0.5	Mean	5.89	6.14	7.88	-4.64
100/0.8/0.5	StdDev	53.83	56.79	79.74	49.77
100/0.8/0.5	Median	3.14	3.24	3.85	-2.38
100/0.8/0.5	Min	1.50	-13.80	-12.35	-1,114.23
100/0.8/0.5	Max	1,206.62	1,272.75	1,786.83	28.69
100/0.8/0.5	Q1	2.47	2.55	3.05	-3.22
100/0.8/0.5	Q3	4.10	4.24	5.05	-1.85
100/0.8/0.8	Mean	-28,047.69	-14,800.51	-11,463.15	-20,651.54
100/0.8/0.8	StdDev	652,535.08	318,430.46	431,219.48	342,604.75
100/0.8/0.8	Median	3.19	2.92	3.36	-1.82
100/0.8/0.8	Min	-14,531,867.07	-6,933,255.59	-9,225,363.00	-7,366,176.32
100/0.8/0.8	Max	1,135,746.93	892,754.60	2,660,929.15	6,311.83
100/0.8/0.8	Q1	2.22	2.08	2.41	-3.11
100/0.8/0.8	Q3	4.95	4.37	4.85	2.09

APPENDIX 4.2 DESCRIPTIVE STATISTICS OF COEFFICIENT ESTIMATES WITH TRUE COEFFICIENT {3 2 0 -1}

$n / \rho / mult$	Statistics	$\hat{a}_1$	$\hat{a}_2$	$\hat{a}_3$	$\hat{a}_4$
20/0/0.15	Mean	3.12	2.10	-0.02	-1.13
20/0/0.15	StdDev	0.58	0.42	0.41	0.28
20/0/0.15	Median	3.01	2.04	-0.21	-1.09
20/0/0.15	Min	1.93	1.23	-0.96	-2.71
20/0/0.15	Max	5.70	4.26	1.36	-0.58
20/0/0.15	Q1	2.72	1.81	-0.36	-1.27
20/0/0.15	Q3	3.42	2.33	0.37	-0.94
20/0/0.5	Mean	84.93	62.30	2.77	-43.46
20/0/0.5	StdDev	634.50	433.69	205.46	337.71
20/0/0.5	Median	2.95	2.15	-0.51	-1.48
20/0/0.5	Min	-51.16	-25.93	-2,502.86	-4,653.73
20/0/0.5	Max	9,653.76	7,047.81	2,417.49	1,188.98
20/0/0.5	Q1	2.10	1.50	-1.23	-2.54
20/0/0.5	Q3	4.68	3.54	1.24	-0.99
20/0/0.8	Mean	-1,145.72	-1,026.19	776.54	215.37
20/0/0.8	StdDev	18,247.98	16,585.25	11,898.01	14,267.29
20/0/0.8	Median	2.08	1.63	-0.54	-1.29
20/0/0.8	Min	-356,313.96	-319,449.45	-12,754.48	-167,221.23
20/0/0.8	Max	5,741.66	9,223.01	197,214.67	270,335.37
20/0/0.8	Q1	1.10	0.90	-1.68	-2.80
20/0/0.8	Q3	3.87	3.21	1.55	-0.49
20/0.5/0.15	Mean	3.11	2.11	0.11	-1.19
20/0.5/0.15	StdDev	0.58	0.49	0.50	0.35
20/0.5/0.15	Median	3.01	2.02	0.34	-1.13
20/0.5/0.15	Min	1.91	1.13	-1.58	-3.18
20/0.5/0.15	Max	6.16	5.82	1.02	-0.47
20/0.5/0.15	Q1	2.73	1.80	-0.39	-1.36
20/0.5/0.15	Q3	3.40	2.34	0.51	-0.95
20/0.5/0.5	Mean	-2,700.31	-2,557.72	963.15	1,598.75
20/0.5/0.5	StdDev	44,858.80	44,771.54	22,061.67	24,942.65
20/0.5/0.5	Median	2.68	2.09	0.67	-1.47
20/0.5/0.5	Min	-982,045.87	-963,655.52	-106,712.36	-3,277.97
20/0.5/0.5	Max	9,725.01	59,441.30	471,635.80	541,937.89
20/0.5/0.5	Q1	1.84	1.29	-1.56	-2.37
20/0.5/0.5	Q3	4.08	3.52	1.45	-0.84
20/0.5/0.8	Mean	1,007.96	-2,004.33	3,182.10	1,029.03
20/0.5/0.8	StdDev	76,158.98	79,918.01	54,732.49	44,819.91
20/0.5/0.8	Median	1.87	1.54	0.65	-1.14
20/0.5/0.8	Min	-1,299,222.25	-1,648,014.38	-275,674.48	-373,997.12
20/0.5/0.8	Max	858,001.14	540,002.82	999,653.87	832,923.52
20/0.5/0.8	Q1	0.84	-0.77	-2.08	-2.57
20/0.5/0.8	Q3	3.36	3.10	2.06	1.46
20/0.8/0.15	Mean	3.13	2.20	0.19	-1.35
20/0.8/0.15	StdDev	0.65	0.77	0.81	0.59
20/0.8/0.15	Median	3.01	2.03	0.54	-1.29
20/0.8/0.15	Min	1.89	1.06	-3.81	-4.94
20/0.8/0.15	Max	7.70	9.85	1.64	1.58
20/0.8/0.15	Q1	2.73	1.72	-0.54	-1.60
20/0.8/0.15	Q3	3.39	2.49	0.78	-1.02
20/0.8/0.5	Mean	-1,634.59	-1,203.61	676.28	598.86
20/0.8/0.5	StdDev	14,799.61	20,047.39	11,418.24	12,024.54
20/0.8/0.5	Median	2.32	1.90	-0.03	-1.43
20/0.8/0.5	Min	-238,127.02	-377,302.63	-82,173.42	-149,800.50
20/0.8/0.5	Max	4,045.45	98,908.68	189,451.63	186,436.90
20/0.8/0.5	Q1	1.43	0.96	-2.50	-2.78
20/0.8/0.5	Q3	3.68	3.90	1.93	1.33
20/0.8/0.8	Mean	-11,446.48	4,675.75	-11,199.40	4,174.54
20/0.8/0.8	StdDev	161,804.14	128,417.46	341,344.86	196,360.36
20/0.8/0.8	Median	1.51	1.17	0.77	-0.91
20/0.8/0.8	Min	-3,401,356.29	-572,443.56	-7,504,793.75	-1,687,548.96
20/0.8/0.8	Max	5,355.49	2,690,783.89	1,077,656.95	3,986,412.17
20/0.8/0.8	Q1	-3.99	-2.35	-2.28	-2.61
20/0.8/0.8	Q3	2.78	2.57	2.39	2.15
50/0/0.15	Mean	3.02	2.03	0.00	-1.09
50/0/0.15	StdDev	0.29	0.22	0.36	0.14
50/0/0.15	Median	3.00	2.01	-0.23	-1.09

$n / \rho / mult$	Statistics	$\hat{a}_1$	$\hat{a}_2$	$\hat{a}_3$	$\hat{a}_4$
50/0/0.15	Min	2.13	1.28	-0.56	-1.55
50/0/0.15	Max	4.24	2.94	0.59	-0.64
50/0/0.15	Q1	2.82	1.88	-0.34	-1.18
50/0/0.15	Q3	3.19	2.15	0.35	-1.00
50/0/0.5	Mean	3.57	2.58	0.06	-1.86
50/0/0.5	StdDev	2.71	2.19	1.71	1.29
50/0/0.5	Median	2.95	2.09	0.60	-1.57
50/0/0.5	Min	1.32	0.80	-15.91	-19.34
50/0/0.5	Max	46.81	40.90	8.04	0.82
50/0/0.5	Q1	2.45	1.72	-1.09	-2.07
50/0/0.5	Q3	3.79	2.80	1.14	-1.24
50/0/0.8	Mean	-2,705.91	-3,308.85	1,835.65	2,419.72
50/0/0.8	StdDev	59,812.64	75,739.32	42,021.63	52,648.70
50/0/0.8	Median	2.77	2.19	-0.73	-1.74
50/0/0.8	Min	-1,337,015.97	-1,693,234.74	-33,302.13	-3,350.18
50/0/0.8	Max	8,555.65	28,517.65	938,966.65	1,176,730.94
50/0/0.8	Q1	2.03	1.61	-1.59	-2.83
50/0/0.8	Q3	4.53	3.54	1.80	-1.21
50/0.5/0.15	Mean	3.02	2.05	0.11	-1.16
50/0.5/0.15	StdDev	0.30	0.26	0.43	0.22
50/0.5/0.15	Median	3.00	2.03	0.38	-1.15
50/0.5/0.15	Min	2.21	1.31	-0.82	-2.14
50/0.5/0.15	Max	4.08	3.12	0.69	-0.67
50/0.5/0.15	Q1	2.82	1.86	-0.39	-1.29
50/0.5/0.15	Q3	3.20	2.22	0.46	-1.00
50/0.5/0.5	Mean	-16,513.35	-14,765.25	7,463.86	7,821.52
50/0.5/0.5	StdDev	265,026.66	233,671.99	118,949.51	123,978.07
50/0.5/0.5	Median	3.01	2.27	0.99	-1.78
50/0.5/0.5	Min	-4,892,430.47	-3,980,456.69	-2,353.78	-1,325.35
50/0.5/0.5	Max	1,057.76	1,013.74	2,119,128.13	2,143,549.24
50/0.5/0.5	Q1	2.33	1.73	-1.19	-2.45
50/0.5/0.5	Q3	3.91	3.27	1.50	-1.32
50/0.5/0.8	Mean	-34,061.31	-14,084.65	-1,425.98	20,765.60
50/0.5/0.8	StdDev	360,426.49	356,685.46	399,582.13	299,368.38
50/0.5/0.8	Median	2.33	1.94	0.87	-1.53
50/0.5/0.8	Min	-5,982,117.01	-5,063,462.39	-8,169,151.55	-1,240,412.93
50/0.5/0.8	Max	25,465.48	4,916,270.40	2,535,745.75	5,634,198.31
50/0.5/0.8	Q1	1.43	1.19	-2.07	-2.57
50/0.5/0.8	Q3	3.64	3.35	2.27	1.57
50/0.8/0.15	Mean	3.03	2.08	0.29	-1.35
50/0.8/0.15	StdDev	0.32	0.38	0.61	0.33
50/0.8/0.15	Median	3.00	2.02	0.60	-1.36
50/0.8/0.15	Min	2.19	1.29	-1.16	-2.69
50/0.8/0.15	Max	4.18	3.59	1.13	-0.62
50/0.8/0.15	Q1	2.81	1.80	-0.52	-1.57
50/0.8/0.15	Q3	3.21	2.33	0.70	-1.12
50/0.8/0.5	Mean	-70,295.04	51,726.15	-68,787.55	27,086.91
50/0.8/0.5	StdDev	922,539.15	773,274.31	1,477,169.44	961,275.10
50/0.8/0.5	Median	2.80	2.42	-1.11	-1.88
50/0.8/0.5	Min	-19,593,818.43	-847,286.93	-31,911,408.05	-7,303,928.39
50/0.8/0.5	Max	1,661.48	16,545,949.05	3,923,891.96	19,462,723.74
50/0.8/0.5	Q1	2.06	1.53	-2.26	-2.69
50/0.8/0.5	Q3	3.74	3.94	1.64	-1.19
50/0.8/0.8	Mean	-65,854.12	-1,223.20	-28,935.21	32,470.60
50/0.8/0.8	StdDev	592,087.17	625,993.27	1,162,225.26	680,809.36
50/0.8/0.8	Median	1.97	1.77	-0.01	-1.30
50/0.8/0.8	Min	-11,760,461.72	-4,574,133.27	-25,469,293.63	-5,462,752.69
50/0.8/0.8	Max	4,636.71	11,247,448.98	2,918,264.80	13,363,500.38
50/0.8/0.8	Q1	-16.99	-2.67	-2.87	-2.56
50/0.8/0.8	Q3	3.23	3.57	3.25	3.48
100/0/0.15	Mean	3.02	2.02	-0.01	-1.08
100/0/0.15	StdDev	0.21	0.16	0.35	0.10
100/0/0.15	Median	3.02	2.01	-0.27	-1.07
100/0/0.15	Min	2.51	1.61	-0.48	-1.38
100/0/0.15	Max	3.60	2.47	0.50	-0.81
100/0/0.15	Q1	2.87	1.91	-0.34	-1.14

$n / \rho / mult$	Statistics	$\hat{a}_1$	$\hat{a}_2$	$\hat{a}_3$	$\hat{a}_4$
100/0/0.15	Q3	3.16	2.12	0.34	-1.02
100/0/0.5	Mean	3.25	2.31	0.02	-1.67
100/0/0.5	StdDev	0.97	0.76	1.29	0.54
100/0/0.5	Median	3.04	2.15	0.61	-1.52
100/0/0.5	Min	1.72	1.12	-3.53	-4.33
100/0/0.5	Max	8.63	7.07	3.29	-0.74
100/0/0.5	Q1	2.59	1.78	-1.10	-1.90
100/0/0.5	Q3	3.66	2.64	1.16	-1.29
100/0/0.8	Mean	-1.12	-3.31	11.39	0.87
100/0/0.8	StdDev	213.68	196.18	127.17	131.55
100/0/0.8	Median	3.03	2.38	-0.72	-1.90
100/0/0.8	Min	-3,442.71	-3,012.04	-333.13	-1,103.64
100/0/0.8	Max	1,716.42	1,492.52	1,786.19	1,922.84
100/0/0.8	Q1	2.36	1.77	-1.68	-2.83
100/0/0.8	Q3	4.34	3.36	1.85	-1.45
100/0.5/0.15	Mean	3.03	2.02	0.18	-1.18
100/0.5/0.15	StdDev	0.20	0.20	0.39	0.16
100/0.5/0.15	Median	3.02	1.99	0.40	-1.19
100/0.5/0.15	Min	2.54	1.57	-0.55	-1.66
100/0.5/0.15	Max	3.67	2.61	0.59	-0.73
100/0.5/0.15	Q1	2.89	1.87	-0.36	-1.28
100/0.5/0.15	Q3	3.15	2.15	0.45	-1.07
100/0.5/0.5	Mean	3.34	2.60	0.39	-1.98
100/0.5/0.5	StdDev	1.23	1.22	1.61	0.88
100/0.5/0.5	Median	3.05	2.27	1.13	-1.88
100/0.5/0.5	Min	1.65	1.08	-11.99	-5.95
100/0.5/0.5	Max	15.30	11.48	2.99	7.53
100/0.5/0.5	Q1	2.59	1.85	-1.13	-2.32
100/0.5/0.5	Q3	3.74	2.97	1.44	-1.56
100/0.5/0.8	Mean	-20,794.81	-5,658.84	-6,550.29	17,701.11
100/0.5/0.8	StdDev	247,863.75	204,149.18	307,848.50	236,702.33
100/0.5/0.8	Median	2.72	2.34	1.20	-1.92
100/0.5/0.8	Min	-4,880,740.76	-2,140,982.67	-6,639,852.29	-2,751.48
100/0.5/0.8	Max	2,598.93	3,370,024.41	1,408,774.09	4,969,517.08
100/0.5/0.8	Q1	2.05	1.64	-2.14	-2.85
100/0.5/0.8	Q3	3.94	3.64	2.01	-1.35
100/0.8/0.15	Mean	3.03	2.01	0.41	-1.39
100/0.8/0.15	StdDev	0.21	0.30	0.51	0.24
100/0.8/0.15	Median	3.02	1.92	0.62	-1.42
100/0.8/0.15	Min	2.52	1.43	-0.78	-2.16
100/0.8/0.15	Max	3.69	3.03	0.95	-0.67
100/0.8/0.15	Q1	2.87	1.80	0.54	-1.55
100/0.8/0.15	Q3	3.16	2.12	0.69	-1.29
100/0.8/0.5	Mean	-60,400.90	46,697.16	-79,090.16	44,614.02
100/0.8/0.5	StdDev	736,347.71	625,455.53	1,086,540.54	675,651.31
100/0.8/0.5	Median	2.97	2.76	-1.27	-2.07
100/0.8/0.5	Min	-13,933,783.13	-1,590,853.96	-18,676,675.24	-3,326,757.94
100/0.8/0.5	Max	22.81	11,179,470.60	1,975,861.37	11,761,111.85
100/0.8/0.5	Q1	2.48	1.94	-1.99	-2.58
100/0.8/0.5	Q3	3.71	3.94	1.58	-1.61
100/0.8/0.8	Mean	-582,429.67	395,322.69	28,400.58	-239,937.80
100/0.8/0.8	StdDev	6,294,075.21	5,375,040.31	8,302,635.38	9,214,683.26
100/0.8/0.8	Median	2.40	2.40	-1.21	-1.84
100/0.8/0.8	Min	-121,284,203.76	-17,541,818.97	-97,512,113.08	-194,614,706.66
100/0.8/0.8	Max	151.02	102,133,978.34	149,725,011.76	51,985,503.58
100/0.8/0.8	Q1	1.59	1.21	-2.68	-2.86
100/0.8/0.8	Q3	3.60	4.03	2.65	2.73

APPENDIX 5.1 HIT RATE, P-VALUE AND RANK

$n/\rho/mult$	Method	$\hat{a}_1$			$\hat{a}_2$			$\hat{a}_3$		
		hr	*p*	*r*	*hr*	*p*	*r*	*hr*	*p*	*r*
20/0/0.15	Normal	0.944	0.269	1	0.944	0.269	1	0.946	0.341	2
20/0/0.15	BC	0.930	**0.020**	4	0.916	**0.000**	4	0.870	**0.000**	5
20/0/0.15	BCA	0.932	**0.032**	3	0.920	**0.001**	3	0.874	**0.000**	4
20/0/0.15	Hybrid	0.918	**0.001**	6	0.914	**0.000**	5	0.952	0.581	1
20/0/0.15	PCTL	0.924	**0.004**	5	0.914	**0.000**	5	0.868	**0.000**	6
20/0/0.15	Jackknife	0.940	0.152	2	0.942	0.206	2	0.932	**0.032**	3
20/0/0.5	Normal	0.980	0.999	1	0.982	0.999	2	0.994	1.000	1
20/0/0.5	BC	0.884	**0.000**	3	0.882	**0.000**	4	0.860	**0.000**	6
20/0/0.5	BCA	0.878	**0.000**	4	0.878	**0.000**	5	0.874	**0.000**	5
20/0/0.5	Hybrid	0.840	**0.000**	6	0.862	**0.000**	6	0.966	0.950	3
20/0/0.5	PCTL	0.980	0.999	1	0.984	1.000	1	0.982	0.999	2
20/0/0.5	Jackknife	0.868	**0.000**	5	0.890	**0.000**	3	0.944	0.269	4
20/0/0.8	Normal	0.980	0.999	2	0.984	1.000	2	0.980	0.999	2
20/0/0.8	BC	0.766	**0.000**	6	0.788	**0.000**	6	0.740	**0.000**	6
20/0/0.8	BCA	0.772	**0.000**	5	0.802	**0.000**	5	0.766	**0.000**	5
20/0/0.8	Hybrid	0.826	**0.000**	3	0.878	**0.000**	3	0.926	**0.007**	3
20/0/0.8	PCTL	0.990	1.000	1	0.992	1.000	1	0.996	1.000	1
20/0/0.8	Jackknife	0.776	**0.000**	4	0.810	**0.000**	4	0.868	**0.000**	4
20/0.5/0.15	Normal	0.936	0.075	1	0.918	**0.001**	1	0.976	0.996	1
20/0.5/0.15	BC	0.928	**0.012**	4	0.904	**0.000**	3	0.904	**0.000**	4
20/0.5/0.15	BCA	0.932	**0.032**	2	0.914	**0.000**	2	0.902	**0.000**	5
20/0.5/0.15	Hybrid	0.900	**0.000**	6	0.884	**0.000**	6	0.970	0.980	2
20/0.5/0.15	PCTL	0.924	**0.004**	5	0.894	**0.000**	5	0.902	**0.000**	5
20/0.5/0.15	Jackknife	0.930	**0.020**	3	0.904	**0.000**	3	0.958	0.794	3
20/0.5/0.5	Normal	0.974	0.993	1	0.978	0.998	1	0.992	1.000	1
20/0.5/0.5	BC	0.872	**0.000**	4	0.880	**0.000**	4	0.852	**0.000**	6
20/0.5/0.5	BCA	0.886	**0.000**	3	0.886	**0.000**	3	0.864	**0.000**	5
20/0.5/0.5	Hybrid	0.858	**0.000**	5	0.866	**0.000**	5	0.972	0.988	3
20/0.5/0.5	PCTL	0.972	0.988	2	0.968	0.968	2	0.982	0.999	2
20/0.5/0.5	Jackknife	0.834	**0.000**	6	0.836	**0.000**	6	0.930	**0.020**	4
20/0.5/0.8	Normal	0.980	0.999	1	0.984	1.000	1	0.990	1.000	2
20/0.5/0.8	BC	0.738	**0.000**	6	0.716	**0.000**	5	0.722	**0.000**	6
20/0.5/0.8	BCA	0.746	**0.000**	5	0.710	**0.000**	6	0.730	**0.000**	5
20/0.5/0.8	Hybrid	0.844	**0.000**	3	0.892	**0.000**	3	0.942	0.206	3
20/0.5/0.8	PCTL	0.976	0.996	2	0.978	0.998	2	0.998	1.000	1
20/0.5/0.8	Jackknife	0.772	**0.000**	4	0.792	**0.000**	4	0.856	**0.000**	4
20/0.8/0.15	Normal	0.936	0.075	1	0.904	**0.000**	1	0.966	0.950	2
20/0.8/0.15	BC	0.932	**0.032**	2	0.896	**0.000**	3	0.872	**0.000**	5
20/0.8/0.15	BCA	0.932	**0.032**	2	0.898	**0.000**	2	0.880	**0.000**	4
20/0.8/0.15	Hybrid	0.898	**0.000**	6	0.832	**0.000**	6	0.978	0.998	1
20/0.8/0.15	PCTL	0.926	**0.007**	4	0.888	**0.000**	4	0.866	**0.000**	6
20/0.8/0.15	Jackknife	0.924	**0.004**	5	0.868	**0.000**	5	0.946	0.341	3
20/0.8/0.5	Normal	0.978	0.998	2	0.982	0.999	1	0.992	1.000	1
20/0.8/0.5	BC	0.858	**0.000**	4	0.814	**0.000**	4	0.812	**0.000**	6
20/0.8/0.5	BCA	0.862	**0.000**	3	0.812	**0.000**	5	0.816	**0.000**	5
20/0.8/0.5	Hybrid	0.852	**0.000**	5	0.900	**0.000**	3	0.960	0.848	3
20/0.8/0.5	PCTL	0.980	0.999	1	0.978	0.998	2	0.980	0.999	2
20/0.8/0.5	Jackknife	0.800	**0.000**	6	0.780	**0.000**	6	0.858	**0.000**	4
20/0.8/0.8	Normal	0.978	0.998	1	0.980	0.999	1	0.988	1.000	2
20/0.8/0.8	BC	0.720	**0.000**	5	0.660	**0.000**	6	0.664	**0.000**	5
20/0.8/0.8	BCA	0.710	**0.000**	6	0.670	**0.000**	5	0.662	**0.000**	6
20/0.8/0.8	Hybrid	0.854	**0.000**	3	0.898	**0.000**	3	0.932	**0.032**	3
20/0.8/0.8	PCTL	0.976	0.996	2	0.976	0.996	2	0.994	1.000	1
20/0.8/0.8	Jackknife	0.772	**0.000**	4	0.806	**0.000**	4	0.848	**0.000**	4
50/0/0.15	Normal	0.938	0.109	1	0.938	0.109	4	0.948	0.419	2
50/0/0.15	BC	0.926	**0.007**	3	0.944	0.269	2	0.880	**0.000**	4
50/0/0.15	BCA	0.920	**0.001**	5	0.946	0.341	1	0.878	**0.000**	5
50/0/0.15	Hybrid	0.920	**0.001**	5	0.928	**0.012**	6	0.960	0.848	1
50/0/0.15	PCTL	0.922	**0.002**	4	0.938	0.109	4	0.876	**0.000**	6
50/0/0.15	Jackknife	0.932	**0.032**	2	0.940	0.152	3	0.944	0.269	3
50/0/0.5	Normal	0.942	0.206	4	0.964	0.925	4	0.996	1.000	1
50/0/0.5	BC	0.962	0.891	2	0.974	0.993	2	0.790	**0.000**	6
50/0/0.5	BCA	0.962	0.891	2	0.972	0.988	3	0.802	**0.000**	5
50/0/0.5	Hybrid	0.832	**0.000**	6	0.880	**0.000**	6	0.994	1.000	2

$n / \rho / mult$	Method	$\hat{a}_1$			$\hat{a}_2$			$\hat{a}_3$		
		hr	*p*	*r*	*hr*	*p*	*r*	*hr*	*p*	*r*
50/0/0.5	PCTL	0.972	0.988	1	0.984	1.000	1	0.810	**0.000**	4
50/0/0.5	Jackknife	0.876	**0.000**	5	0.932	**0.032**	5	0.986	1.000	3
50/0/0.8	Normal	0.974	0.993	2	0.986	1.000	2	0.994	1.000	1
50/0/0.8	BC	0.882	**0.000**	4	0.894	**0.000**	5	0.878	**0.000**	6
50/0/0.8	BCA	0.892	**0.000**	3	0.900	**0.000**	4	0.888	**0.000**	5
50/0/0.8	Hybrid	0.796	**0.000**	6	0.870	**0.000**	6	0.970	0.980	3
50/0/0.8	PCTL	0.978	0.998	1	0.994	1.000	1	0.986	1.000	2
50/0/0.8	Jackknife	0.832	**0.000**	5	0.902	**0.000**	3	0.952	0.581	4
50/0.5/0.15	Normal	0.938	0.109	2	0.920	**0.001**	4	0.930	**0.020**	3
50/0.5/0.15	BC	0.946	0.341	1	0.918	**0.001**	5	0.856	**0.000**	4
50/0.5/0.15	BCA	0.924	**0.004**	5	0.928	**0.012**	1	0.856	**0.000**	4
50/0.5/0.15	Hybrid	0.922	**0.002**	6	0.896	**0.000**	6	0.960	0.848	1
50/0.5/0.15	PCTL	0.932	**0.032**	3	0.924	**0.004**	2	0.856	**0.000**	4
50/0.5/0.15	Jackknife	0.930	**0.020**	4	0.924	**0.004**	2	0.934	0.050	2
50/0.5/0.5	Normal	0.960	0.848	2	0.964	0.925	2	0.998	1.000	1
50/0.5/0.5	BC	0.958	0.794	3	0.962	0.891	4	0.790	**0.000**	6
50/0.5/0.5	BCA	0.958	0.794	3	0.964	0.925	2	0.810	**0.000**	5
50/0.5/0.5	Hybrid	0.856	**0.000**	6	0.836	**0.000**	6	0.992	1.000	2
50/0.5/0.5	PCTL	0.972	0.988	1	0.974	0.993	1	0.814	**0.000**	4
50/0.5/0.5	Jackknife	0.882	**0.000**	5	0.866	**0.000**	5	0.982	0.999	3
50/0.5/0.8	Normal	0.980	0.999	2	0.984	1.000	2	0.994	1.000	1
50/0.5/0.8	BC	0.874	**0.000**	4	0.866	**0.000**	5	0.848	**0.000**	6
50/0.5/0.8	BCA	0.876	**0.000**	3	0.874	**0.000**	4	0.864	**0.000**	5
50/0.5/0.8	Hybrid	0.826	**0.000**	5	0.894	**0.000**	3	0.958	0.794	3
50/0.5/0.8	PCTL	0.984	1.000	1	0.990	1.000	1	0.986	1.000	2
50/0.5/0.8	Jackknife	0.810	**0.000**	6	0.848	**0.000**	6	0.890	**0.000**	4
50/0.8/0.15	Normal	0.934	0.050	1	0.860	**0.000**	5	0.872	**0.000**	3
50/0.8/0.15	BC	0.924	**0.004**	3	0.894	**0.000**	1	0.766	**0.000**	6
50/0.8/0.15	BCA	0.922	**0.002**	6	0.894	**0.000**	1	0.774	**0.000**	4
50/0.8/0.15	Hybrid	0.924	**0.004**	3	0.828	**0.000**	6	0.900	**0.000**	1
50/0.8/0.15	PCTL	0.924	**0.004**	3	0.894	**0.000**	1	0.768	**0.000**	5
50/0.8/0.15	Jackknife	0.932	**0.032**	2	0.862	**0.000**	4	0.876	**0.000**	2
50/0.8/0.5	Normal	0.976	0.996	2	0.986	1.000	1	0.990	1.000	1
50/0.8/0.5	BC	0.950	0.500	3	0.926	**0.007**	3	0.860	**0.000**	6
50/0.8/0.5	BCA	0.944	0.269	4	0.926	**0.007**	3	0.866	**0.000**	5
50/0.8/0.5	Hybrid	0.862	**0.000**	5	0.868	**0.000**	5	0.966	0.950	2
50/0.8/0.5	PCTL	0.980	0.999	1	0.980	0.999	2	0.892	**0.000**	3
50/0.8/0.5	Jackknife	0.842	**0.000**	6	0.776	**0.000**	6	0.890	**0.000**	4
50/0.8/0.8	Normal	0.982	0.999	2	0.990	1.000	1	0.992	1.000	1
50/0.8/0.8	BC	0.854	**0.000**	3	0.824	**0.000**	5	0.802	**0.000**	6
50/0.8/0.8	BCA	0.852	**0.000**	4	0.828	**0.000**	4	0.816	**0.000**	4
50/0.8/0.8	Hybrid	0.848	**0.000**	5	0.934	0.050	3	0.948	0.419	3
50/0.8/0.8	PCTL	0.990	1.000	1	0.990	1.000	1	0.984	1.000	2
50/0.8/0.8	Jackknife	0.754	**0.000**	6	0.770	**0.000**	6	0.806	**0.000**	5
100/0/0.15	Normal	0.952	0.581	4	0.944	0.269	2	0.886	**0.000**	2
100/0/0.15	BC	0.956	0.731	1	0.936	0.075	5	0.828	**0.000**	5
100/0/0.15	BCA	0.952	0.581	4	0.938	0.109	4	0.824	**0.000**	6
100/0/0.15	Hybrid	0.946	0.341	6	0.934	0.050	6	0.928	**0.012**	1
100/0/0.15	PCTL	0.954	0.659	2	0.948	0.419	1	0.834	**0.000**	4
100/0/0.15	Jackknife	0.954	0.659	2	0.944	0.269	2	0.882	**0.000**	3
100/0/0.5	Normal	0.936	0.075	4	0.972	0.988	1	1.000	1.000	1
100/0/0.5	BC	0.960	0.848	1	0.942	0.206	4	0.418	**0.000**	4
100/0/0.5	BCA	0.960	0.848	1	0.944	0.269	3	0.418	**0.000**	4
100/0/0.5	Hybrid	0.880	**0.000**	6	0.940	0.152	6	1.000	1.000	1
100/0/0.5	PCTL	0.960	0.848	1	0.942	0.206	4	0.418	**0.000**	4
100/0/0.5	Jackknife	0.928	**0.012**	5	0.966	0.950	2	0.992	1.000	3
100/0/0.8	Normal	0.954	0.659	4	0.986	1.000	2	0.998	1.000	1
100/0/0.8	BC	0.968	0.968	2	0.976	0.996	4	0.644	**0.000**	6
100/0/0.8	BCA	0.968	0.968	2	0.978	0.998	3	0.666	**0.000**	5
100/0/0.8	Hybrid	0.856	**0.000**	6	0.930	**0.020**	6	0.998	1.000	1
100/0/0.8	PCTL	0.980	0.999	1	0.990	1.000	1	0.722	**0.000**	4
100/0/0.8	Jackknife	0.872	**0.000**	5	0.952	0.581	5	0.980	0.999	3
100/0.5/0.15	Normal	0.952	0.581	1	0.914	**0.000**	4	0.862	**0.000**	3
100/0.5/0.15	BC	0.940	0.152	4	0.918	**0.001**	1	0.792	**0.000**	4

$n / \rho / mult$	Method	$\hat{a}_1$			$\hat{a}_2$			$\hat{a}_3$		
		hr	p	r	hr	p	r	hr	p	r
100/0.5/0.15	BCA	0.938	0.109	5	0.918	**0.001**	1	0.792	**0.000**	4
100/0.5/0.15	Hybrid	0.944	0.269	3	0.894	**0.000**	6	0.890	**0.000**	1
100/0.5/0.15	PCTL	0.936	0.075	6	0.918	**0.001**	1	0.790	**0.000**	6
100/0.5/0.15	Jackknife	0.950	0.500	2	0.914	**0.000**	4	0.864	**0.000**	2
100/0.5/0.5	Normal	0.944	0.269	1	0.936	0.075	4	1.000	1.000	1
100/0.5/0.5	BC	0.938	0.109	4	0.944	0.269	1	0.456	**0.000**	6
100/0.5/0.5	BCA	0.940	0.152	3	0.942	0.206	2	0.464	**0.000**	5
100/0.5/0.5	Hybrid	0.892	**0.000**	6	0.850	**0.000**	6	1.000	1.000	1
100/0.5/0.5	PCTL	0.942	0.206	2	0.940	0.152	3	0.470	**0.000**	4
100/0.5/0.5	Jackknife	0.922	**0.002**	5	0.894	**0.000**	5	0.980	0.999	3
100/0.5/0.8	Normal	0.968	0.968	2	0.992	1.000	1	0.998	1.000	1
100/0.5/0.8	BC	0.962	0.891	3	0.956	0.731	3	0.808	**0.000**	6
100/0.5/0.8	BCA	0.962	0.891	3	0.956	0.731	3	0.816	**0.000**	5
100/0.5/0.8	Hybrid	0.854	**0.000**	6	0.908	**0.000**	5	0.992	1.000	2
100/0.5/0.8	PCTL	0.982	0.999	1	0.990	1.000	2	0.828	**0.000**	4
100/0.5/0.8	Jackknife	0.868	**0.000**	5	0.886	**0.000**	6	0.958	0.794	3
100/0.8/0.15	Normal	0.952	0.581	1	0.814	**0.000**	4	0.724	**0.000**	2
100/0.8/0.15	BC	0.942	0.206	5	0.850	**0.000**	2	0.620	**0.000**	6
100/0.8/0.15	BCA	0.942	0.206	5	0.850	**0.000**	2	0.634	**0.000**	4
100/0.8/0.15	Hybrid	0.948	0.419	3	0.798	**0.000**	6	0.770	**0.000**	1
100/0.8/0.15	PCTL	0.946	0.341	4	0.852	**0.000**	1	0.628	**0.000**	5
100/0.8/0.15	Jackknife	0.950	0.500	2	0.814	**0.000**	4	0.722	**0.000**	3
100/0.8/0.5	Normal	0.962	0.891	2	0.976	0.996	1	0.988	1.000	2
100/0.8/0.5	BC	0.958	0.794	3	0.960	0.848	2	0.686	**0.000**	6
100/0.8/0.5	BCA	0.956	0.731	4	0.956	0.731	4	0.688	**0.000**	5
100/0.8/0.5	Hybrid	0.910	**0.000**	5	0.818	**0.000**	5	0.990	1.000	1
100/0.8/0.5	PCTL	0.964	0.925	1	0.960	0.848	2	0.698	**0.000**	4
100/0.8/0.5	Jackknife	0.902	**0.000**	6	0.802	**0.000**	6	0.920	**0.001**	3
100/0.8/0.8	Normal	0.990	1.000	1	0.998	1.000	1	0.996	1.000	1
100/0.8/0.8	BC	0.944	0.269	3	0.922	**0.002**	4	0.880	**0.000**	4
100/0.8/0.8	BCA	0.944	0.269	3	0.912	**0.000**	5	0.878	**0.000**	5
100/0.8/0.8	Hybrid	0.868	**0.000**	5	0.950	0.500	3	0.966	0.950	2
100/0.8/0.8	PCTL	0.990	1.000	1	0.994	1.000	2	0.914	**0.000**	3
100/0.8/0.8	Jackknife	0.786	**0.000**	6	0.776	**0.000**	6	0.802	**0.000**	6

Note, the hr is the hit rate, the p is the p-value calculated from the hypothesis test $H_0 : hr \geq 0.95$ vs $H_a : hr < 0.95$ and r is the rank of hr as described in Section 5.1.3. The p-values that are less than 0.05 are marked bold.

APPENDIX 9.1 IRIS VIRGINICA DATA

Obs#	X_1	X_2	X_3	X_4
1	6.3	3.3	6.0	2.5
2	5.8	2.7	5.1	1.9
3	7.1	3.0	5.9	2.1
4	6.3	2.9	5.6	1.8
5	6.5	3.0	5.8	2.2
6	7.6	3.0	6.6	2.1
7	4.9	2.5	4.5	1.7
8	7.3	2.9	6.3	1.8
9	6.7	2.5	5.8	1.8
10	7.2	3.6	6.1	2.5
11	6.5	3.2	5.1	2.0
12	6.4	2.7	5.3	1.9
13	6.8	3.0	5.5	2.1
14	5.7	2.5	5.0	2.0
15	5.8	2.8	5.1	2.4
16	6.4	3.2	5.3	2.3
17	6.5	3.0	5.5	1.8
18	7.7	3.8	6.7	2.2
19	7.7	2.6	6.9	2.3
20	6.0	2.2	5.0	1.5
21	6.9	3.2	5.7	2.3
22	5.6	2.8	4.9	2.0
23	7.7	2.8	6.7	2.0
24	6.3	2.7	4.9	1.8
25	6.7	3.3	5.7	2.1
26	7.2	3.2	6.0	1.8
27	6.2	2.8	4.8	1.8
28	6.1	3.0	4.9	1.8
29	6.4	2.8	5.6	2.1
30	7.2	3.0	5.8	1.6
31	7.4	2.8	6.1	1.9
32	7.9	3.8	6.4	2.0
33	6.4	2.8	5.6	2.2
34	6.3	2.8	5.1	1.5
35	6.1	2.6	5.6	1.4
36	7.7	3.0	6.1	2.3
37	6.3	3.4	5.6	2.4
38	6.4	3.1	5.5	1.8
39	6.0	3.0	4.8	1.8
40	6.9	3.1	5.4	2.1
41	6.7	3.1	5.6	2.4
42	6.9	3.1	5.1	2.3
43	5.8	2.7	5.1	1.9
44	6.8	3.2	5.9	2.3
45	6.7	3.3	5.7	2.5
46	6.7	3.0	5.2	2.3
47	6.3	2.5	5.0	1.9
48	6.5	3.0	5.2	2.0
49	6.2	3.4	5.4	2.3
50	5.9	3.0	5.1	1.8

APPENDIX 9.2 LEAVE-ONE-OUT COEFFICIENT ESTIMATES AND OBJECTIVE FUNCTION VALUE OF IRIS VIRGINICA DATA

Obs# deleted	$\hat{a}_{1(obs\#)}$	$\hat{a}_{2(obs\#)}$	$\hat{a}_{3(obs\#)}$	$\hat{a}_{4(obs\#)}$	$L_{(obs\#)}$
1	-5.3820	5.5830	5.6690	-5.7020	3.2874
2	-17.5060	18.6090	18.4180	-20.3430	3.4273
3	-12.5840	13.4840	13.2320	-14.6670	3.4211
4	-22.9570	24.2690	24.2870	-26.9170	3.3173
5	-9.0470	9.6430	9.5730	-10.4610	3.4258
6	-8.5450	9.1690	9.0590	-10.0030	3.4388
7	5.3160	-5.4250	-5.4290	6.0210	3.3375
8	-6.9820	7.4770	7.5020	-8.3180	3.4069
9	-8.6080	9.3120	9.0690	-10.0470	3.4353
10	-7.2200	7.7990	7.6410	-8.4090	3.4360
11	-9.0230	9.6000	9.5740	-10.5040	3.4332
12	-7.0390	7.6010	7.4490	-8.2050	3.4168
13	-10.3700	11.0450	10.9610	-12.0690	3.3924
14	-5.9330	6.4840	6.3080	-7.0300	3.4253
15	-5.9820	6.5950	6.4140	-7.3320	3.3923
16	-9.6930	10.3650	10.2940	-11.4290	3.4284
17	-12.4130	13.3990	13.1170	-14.7140	3.3923
18	-1.7400	2.0390	1.9210	-2.0870	3.1341
19	22.2120	-25.2530	-22.3210	26.6160	3.3477
20	-4.7810	5.2330	5.0600	-5.5180	3.4174
21	-14.1030	15.0340	14.9020	-16.5880	3.4150
22	-16.7280	17.8330	17.5590	-19.3840	3.4301
23	-11.2550	12.1180	11.8360	-13.1370	3.4380
24	-4.6800	4.9910	4.9990	-5.3350	3.3545
25	-7.3610	8.0030	7.7750	-8.6020	3.3970
26	-6.1870	6.7970	6.6220	-7.4870	3.3771
27	-5.2220	5.5460	5.5850	-5.9890	3.3894
28	-11.7880	12.7090	12.4170	-13.8460	3.4357
29	-9.2930	9.9980	9.8240	-10.8900	3.4390
30	-8.8060	9.5700	9.3690	-10.5770	3.4247
31	-14.2610	15.3150	14.8480	-16.3460	3.4050
32	-2.5150	2.9730	2.7300	-3.1560	3.3089
33	-8.8210	9.5610	9.3250	-10.4260	3.4291
34	-24.3330	26.3180	25.6430	-29.2230	3.4128
35	6.2210	-6.2510	-6.6630	7.6400	3.0805
36	7.4230	-7.7450	-7.5350	8.4370	3.1250
37	-7.1610	7.7150	7.4960	-8.0610	3.3791
38	-14.8550	16.1380	15.6410	-17.6730	3.3132
39	-11.9970	12.9320	12.6310	-14.0830	3.4361
40	-12.0750	12.6300	12.7900	-13.8850	3.3587
41	-14.3780	15.4350	15.2330	-17.2100	3.3798
42	-18.1810	17.9210	19.6410	-20.6980	3.0034
43	-17.5040	18.6070	18.4160	-20.3400	3.4273
44	-8.5280	9.1580	9.0230	-9.9360	3.4368
45	-12.9750	13.9000	13.7400	-15.3940	3.4276
46	-10.1340	10.5770	10.8720	-11.9550	3.1690
47	-3.5680	3.9160	3.8070	-4.1090	3.2436
48	-8.1500	8.6820	8.6520	-9.4790	3.4167
49	-8.4720	9.1930	8.8340	-9.6270	3.3777
50	109.3570	-117.5350	-113.5570	128.0530	3.3253

APPENDIX 9.3 SAS CODES

```
options pageno=1 ls=80 ps=60 nocenter;
data _null_;
       call symput("thedate", put(today(), yymmdd6.));
run;
*************************Please change*******************************;
/*Specify the path that you save all the macros and data sets*/
%let current=C:\your file path\;
/*Specify the initial guess number "nrep"; the bootstrap number "b" and
its significant level "alpha"*/
%let nrep=200;
%let b=1000;
%let alpha=0.05;
/*Specify the name of the data set "datasetname" that will be applied
RMA*/
/*Clean the data set into the form x1-x&nvar*/
/*Write sas data "super" into the "current" path and generate scatter
plots of "datasetname"*/
%include "&current.macro_CleanData.sas";
%CleanData(realdata=datasetname);
*************************Please change*******************************;
/*Assign variable number "nvar" and observation number "nsamp" to macro
variables*/
%global nvar nsamp;
data _null_;
       set "&current.super";
       call symputx("nvar",nvar);
       call symputx("nsamp",nsamp);
run;
%put &nvar &nsamp;
*****************************Bootstrap*********************************;
/*Print log and output to "current" path*/
proc printto
       log="&current.&thedate Bootstrap_CI.log" new
       print="&current.&thedate Bootstrap_CI.lst" new;
run;
%include "&current.jackkboot.sas";
data orin;
       set "&current.super";
       title " ";
run;
/*Perform RMA on Original Data and obtain "Solout" and allinitial*/
/*"Solout": external sas set containing RMA estimates and objective
function of "datasetname"*/
/*allinitial: sas set of initial guesses for RMA regression*/
%include "&current.macro_RMA_Orin.sas";
%include "&current.macro_RMA_2Var_Orin.sas";
%include "&current.macro_RMA_3Var_Orin.sas";
%include "&current.macro_RMA_4Var_Orin.sas";
%RMA_Orin(step=&nvar,nrep=&nrep);
data sol;
       set sol;
run;
proc sort data=sol out=solout;
       by obj_func;
run;
data "&current.solout";
       set solout;
run;
/*Obtain 6 Bootstrap CIs*/;
%macro Bootstrap(nvar=,nrep=,b=,alpha=);

       %include "&current.macro_RMA_Boot.sas";
```

```
	%include "&current.macro_RMA_2Var_Boot.sas";
	%include "&current.macro_RMA_3Var_Boot.sas";
	%include "&current.macro_RMA_4Var_Boot.sas";
/*Bootstrapping for data orin, output data is "bootdist" (system)*/
	%macro analyze(data=,out=);
		%RMA_Boot(step=&nvar, nrep=&nrep);
	%mend;
	title 'Normal ("Standard") Confidence Interval with Bias
Correction';
	%boot(data=orin,alpha=&alpha,samples=&b,random=77842628,stat=est1-
est&nvar);
	title 'All Confidence Interval with Bias Correction for simulation
&s';
	%allci(stat=est1-est&nvar,alpha=&alpha);
	proc sort data=bootdist out=bootout;
		by _sample_;
		title " ";
	run;
	data allciout;
		set allci;
	run;
	data bootstatout;
		set bootstat;
	run;
%mend;
%Bootstrap(nvar=&nvar,nrep=&nrep,b=&b,alpha=&alpha);
/*Obtain external sas set "resultboot" storing bootstrap statistics*/
data "&current.resultboot" resultboot;
	set bootstatout;
	title 'Normal ("Standard") Confidence Interval with Bias
Correction';
	label
		name="Name"
		value="Observed Statistic"
		bootmean="Bootstrap Mean"
		bias="Approximate Bias"
		stderr="Approximate Standard Error"
		alcl="Approximate Lower Confidence Limit"
		biasco="Bias-Corrected Statistic"
		aucl="Approximate Upper Confidence Limit"
		confid="Confidence Level %"
		method="Method for Confidence Interval"
		min="Minimum Resampled Estimate"
		max="Maximum Resampled Estimate"
		n="Number of Resamples"		;
run;
proc print data=resultboot;
run;
/*Obtain external sas set "allciout" storing bootstrap CIs*/
data "allciout" allciout;
	set allciout;
	label
		name="Name"
		value="Observed Statistic"
		alcl="Approximate Lower Confidence Limit"
		aucl="Approximate Upper Confidence Limit"
		confid="Confidence Level (%)"
		method="Method for Confidence Interval"
		n="Number of Resamples"
		_lo="Lower Percentile Point"
		_up="Upper Percentile Point"
		_z0="Bias Correction (Z0)"
```

```
              _accel="Acceleration"            ;
run;
proc print data=allciout;
run;
********************Residual Plots and De-trending*******************;
/*Print log and output to "current" path*/
proc printto
       log="&thedate ResidualPlots_Detrend.log" new
       print="&thedate ResidualPlots_Detrend.lst" new;
run;
/*Obtain residual plots and de-trended residual plots*/
%macro Detrend(nvar=,nrep=);
       data orin;
              set "&current.super";
       run;
       data est;
              set "&current.solout";
              do i=1 to &nsamp;
                     output;
              end;
              drop i;
       run;
/*Create external sas data "before" containing RMA estimates, fitted
values and residuals*/
/*Regress residual e1 on original variables and predicted variables*/
       data before "&current.before";
              merge orin est;
              sum=0;
              %do i=1 %to &nvar;
                     sum+est&i*x&i;
              %end;
              %do i=1 %to &nvar;
                     xhat&i=(1-sum+est&i*x&i)/est&i;
                     e&i=x&i-xhat&i;
              %end;
              keep x: xhat: e:;
       run;
       %do i=1 %to &nvar;
              proc glm data=before;
                     model e1=x&i;
                     output out=xout&i p=ehat rstudent=resid_r;
                     title "";
              run;
              proc glm data=before;
                     model e1=xhat&i;
                     output out=xhatout&i p=ehat rstudent=resid_r;
                     title " ";
              run;
       %end;
       goptions device=gif ftext='arial' htext=16pt gunit=in hsize=10
vsize=8;
       options orientation=landscape;
       options nodate nonumber;
       symbol1 interpol=rl value=plus color=black height=16pt;
       ods pdf file="&current.&thedate ResidualPlots and De-trending.pdf"
style=sansprinter;
       ods noresults;
       ods proclabel='gplot';
       %do i=1 %to &nvar;
              proc gplot data=before;
                     title font='arial' height=16pt "Residual Plot";
                     plot e1*(x&i xhat&i);
```

```
            run;
      %end;
      %do i=1 %to &nvar;
            proc gplot data=xout&i;
                  title font='arial' height=16pt "Residual Plot after
De-trending";
                  plot resid_r*x&i;
            run;
            proc gplot data=xhatout&i;
                  title font='arial' height=16pt "Residual Plot after
De-trending";
                  plot resid_r*xhat&i;
            run;
      %end;
      ods pdf close;
%mend;
%Detrend(nvar=&nvar,nrep=200);
*****************Leave-One-Out Influential Detection***************;
/*Print log and output to "current" path*/
proc printto
  log="&current.&thedate LOO_CoeffEst.log" new
  print="&current.&thedate LOO_CoeffEst.lst" new;
run;
/*Perform Leave-one-out on original data set and obtain plots of
"distances" vs obs#*/
%macro LOO_CoeffEst(nvar=,nrep=);
      data orin;
            set "&current.super";
            title " ";
      run;
/*Obtain external sas set "jackout"*/
/*"jackout" contains RMA estimates and objective function of each data
with the ith obs deleted*/
      %include "&current.macro_RMA_Jack.sas";
      %include "&current.macro_RMA_2Var_Jack.sas";
      %include "&current.macro_RMA_3Var_Jack.sas";
      %include "&current.macro_RMA_4Var_Jack.sas";
      %include "&current.macro_jackout.sas";
      %Jackout(nvar=&nvar,nrep=&nrep);
      data preest;
          set "&current.solout";
          do i=1 to &nsamp;
             output;
          end;
             drop i;
      run;
      data jackout;
          set "&current.jackout";
      run;
      data est;
            merge preest orin;
          rename est1-est&nvar=all_est1-all_est&nvar
obj_func=all_obj_func;
            _sample_=_n_;
      run;
/*Define "distances" in Chapter 7*/
      data LOO;
            title" ";
            merge jackout est;
            all_sum=0;
            %do i=1 %to &nvar;
                  all_sum+all_est&i*x&i;
```

```
		%end;

		%do i=1 %to &nvar;
			all_xhat&i=(1-all_sum+all_est&i*x&i)/all_est&i;
			all_e&i=x&i-all_xhat&i;
		%end;
		sum=0;
		%do i=1 %to &nvar;
			sum+est&i*x&i;
		%end;
		%do i=1 %to &nvar;
			xhat&i=(1-sum+est&i*x&i)/est&i;
			e&i=x&i-xhat&i;
		%end;
		%do i=1 %to &nvar;
			DF&i=all_xhat&i-xhat&i;
			DC&i=all_est&i-est&i;
		%end;
		EC=DC1**2+DC2**2+DC3**2;
		DL=all_obj_func-obj_func;
		keep DF: DC: EC DL _sample_;
	run;
	%do i=1 %to &nsamp;
		data jackout&i;
			set jackout;
			if _sample_=&i;
			do j=1 to &nsamp;
				output;
			end;
		run;
		data LOO&i;
			title" ";
			merge jackout&i est;

			all_sum=0;
			%do j=1 %to &nvar;
				all_sum+all_est&j*x&j;
			%end;

			%do j=1 %to &nvar;
				all_xhat&j=(1-
all_sum+all_est&j*x&j)/all_est&j;
			%end;
			sum=0;
			%do j=1 %to &nvar;
				sum+est&j*x&j;
			%end;
			%do j=1 %to &nvar;
				xhat&j=(1-sum+est&j*x&j)/est&j;
			%end;
			%do j=1 %to &nvar;
				DF&j=all_xhat&j-xhat&j;
				DFsq&j=DF&j**2;
			%end;
			keep DFsq1-DFsq&nvar _sample_;
		run;
		proc means data=LOO&i sum noprint;
			output out=outsum&i;
		run;
		data sumLOO&i;
			set outsum&i;
			if _stat_ ne "MEAN" then delete;
```

```
                        %do j=1 %to &nvar;
                               EF&j=DFsq&j*&nsamp;
                        %end;
                        keep EF:;
                run;
        %end;
        data sumLOO;
                set sumLOO1;
        run;
        %do i=2 %to &nsamp;
                proc append base=sumLOO data=sumLOO&i;
                run;
        %end;
        data sumLOO;
                set sumLOO;
                _sample_=_n_;
        run;
        goptions device=gif ftext='arial' htext=16pt gunit=in hsize=10
vsize=8;
        options orientation=landscape;
        options nodate nonumber;
        symbol1 value=plus height=16pt color=black;
        ods pdf file="&current.Leave-one-out Plots.pdf" style=sansprinter;
        ods noresults;
        ods proclabel='gplot';
        proc gplot data=LOO;
                title height=8pt " ";
                plot (DC1-DC&nvar EC DL) * _sample_;
                plot (DF1-DF&nvar) * _sample_;
        run;
        proc gplot data=sumLOO;
                title height=8pt " ";
                plot (EF1-EF&nvar) * _sample_;
        run;
        ods pdf close;
%mend;
%LOO_CoeffEst(nvar=&nvar,nrep=200);
**********Forward Subset Selection**********;
/*Print log and output to "current" path*/
proc printto
        log="&thedate SubsetSelect.log" new
        print="&thedate SubsetSelect.lst" new;
run;
/*Perform the forward subset selection based on the sum of squared
residuals*/
%macro SubsetSelect(realdata=,numrep=);
/*Identify the two variables with the largest correlation*/
        %include "&current.macro_LargestCorr.sas";
        %include "&current.macro_VarSelect.sas";
/*Obtain external sas sets "olsout" "rsqout" in step 2*/
/*"olsout": estimates of the 4 OLS regressing X_i on others in canonical
form*/
/*rsqout: R-squared of the 4 OLS regressions*/
        %include "&current.macro_EstOrin2.sas";
/*Obtain sum of squared residuals in step 2*/
        %include "&current.macro_ResidualSq2.sas";
        %LargestCorr(step=2);
        %EstOrin2(step=2,nrep=&numrep);
        %ResidualSq2(step=2);
/*Obtain external sas sets "olsout" "rsqout" in each step after step 2*/
        %include "&current.macro_EstOrin.sas";
/*Obtain sum of squared residuals in each step after step 2*/
```

```
      %include "&current.macro_ResidualSq.sas";
/*Idendify the variable with the smallest sum of squared residuals to
enter into the model*/
      %include "&current.macro_Indicator.sas";
      %do p=3 %to &nvar;
            %Indicator(step=&p,nrep=&numrep);
      %end;
/*Obtain plots of sum of squared residuals versus number of variables in
the model*/
      %include "&current.macro_PlotSum.sas";
      %PlotSum();
%mend SubsetSelect;
%SubsetSelect(realdata=iris3,numrep=200);
```

Jackboot

```
%macro jack(      /* Jackknife resampling analysis */
   data=,         /* Input data set. If the data set does not support
                     direct access via the POINT= option, do NOT use
                     the %BYSTMT macro in the %ANALYZE macro. */
   stat=_numeric_,/* Numeric variables in the OUT= data set created
                     by the %ANALYZE macro that contain the values
                     of statistics for which you want to compute
                     jackknife distributions. */
   id=,           /* One or more numeric or character variables that
                     uniquely identify the observations of the OUT=
                     data set within each BY group. No ID variables
                     are needed if the OUT= data set has only one
                     observation per BY group.
                     The ID variables may not be named _TYPE_, _NAME_,
                     or _STAT_. */
   biascorr=1,    /* 1 for bias correction; 0 otherwise. */
   alpha=.05,     /* significance (i.e., one minus confidence) level
                     for confidence intervals; blank to suppress
                     confidence intervals. */
   print=1,       /* 1 to print the jackknife estimates;
                     0 otherwise. */
   chart=1        /* 1 to chart the jackknife resampling distributions;
                     0 otherwise. */
   );
   %if %bquote(&data)= %then %do;
      %put ERROR in JACK: The DATA= argument must be specified.;
      %goto exit;
   %end;
   %global _jackdat; %let _jackdat=&data;
   %global vardef;
   %let vardef=DF;
   %local jack by useby;
   %let useby=0;
   *** compute the actual values of the statistics;
   %let by=;
   %analyze(data=&data,out=JACKACT);
   *** find number of observations in the input data set;
   %local nobs;
   data _null_;
      call symput('nobs',trim(left(put(_nobs,12.))));
      if 0 then set &data nobs=_nobs;
      stop;
   run;
   %if &useby %then %do;
      %jackby(data=&data,print=1);
      %let by=_sample_;
```

```
      %analyze(data=JACKDATA,out=JACKDIST);
   %end;
   %else %do;
      %jackslow(data=&data);
   %end;
   %if &chart %then %do;
      %if %bquote(&id)^= %then %do;
         proc sort data=JACKDIST; by &id; run;
         proc chart data=JACKDIST(drop=_sample_);
            vbar &stat;
            by &id;
         run;
      %end;
      %else %do;
         proc chart data=JACKDIST(drop=_sample_);
            vbar &stat;
         run;
      %end;
   %end;
%jackse(stat=&stat,id=&id,alpha=&alpha,biascorr=&biascorr,print=&print)
%exit:;
%mend jack;
%macro jackby( /* Jackknife resampling */
   data=&_jackdat,
   print=1
   );
   data JACKDATA/view=JACKDATA;
      do _sample_=1 to &nobs;
         do _i=1 to &nobs;
            if _i^=_sample_ then do;
               _obs_=_i;
               set &data point=_i;
               output;
            end;
         end;
      end;
      stop;
   run;
   %if &print %then %do;
      proc print data=JACKDATA; id _sample_ _obs_; run;
   %end;
%exit:;
%mend jackby;
%macro jackslow( /* Uniform jackknife sampling and analysis
                    without BY processing */
   data=&_jackdat
   );
   %put %cmpres(WARNING: Jackknife analysis will be slow because the
        ANALYZE macro did not use the BYSTMT macro.);
   data JACKDIST; set JACKACT; _sample_=0; delete; run;
   options nonotes;
   %local sample;
   %do sample=1 %to &nobs;
      %put Jackknife sample &sample;
      data _TMPD_;
         drop _i;
         do _i=1 to &nobs;
            set &data;
            if _i^=&sample then output;
         end;
         stop;
      run;
```

```
      %analyze(data=_TMPD_,out=_TMPS_);
      data _TMPS_; set _TMPS_; _sample_=&sample; run;
      proc append data=_TMPS_ base=JACKDIST; run;
   %end;
%exit:;
   options notes;
%mend jackslow;
****************************** JACKSE ******************************;
%macro jackse( /* Jackknife estimates of standard error, bias, and
                  normal confidence intervals */
   stat=,
   id=,
   alpha=.05,
   biascorr=1,
   print=1
   );
   %global _jackdat;
   %if %bquote(&_jackdat)= %then %do;
      %put ERROR in JACKSE: You must run JACK before JACKSE;
      %goto exit;
   %end;
   %if %bquote(&alpha)^= %then %do;
      *** compute confidence level;
      %local conf;
      data _null_;
         conf=100*(1-&alpha);
         call symput('conf',trim(left(put(conf,best8.))));
      run;
   %end;
   %if %bquote(&id)^= %then %do;
      *** sort the actual statistics;
      proc sort data=JACKACT;
         by &id;
      run;
   %end;
   *** transpose the actual statistics in each observation;
   proc transpose data=JACKACT out=JACKACT2 prefix=value;
      %if %bquote(&stat)^= %then %do;
         var &stat;
      %end;
      %if %bquote(&id)^= %then %do;
         by &id;
      %end;
   run;
   proc sort data=JACKACT2;
      by %if %bquote(&id)^= %then &id; _name_ ;
   run;
   %if %bquote(&id)^= %then %do;
      proc sort data=JACKDIST;
         by &id;
      run;
   %end;
   *** compute mean, std, min, max of resampling distribution;
   proc means data=JACKDIST(drop=_sample_) noprint vardef=n;
      %if %bquote(&stat)^= %then %do;
         var &stat;
      %end;
      output out=JACKTMP2(drop=_type_ _freq_);
      %if %bquote(&id)^= %then %do;
         by &id;
      %end;
   run;
```

```
*** transpose statistics for resampling distribution;
proc transpose data=JACKTMP2 out=JACKTMP3;
   %if %bquote(&stat)^= %then %do;
      var &stat;
   %end;
   id _stat_;
   %if %bquote(&id)^= %then %do;
      by &id;
   %end;
run;
proc sort data=JACKTMP3;
   by %if %bquote(&id)^= %then &id; _name_ ;
run;
data JACKSTAT;
   retain &id name value jackmean
          %if &biascorr %then bias;
          stderr
          %if %bquote(&alpha)^= %then alcl;
          %if &biascorr %then biasco;
          %if %bquote(&alpha)^= %then aucl confid method;
          min max n;
   merge JACKACT2(rename=(_name_=name value1=value))
         JACKTMP3(rename=(_name_=name mean=jackmean std=stderr));
   by %if %bquote(&id)^= %then &id; name;
   %if %bquote(&alpha)^= %then %do;
      length method $20;
      retain z; drop z;
      if _n_=1 then do;
         z=probit(1-&alpha/2); put z=;
         confid=&conf;
         method='Jackknife';
      end;
   %end;
   stderr=stderr*sqrt(&nobs-1);
   %if &biascorr %then %do;
      bias=(jackmean-value)*(&nobs-1);
      biasco=value-bias;
      %if %bquote(&alpha)^= %then %do;
         alcl=biasco-z*stderr;
         aucl=biasco+z*stderr;
      %end;
   %end;
   %else %if %bquote(&alpha)^= %then %do;
      alcl=value-z*stderr;
      aucl=value+z*stderr;
   %end;
   label name  ='Name'
         value ='Observed Statistic'
         jackmean='Jackknife Mean'
         %if &biascorr %then %do;
            bias  ='Estimated Bias'
            biasco='Bias-Corrected Statistic'
         %end;
         stderr='Estimated Standard Error'
         %if %bquote(&alpha)^= %then %do;
            alcl  ='Estimated Lower Confidence Limit'
            aucl  ='Estimated Upper Confidence Limit'
            method='Method for Confidence Interval'
            confid='Confidence Level (%)'
         %end;
         min   ='Minimum Resampled Estimate'
         max   ='Maximum Resampled Estimate'
```

```
            n      ='Number of Resamples'
            ;
   run;
   %if &print %then %do;
      proc print data=JACKSTAT label;
         id %if %bquote(&id)^= %then &id; name;
      run;
   %end;

%exit:;

%mend jackse;
****************************** BOOT ******************************;
%macro boot(      /* Bootstrap resampling analysis */
   data=,         /* Input data set, not a view or a tape file. */
   samples=200,   /* Number of resamples to generate. */
   residual=,     /* Name of variable in the input data set that
                     contains residuals; may not be used with SIZE= */
   equation=,     /* Equation (in the form of an assignment statement)
                     for computing the response variable */
   size=,         /* Size of each resample; default is size of the
                     input data set. The SIZE= argument may not be
                     used with BALANCED=1 or with a nonblank value
                     for RESIDUAL= */
   balanced=,     /* 1 for balanced resampling; 0 for uniform
                     resampling. By default, balanced resampling
                     is used unless the SIZE= argument is specified,
                     in which case uniform resampling is used. */
   random=0,      /* Seed for pseudorandom numbers. */
   stat=_numeric_,/* Numeric variables in the OUT= data set created
                     by the %ANALYZE macro that contain the values
                     of statistics for which you want to compute
                     bootstrap distributions. */
   id=,           /* One or more numeric or character variables that
                     uniquely identify the observations of the OUT=
                     data set within each BY group. No ID variables
                     are needed if the OUT= data set has only one
                     observation per BY group.
                     The ID variables may not be named _TYPE_, _NAME_,
                     or _STAT_ */
   biascorr=1,    /* 1 for bias correction; 0 otherwise */
   alpha=.05,     /* significance (i.e., one minus confidence) level
                     for confidence intervals; blank to suppress normal
                     confidence intervals */
   print=1,       /* 1 to print the bootstrap estimates;
                     0 otherwise. */
   chart=1        /* 1 to chart the bootstrap resampling distributions;
                     0 otherwise. */
   );
   %if %bquote(&data)= %then %do;
      %put ERROR in BOOT: The DATA= argument must be specified.;
      %goto exit;
   %end;
   %global _bootdat; %let _bootdat=&data;
   %local by useby;
   %let useby=0;
   %global usevardf vardef;
   %let usevardf=0;
   *** compute the actual values of the statistics;
   %let vardef=DF;
   %let by=;
   %analyze(data=&data,out=_ACTUAL_);
```

```
   *** compute plug-in estimates;
   %if &usevardf %then %do;
      %let vardef=N;
      %analyze(data=&data,out=_PLUGIN_);
      %let vardef=DF;
   %end;
   %if &useby=0 %then %let balanced=0;
   %if %bquote(&size)^= %then %do;
      %if %bquote(&balanced)= %then %let balanced=0;
      %else %if &balanced %then %do;
         %put %cmpres(ERROR in BOOT: The SIZE= argument may not be used
              with BALANCED=1.);
         %goto exit;
      %end;
      %if %bquote(&residual)^= %then %do;
         %put %cmpres(ERROR in BOOT: The SIZE= argument may not be used
              with RESIDUAL=.);
         %goto exit;
      %end;
   %end;
   %else %if %bquote(&balanced)= %then %let balanced=1;
   *** find number of observations in the input data set;
   %global _nobs;
   data _null_;
      call symput('_nobs',trim(left(put(_nobs,12.))));
      if 0 then set &data nobs=_nobs;
      stop;
   run;
   %if &balanced %then
      %bootbal(data=&data,samples=&samples,
               random=&random,print=0);
   %else %if &useby %then
      %bootby(data=&data,samples=&samples,
              random=&random,size=&size,print=0);
   %if &balanced | &useby %then %do;
      %let by=_sample_;
      %analyze(data=BOOTDATA,out=BOOTDIST);
   %end;
   %else
      %bootslow(data=&data,samples=&samples,
                random=&random,size=&size);
   %if &chart %then %do;
      %if %bquote(&id)^= %then %do;
         proc sort data=BOOTDIST; by &id; run;
         proc chart data=BOOTDIST(drop=_sample_);
            vbar &stat;
            by &id;
         run;
      %end;
      %else %do;
         proc chart data=BOOTDIST(drop=_sample_);
            vbar &stat;
         run;
      %end;
   %end;
%bootse(stat=&stat,id=&id,alpha=&alpha,biascorr=&biascorr,print=&print)
%exit:;
%mend boot;
%macro bootbal( /* Balanced bootstrap resampling */
   data=&_bootdat,
   samples=200,
   random=0,
```

```
   print=0,
   );
   * Gleason, J.R. (1988) "Algorithms for balanced bootstrap
     simulations," American Statistician, 42, 263-266;
   data BOOTDATA/view=BOOTDATA;
      %bootin;
      drop _a _cbig _ii _j _jbig _k _s;
      array _c(&_nobs) _temporary_;  /* cell counts */
      array _p(&_nobs) _temporary_;  /* pointers */
      do _j=1 to &_nobs;
         _c(_j)=&samples;
      end;
      do _j=1 to &_nobs;
         _p(_j)=_j;
      end;
      _k=&_nobs;                    /* number of nonempty cells left */
      _jbig=_k;                     /* index of largest cell */
      _cbig=&samples;               /* _cbig >= _c(_j) */
      do _sample_=1 to &samples;
         do _i=1 to &_nobs;
            do until(_s<=_c(_j));
               _j=ceil(ranuni(&random)*_k);    /* choose a cell */
               _s=ceil(ranuni(&random)*_cbig); /* accept cell? */
            end;
            _l=_p(_j);
            _obs_=_l;
            _c(_j)+-1;
* put _sample_= _i= _k= _l= @30 %do i=1 %to &_nobs; _c(&i) %end;;
            if _j=_jbig then do;
               _a=floor((&samples-_sample_+_k)/_k);
               if _cbig-_c(_j)>_a then do;
                  do _ii=1 to _k;
                     if _c(_ii)>_c(_jbig) then _jbig=_ii;
                  end;
                  _cbig=_c(_jbig);
               end;
            end;
            if _c(_j)=0 then do;
               if _jbig=_k then _jbig=_j;
               _p(_j)=_p(_k);
               _c(_j)=_c(_k);
               _k+-1;
            end;
            %bootout(_l);
         end;
      end;
      stop;
   run;
   %if &print %then %do;
      proc print data=BOOTDATA; id _sample_ _obs_; run;
   %end;
%exit:;
%mend bootbal;
%macro bootby( /* Uniform bootstrap resampling */
   data=&_bootdat,
   samples=200,
   random=0,
   size=,
   print=0
   );
   %if %bquote(&size)= %then %let size=&_nobs;
   data BOOTDATA/view=BOOTDATA;
```

```
      %bootin;
      do _sample_=1 to &samples;
         do _i=1 to &size;
            _p=ceil(ranuni(&random)*&_nobs);
            _obs_=_p;
            %bootout(_p);
         end;
      end;
      stop;
   run;
   %if &print %then %do;
      proc print data=BOOTDATA; id _sample_ _obs_; run;
   %end;
%exit:;
%mend bootby;
%macro bootslow( /* Uniform bootstrap resampling and analysis
                    without BY processing */
   data=&_bootdat,
   samples=20,
   random=0,
   size=
   );
   %put %cmpres(WARNING: Bootstrap analysis will be slow because the
        ANALYZE macro did not use the BYSTMT macro.);
   %if %bquote(&size)= %then %let size=&_nobs;
   data BOOTDIST; set _ACTUAL_; _sample_=0; delete; run;
   options nonotes;
   %local sample;
   %do sample=1 %to &samples;
      %put Bootstrap sample &sample;
      data _TMPD_;
         %bootin;
         do _i=1 to &size;
            _p=ceil(ranuni(%eval(&random+&sample))*&_nobs);
            %bootout(_p);
         end;
         stop;
      run;
      %analyze(data=_TMPD_,out=_TMPS_);
      data _TMPS_; set _TMPS_; _sample_=&sample; run;
      proc append data=_TMPS_ base=BOOTDIST; run;
   %end;
%exit:;
   options notes;
%mend bootslow;
****************************** BOOTSE ******************************;
%macro bootse( /* Bootstrap estimates of standard error, bias, and
                  normal confidence intervals */
   stat=,
   id=,
   alpha=.05,
   biascorr=1,
   print=1
   );
   %global _bootdat;
   %if %bquote(&_bootdat)= %then %do;
      %put ERROR in BOOTSE: You must run BOOT before BOOTSE;
      %goto exit;
   %end;
   %if %bquote(&alpha)^= %then %do;
      *** compute confidence level;
      %local conf;
```

```
   data _null_;
      conf=100*(1-&alpha);
      call symput('conf',trim(left(put(conf,best8.))));
   run;
%end;
%if %bquote(&id)^= %then %do;
   *** sort the actual statistics;
   proc sort data=_ACTUAL_;
      by &id;
   run;
   %if &usevardf %then %do;
      *** sort the plug-in estimates;
      proc sort data=_PLUGIN_;
         by &id;
      run;
   %end;
%end;
*** transpose the actual statistics in each observation;
proc transpose data=_ACTUAL_ out=_ACTTR_ prefix=value;
   %if %bquote(&stat)^= %then %do;
      var &stat;
   %end;
   %if %bquote(&id)^= %then %do;
      by &id;
   %end;
run;
proc sort data=_ACTTR_;
   by %if %bquote(&id)^= %then &id; _name_ ;
run;
%if &usevardf %then %do;
   *** transpose the plug-in estimates in each observation;
   proc transpose data=_PLUGIN_ out=_PLUGTR_ prefix=value;
      %if %bquote(&stat)^= %then %do;
         var &stat;
      %end;
      %if %bquote(&id)^= %then %do;
         by &id;
      %end;
   run;
   proc sort data=_PLUGTR_;
      by %if %bquote(&id)^= %then &id; _name_ ;
   run;
%end;
%if %bquote(&id)^= %then %do;
   proc sort data=BOOTDIST;
      by &id;
   run;
%end;
*** compute mean, std, min, max of resampling distribution;
proc means data=BOOTDIST(drop=_sample_) noprint;
   %if %bquote(&stat)^= %then %do;
      var &stat;
   %end;
   output out=_TMP2_(drop=_type_ _freq_);
   %if %bquote(&id)^= %then %do;
      by &id;
   %end;
run;
*** transpose statistics for resampling distribution;
proc transpose data=_TMP2_ out=_TMP3_;
   %if %bquote(&stat)^= %then %do;
      var &stat;
```

```
   %end;
   id _stat_;
   %if %bquote(&id)^= %then %do;
      by &id;
   %end;
run;
proc sort data=_TMP3_;
   by %if %bquote(&id)^= %then &id; _name_ ;
run;
data BOOTSTAT;
   retain &id name value bootmean
          %if &biascorr %then bias;
          stderr
          %if %bquote(&alpha)^= %then alcl;
          %if &biascorr %then biasco;
          %if %bquote(&alpha)^= %then aucl confid method;
          min max n;
   merge _ACTTR_(rename=(_name_=name value1=value))
         %if &usevardf %then
            _PLUGTR_(rename=(_name_=name value1=plugin));
         _TMP3_(rename=(_name_=name mean=bootmean std=stderr));
   by %if %bquote(&id)^= %then &id; name;
   %if %bquote(&alpha)^= %then %do;
      length method $20;
      retain z; drop z;
      if _n_=1 then do;
         z=probit(1-&alpha/2); put z=;
         confid=&conf;
         method='Bootstrap Normal';
      end;
   %end;
   %if &biascorr %then %do;
      bias=bootmean-%if &usevardf %then plugin; %else value;;
      biasco=value-bias;
      %if %bquote(&alpha)^= %then %do;
         alcl=biasco-z*stderr;
         aucl=biasco+z*stderr;
      %end;
   %end;
   %else %if %bquote(&alpha)^= %then %do;
      alcl=value-z*stderr;
      aucl=value+z*stderr;
   %end;
   label name  ='Name'
         value ='Observed Statistic'
         bootmean='Bootstrap Mean'
         %if &usevardf %then %do;
            plugin='Plug-In Estimate'
         %end;
         %if &biascorr %then %do;
            bias  ='Approximate Bias'
            biasco='Bias-Corrected Statistic'
         %end;
         stderr='Approximate Standard Error'
         %if %bquote(&alpha)^= %then %do;
            alcl  ='Approximate Lower Confidence Limit'
            aucl  ='Approximate Upper Confidence Limit'
            confid='Confidence Level (%)'
            method='Method for Confidence Interval'
         %end;
         min   ='Minimum Resampled Estimate'
         max   ='Maximum Resampled Estimate'
```

```
            n      ='Number of Resamples'
            ;
   run;
   %if &print %then %do;
      proc print data=BOOTSTAT label;
         id %if %bquote(&id)^= %then &id; name;
      run;
   %end;
%exit:;
%mend bootse;

****************************** BOOTCI ******************************;
%macro bootci(    /* Bootstrap percentile-based confidence intervals.
                     Creates output data set BOOTCI. */
   method,        /* One of the following methods must be specified:
                        PERCENTILE or PCTL
                        HYBRID
                        T
                        BC
                        BCA     Requires the %JACK macro
                     */
   stat=,         /* Numeric variables in the OUT= data set created
                     by the %ANALYZE macro that contain the values
                     of statistics for which you want to compute
                     bootstrap distributions. */
   student=,      /* For the T method only, numeric variables in the
                     OUT= data set created by the %ANALYZE macro that
                     contain the standard errors of the statistics for
which
                     you want to compute bootstrap distributions.
                     There must be a one-to-one between the VAR=
                     variables and the STUDENT= variables */
   id=,           /* One or more numeric or character variables that
                     uniquely identify the observations of the OUT=
                     data set within each BY group. No ID variables
                     are needed if the OUT= data set has only one
                     observation per BY group.
                     The ID variables may not be named _TYPE_, _NAME_,
                     or _STAT_ */
   alpha=.05,     /* significance (i.e., one minus confidence) level
                     for confidence intervals */
   print=1);      /* 1 to print the bootstrap confidence intervals;
                     0 otherwise. */

   %global _bootdat;
   %if %bquote(&_bootdat)= %then %do;
      %put ERROR in BOOTCI: You must run BOOT before BOOTCI;
      %goto exit;
   %end;
   *** check method;
   data _null_;
      length method $10;
      method=upcase(symget('method'));
      if method=' ' then do;
         put 'ERROR in BOOTCI: You must specify one of the methods '
             'PCTL, HYBRID, T, BC or BCa';
         abort;
      end;
      else if method='PERCENTILE' then method='PCTL';
      else if method not in ('PCTL' 'HYBRID' 'BC' 'BCA' 'T')
         then do;
         put "ERROR in BOOTCI: Unrecognized method '" method "'";
```

```
      abort;
   end;
   call symput('qmethod',method);
run;
%if &qmethod=T %then %do;
   %if %bquote(&stat)= | %bquote(&student)= %then %do;
      data _null_;
put 'ERROR: VAR= and STUDENT= must be specified with the T method';
      run;
      %goto exit;
   %end;
%end;
*** sort resampling distributions;
%if %bquote(&id)^= %then %do;
   proc sort data=BOOTDIST;
      by &id _sample_;
   run;
%end;
*** transpose resampling distributions;
proc transpose data=BOOTDIST prefix=col
   out=BOOTTRAN(rename=(col1=value _name_=name));
   %if %bquote(&stat)^= %then %do;
      var &stat;
   %end;
   by %if %bquote(&id)^= %then &id; _sample_;
run;
%if &qmethod=T %then %do;
   *** transpose studentizing statistics;
   proc transpose data=BOOTDIST prefix=col
      out=BOOTSTUD(rename=(col1=student _name_=studname));
         var &student;
      by %if %bquote(&id)^= %then &id; _sample_;
   run;
   data BOOTTRAN;
      merge BOOTTRAN BOOTSTUD;
      label student='Value of Studentizing Statistic'
            studname='Name of Studentizing Statistic';
   run;
%end;
proc sort data=BOOTTRAN;
   by
      %if %bquote(&id)^= %then &id;
      name
      %if &qmethod=BC | &qmethod=BCA %then value;
      %else %if &qmethod=T %then _sample_;
   ;
run;
%if &qmethod=T %then %do;
   *** transpose the actual statistics in each observation
       must get data set in unsorted order for merge;
   proc transpose data=_ACTUAL_ out=_ACTTR_ prefix=value;
      %if %bquote(&stat)^= %then %do;
         var &stat;
      %end;
      %if %bquote(&id)^= %then %do;
         by &id;
      %end;
   run;
   *** transpose the actual studentizing statistics;
   proc transpose data=_ACTUAL_ prefix=col
         out=_ACTSTUD(rename=(_name_=studname col1=student));
         var &student;
```

```
      %if %bquote(&id)^= %then %do;
         by &id;
      %end;
   run;
   *** merge statistics with studentizing statistics;
   data _ACT_T_;
      merge _ACTTR_ _ACTSTUD;
      label student='Value of Studentizing Statistic'
            studname='Name of Studentizing Statistic';
   run;
   proc sort data=_ACT_T_;
      by %if %bquote(&id)^= %then &id; _name_ ;
   run;
   data BOOTTRAN;
      merge BOOTTRAN _ACT_T_(rename=(_name_=name));
      by
         %if %bquote(&id)^= %then &id;
         name
      ;
      value=(value-value1)/student;
   run;
%end;
%if &qmethod=BC | &qmethod=BCA %then %do;
   %if &qmethod=BCA %then %do;
      %jack(data=&_bootdat,stat=&stat,id=&id,alpha=&alpha,
            chart=0,print=&print);
      *** estimate acceleration for BCa;
      proc means data=JACKDIST noprint vardef=df;
         %if %bquote(&stat)^= %then %do;
            var &stat;
         %end;
         output out=JACKSKEW(drop=_type_ _freq_ _sample_) skewness=;
         %if %bquote(&id)^= %then %do;
            by &id;
         %end;
      run;
      *** transpose skewness;
      proc transpose data=JACKSKEW prefix=col
         out=_ACCEL_(rename=(col1=skewness _name_=name));
         %if %bquote(&stat)^= %then %do;
            var &stat;
         %end;
         %if %bquote(&id)^= %then %do;
            by &id;
         %end;
      run;
      proc sort data=_ACCEL_;
         by %if %bquote(&id)^= %then &id; name ;
      run;
   %end;
   *** estimate median bias for BC;
   data _BC_;
      retain _alpha _conf;
      drop value value1;
      if _n_=1 then do;
         _alpha=&alpha;
         _conf=100*(1-_alpha);
         call symput('conf',trim(left(put(_conf,best8.))));
      end;
      merge _ACTTR_(rename=(_name_=name))
            BOOTTRAN;
      by %if %bquote(&id)^= %then &id; name;
```

```
      if first.name then do; n=0; _z0=0; end;
      n+1;
      _z0+(value<value1)+.5*(value=value1);
      if last.name then do;
         _z0=probit(_z0/n);
         output;
      end;
   run;
   *** compute percentiles;
   data BOOTPCTL;
      retain _i _lo _up _nplo _jlo _glo _npup _jup _gup
             alcl aucl;
      drop _alpha _sample_ _conf _i _nplo _jlo _glo _npup _jup _gup
           value;
      merge BOOTTRAN _BC_ %if &qmethod=BCA %then _ACCEL_;;
      by %if %bquote(&id)^= %then &id; name;
      label _lo='Lower Percentile Point'
            _up='Upper Percentile Point'
            _z0='Bias Correction (Z0)';
      if first.name then do;
         %if &qmethod=BC %then %do;
            _lo=probnorm(_z0+(_z0+probit(_alpha/2)));
            _up=probnorm(_z0+(_z0+probit(1-_alpha/2)));
         %end;
         %else %if &qmethod=BCA %then %do;
            drop skewness;
            retain _accel;
            label _accel='Acceleration';
            _accel=skewness/(-6*sqrt(&_nobs))*
                   (&_nobs-2)/&_nobs/sqrt((&_nobs-1)/&_nobs);
            _i=_z0+probit(_alpha/2);
            _lo=probnorm(_z0+_i/(1-_i*_accel));
            _i=_z0+probit(1-_alpha/2);
            _up=probnorm(_z0+_i/(1-_i*_accel));
         %end;
         _nplo=min(n-.5,max(.5,fuzz(n*_lo)));
         _jlo=floor(_nplo); _glo=_nplo-_jlo;
         _npup=min(n-.5,max(.5,fuzz(n*_up)));
         _jup=floor(_npup); _gup=_npup-_jup;
         _i=0;
      end;
      _i+1;
      if _glo then do;
         if _i=_jlo+1 then alcl=value;
      end;
      else do;
         if _i=_jlo then alcl=value;
         else if _i=_jlo+1 then alcl=(alcl+value)/2;
      end;
      if _gup then do;
         if _i=_jup+1 then aucl=value;
      end;
      else do;
         if _i=_jup then aucl=value;
         else if _i=_jup+1 then aucl=(aucl+value)/2;
      end;
      if last.name then do;
         output;
      end;
   run;
%end;
%else %do;
```

```
      %local conf pctlpts pctlpre pctlname;
      %let pctlpre=a;
      %let pctlname=lcl ucl;
      data _null_;
         _alpha=&alpha;
         _conf=100*(1-_alpha);
         call symput('conf',trim(left(put(_conf,best8.))));
         %if &qmethod=PCTL %then %do;
            _lo=_alpha/2;
            _up=1-_lo;
         %end;
         %else %if &qmethod=HYBRID | &qmethod=T %then %do;
            _up=_alpha/2;
            _lo=1-_up;
         %end;
         _lo=100*_lo;
         _up=100*_up;
         call symput('pctlpts',trim(left(put(_lo,best8.)))||' '||
                               trim(left(put(_up,best8.))));
      run;
      proc univariate data=BOOTTRAN noprint pctldef=5;
         var value;
         output out=BOOTPCTL n=n
            pctlpts=&pctlpts pctlpre=&pctlpre pctlname=&pctlname;
         by %if %bquote(&id)^= %then &id; name;
      run;
   %end;
   data BOOTCI;
      retain &id name value alcl aucl confid method n;
      merge
         %if &qmethod=T
            %then _ACT_T_(rename=(_name_=name value1=value));
            %else _ACTTR_(rename=(_name_=name value1=value));
         BOOTPCTL;
      by %if %bquote(&id)^= %then &id; name;
      %if &qmethod=HYBRID %then %do;
         aucl=2*value-aucl;
         alcl=2*value-alcl;
      %end;
      %else %if &qmethod=T %then %do;
         aucl=value-aucl*student;
         alcl=value-alcl*student;
      %end;
      confid=&conf;
      length method $20;
      method='Bootstrap '||symget('method');
      label name  ='Name'
            value ='Observed Statistic'
            alcl  ='Approximate Lower Confidence Limit'
            aucl  ='Approximate Upper Confidence Limit'
            confid='Confidence Level (%)'
            method='Method for Confidence Interval'
            n     ='Number of Resamples'
            ;
   run;
   %if &print %then %do;
      proc print data=BOOTCI label;
         id %if %bquote(&id)^= %then &id; name;
      run;
   %end;
%exit:
%mend bootci;
```

```
****************************** ALLCI ******************************;
%macro allci(      /* Computes all types of confidence intervals
                      available in BOOTCI. Creates output data set
                      ALLCI. */
   stat=,          /* Numeric variables in the OUT= data set created
                      by the %ANALYZE macro that contain the values
                      of statistics for which you want to compute
                      bootstrap distributions. */
   student=,       /* For the T method only, numeric variables in the
                      OUT= data set created by the %ANALYZE macro that
                      contain the standard errors of the statistics for
which
                      you want to compute bootstrap distributions.
                      There must be a one-to-one between the VAR=
                      variables and the STUDENT= variables */
   id=,            /* One or more numeric or character variables that
                      uniquely identify the observations of the OUT=
                      data set within each BY group. No ID variables
                      are needed if the OUT= data set has only one
                      observation per BY group.
                      The ID variables may not be named _TYPE_, _NAME_,
                      or _STAT_ */
   alpha=.05,      /* significance (i.e., one minus confidence) level
                      for confidence intervals */
   keep=,          /* Variables to keep in the output data set
                      containing the confidence intervals; can be used
                      to avoid warnings from PROC TRANSPOSE */
   print=1);       /* 1 to print the bootstrap confidence intervals;
                      0 otherwise. */
   %if %bquote(&keep)^= %then %let keep=(keep=&keep);
   %bootci(bca,stat=&stat,id=&id,alpha=&alpha,print=0)
   data ALLCI; set bootci&keep; run;
   %bootci(bc,stat=&stat,id=&id,alpha=&alpha,print=0)
   proc append data=bootci&keep base=ALLCI force; run;
   %bootci(pctl,stat=&stat,id=&id,alpha=&alpha,print=0)
   proc append data=bootci&keep base=ALLCI force; run;
   %bootci(hybrid,stat=&stat,id=&id,alpha=&alpha,print=0)
   proc append data=bootci&keep base=ALLCI force; run;
   %if %bquote(&student)^= %then %do;
      %bootci(t,stat=&stat,id=&id,student=&student,alpha=&alpha,print=0)
      proc append data=bootci&keep base=ALLCI force; run;
   %end;
   proc append data=bootstat&keep base=ALLCI force; run;
   proc append data=jackstat&keep base=ALLCI force; run;
   %if &print %then %do;
      proc print data=ALLCI label;
         id %if %bquote(&id)^= %then &id; name;
      run;
   %end;
%mend allci;
%macro bystmt;
   %let useby=1;
   by &by;
%mend bystmt;
%macro vardef;
   %let usevardf=1;
   vardef=&vardef
%mend vardef;
%macro bootin; /* INTERNAL USE ONLY
       input an observation from the original data set */
   %if %bquote(&residual)^= %then %do;
      array _r(&_nobs) _temporary_; /* residuals */
```

```
      do _i=1 to &_nobs;
         set &data point=_i;
         _r(_i)=&residual;
      end;
   %end;
   %else %do;
      drop _i;
   %end;
%mend bootin;
%macro bootout(obs); /* INTERNAL USE ONLY
       output an observation to the resampled data set */
   %if %bquote(&residual)^= %then %do;
      set &data point=_i;
      &residual=_r(&obs);
      &equation;
   %end;
   %else %do;
      set &data point=&obs;
   %end;
   output;
%mend bootout;
```

Macro CleanData

```
options pageno=1 ls=80 ps=60 nocenter;
data _null_;
       call symput("thedate", put(today(), yymmdd6.));
run;
%macro CleanData(realdata=);
data super00;
       set "&current.&realdata";
       title " ";
run;
proc contents data=super00 out=superout;
run;
data superout;
       set superout;
       keep varnum nobs;
run;
proc iml;
       use superout;
       read all into matsuperout;
       n=nrow(matsuperout);
       lastobs=matsuperout[n,];
       create parameter from lastobs[colname={"nvar","nsamp"}];
       append from lastobs;

       use super00;
       read all into matsuper;
       create super0 from matsuper;
       append from matsuper;
run;
data parameter;
       set parameter;
       call symputx("nvar",nvar);
       call symputx("nsamp",nsamp);
run;
data super00;
       set super0;
       rename col1-col&nvar=x1-x&nvar;
       nvar=&nvar;
       nsamp=&nsamp;
```

```
run;
proc iml;
	use super00;
	read all into tempsuper00;
	tempsuper=tempsuper00[,&nvar+1:&nvar+2] || tempsuper00[,1:&nvar];
	create super from tempsuper;
	append from tempsuper;
quit;
data super "&current.super";
	set super;
	rename col1=nvar col2=nsamp col3-col%eval(2+&nvar)=x1-x&nvar;
run;
proc print data=super(obs=10);
run;
ods pdf file="&current.&thedate CleanData.pdf" style=sansprinter;
ods noresults;
ods proclabel='gplot';
symbol1 value=plus color=black height=10pt pointlabel=none;
goptions device=gif ftext='arial' htext=16pt gunit=in hsize=8 vsize=6;
options orientation=landscape;
options nodate nonumber;
	proc gplot data=super;
		title font='arial' height=16pt "data super : bivariate
scatter plot of original data set";
		title2 font='arial' height=16pt "n=&nvar nsamp=&nsamp ";
		%do i=1 %to %eval(&nvar-1);
			%do j=2 %to &nvar;
				%if &i<&j %then %do;
				plot x&i*x&j;
				%end;
			%end;
		%end;
	run;
ods pdf close;
%mend;
```

Macro_RMA_Orin

```
options pageno=1 ls=80 ps=60 nocenter;
data _null_;
	call symput("thedate", put(today(), yymmdd6.));
run;
/*perform rma on original data*/;
%macro RMA_Orin(step=,nrep=);

	%if &step=2 %then %RMA_2Var_Orin(nest=&step,nrep=&nrep);

	%if &step=3 %then %RMA_3Var_Orin(nest=&step,nrep=&nrep);

	%if &step>=4 %then %RMA_4Var_Orin(nest=&step,nrep=&nrep);

%mend RMA_Orin;
```

Macro_RMA_2Var_Orin

```
options pageno=1 ls=80 ps=60 nocenter;
data _null_;
	call symput("thedate", put(today(), yymmdd6.));
run;
%macro RMA_2Var_Orin(nest=,nrep=);
proc iml;
start regress(original) global(initial);
initial1=j(ncol(original),ncol(original),0);
```

```
initial2=j(ncol(original),ncol(original),0);
initial=j(ncol(original),1,0);
rsq=j(1,ncol(original),0);
       do m=1 to ncol(original);
                     yy=loc(do(1,ncol(original),1)^=m);
                     x=j(nrow(original),1,1)||original[,yy];
                     y=original[,m];
                     beta=inv(x`*x)*x`*y;
                     resid=y-x*beta;
                     sse=ssq(resid);
                     cssy=ssq(y-sum(y)/nrow(original));
                     rsquare=(cssy-sse)/cssy;
                     rsq[m]=rsquare;
              do n=1 to ncol(original);
                     initial1[n,m]=beta[n];
              end;
                     rsq[m]=rsquare;
       end;
       do i=1 to ncol(initial1);
                     initial2[i,]=-initial1[i,]/initial1[1,];
              do j=i+1 to ncol(initial1);
                     initial2[i,j]=-initial1[i+1,j]/initial1[1,j];
              end;
                     initial2[i,1]=-initial1[i,1]/initial1[1,1];
                     initial2[i,i]=1/initial1[1,i];
       end;
       create allols from initial2 [colname={'X1asDepend','X2asDepend'}];
       append from initial2;
       create allrsq from rsq [colname={'X1asDepend','X2asDepend'}];
       append from rsq;
       maxrsq=max(rsq);
       initial=initial2[,loc(rsq=maxrsq)];
       create allinitial from initial;
       append from initial;
finish regress;
start minls(b) global(xxorig);
       original=xxorig;
       f=j(&nsamp,1,0);
       prod=1;
       const=1;
       do j = 1 to &nest;
              prod=prod*b[j];
       end;
       prod=abs(prod);
       do i = 1 to &nsamp;
              pred=0;
              do j = 1 to &nest;
                     pred=pred +b[j]*original[i,j];
              end;
              pow=2/&nest;
              f[i]= (sqrt(((abs(-const+pred))**&nest)/prod))**pow;
       end;
       return(f);
finish minls;
start minlg(b) global(xxorig);
       original=xxorig;
       f=0;
       prod=1;
       const=1;
       do j = 1 to &nest;
              prod=prod*b[j];
       end;
```

```
        prod=abs(prod);
        do i = 1 to &nsamp;
                pred=0;
                do j = 1 to &nest;
                        pred=pred +b[j]*original[i,j];
                end;
                v= ((abs(-const+pred)))**&nest/(prod);
                f=f+v**(2/&nest);
        end;
        f=(f*.5);
        return(f);
finish minlg;
start TwoVarExact(original) global(xxorig,initial,b);
        original=xxorig;
        xbar=j(&nsamp,&nest,0);
        std=j(1,&nest,0);
        do i=1 to &nsamp;
                do j=1 to &nest;
                        xbar[i,j]=original[+,j]/&nsamp;
                end;
        end;
        xxs=original-xbar;
        sign=sum(xxs[,1]#xxs[,2])/abs(sum(xxs[,1]#xxs[,2]));
        slope=sign*sqrt(sum(xxs[,2]#xxs[,2])/sum(xxs[,1]#xxs[,1]));
        intercept=(xbar[,2]-slope*xbar[,1])[1];
        b=j(1,&nest,0);
        %if intercept ne 0 %then %do;
                b[1]=-slope/intercept;
                b[2]=1/intercept;
        %end;
        %if intercept=0 %then %do;
                intercept=0.000001;
                b[1]=-slope/intercept;
                b[2]=1/intercept;
        %end;
finish TwoVarExact;
sumresult=j(1,&nest+2+1,0);
        title "";
        use orin;
        read all into aa;
        cc=j(nrow(aa),%eval(2+&nest),.);
        do i=1 to nrow(aa);
                bb=loc(aa[i,1:%eval(2+&nvar)]^=.);
                cc[i,]=aa[i,bb];
        end;
        x=cc[,3:%eval(2+&nest)];
        xxorig=x;
        original=x;
        run regress(original);
        run TwoVarExact(original);
        fopt1=minlg(b);
        sumresult[,1]=&nvar;
        sumresult[,2]=&nsamp;
        sumresult[,3:(&nest+2)]=b;
        sumresult[,&nest+2+1]=fopt1;
create sol from sumresult
[colname={'nvar','nsamp','est1','est2','obj_func'}];
append from sumresult;
quit;
%mend RMA_2Var_Orin;
```

Macro_RMA_3Var_Orin

```
options pageno=1 ls=80 ps=60 nocenter;
data _null_;
       call symput("thedate", put(today(), yymmdd6.));
run;
%macro RMA_3Var_Orin(nest=,nrep=);
proc iml;
start regress(original) global(initial);
initial1=j(ncol(original),ncol(original),0);
initial2=j(ncol(original),ncol(original),0);
initial=j(ncol(original),1,0);
rsq=j(1,ncol(original),0);
       do m=1 to ncol(original);
                     yy=loc(do(1,ncol(original),1)^=m);
                     x=j(nrow(original),1,1)||original[,yy];
                     y=original[,m];
                     beta=inv(x`*x)*x`*y;
                     resid=y-x*beta;
                     sse=ssq(resid);
                     cssy=ssq(y-sum(y)/nrow(original));
                     rsquare=(cssy-sse)/cssy;
                     rsq[m]=rsquare;
              do n=1 to ncol(original);
                     initial1[n,m]=beta[n];
              end;
                     rsq[m]=rsquare;
       end;
       do i=1 to ncol(initial1);
                     initial2[i,]=-initial1[i,]/initial1[1,];
              do j=i+1 to ncol(initial1);
                     initial2[i,j]=-initial1[i+1,j]/initial1[1,j];
              end;
                     initial2[i,1]=-initial1[i,1]/initial1[1,1];
                     initial2[i,i]=1/initial1[1,i];
       end;
       create allols from initial2
[colname={'X1asDepend','X2asDepend','X3asDepend'}];
       append from initial2;
       create allrsq from rsq
[colname={'X1asDepend','X2asDepend','X3asDepend'}];
       append from rsq;
       maxrsq=max(rsq);
       initial=initial2[,loc(rsq=maxrsq)];
       create allinitial from initial;
       append from initial;
finish regress;
start minls(b) global(xxorig);
       original=xxorig;
       f=j(&nsamp,1,0);
       prod=1;
       const=1;
       do j = 1 to &nest;
              prod=prod*b[j];
       end;
       prod=abs(prod);
       do i = 1 to &nsamp;
              pred=0;
              do j = 1 to &nest;
                     pred=pred +b[j]*original[i,j];
              end;
              pow=2/&nest;
```

```
            f[i]= (sqrt(((abs(-const+pred))**&nest)/prod))**pow;
      end;
      return(f);
finish minls;
start minlg(b) global(xxorig);
      original=xxorig;
      f=0;
      prod=1;
      const=1;
      do j = 1 to &nest;
            prod=prod*b[j];
      end;
      prod=abs(prod);
      do i = 1 to &nsamp;
            pred=0;
            do j = 1 to &nest;
                  pred=pred +b[j]*original[i,j];
            end;
            v= ((abs(-const+pred)))**&nest/(prod);
            f=f+v**(2/&nest);
      end;
      f=(f*.5);
      return(f);
finish minlg;
start ThreeVarExact(original) global(xxorig,initial,b);
      original=xxorig;
      xbar=j(&nsamp,&nest,0);
      std=j(1,&nest,0);
      do i=1 to &nsamp;
            do j=1 to &nest;
                  xbar[i,j]=original[+,j]/&nsamp;
            end;
      end;
      xxs=original-xbar;
      xxs2=xxs#xxs;
      do j=1 to &nest;
            std[j]=sqrt(xxs2[+,j]/(&nsamp-1));
            xxs[,j]=xxs[,j]/std[j];
      end;
      crr=xxs`*xxs/(&nsamp-1);
      crrvec=j(3,1,0);
      crrvec[1]=crr[1,2];
      crrvec[2]=crr[1,3];
      crrvec[3]=crr[2,3];
      la=min(crrvec);
      nu=max(crrvec);
      mu=median(crrvec);
      if la = crr[1,2] then if nu = crr[1,3] then order= {3 1 2};
      if la = crr[1,2] then if nu = crr[2,3] then order= {3 2 1};
      if la = crr[1,3] then if nu = crr[1,2] then order= {2 1 3};
      if la = crr[1,3] then if nu = crr[2,3] then order= {2 3 1};
      if la = crr[2,3] then if nu = crr[1,2] then order= {1 2 3};
      if la = crr[2,3] then if nu = crr[1,3] then order= {1 3 2};
      xbar=xbar[order];
      std=std[order];
      aa=j(5,1,0);
      aa[1]=1-la**2;
      aa[2]=la*(mu-la*nu);
      aa[3]=2*(la*mu*nu-1);
      aa[4]=mu*(la-mu*nu);
      aa[5]=1-mu**2;
      sol=polyroot(aa);
```

```
	prodsol=sol[#,1];
	if prodsol > 0 | prodsol < 0 then do;
		%do k=1 %to 4;
			alpha=sol[&k,1];
			beta=(1-alpha**2)/(mu-la*alpha);
			xyz=j(1,3,0);
			xyz[1]=1;
			xyz[2]=alpha;
			xyz[3]=-beta;
			aaa=xyz[1]/std[1];
			bbb=xyz[2]/std[2];
			ccc=xyz[3]/std[3];
			ddd=(aaa*xbar[1]+bbb*xbar[2]+ccc*xbar[3]);
			coefff=j(1,3,0);
			coefff[1]=aaa/ddd;
			coefff[2]=bbb/ddd;
			coefff[3]=ccc/ddd;
			coefff2=j(1,3,0);
			coefff2[order[1]]=coefff[1];
			coefff2[order[2]]=coefff[2];
			coefff2[order[3]]=coefff[3];
			b&k=j(1,3,0);
			b&k[1]=coefff2[1];
			b&k[2]=coefff2[2];
			b&k[3]=coefff2[3];
	result=aa[1]*sol[,1]##4+aa[2]*sol[,1]##3+aa[3]*sol[,1]##2+aa[4]*so
l[,1]+aa[5]*j(4,1,1);
		minf&k=minlg(b&k);
		%end;
		tttt=(minf1||b1) // (minf2||b2) // (minf3||b3) //
(minf4||b4);
		minf=min(minf1,minf2,minf3,minf4);
		locminf=loc(tttt[,1]=minf);
		b=tttt[locminf,2:4];
	end;
	if prodsol=0 then do;
		optn1= &nsamp//{0};
		result1=j(&nrep,2*&nest+1,0);
		seed44=89425;
		do r=1 to &nrep;
			ini=j(1,&nest,0);
			do v = 1 to &nest;
				ini[v]=initial[v]+2*(ranuni(seed44)-0.5);
			end;
			call nlplm(rc1,xres1,"minls",ini,optn1);
			fopt=minlg(xres1);
			result1[r,1]=fopt;
			result1[r,2:(&nest+1)]=ini;
			result1[r,(&nest+2):(2*&nest+1)]=xres1;
		end;
		minfopt=min(result1[,1]);
		locminfopt=loc(result1[,1]=minfopt)[1];
		iniguess=j(1,&nest,0);
		iniguess=result1[locminfopt,2:(&nest+1)];
		b=j(1,&nest,0);
		b=result1[locminfopt,(&nest+2):(2*&nest+1)];
	end;
finish ThreeVarExact;
sumresult=j(1,&nest+2+1,0);
	title "";
	use orin;
	read all into aa;
```

```
        cc=j(nrow(aa),%eval(2+&nest),.);
        do i=1 to nrow(aa);
                bb=loc(aa[i,1:%eval(2+&nvar)]^=.);
                cc[i,]=aa[i,bb];
        end;
        x=cc[,3:%eval(2+&nest)];
        xxorig=x;
        original=x;
        run regress(original);
        run ThreeVarExact(original);
        fopt1=minlg(b);
        sumresult[,1]=&nvar;
        sumresult[,2]=&nsamp;
        sumresult[,3:(&nest+2)]=b;
        sumresult[,&nest+2+1]=fopt1;
create sol from sumresult
[colname={'nvar','nsamp','est1','est2','est3','obj_func'}];
append from sumresult;
quit;
%mend RMA_3Var_Orin;
```

Macro_RMA_4Var_Orin

```
options pageno=1 ls=80 ps=60 nocenter;
data _null_;
        call symput("thedate", put(today(), yymmdd6.));
run;
%macro RMA_4Var_Orin(nest=,nrep=);
proc iml;
start regress(original) global(initial);
initial1=j(ncol(original),ncol(original),0);
initial2=j(ncol(original),ncol(original),0);
initial=j(ncol(original),1,0);
rsq=j(1,ncol(original),0);
        do m=1 to ncol(original);
                        yy=loc(do(1,ncol(original),1)^=m);
                        x=j(nrow(original),1,1)||original[,yy];
                        y=original[,m];
                        beta=inv(x`*x)*x`*y;
                        resid=y-x*beta;
                        sse=ssq(resid);
                        cssy=ssq(y-sum(y)/nrow(original));
                        rsquare=(cssy-sse)/cssy;
                        rsq[m]=rsquare;
                do n=1 to ncol(original);
                        initial1[n,m]=beta[n];
                end;
                        rsq[m]=rsquare;
        end;
        do i=1 to ncol(initial1);
                        initial2[i,]=-initial1[i,]/initial1[1,];
                do j=i+1 to ncol(initial1);
                        initial2[i,j]=-initial1[i+1,j]/initial1[1,j];
                end;
                        initial2[i,1]=-initial1[i,1]/initial1[1,1];
                        initial2[i,i]=1/initial1[1,i];
        end;
        cnameall={'X1asDepend','X2asDepend','X3asDepend','X4asDepend','X5a
sDepend','X6asDepend','X7asDepend','X8asDepend','X9asDepend','X10asDepend
','X11asDepend','X12asDepend','X13asDepend','X14asDepend','X15asDepend','
X16asDepend'};
        cnameselect=j(&nest,1,.);
```

```
       cnameselect=cnameall[1:&nest];
       create allols from initial2 [colname=cnameselect];
       append from initial2;
       create allrsq from rsq [colname=cnameselect];
       append from rsq;
       maxrsq=max(rsq);
       initial=initial2[,loc(rsq=maxrsq)];
       create allinitial from initial;
       append from initial;
finish regress;
start minls(b) global(xxorig);
       original=xxorig;
       f=j(&nsamp,1,0);
       prod=1;
       const=1;
       do j = 1 to &nest;
              prod=prod*b[j];
       end;
       prod=abs(prod);
       do i = 1 to &nsamp;
              pred=0;
              do j = 1 to &nest;
                     pred=pred +b[j]*original[i,j];
              end;
              pow=2/&nest;
              f[i]= (sqrt(((abs(-const+pred))**&nest)/prod))**pow;
       end;
       return(f);
finish minls;
start minlg(b) global(xxorig);
       original=xxorig;
       f=0;
       prod=1;
       const=1;
       do j = 1 to &nest;
              prod=prod*b[j];
       end;
       prod=abs(prod);
       do i = 1 to &nsamp;
              pred=0;
              do j = 1 to &nest;
                     pred=pred +b[j]*original[i,j];
              end;
              v= ((abs(-const+pred)))**&nest/(prod);
              f=f+v**(2/&nest);
       end;
       f=(f*.5);
       return(f);
finish minlg;
start rma_4var(original) global(xxorig,initial,b);
       original=xxorig;

       optn1= &nsamp//{0};
       result1=j(&nrep,2*&nest+1,0);
       seed44=89425;
       do r=1 to &nrep;
              ini=j(1,&nest,0);
              do v = 1 to &nest;
                     ini[v]=initial[v]+2*(ranuni(seed44)-0.5);
              end;
              call nlplm(rc1,xres1,"minls",ini,optn1);
              fopt=minlg(xres1);
```

```
                result1[r,1]=fopt;
                result1[r,2:(&nest+1)]=ini;
                result1[r,(&nest+2):(2*&nest+1)]=xres1;
        end;
        minfopt=min(result1[,1]);
        locminfopt=loc(result1[,1]=minfopt)[1];
        iniguess=j(1,&nest,0);
        iniguess=result1[locminfopt,2:(&nest+1)];
        b=j(1,&nest,0);
        b=result1[locminfopt,(&nest+2):(2*&nest+1)];
finish rma_4var;
sumresult=j(1,&nest+2+1,0);
        title "";
        use orin;
        read all into aa;
        cc=j(nrow(aa),%eval(2+&nest),.);
        do i=1 to nrow(aa);
                bb=loc(aa[i,1:%eval(2+&nvar)]^=.);
                cc[i,]=aa[i,bb];
        end;
        x=cc[,3:%eval(2+&nest)];
        xxorig=x;
        original=x;
        run regress(original);
        run rma_4var(original);
        fopt1=minlg(b);
        sumresult[,1]=&nvar;
        sumresult[,2]=&nsamp;
        sumresult[,3:(&nest+2)]=b;
        sumresult[,&nest+2+1]=fopt1;
cnameall2={'nvar','nsamp','est1','est2','est3','est4','est5','est6','est7
','est8','est9','est10','est11','est12','est13','est14','est15','est16','
obj_func'};
cnameselect2=j(%eval(&nest+1),1,.);
cnameselect2=cnameall2[1:%eval(&nest+2)]//cnameall2[19];
create sol from sumresult [colname=cnameselect2];
append from sumresult;
quit;
%mend RMA_4Var_Orin;
```

Macro_RMA_Boot

```
options pageno=1 ls=80 ps=60 nocenter;
data _null_;
        call symput("thedate", put(today(), yymmdd6.));
run;
%macro RMA_Boot(step=,nrep=);

        %if &step=2 %then %RMA_2Var_Boot(nest=&step,nrep=&nrep);

        %if &step=3 %then %RMA_3Var_Boot(nest=&step,nrep=&nrep);

        %if &step>=4 %then %RMA_4Var_Boot(nest=&step,nrep=&nrep);

%mend RMA_Boot;
```

Macro_RMA_2Var_Boot

```
options pageno=1 ls=80 ps=60 nocenter;
data _null_;
        call symput("thedate", put(today(), yymmdd6.));
run;
%macro RMA_2Var_Boot(nest=,nrep=);
```

```
proc iml;
start minls(b) global(xxorig);
       original=xxorig;
       f=j(nrow(original),1,0);
       prod=1;
       const=1;
       do j = 1 to &nest;
              prod=prod*b[j];
       end;
       prod=abs(prod);
       do i = 1 to nrow(original);
              pred=0;
              do j = 1 to &nest;
                     pred=pred +b[j]*original[i,j];
              end;
              pow=2/&nest;
              f[i]= (sqrt(((abs(-const+pred))**&nest)/prod))**pow;
       end;
       return(f);
finish minls;
start minlg(b) global(xxorig);
       original=xxorig;
       f=0;
       prod=1;
       const=1;
       do j = 1 to &nest;
              prod=prod*b[j];
       end;
       prod=abs(prod);
       do i = 1 to nrow(original);
              pred=0;
              do j = 1 to &nest;
                     pred=pred +b[j]*original[i,j];
              end;
              v= ((abs(-const+pred)))**&nest/(prod);
              f=f+v**(2/&nest);
       end;
       f=(f*.5);
       return(f);
finish minlg;
start TwoVarExact(original) global(xxorig,initial,b);
       original=xxorig;
       xbar=j(nrow(original),&nest,0);
       std=j(1,&nest,0);
       do i=1 to nrow(original);
              do j=1 to &nest;
                     xbar[i,j]=original[+,j]/nrow(original);
              end;
       end;
       xxs=original-xbar;
       sign=sum(xxs[,1]#xxs[,2])/abs(sum(xxs[,1]#xxs[,2]));
       slope=sign*sqrt(sum(xxs[,2]#xxs[,2])/sum(xxs[,1]#xxs[,1]));
       intercept=(xbar[,2]-slope*xbar[,1])[1];
       b=j(1,&nest,0);
       %if intercept ne 0 %then %do;
              b[1]=-slope/intercept;
              b[2]=1/intercept;
       %end;
       %if intercept=0 %then %do;
              intercept=0.000001;
              b[1]=-slope/intercept;
              b[2]=1/intercept;
```

```
        %end;
finish TwoVarExact;
sumresult=j(1,&nest+2+1,0);
        title "";
        use allinitial;
        read all into initial;
        use &data;
        read all into aa;
        xstart=ncol(aa)-&nest+1;
        xend=ncol(aa);
        x=aa[,xstart:xend];
        xxorig=x;
        original=x;
        run TwoVarExact(original);
        fopt1=minlg(b);
        sumresult[,1]=&nvar;
        sumresult[,2]=&nsamp;
        sumresult[,3:(&nest+2)]=b;
        sumresult[,&nest+2+1]=fopt1;
create &out from sumresult
[colname={'nvar','nsamp','est1','est2','obj_func'}];
append from sumresult;
quit;
%mend RMA_2Var_Boot;
```

Macro_RMA_3Var_Boot

```
options pageno=1 ls=80 ps=60 nocenter;
data _null_;
        call symput("thedate", put(today(), yymmdd6.));
run;
%macro RMA_3Var_Boot(nest=,nrep=);
proc iml;
start minls(b) global(xxorig);
        original=xxorig;
        f=j(nrow(original),1,0);
        prod=1;
        const=1;
        do j = 1 to &nest;
                prod=prod*b[j];
        end;
        prod=abs(prod);
        do i = 1 to nrow(original);
                pred=0;
                do j = 1 to &nest;
                        pred=pred +b[j]*original[i,j];
                end;
                pow=2/&nest;
                f[i]= (sqrt(((abs(-const+pred))**&nest)/prod))**pow;
        end;
        return(f);
finish minls;
start minlg(b) global(xxorig);
        original=xxorig;
        f=0;
        prod=1;
        const=1;
        do j = 1 to &nest;
                prod=prod*b[j];
        end;
        prod=abs(prod);
        do i = 1 to nrow(original);
```

```
            pred=0;
            do j = 1 to &nest;
                  pred=pred +b[j]*original[i,j];
            end;
            v= ((abs(-const+pred)))**&nest/(prod);
            f=f+v**(2/&nest);
      end;
      f=(f*.5);
      return(f);
finish minlg;
start ThreeVarExact(original) global(xxorig,initial,b);
      original=xxorig;

      xbar=j(nrow(original),&nest,0);
      std=j(1,&nest,0);
      do i=1 to nrow(original);
            do j=1 to &nest;
                  xbar[i,j]=original[+,j]/nrow(original);
            end;
      end;
      xxs=original-xbar;
      xxs2=xxs#xxs;
      do j=1 to &nest;
            std[j]=sqrt(xxs2[+,j]/(nrow(original)-1));
            xxs[,j]=xxs[,j]/std[j];
      end;
      crr=xxs`*xxs/(nrow(original)-1);
      crrvec=j(3,1,0);
      crrvec[1]=crr[1,2];
      crrvec[2]=crr[1,3];
      crrvec[3]=crr[2,3];
      la=min(crrvec);
      nu=max(crrvec);
      mu=median(crrvec);
      if la = crr[1,2] then if nu = crr[1,3] then order= {3 1 2};
      if la = crr[1,2] then if nu = crr[2,3] then order= {3 2 1};
      if la = crr[1,3] then if nu = crr[1,2] then order= {2 1 3};
      if la = crr[1,3] then if nu = crr[2,3] then order= {2 3 1};
      if la = crr[2,3] then if nu = crr[1,2] then order= {1 2 3};
      if la = crr[2,3] then if nu = crr[1,3] then order= {1 3 2};
      xbar=xbar[order];
      std=std[order];
      aa=j(5,1,0);
      aa[1]=1-la**2;
      aa[2]=la*(mu-la*nu);
      aa[3]=2*(la*mu*nu-1);
      aa[4]=mu*(la-mu*nu);
      aa[5]=1-mu**2;
      sol=polyroot(aa);
      prodsol=sol[#,1];
      if prodsol > 0 | prodsol < 0 then do;
            %do k=1 %to 4;
                  alpha=sol[&k,1];
                  beta=(1-alpha**2)/(mu-la*alpha);
                  xyz=j(1,3,0);
                  xyz[1]=1;
                  xyz[2]=alpha;
                  xyz[3]=-beta;
                  aaa=xyz[1]/std[1];
                  bbb=xyz[2]/std[2];
                  ccc=xyz[3]/std[3];
                  ddd=(aaa*xbar[1]+bbb*xbar[2]+ccc*xbar[3]);
```

```
                        coefff=j(1,3,0);
                        coefff[1]=aaa/ddd;
                        coefff[2]=bbb/ddd;
                        coefff[3]=ccc/ddd;
                        coefff2=j(1,3,0);
                        coefff2[order[1]]=coefff[1];
                        coefff2[order[2]]=coefff[2];
                        coefff2[order[3]]=coefff[3];
                        b&k=j(1,3,0);
                        b&k[1]=coefff2[1];
                        b&k[2]=coefff2[2];
                        b&k[3]=coefff2[3];
        result=aa[1]*sol[,1]##4+aa[2]*sol[,1]##3+aa[3]*sol[,1]##2+aa[4]*so
l[,1]+aa[5]*j(4,1,1);
                minf&k=minlg(b&k);
                %end;
                tttt=(minf1||b1) // (minf2||b2) // (minf3||b3) //
(minf4||b4);
                minf=min(minf1,minf2,minf3,minf4);
                locminf=loc(tttt[,1]=minf);
                b=tttt[locminf,2:4];
        end;
        if prodsol=0 then do;
                optn1= nrow(original)//{0};
                result1=j(&nrep,2*&nest+1,0);
                seed44=89425;
                do r=1 to &nrep;
                        ini=j(1,&nest,0);
                        do v = 1 to &nest;
                                ini[v]=initial[v]+2*(ranuni(seed44)-0.5);
                        end;
                        call nlplm(rc1,xres1,"minls",ini,optn1);
                        fopt=minlg(xres1);
                        result1[r,1]=fopt;
                        result1[r,2:(&nest+1)]=ini;
                        result1[r,(&nest+2):(2*&nest+1)]=xres1;
                end;
                minfopt=min(result1[,1]);
                locminfopt=loc(result1[,1]=minfopt)[1];
                iniguess=j(1,&nest,0);
                iniguess=result1[locminfopt,2:(&nest+1)];
                b=j(1,&nest,0);
                b=result1[locminfopt,(&nest+2):(2*&nest+1)];
        end;
finish ThreeVarExact;
sumresult=j(1,&nest+2+1,0);
        title "";
        use allinitial;
        read all into initial;
        use &data;
        read all into aa;
        xstart=ncol(aa)-&nest+1;
        xend=ncol(aa);
        x=aa[,xstart:xend];
        xxorig=x;
        original=x;
        run ThreeVarExact(original);
        fopt1=minlg(b);
        sumresult[,1]=&nvar;
        sumresult[,2]=&nsamp;
        sumresult[,3:(&nest+2)]=b;
        sumresult[,&nest+2+1]=fopt1;
```

```
create &out from sumresult
[colname={'nvar','nsamp','est1','est2','est3','obj_func'}];
append from sumresult;
quit;
%mend RMA_3Var_Boot;
```

Macro_RMA_4Var_Boot

```
options pageno=1 ls=80 ps=60 nocenter;
data _null_;
      call symput("thedate", put(today(), yymmdd6.));
run;
%macro RMA_4Var_Boot(nest=,nrep=);
proc iml;

start minls(b) global(xxorig);
      original=xxorig;
      f=j(nrow(original),1,0);
      prod=1;
      const=1;
      do j = 1 to &nest;
            prod=prod*b[j];
      end;
      prod=abs(prod);
      do i = 1 to nrow(original);
            pred=0;
            do j = 1 to &nest;
                  pred=pred +b[j]*original[i,j];
            end;
            pow=2/&nest;
            f[i]= (sqrt(((abs(-const+pred))**&nest)/prod))**pow;
      end;
      return(f);
finish minls;
start minlg(b) global(xxorig);
      original=xxorig;
      f=0;
      prod=1;
      const=1;
      do j = 1 to &nest;
            prod=prod*b[j];
      end;
      prod=abs(prod);
      do i = 1 to nrow(original);
            pred=0;
            do j = 1 to &nest;
                  pred=pred +b[j]*original[i,j];
            end;
            v= ((abs(-const+pred)))**&nest/(prod);
            f=f+v**(2/&nest);
      end;
      f=(f*.5);
      return(f);
finish minlg;
start rma_4var(original) global(xxorig,initial,b);
      original=xxorig;
      optn1= nrow(original)//{0};
      result1=j(&nrep,2*&nest+1,0);
      seed44=89425;
      do r=1 to &nrep;
            ini=j(1,&nest,0);
            do v = 1 to &nest;
```

```
                ini[v]=initial[v]+2*(ranuni(seed44)-0.5);
            end;
            call nlplm(rc1,xres1,"minls",ini,optn1);
            fopt=minlg(xres1);
            result1[r,1]=fopt;
            result1[r,2:(&nest+1)]=ini;
            result1[r,(&nest+2):(2*&nest+1)]=xres1;
        end;
        minfopt=min(result1[,1]);
        locminfopt=loc(result1[,1]=minfopt)[1];
        iniguess=j(1,&nest,0);
        iniguess=result1[locminfopt,2:(&nest+1)];
        b=j(1,&nest,0);
        b=result1[locminfopt,(&nest+2):(2*&nest+1)];
finish rma_4var;
        title "";
        use allinitial;
        read all into initial;
        use &data;
        read all into aa;
        xstart=ncol(aa)-&nest+1;
        xend=ncol(aa);
        x=aa[,xstart:xend];
        xxorig=x;
        original=x;
        run rma_4var(original);
        fopt1=minlg(b);
        sumresult=j(1,&nest+1+2,0);
        sumresult[,1]=&nvar;
        sumresult[,2]=&nsamp;
        sumresult[,3:(&nest+2)]=b;
        sumresult[,&nest+1+2]=fopt1;
cnameall2={'nvar','nsamp','est1','est2','est3','est4','est5','est6','est7
','est8','est9','est10','est11','est12','est13','est14','est15','est16','
obj_func'};
cnameselect2=j(%eval(&nest+1+2),1,.);
cnameselect2=cnameall2[1:%eval(&nest+2)]//cnameall2[17+2];
create &out from sumresult [colname=cnameselect2];
append from sumresult;
quit;
%mend RMA_4Var_Boot;
```

Macro_Jackout

```
options pageno=1 ls=80 ps=60 nocenter;
data _null_;
        call symput("thedate", put(today(), yymmdd6.));
run;
%macro Jackout(nvar=,nrep=);
        %macro analyze(data=,out=);
                %RMA_Jack(step=&nvar, nrep=&nrep);
        %mend;
/*Select data of interest for bootstrapping*/
/*Perform RMA on each bootstrapped data and obtain an CI with 6 methods*/
        title2 'Leave-One_Out Approach';
        %jack(data=orin,stat=est1-est&nvar);
        proc sort data=jackdist out=jackout;
                by _sample_;
        run;
        data "&current.jackout";
                set jackout;
                title 'Jackknife Resampling';
```

```
        run;
%mend Jackout;
```

Macro_RMA_Jack

```
options pageno=1 ls=80 ps=60 nocenter;

data _null_;
        call symput("thedate", put(today(), yymmdd6.));
run;
%macro RMA_Jack(step=,nrep=);

        %if &step=2 %then %RMA_2Var_Jack(nest=&step,nrep=&nrep);
        %if &step=3 %then %RMA_3Var_Jack(nest=&step,nrep=&nrep);
        %if &step>=4 %then %RMA_4Var_Jack(nest=&step,nrep=&nrep);
%mend RMA_Jack;
```

Macro_RMA_2Var_Jack

```
options pageno=1 ls=80 ps=60 nocenter;
data _null_;
        call symput("thedate", put(today(), yymmdd6.));
run;
%macro RMA_2Var_Jack(nest=,nrep=);
proc iml;
start regress(original) global(initial);
initial1=j(ncol(original),ncol(original),0);
initial2=j(ncol(original),ncol(original),0);
initial=j(ncol(original),1,0);
rsq=j(1,ncol(original),0);
        do m=1 to ncol(original);
                        yy=loc(do(1,ncol(original),1)^=m);
                        x=j(nrow(original),1,1)||original[,yy];
                        y=original[,m];
                        beta=inv(x`*x)*x`*y;
                        resid=y-x*beta;
                        sse=ssq(resid);
                        cssy=ssq(y-sum(y)/nrow(original));
                        rsquare=(cssy-sse)/cssy;
                        rsq[m]=rsquare;
                do n=1 to ncol(original);
                        initial1[n,m]=beta[n];
                end;
                        rsq[m]=rsquare;
        end;
        do i=1 to ncol(initial1);
                        initial2[i,]=-initial1[i,]/initial1[1,];
                do j=i+1 to ncol(initial1);
                        initial2[i,j]=-initial1[i+1,j]/initial1[1,j];
                end;
                        initial2[i,i]=1/initial1[1,i];
        end;
        create allols from initial2 [colname={'X1asDepend','X2asDepend'}];
        append from initial2;
        create allrsq from rsq [colname={'X1asDepend','X2asDepend'}];
        append from rsq;
        maxrsq=max(rsq);
        initial=initial2[,loc(rsq=maxrsq)];
        create allinitial from initial;
        append from initial;
finish regress;
start minls(b) global(xxorig);
```

```
       original=xxorig;
       f=j(nrow(original),1,0);
       prod=1;
       const=1;
       do j = 1 to &nest;
              prod=prod*b[j];
       end;
       prod=abs(prod);
       do i = 1 to nrow(original);
              pred=0;
              do j = 1 to &nest;
                     pred=pred +b[j]*original[i,j];
              end;
              pow=2/&nest;
              f[i]= (sqrt(((abs(-const+pred))**&nest)/prod))**pow;
       end;
       return(f);
finish minls;
start minlg(b) global(xxorig);
       original=xxorig;
       f=0;
       prod=1;
       const=1;
       do j = 1 to &nest;
              prod=prod*b[j];
       end;
       prod=abs(prod);
       do i = 1 to nrow(original);
              pred=0;
              do j = 1 to &nest;
                     pred=pred +b[j]*original[i,j];
              end;
              v= ((abs(-const+pred)))**&nest/(prod);
              f=f+v**(2/&nest);
       end;
       f=(f*.5);
       return(f);
finish minlg;
start TwoVarExact(original) global(xxorig,initial,b);
       original=xxorig;
       xbar=j(nrow(original),&nest,0);
       std=j(1,&nest,0);
       do i=1 to nrow(original);
              do j=1 to &nest;
                     xbar[i,j]=original[+,j]/nrow(original);
              end;
       end;
       xxs=original-xbar;
       sign=sum(xxs[,1]#xxs[,2])/abs(sum(xxs[,1]#xxs[,2]));
       slope=sign*sqrt(sum(xxs[,2]#xxs[,2])/sum(xxs[,1]#xxs[,1]));
       intercept=(xbar[,2]-slope*xbar[,1])[1];
       b=j(1,&nest,0);
       %if intercept ne 0 %then %do;
              b[1]=-slope/intercept;
              b[2]=1/intercept;
       %end;
       %if intercept=0 %then %do;
              intercept=0.000001;
              b[1]=-slope/intercept;
              b[2]=1/intercept;
       %end;
finish TwoVarExact;
```

```
sumresult=j(1,&nest+2+1,0);
       title "";
       use &data;
       read all into aa;
       xstart=ncol(aa)-&nest+1;
       xend=ncol(aa);
       x=aa[,xstart:xend];
       xxorig=x;
       original=x;
       run regress(original);
       run TwoVarExact(original);

       fopt1=minlg(b);
       sumresult[,1]=&nvar;
       sumresult[,2]=&nsamp;
       sumresult[,3:(&nest+2)]=b;
       sumresult[,&nest+2+1]=fopt1;
create &out from sumresult
[colname={'nvar','nsamp','est1','est2','obj_func'}];
append from sumresult;
quit;
%mend RMA_2Var_Jack;
```

Macro_RMA_3Var_Jack

```
options pageno=1 ls=80 ps=60 nocenter;
data _null_;
       call symput("thedate", put(today(), yymmdd6.));
run;
%macro RMA_3Var_Jack(nest=,nrep=);
proc iml;
start regress(original) global(initial);
initial1=j(ncol(original),ncol(original),0);
initial2=j(ncol(original),ncol(original),0);
initial=j(ncol(original),1,0);
rsq=j(1,ncol(original),0);
       do m=1 to ncol(original);
                     yy=loc(do(1,ncol(original),1)^=m);
                     x=j(nrow(original),1,1)||original[,yy];
                     y=original[,m];
                     beta=inv(x`*x)*x`*y;
                     resid=y-x*beta;
                     sse=ssq(resid);
                     cssy=ssq(y-sum(y)/nrow(original));
                     rsquare=(cssy-sse)/cssy;
                     rsq[m]=rsquare;
              do n=1 to ncol(original);
                     initial1[n,m]=beta[n];
              end;
                     rsq[m]=rsquare;
       end;
       do i=1 to ncol(initial1);
                     initial2[i,]=-initial1[i,]/initial1[1,];
              do j=i+1 to ncol(initial1);
                     initial2[i,j]=-initial1[i+1,j]/initial1[1,j];
              end;
                     initial2[i,i]=1/initial1[1,i];
       end;
       create allols from initial2
[colname={'X1asDepend','X2asDepend','X3asDepend'}];
       append from initial2;
```

```
       create allrsq from rsq
[colname={'X1asDepend','X2asDepend','X3asDepend'}];
       append from rsq;
       maxrsq=max(rsq);
       initial=initial2[,loc(rsq=maxrsq)];
       create allinitial from initial;
       append from initial;
finish regress;
start minls(b) global(xxorig);
       original=xxorig;
       f=j(nrow(original),1,0);
       prod=1;
       const=1;
       do j = 1 to &nest;
              prod=prod*b[j];
       end;
       prod=abs(prod);
       do i = 1 to nrow(original);
              pred=0;
              do j = 1 to &nest;
                     pred=pred +b[j]*original[i,j];
              end;
              pow=2/&nest;
              f[i]= (sqrt(((abs(-const+pred))**&nest)/prod))**pow;
       end;
       return(f);
finish minls;
start minlg(b) global(xxorig);
       original=xxorig;
       f=0;
       prod=1;
       const=1;
       do j = 1 to &nest;
              prod=prod*b[j];
       end;
       prod=abs(prod);
       do i = 1 to nrow(original);
              pred=0;
              do j = 1 to &nest;
                     pred=pred +b[j]*original[i,j];
              end;
              v= ((abs(-const+pred)))**&nest/(prod);
              f=f+v**(2/&nest);
       end;
       f=(f*.5);
       return(f);
finish minlg;
start ThreeVarExact(original) global(xxorig,initial,b);
       original=xxorig;
       xbar=j(nrow(original),&nest,0);
       std=j(1,&nest,0);
       do i=1 to nrow(original);
              do j=1 to &nest;
                     xbar[i,j]=original[+,j]/nrow(original);
              end;
       end;
       xxs=original-xbar;
       xxs2=xxs#xxs;
       do j=1 to &nest;
              std[j]=sqrt(xxs2[+,j]/(nrow(original)-1));
              xxs[,j]=xxs[,j]/std[j];
       end;
```

```
        crr=xxs`*xxs/(nrow(original)-1);
        crrvec=j(3,1,0);
        crrvec[1]=crr[1,2];
        crrvec[2]=crr[1,3];
        crrvec[3]=crr[2,3];
        la=min(crrvec);
        nu=max(crrvec);
        mu=median(crrvec);
        if la = crr[1,2] then if nu = crr[1,3] then order= {3 1 2};
        if la = crr[1,2] then if nu = crr[2,3] then order= {3 2 1};
        if la = crr[1,3] then if nu = crr[1,2] then order= {2 1 3};
        if la = crr[1,3] then if nu = crr[2,3] then order= {2 3 1};
        if la = crr[2,3] then if nu = crr[1,2] then order= {1 2 3};
        if la = crr[2,3] then if nu = crr[1,3] then order= {1 3 2};
        xbar=xbar[order];
        std=std[order];
        aa=j(5,1,0);
        aa[1]=1-la**2;
        aa[2]=la*(mu-la*nu);
        aa[3]=2*(la*mu*nu-1);
        aa[4]=mu*(la-mu*nu);
        aa[5]=1-mu**2;
        sol=polyroot(aa);
        prodsol=sol[#,1];
        if prodsol > 0 | prodsol < 0 then do;
                %do k=1 %to 4;
                        alpha=sol[&k,1];
                        beta=(1-alpha**2)/(mu-la*alpha);
                        xyz=j(1,3,0);
                        xyz[1]=1;
                        xyz[2]=alpha;
                        xyz[3]=-beta;
                        aaa=xyz[1]/std[1];
                        bbb=xyz[2]/std[2];
                        ccc=xyz[3]/std[3];
                        ddd=(aaa*xbar[1]+bbb*xbar[2]+ccc*xbar[3]);
                        coefff=j(1,3,0);
                        coefff[1]=aaa/ddd;
                        coefff[2]=bbb/ddd;
                        coefff[3]=ccc/ddd;
                        coefff2=j(1,3,0);
                        coefff2[order[1]]=coefff[1];
                        coefff2[order[2]]=coefff[2];
                        coefff2[order[3]]=coefff[3];
                        b&k=j(1,3,0);
                        b&k[1]=coefff2[1];
                        b&k[2]=coefff2[2];
                        b&k[3]=coefff2[3];
        result=aa[1]*sol[,1]##4+aa[2]*sol[,1]##3+aa[3]*sol[,1]##2+aa[4]*so
l[,1]+aa[5]*j(4,1,1);
                minf&k=minlg(b&k);
                %end;
                tttt=(minf1||b1) // (minf2||b2) // (minf3||b3) //
(minf4||b4);
                minf=min(minf1,minf2,minf3,minf4);
                locminf=loc(tttt[,1]=minf);
                b=tttt[locminf,2:4];
        end;
        if prodsol=0 then do;
                optn1= nrow(original)//{0};
                result1=j(&nrep,2*&nest+1,0);
                seed44=89425;
```

```
                do r=1 to &nrep;
                        ini=j(1,&nest,0);
                        do v = 1 to &nest;
                                ini[v]=initial[v]+2*(ranuni(seed44)-0.5);
                        end;
                        call nlplm(rc1,xres1,"minls",ini,optn1);
                        fopt=minlg(xres1);
                        result1[r,1]=fopt;
                        result1[r,2:(&nest+1)]=ini;
                        result1[r,(&nest+2):(2*&nest+1)]=xres1;
                end;
                minfopt=min(result1[,1]);
                locminfopt=loc(result1[,1]=minfopt)[1];
                iniguess=j(1,&nest,0);
                iniguess=result1[locminfopt,2:(&nest+1)];
                b=j(1,&nest,0);
                b=result1[locminfopt,(&nest+2):(2*&nest+1)];
        end;
finish ThreeVarExact;
sumresult=j(1,&nest+2+1,0);
        title "";
        use &data;
        read all into aa;
        xstart=ncol(aa)-&nest+1;
        xend=ncol(aa);
        x=aa[,xstart:xend];
        xxorig=x;
        original=x;
        run regress(original);
        run ThreeVarExact(original);
        fopt1=minlg(b);
        sumresult[,1]=&nvar;
        sumresult[,2]=&nsamp;
        sumresult[,3:(&nest+2)]=b;
        sumresult[,&nest+2+1]=fopt1;
create &out from sumresult
[colname={'nvar','nsamp','est1','est2','est3','obj_func'}];
append from sumresult;
quit;
%mend RMA_3Var_Jack;
```

Macro_RMA_4Var_Jack

```
options pageno=1 ls=80 ps=60 nocenter;
data _null_;
        call symput("thedate", put(today(), yymmdd6.));
run;
%macro RMA_4Var_Jack(nest=,nrep=);
proc iml;
start regress(original) global(initial);
initial1=j(ncol(original),ncol(original),0);
initial2=j(ncol(original),ncol(original),0);
initial=j(ncol(original),1,0);
rsq=j(1,ncol(original),0);
        do m=1 to ncol(original);
                        yy=loc(do(1,ncol(original),1)^=m);
                        x=j(nrow(original),1,1)||original[,yy];
                        y=original[,m];
                        beta=inv(x`*x)*x`*y;
                        resid=y-x*beta;
                        sse=ssq(resid);
                        cssy=ssq(y-sum(y)/nrow(original));
```

```
                rsquare=(cssy-sse)/cssy;
                rsq[m]=rsquare;
        do n=1 to ncol(original);
                initial1[n,m]=beta[n];
        end;
                rsq[m]=rsquare;
    end;
    do i=1 to ncol(initial1);
                initial2[i,]=-initial1[i,]/initial1[1,];
        do j=i+1 to ncol(initial1);
                initial2[i,j]=-initial1[i+1,j]/initial1[1,j];
        end;
                initial2[i,i]=1/initial1[1,i];
    end;
    maxrsq=max(rsq);
    initial=initial2[,loc(rsq=maxrsq)];
    create allinitial from initial;
    append from initial;
finish regress;
start minls(b) global(xxorig);
    original=xxorig;
    f=j(nrow(original),1,0);
    prod=1;
    const=1;
    do j = 1 to &nest;
        prod=prod*b[j];
    end;
    prod=abs(prod);
    do i = 1 to nrow(original);
        pred=0;
        do j = 1 to &nest;
            pred=pred +b[j]*original[i,j];
        end;
        pow=2/&nest;
        f[i]= (sqrt(((abs(-const+pred))**&nest)/prod))**pow;
    end;
    return(f);
finish minls;
start minlg(b) global(xxorig);
    original=xxorig;
    f=0;
    prod=1;
    const=1;
    do j = 1 to &nest;
        prod=prod*b[j];
    end;
    prod=abs(prod);
    do i = 1 to nrow(original);
        pred=0;
        do j = 1 to &nest;
            pred=pred +b[j]*original[i,j];
        end;
        v= ((abs(-const+pred)))**&nest/(prod);
        f=f+v**(2/&nest);
    end;
    f=(f*.5);
    return(f);
finish minlg;
start rma_4var(original) global(xxorig,initial,b);
    original=xxorig;
    optn1= nrow(original)//{0};
    result1=j(&nrep,2*&nest+1,0);
```

```
        seed44=89425;
        do r=1 to &nrep;
                ini=j(1,&nest,0);
                do v = 1 to &nest;
                        ini[v]=initial[v]+2*(ranuni(seed44)-0.5);
                end;
                call nlplm(rc1,xres1,"minls",ini,optn1);
                fopt=minlg(xres1);
                result1[r,1]=fopt;
                result1[r,2:(&nest+1)]=ini;
                result1[r,(&nest+2):(2*&nest+1)]=xres1;
        end;
        minfopt=min(result1[,1]);
        locminfopt=loc(result1[,1]=minfopt)[1];
        iniguess=j(1,&nest,0);
        iniguess=result1[locminfopt,2:(&nest+1)];
        b=j(1,&nest,0);
        b=result1[locminfopt,(&nest+2):(2*&nest+1)];
finish rma_4var;
        title "";
        use &data;
        read all into aa;
        xstart=ncol(aa)-&nest+1;
        xend=ncol(aa);
        x=aa[,xstart:xend];
        xxorig=x;
        original=x;
        run regress(original);
        run rma_4var(original);
        fopt1=minlg(b);
        sumresult=j(1,&nest+1+2,0);
        sumresult[,1]=&nvar;
        sumresult[,2]=&nsamp;
        sumresult[,3:(&nest+2)]=b;
        sumresult[,&nest+1+2]=fopt1;
cnameall2={'nvar','nsamp','est1','est2','est3','est4','est5','est6','est7
','est8','est9','est10','est11','est12','est13','est14','est15','est16','
obj_func'};
cnameselect2=j(%eval(&nest+1+2),1,.);
cnameselect2=cnameall2[1:%eval(&nest+2)]//cnameall2[17+2];
create &out from sumresult [colname=cnameselect2];
append from sumresult;
quit;
%mend RMA_4Var_Jack;
```

Macro_LargestCorr

```
options pageno=1 ls=80 ps=60 nocenter;
data _null_;
        call symput("thedate", put(today(), yymmdd6.));
run;
%macro LargestCorr(step=);
        data corrdata0;
                set super;
        run;
        proc corr data=corrdata0 outp=corrout;
                var x1-x&nvar;
        run;
        data corrdata;
                set corrout;
                if _type_ = "CORR";
                keep x1-x&nvar;
```

```
	run;
	proc iml;
		use corrdata;
		read all into corrdata;
		offdiag=loc(corrdata[]^=1);
		zzz=max(abs(corrdata[offdiag]));
		maxcorrloc=min(loc(corrdata[,]=zzz | corrdata[,]=-1*zzz));
		print maxcorrloc;
		iin=j(1,2,0);
		iin[1]=ceil(maxcorrloc/&nvar);
		iin[2]=maxcorrloc-(iin[1]-1)*&nvar;
		create iin0 from iin;
		append from iin;
	quit;
data iin;
	set iin0;
	if col1>col2 then do;
		rename col1=ind2 col2=ind1;
	end;
	else do;
		rename col1-col2=ind1-ind2;
	end;
run;
proc print data=iin;
	title "the first two variables in the model (with the highest
correlation)";
run;
%VarSelect(step=&step);
data orin0;
	set preorinmat;
	rename col1=nvar col2=nsamp col3-col%eval(2+&nvar)=x1-x&nvar
col%eval(2+&nvar+1)-col%eval(2+&nvar+&step)=ind1-ind&step;
	title "";
run;
%mend LargestCorr;
```

Macro_VarSelect

```
options pageno=1 ls=80 ps=60 nocenter;
data _null_;
    call symput("thedate", put(today(), yymmdd6.));
run;
/*select i1, i2 to be in the model, save i1, i2 to iin -- done in
macro_LargestCorr*/
/*do jj=1 to p;*/
/*do j=1 to pj*/
/*if j not an element of iin then compute rma for iin+j & get residuals
for var xj*/
/*then squared all the residuals to get SSR*/
/*put j and SSR in a matrix*/
/*(j1 SSR1, j2 SSR2,...., jp-2 SSRp-2)*/
/*find the smallest SSRk and create a new vector that has iin and jk in
it*/
/*end j & jj loops*/
%macro VarSelect(step=);
	data preiin;
		set iin;
			do i=1 to &nsamp;
				output;
			end;
		drop i;
	run;
```

```
	data preorin;
		merge super preiin;
	run;
	proc iml;
		use preorin;
		read all into temp;
		preorinmat=temp[,1:2] || j(&nsamp,&nvar,.) ||
temp[,2+&nvar+1:2+&nvar+&step];
		do j=1 to &nsamp;
			do k=1 to &step;
				do i=1 to &nvar;
					if temp[j,k+&nvar+2]=i then
preorinmat[j,i+2]=temp[j,i+2];
				end;
			end;
		end;
		create preorinmat from preorinmat;
		append from preorinmat;
	quit;

%mend VarSelect;
```

Macro_EstOrin2

```
options pageno=1 ls=80 ps=60 nocenter;
data _null_;
	call symput("thedate", put(today(), yymmdd6.));
run;
%macro EstOrin2(step=,nrep=);
data orin;
	title "";
	set orin0;
run;
%RMA_Orin(step=&step,nrep=&nrep);
data sol;
	set sol;
run;
proc sort data=sol out=solout;
	by obj_func;
run;
data olsout "&current.olsout";
	set allols;
run;
data rsqout "&current.rsqout";
	set allrsq;
run;
data solout "&current.solout";
	set solout;
run;
proc print data=olsout;
	title "Canonical Form of all OLS Estimates";
run;
proc print data=rsqout;
	title "Rsq of all OLS Estimates";
run;
proc print data=solout;
	title "RMA coeff estimate for original data";
run;
%mend EstOrin2;
```

Macro_ResidualSq2

```
options pageno=1 ls=80 ps=60 nocenter;
data _null_;
      call symput("thedate", put(today(), yymmdd6.));
run;
%macro ResidualSq2(step=);
      data _null_;
            set super;
            call symputx("nvar",nvar);
            call symputx("nsamp",nsamp);
      run;
      %put &nvar &nsamp;
      data preest;
            set solout;
            do i=1 to &nsamp;
                  output;
            end;
            drop i;
      run;
      data solout00;
            merge orin0 preest;
      run;
      proc iml;
            use solout00;
            read all into m;
            l=j(nrow(m),&nvar,.);
            do i=1 to nrow(m);
                  n=loc(m[i,3:2+&nvar]^=.);
      l[i,n]=m[i,%eval(2+&nvar+&step+1):%eval(2+&nvar+&step+&step)];
            end;
            solout=m[,1:2+&nvar] || l ||
m[,%eval(2+&nvar+&step+&step+1)] ||
m[,%eval(2+&nvar+1):%eval(2+&nvar+&step)];
            create soloutall from solout;
            append from solout;
      quit;
      data soloutall;
            set soloutall;
            rename col1=nvar col2=nsamp
                        col3-col%eval(2+&nvar)=x1-x&nvar
                        col%eval(2+&nvar+1)-
col%eval(2+&nvar+&nvar)=est1-est&nvar
                        col%eval(2+&nvar+&nvar+1)=obj_func
                        col%eval(2+&nvar+&nvar+2)-
col%eval(2+&nvar+&nvar+2+&step)=ind1-ind&step;
      run;
      data resid_all;
            set soloutall;
            sum=0;
            %do i=1 %to &nvar;
                  sum+est&i*x&i;
            %end;
            %do i=1 %to &nvar;
                  xhat&i=(1-sum+est&i*x&i)/est&i;
                  e&i=x&i-xhat&i;
            %end;
            %do i=1 %to &nvar;
                  esq&i=e&i**2;
            %end;
            keep ind1-ind&nvar x1-x&nvar est1-est&nvar e1-e&nvar esq1-
esq&nvar obj_func;
      run;
      %do i=1 %to &nvar;
```

```
              data residsq_all&i;
                     set resid_all;
                     by ind1;
                     if first.ind1 then sum&i=.;
                            sum&i+esq&i;
                     if last.ind1 then output;
                     keep sum&i ind1-ind&nvar obj_func;
              run;
       %end;
       data residsq_all;
              merge residsq_all1-residsq_all&nvar;
       run;
       proc print data=residsq_all;
              title "Sum of ResidualSq of Each Coeff Estimates";
       run;
       data forplot&step "&current.forplot&step";
              set residsq_all;
       run;
%mend ResidualSq2;
```

Macro_EstOrin

```
options pageno=1 ls=80 ps=60 nocenter;
data _null_;
       call symput("thedate", put(today(), yymmdd6.));
run;
/*perform rma on original data*/;
%macro EstOrin(step=,nrep=);
data orin;
       title "";
       set orin0&r;
run;
%RMA_Orin(step=&step,nrep=&nrep);
data sol;
       set sol;
run;
proc sort data=sol out=solout;
       by obj_func;
run;
data olsout "&current.olsout";
       set allols;
run;
data rsqout "&current.rsqout";
       set allrsq;
run;
data solout "&current.solout";
       set solout;
run;
proc print data=olsout;
       title "Canonical Form of all OLS Estimates";
run;
proc print data=rsqout;
       title "Rsq of all OLS Estimates";
run;
proc print data=solout;
       title "RMA coeff estimate for original data";
run;
%mend EstOrin;
```

Macro_ResidualSq

```
options pageno=1 ls=80 ps=60 nocenter;
```

```
data _null_;
	call symput("thedate", put(today(), yymmdd6.));
run;
%macro ResidualSq(step=);
	data preest;
		set solout;
		do i=1 to &nsamp;
			output;
		end;
		drop i;
	run;
	data solout00;
		merge orin0&r preest;
	run;
	proc iml;
		use solout00;
		read all into m;
		l=j(nrow(m),&nvar,.);
		do i=1 to nrow(m);
			n=loc(m[i,3:2+&nvar]^=.);
	l[i,n]=m[i,%eval(2+&nvar+&step+1):%eval(2+&nvar+&step+&step)];
		end;
		solout=m[,1:2+&nvar] || l ||
m[,%eval(2+&nvar+&step+&step+1)] ||
m[,%eval(2+&nvar+1):%eval(2+&nvar+&step)];
		create soloutall from solout;
		append from solout;
	quit;
	data soloutall;
		set soloutall;
		rename col1=nvar col2=nsamp
				col3-col%eval(2+&nvar)=x1-x&nvar
				col%eval(2+&nvar+1)-
col%eval(2+&nvar+&nvar)=est1-est&nvar
				col%eval(2+&nvar+&nvar+1)=obj_func
				col%eval(2+&nvar+&nvar+2)-
col%eval(2+&nvar+&nvar+2+&step)=ind1-ind&step;
	run;
	data resid_all;
		set soloutall;
		sum=0;
		%do i=1 %to &nvar;
			sum+est&i*x&i;
		%end;
		%do i=1 %to &nvar;
			xhat&i=(1-sum+est&i*x&i)/est&i;
			e&i=x&i-xhat&i;
		%end;
		%do i=1 %to &nvar;
			esq&i=e&i**2;
		%end;
		keep ind1-ind&nvar x1-x&nvar est1-est&nvar e1-e&nvar esq1-
esq&nvar obj_func;
	run;
	%do i=1 %to &nvar;
		data residsq_all&i;
			set resid_all;
			by ind1;
			if first.ind1 then sum&i=.;
				sum&i+esq&i;
			if last.ind1 then output;
			keep sum&i ind1-ind&nvar obj_func;
```

```
		run;
	%end;
	data residsq_allout&r;
		merge residsq_all1-residsq_all&nvar;
	run;
	proc print data=residsq_allout&r;
		title "Sum of ResidualSq of Each Coeff Estimates";
	run;
%mend ResidualSq;
```

Macro_Indicator

```
options pageno=1 ls=80 ps=60 nocenter;
data _null_;
    call symput("thedate", put(today(), yymmdd6.));
run;
%macro Indicator(step=,nrep=);
%let steplast=%eval(&step-1);
%let stepone=1;
%let indlast=%eval(&nvar-&step+1);
%let stepminusone=%eval(&indlast-&stepone);
proc iml;
	use iin;
	read all into a;
	iinmat=a[,1:&steplast];
	b=do(1,&nvar,1);
	iintemp=remove(b,iinmat);
	create iinmat from iintemp;
	append from iintemp;
quit;
%if &step=&nvar %then %do;
	data iinmat;
		set iinmat;
		rename col1=ind&nvar;
	run;
	data iinmatiin;
		merge iin iinmat;
	run;
	data iin;
		set iinmatiin;
		keep ind1-ind&nvar;
	run;
	%VarSelect(step=&step);
	%let paraend=%eval(2+&nvar);
	%let indstart=%eval(2+&nvar+1);
	%let indend=%eval(2+&nvar+&step);
	data orin0;
		set preorinmat;
		rename col1=nvar col2=nsamp col3-col&paraend=x1-x&nvar
col&indstart-col&indend=ind1-ind&step;
		title "";
	run;
	%EstOrin2(step=&step,nrep=&nrep);
	%ResidualSq2(step=&step);
	data tomatch0;
		set residsq_all;
		rename sum1-sum&nvar=ssum1-ssum&nvar;
	run;
	data tomatch1;
		set forplot%eval(&nvar-1);
		rename sum1-sum&nvar=suum1-suum&nvar obj_func=obj_func0;
	run;
```

```
	data tomatch2;
		merge tomatch0 tomatch1;
	run;
	data tomatch3;
		set tomatch2;
		%do i=1 %to &nvar;
			if suum&i=. then sum&i=.;
		    if suum&i ne . then sum&i=ssum&i;
		%end;
		keep ind1-ind&nvar sum1-sum&nvar obj_func;
	run;
	data forplot&step "&current.forplot&step";
		set tomatch3;
	run;
%end;
%if &step<&nvar %then %do;
	data iinmat;
		set iinmat;
		rename col&stepone-col&indlast=ind&step&&stepone-
ind&step&&indlast;
	run;
	data iinmatiin;
		merge iin iinmat;
	run;
	%do r=1 %to &indlast;
		data iin;
			set iinmatiin;
			keep ind1-ind&steplast ind&step&&r;
			rename ind&step&&r=ind&step;
		run;
		%VarSelect(step=&step);
		%let paraend=%eval(2+&nvar);
		%let indstart=%eval(2+&nvar+1);
		%let indend=%eval(2+&nvar+&step);
		data orin0&r;
			set preorinmat;
			rename col1=nvar col2=nsamp col3-col&paraend=x1-
x&nvar col&indstart-col&indend=ind1-ind&step;
			title "";
		run;
		%EstOrin(step=&step,nrep=&nrep);
		%ResidualSq(step=&step);
		%let one=1;
		data residsq_allout0&r;
			set residsq_allout&r;
			title "";
			rename ind1-ind&step=ind&r&&one-ind&r&&step sum1-
sum&nvar=sum&r&&one-sum&r&&nvar obj_func=obj_func&r;
		run;
	%end;
	data try;
		merge residsq_allout01-residsq_allout0%eval(&nvar-&step+1);
	run;
	data residsq_allout;
		set try;
		%do i=2 %to %eval(&nvar-&step+1);
			%do j=1 %to &nvar;
			if ind11=ind&i&&one & ind11=&j then do;
				if sum1&j <= sum&i&&j then do;
					%do k=1 %to &step;
						ind&k=ind&one&&k;
						obj_func=obj_func&one;
```

```
                        %end;
                    end;
                    else do;
                        %do k=1 %to &step;
                            ind&k=ind&i&&k;
                            obj_func=obj_func&i;
                        %end;
                    end;
                    sum&j=min(sum1&j,sum&i&&j);
                end;
                %end;
            %end;
            keep ind1-ind&step sum1-sum&nvar obj_func;
        run;
        proc print data=residsq_allout;
            title "Select Variables ind1-ind&step at Step &step";
        run;
        data iin;
            set residsq_allout;
            keep ind1-ind&step;
            title "";
        run;
        data iin;
            set iin;
        run;
        proc print;
        run;
        data forplot&step "&current.forplot&step";
            set residsq_allout;
        run;
%end;
%mend Indicator;
```

Macro_PlotSum

```
options pageno=1 ls=80 ps=60 nocenter;
data _null_;
      call symput("thedate", put(today(), yymmdd6.));
run;
%macro PlotSum();
%do i=2 %to &nvar;
      data forplotstep&i;
            set "&current.forplot&i";
            if sum1 ne . then sum1&i=sum1;
            %do j=2 %to &nvar;
                  else if sum&j ne . then sum&j&&i=sum&j;
            %end;
            rename obj_func=obj_func&i;
      run;
%end;
data forplot0_1;
      merge forplotstep2-forplotstep&nvar;
      keep sum12-sum&nvar&&nvar;
run;
data forplot0_2;
      merge forplotstep2-forplotstep&nvar;
      keep obj_func2-obj_func&nvar;
run;
proc iml;
      use forplot0_1;
      read all into pp1;
      pplot1=t(pp1);
```

```
	tt=j(&nvar-1,ncol(pplot1),.);
	do i=1 to ncol(pplot1);
		nmloc=loc(pplot1[,i]);
		tt[,i]=pplot1[,i][nmloc];
	end;
	create forplot00 from tt;
	append from tt;
	use forplot0_2;
	read all into pp2;
	pplot2=t(pp2);
	create forplotobj from pplot2;
	append from pplot2;
quit;
data forplotobj;
	set forplotobj;
	rename col1=ObjFunc;
run;
data forplot;
	set forplot00;
	rename col1=SumResidSq;
run;
data forplotiin0;
	set forplot&nvar;
	ind12=cats(of ind1 ind2)*1.0;
	keep ind12 ind3-ind&nvar;
run;
proc iml;
	use forplotiin0;
	read all into ii;
	pii=ii;
	pii=ii[,ncol(ii)] || ii[,1:(ncol(ii)-1)];
	tpii=t(pii);
	create forplotiin00 from tpii;
	append from tpii;
quit;
data forplotiin;
	set forplotiin00;
	rename col1=indsim;
run;
data forplotsimiin;
	merge forplot forplotiin forplotobj;
	NumVar=_n_+1;
run;
data plotdata;
	set forplotsimiin;
	keep SumResidSq indsim ObjFunc NumVar;
run;
goptions device=gif ftext='arial' htext=16pt gunit=in hsize=10 vsize=8;
options orientation=landscape;
options nodate nonumber;
ods pdf file="&current.&thedate PlotSum_ResidSq.pdf" style=sansprinter;
ods noresults;
ods proclabel='gplot';
symbol1 value=plus height=16pt pointlabel=(justify=right position=top
'#indsim:#SumResidSq');
proc gplot data=plotdata;
	title font='arial' height=16pt "Plot Sum of ResidSq vs Number of
Variables Selected";
	format SumResidSq 8.2;
	plot SumResidSq*NumVar=indsim;
run;
ods pdf close;
```

```
goptions device=gif ftext='arial' htext=16pt gunit=in hsize=10 vsize=8;
options orientation=landscape;
options nodate nonumber;
ods pdf file="&current.&thedate PlotSum_ObjFunc.pdf" style=sansprinter;
ods noresults;
ods proclabel='gplot';
symbol1 value=plus height=16pt pointlabel=(justify=right position=top
'#indsim:#ObjFunc');
proc gplot data=plotdata;
       title font='arial' height=16pt "Plot Sum of ResidSq vs Number of
Variables Selected";
       format ObjFunc 8.2;
       plot ObjFunc*NumVar=indsim;
run;
ods pdf close;
%mend PlotSum;
```