A SIMULATION EVALUATION OF BACKWARD ELIMINATION AND STEPWISE VARIABLE SELECTION IN REGRESSION ANALYSIS

by

XIN LI

B.A., Shandong Polytechnic University, China, 2004

A REPORT

Submitted in partial fulfillment of the requirements for the degree

MASTER OF SCIENCE

Department of Statistics
College of Arts and Sciences

KANSAS STATE UNIVERSITY
Manhattan, Kansas

2012

Approved by:

Major Professor
Paul Nelson

Copyright

Abstract

A first step in model building in regression analysis often consists of selecting a parsimonious set of independent variables from a pool of candidate independent variables. This report uses simulation to study and compare the performance of two widely used sequential, variable selection algorithms, stepwise and backward elimination. A score is developed to assess the ability of any variable selection method to terminate with the correct model. It is found that backward elimination performs slightly better than stepwise, increasing sample size leads to a relatively small improvement in both methods and that the magnitude of the variance of the error term is the major factor determining the performance of both.

Table of Contents

List of Figures v
List of Tables vi
Acknowledgements vii
Dedication viii
Chapter 1 - Introduction 1
1.1 Model building 1
1.2 Model Building in Regression 2
1.3 Research Plan 7
1.4 Algorithm Used in Simulation Study 8
Chapter 2 - Simulation Study 15
2.1 Simulation Study Frame 15
2.2 Analysis of the Simulation 19
2.2.1 Box Plots 25
2.2.2 Lsmeans and/or Means analysis 30
2.2.3 Regression Analysis of Backward Elimination Score 33
2.3 A Comparison of Backward and Stepwise Variable Selection 37
2.3.1 Boxplot of Diff 38
2.3.2 Lsmeans and/or Means Analysis of Diff 41
2.3.3 Regression Analysis of Diff 44
Chapter 3 - Conclusion and Further Study 47
3.1 Conclusion 47
3.2 Limitations and Further Study 51
Bibliography 52

List of Figures

Figure 2.1: Histogram of Studentized Residuals 22
Figure 2.2: Normal Probability Plot 23
Figure 2.3: Residual Plot 24
Figure 2.4: Box Plot of Back_Score by Error Variance 25
Figure 2.5: Box Plot of Back_Score and Step_Score by Sample Size 27
Figure 2.6: Box plot of Back_score by Extra predictor 28
Figure 2.7: Plot of SSize by Evari Interaction of Lsmean Against Error Variance 31
Figure 2.8: Plot of Sample Size by Evari Interaction of Lsmean Against Sample Size ... 32
Figure 2.9: Three Dimensional Scatter Plot of SSize by EVariance Interaction 33
Figure 2.10: Box Plot of Score Difference by Sample Size 38
Figure 2.11: Box Plot of Score Difference by Error Variance 39
Figure 2.12: Box Plot of Score Difference by Extra Predictor 40
Figure 2.13: Studentized Residuals Plot of Score Difference 44
Figure 2.14: Three Dimensional Scatter Plot of SSize by EVariance for Diff 45

List of Tables

Table 1.1: Data Example 10
Table 1.2: The Partial F-statistics to Illustrate Backward Elimination 11
Table 1.3: The Partial F-statistics to Illustrate Backward Elimination 12
Table 1.4: The Partial F-statistics to Illustrate Backward Elimination 12
Table 1.5: The Partial F-statistics to Illustrate Backward Elimination 13
Table 2.1: Parameter Settings 16
Table 2.2: Partial Results of Means of 30 Responses Replicated 1000 Times 17
Table 2.3: Summary of R-Square, RMSE and CV for Different Models 20
Table 2.4: Analysis of Variance Main Effects Mode Based on Backward Score 21
Table 2.5: Tests for Normality 21
Table 2.6: LSMEANS of Backward Scores and Pairwise Comparisons of SSizes 30
Table 2.7: Summary of Impact of SSize, EVariance and EPredictor 32
Table 2.8: Parameter Estimates of Regression Analysis of Backward Score 34
Table 2.9: Parameter Estimates of Regression Analysis on Backward Score 35
Table 2.10: Lsmean Marginal Means Diff of Sample Size 41
Table 2.11: Summary of Pairwise Comparison Lsmean Means Diff of SSize 41
Table 2.12: Lsmean Marginal Means Diff of Error Variance 42
Table 2.13: Summary of Pairwise Comparison LSMEANS of Diff of SSize 43
Table 2.14: LSMEAN Marginal Means Diff of Type of Extra Predictor 43
Table 2.15: Parameter Estimates of Regression Analysis for Main Effects 46
Table 3.1: Summary of Means for 40 combinations of treatment effects 48
Information for Table 3.2: General information 66

Acknowledgements

I want to give my heartfelt thanks to my major professor, Dr. Paul Nelson, who lead me go through the entire process. His working attitude will inspire me in my entire life. I want to express my sincerely appreciate to my committee member Dr. Weixing Song, without him, I may don't have chance to get into the fabulous statistical world. I want to give special thanks my committee member Dr. Juan Du, who supports me from multiple aspects.

I would like to extend my gratitude to my departmentmates, they make my life different.

Finally, thank you Mom and Daddy! Thank you parents in law! Thank you Pei!

Dedication

I dedicate this to My wife Pei Liu!

Chapter 1 - Introduction

1.1 Model building

Constructing a tractable statistical model that is a good approximation to the one that generated a sample is often the first step toward making valid inferences. This report concerns model building in a regression setting, where the goal is to construct a parsimonious model consisting of a few 'good' explanatory variables that fit the data well. Parsimony can make data analysis more easily understood and interpretable and avoid multicollinearity among the explanatory variables.

Starting with a hypothesized full model consisting of explanatory variables X_1, X_2, …,X_p, model building in regression uses the available data to find a hopefully small subset of these that work just as well as all of them. Model building algorithms that require examining all $2^p - 1$ possible models have been, until the modern era, impractical because of the large amount of computer time and storage space required. The advent of powerful desktop computing has already overcome some of these time and space limitations and computer intensive methods will play an even bigger role as this technology advances. The model building procedures investigated in this report are based on sequential algorithms that typically terminate much more quickly than procedures based on examining all possible regressions. Backward elimination and stepwise variable selection, the methods studied here, are two such sequential procedures. They are based on partial F-tests whose values can be expressed in terms of coefficients of determination R^2, which sequentially seeks to accomplish this goal without having to examine all $2^p - 1$ sub-models, a potentially big saving in time and effort.

1.2 Model Building in Regression

Regression analysis relates the mean value of a response Y to k-explanatory variables, whose values $\mathbf{w} = \{w_i\}$ in this report are fixed or conditionally considered fixed. The available data consists of the vectors $\{(y_i, \mathbf{w}_i = (w_{i1}, w_{i2}, ..., w_{ik});\ i = 1, 2, ..., n\}$ obtained from a random sample of experimental units. The observed responses $\{y_i\}$ are taken to be realizations of independent random variables $\{Y_i\}$. We assume that there is a function $\psi(\mathbf{w})$ such that $Y_i = \psi(\mathbf{w}_i) + \eta_i$ with $E(\eta_i) = 0$ and $Var(Y_i)$ a constant, $i = 1, 2, ..., n$. These are very strong assumptions which implicitly exclude the possibility that there are other important explanatory variables which cannot be expressed as functions of $\mathbf{w}$. The goal of model building in this context can be expressed as constructing from the data a parsimonious approximation of $\psi(\mathbf{w})$.

Backward elimination implements this process of approximation by assuming the existence of a provisional full rank model of the form:

$$Y_i = \beta_0 + \sum_{j=1}^{p} \beta_j x_{ij} + \varsigma_i \qquad (1)$$

where for functions

$\{\psi_j(\mathbf{w})\}$, $x_j = \psi_j(\mathbf{w})$, $j = 1, 2, ..., p \geq k$; $x_{ij} = \psi_j(\mathbf{w}_i)$, $i = 1,2,\ldots,n$, n > p+1,

$$E(\varsigma_i) = 0,\ Var(\varsigma_i) \text{ is a constant} \qquad (2)$$

and $\{\varsigma_i\}$ are normally distributed. All of our models will contain a constant term and we assume that the rank of the design matrix in (1) is $p + 1$.

In Backward Elimination, described in detail below, if $\beta_j = 0$, explanatory variable 'x_j' can be dropped from (1). Sequentially deleting all of those $\{x_j\}$ for which $\beta_j = 0$ hopefully results in a parsimonious, *just as good, model* of the form

$$Y = \gamma_0 + \sum_{j=1}^{m} \gamma_j z_j + \varsigma \qquad (3)$$

with $m < p$, having more degrees of freedom with which to estimate experimental error, where $\{z_j\}$ is a subset of $\{x_j\}$. Forward Selection, on which the stepwise algorithm is based, on the other hand, starts with $Y = \gamma_0 + \varsigma$ and sequentially adds variables.

Backward elimination operates by sequentially testing hypotheses of the form

$$H_0 : \gamma_j = 0 \text{ vs } H_1 : \gamma_j \neq 0 \qquad (4)$$

as follows. Starting with (1), suppose that the process of variable deletion has so far resulted in (3) with $1 < m \leq p$. Carry out the tests in (4) by rejecting H_0 at one at a time type I error rate α if

$$F_j = [SSE(z_i; i = 1, 2, ..., m, i \neq j) - SSE(z_i; i = 1, 2, ..., m)] / MSE(z_i; i = 1, 2, ..., m)$$

$$\geq F_\alpha(1, n - m - 1) \ .$$

Then, delete z_l if $F_l = \min\{F_j; j = 1, 2, ..., m\} < F_\alpha(1, n - m - 1)$.

Forward variable selection starts with the most parsimonious model,

$$Y = \gamma_0 + \varsigma \qquad (1)$$

and sequentially adds variables, hopefully results in a model of the form

$$Y = \gamma_0 + \sum_{j=1}^{m} \gamma_j z_j + \varsigma \qquad (2)$$

where $\{z_j\}$ is a subset of $\{x_j\}$, and then by testing the hypotheses of the form

$$H_0 : \gamma_j = 0 \text{ vs } H_1 : \gamma_j \neq 0 \qquad (3)$$

Carry out the test in (3) by rejecting H_0 at type I error rate α if

$$F_j = [SSE(z_i; i = 1, 2, ..., m, i \neq j) - SSE(z_i; i = 1, 2, ..., m)] / MSE(z_i; i = 1, 2, ..., m)$$

$$\geq F_\alpha(1, n - m - 1) \ .$$

Then, exclude z_l if $F_l = \min\{F_j; j = 1, 2, ..., m\} < F_\alpha(1, n - m - 1)$.

Continue on this process until all potential variables are tested. In the forward variable selection procedure, variable only allow to enter, once the variable included will not be discard from any following steps

Stepwise selection is an enhanced version of Forward Selection. It allows variables that entered in previous stages to be deleted at later ones. Stepwise selection uses a variable filtering procedure that at each stage allows the deletion of the variable whose p-value for testing that its coefficient is zero is smallest and falls below a pre-specified level. The algorithm then proceeds to the next step, which ensures that all variables in the model are significant at the given level significance. Finally, the stepwise selection algorithm terminates when none of the variables outside the model qualify for admission and none of the variables in the model qualify for deletion. Under some special situations, stepwise selection becomes trapped in an Infinite loop. By default, SAS will terminate the procedure at the end of the second cycle.

Other widely used methods of model building in regression analysis are: Mallow's Cp Criterion, the Largest Coefficient of Determination R^2 and Adjusted R^2, PRESS (PREdiction Sums of Squares) Criterion.

Mallow's C_r Statistic for assessing the fit of a sub-model (reduced) relative to the full model in (1) is given by

$$C_r = (SSE_R/MSE_F) - (n - 2(r+1)),$$

where,

SSE_R = the sum of squared errors for the reduced model,

MSE_R = the mean of squared errors for the full model,

n = the number of observations,

$p+1$ = the rank of the design matrix of (1),

$r+1$ = the rank of the design matrix of the reduced model,

Selection Criterion: Models with *small r* that are close to 1 are considered to be good.

The coefficient of Determination R^2 is the proportion of variability in a data that is accounted for by the model. In some cases, the coefficient of determination is a useful statistic that can be used as a tool to assess the goodness of fit of a model. There are several widely used formulas for the coefficient of determination, such as

$$R^2 = 1 - SS_{err} / SS_{tot,}$$

where, $SS_{err} = \sum (y_i - \hat{y}_i)^2$, the sum of squares of residuals and

$SS_{tot} = \sum (\hat{y}_i - \bar{y})^2$, the total sum of squares

Adjusted R^2 is a modification on the Coefficient of Determination R^2 that creates a penalty for including independent variables that have little explanatory value and it can be used to compare two models. The definition of adjusted R^2 is given by :

$$\text{Adjusted } R^2 = 1 - (SS_{err} / SS_{tot})(df_t / df_e),$$

where,

df_t = the degrees of freedom of the estimate of the population variance for the dependent variable, equal to n – 1.

df_e = the degrees of freedom of the estimate of the underlying population error variance, equal to n – p – 1.

In comparing two models based on Adjusted R^2 or R^2 the model with the larger value is considered to be the better one.

PRESS (PREdiction Sums of Squares) is another statistic that assesses the fit of a model using a form of cross validation. Observations are sequentially deleted and the model refit using the remaining n-1 observations. Let $\hat{y}_{i,-i}$ be the predicted value of y_i obtained by fitting the model with the i^{th} data line deleted. The PRESS statistic is then given by

$$\text{PRESS} = \sum_{i=1}^{n} (y_i - \hat{y}_{i,-i})^2.$$

In comparing two models, the model with the smaller Press is considered to be the better one.

1.3 Research Plan

Although widely used, I have not found any comprehensive studies of how well any of these variable selection procedures perform or even a rough definition of what *performing well* means. The only tangentially relevant article I found is Austin and TU (2004). However, they simply summarized the performance of three variable selection methods, rather than evaluating the performance of any one of them. My report will attempt to fill this gap by using simulation to evaluate and compare the performances of Backward and Stepwise variable selection. I will define and use a score function to assess how well a model building algorithm performs in terms of its ability to produce the model that actually generated the data.

1.4 Algorithm Used in Simulation Study

For a range of representative settings:

$\{\mathbf{w}_i = (w_{i1}, w_{i2}, ..., w_{ik});\ i = 1, 2, ..., n\}$, k, $Var(\varsigma) \equiv \sigma^2$ and n,

(I) Generate $\{\beta_j^{\#}\}$ as independent random variables uniformly distributed on (0, 1).

(II) Generate $\{\mathbf{w}_i\}$ as independent random variables uniformly distributed on (0, 1) and independent of the error terms $\{\varsigma_i\}$, which are generated as independent normal random variables, each with mean zero and variance $\equiv \sigma^2$. Form the responses

$$Y_i = \beta_0^{\#} + \sum_{j=1}^{k} \beta_j^{\#} w_{ij} + \varsigma_i, \quad \text{i = 1, 2, …, n.} \qquad (4)$$

(II) Carry out backward elimination and stepwise selection using the model in (1) where $\{x_j\}$ consist of two types of *extra* predictors constructed to augment $\mathbf{w}$..

Related: $\{x_j\}$ are obtained by forming a full second order model from $\mathbf{w}$. For k = 3 this becomes:

$$x_1 = w_1, x_2 = w_2, x_3 = w_3, x_4 = w_1 w_2, x_5 = w_1 w_3, x_6 = w_2 w_3, x_7 = x_1^2, x_8 = x_2^2, x_9 = x_3^2.$$

Unrelated: $\{x_j\}$ are obtained by forming a full model from $\mathbf{w}$ by adding variables that are not functions of $\mathbf{w}$. Specifically, for k = 3, take:

$x_1 = w_1, x_2 = w_2, x_3 = w_3$ and $\{x_4, x_5, x_6, x_7, x_8, x_9\}$ independent of $\mathbf{w}$

(III) Assess how well backward elimination and stepwise selection perform in terms of resulting in the 'correct' model having the form given by (4). To accomplish this, I assign a 'score' to each data set, defined as follows. Start with Score = 0 and let

λ = number of $\{x_j;\ j = 1, 2, ..., k\}$ not included.

δ = number of 'extra' predictors included,

$$Score = (2\lambda + \delta)/k.$$

Note that '*Score* = 0' is the best outcome and the bigger score gets, the poorer variable selection performs. The term 2λ is a penalty for not selecting 2λ of the true predictors. Other functions of λ could, of course, be used. My definition of score is to be interpreted as a tentative first step in devising ways to numerically quantify the performance of variable selection procedures.

(IV) Independently repeat (I)-(III) N (=1000) times and let *Score(i)* be the score recorded for the ith data set, resulting in {Score(i), = 1,2,…, *N*}.

Note, to distinguish between the variable selection procedures, I use Score_b to denote the score for backward elimination, and use Score_s to denote the score for stepwise selection.

(V) Summarize {Score (i), = 1, 2, …, N} using quantiles, mean and standard deviation.

(VI) Repeat (I)-(V) for a variety of settings and summarize the results.

(VII) Compare the results for backward elimination and Stepwise selection, and then make conclusions.

I begin with an example to illustrate Backward Elimination based on simulated data. I carried out steps I-VI with *N* =1 data set, n = 30 observations, k =3, $\alpha = 0.05$, σ =0.1 and type = "related". Backward elimination was implemented using SAS. The steps the algorithm went through and final results are presented below.

Table 1.1: Data Example of 3 True Predictors (x1, x2, x3) and Response Variable (y)

Obs	x1	x2	x3	y
1	0.19352	0.66566	0.75740	1.33039
2	0.57989	0.26231	0.63654	1.25511
3	0.97856	0.22937	0.73506	1.57215
4	0.24490	0.79618	0.77608	1.43113
5	0.16184	0.50126	0.94163	1.24434
… … … …				
29	0.01581	0.28416	0.42685	0.52080
30	0.69745	0.99144	0.96946	2.03695

Throughout the variable selection process, vi (where i=1,2,3) represent the true predictor that was used to generate the data. While vij (where i=1,2,3; j=1,2,3) represent the quadratic and pairwise cross products of the true predictors, and they are treated as disturbing factors.

Step 1: All Variables Entered

The partial F-statistics for each of the explanatory variables are displayed in **Table 1.2** below. The explanatory variable "v11" has the smallest partial F-statistic (0.06) Hence, v11" and the backward elimination algorithm then investigates whether v11 should be removed from the full model or not. Since the p-value for the variable "v1" is 0.8151, we do not reject the null hypothesis H0: Beta (v11) = 0 at significance level 0.05 and decide to remove the variable "v11" from the full model.

Table 1.2: The Partial F-statistics to Illustrate Backward Elimination

Variable	Parameter Estimate	Standard Error	Type II SS	F Value	Pr > F
Intercept	0.00779	0.16298	0.00002516	0.00	0.9623
v1	0.70702	0.29875	0.06163	5.60	0.0282
v2	0.16498	0.38819	0.00199	0.18	0.6754
v3	0.23336	0.50837	0.00232	0.21	0.6512
v12	-0.09596	0.25128	0.00160	0.15	0.7066
v13	-0.40346	0.30744	0.01895	1.72	0.2043
v23	0.26517	0.33644	0.00684	0.62	0.4398
v11	-0.06444	0.27192	0.00061791	0.06	0.8151
v22	0.21089	0.34794	0.00404	0.37	0.5513
v33	0.32363	0.42396	0.00641	0.58	0.4542

Step 2:

Calculate the partial F-statistic for each of the remaining explanatory variables is displayed in the following table. The explanatory variable “v3” has the smallest partial F-statistic (0.19). Hence, the backward elimination algorithm selects the variable “v3” and investigates whether it should be removed from the full model or not. Since the p-value for the variable “v1” is 0.6667, we do not reject the null hypothesis H0: Beta (v3) = 0 and decide to remove the variable “v3” from the full model at the pre-specified significance level 0.05

Table 1.3: The Partial F-statistics to Illustrate Backward Elimination

Variable	Parameter Estimate	Standard Error	Type II SS	F Value	Pr > F
Intercept	0.01434	0.15697	0.00008773	0.01	0.9281
v1	0.65384	0.19271	0.12097	11.51	0.0027
v2	0.17269	0.37804	0.00219	0.21	0.6525
v3	0.21424	0.49052	0.00200	0.19	0.6667
v12	-0.10798	0.24051	0.00212	0.20	0.6581
v13	-0.40759	0.29997	0.01940	1.85	0.1886
v23	0.25520	0.32622	0.00643	0.61	0.4428
v22	0.21687	0.33913	0.00430	0.41	0.5294
v33	0.34980	0.40002	0.00804	0.76	0.3918

Continuing on this process until all variables left in the model are significant at the specified significance level 0.05, we conclude that all of the explanatory variables are significant at the significance level 5%, as displayed in the following table:

Table 1.4: The Partial F-statistics to Illustrate Backward Elimination

Variable	Parameter Estimate	Standard Error	Type II SS	F Value	Pr > F
Intercept	0.14714	0.04646	0.10179	10.03	0.0039

Variable	Parameter Estimate	Standard Error	Type II SS	F Value	Pr > F
v1	0.38645	0.06398	0.37038	36.49	<.0001
v22	0.44844	0.06344	0.50728	49.97	<.0001
v33	0.48947	0.07413	0.44254	43.59	<.0001

Last Step:

Summary: Backward Elimination with Significance Level 5%.

Table 1.5: The Partial F-statistics to Illustrate Backward Elimination

Step	Variable Removed	Number Vars In	Partial R-Square	Model R-Square	C(p)	F Value	Pr > F
1	v11	8	0.0003	0.8754	8.0562	0.06	0.8151
2	v3	7	0.0011	0.8743	6.2383	0.19	0.6667
3	v2	6	0.0005	0.8738	4.3157	0.08	0.7745
4	v12	5	0.0025	0.8713	2.7157	0.45	0.5077
5	v23	4	0.0119	0.8594	2.6302	2.22	0.1494
6	v13	3	0.0084	0.8510	1.9872	1.50	0.2322

The final model contains only one of the three covariates used to generate the data and two extra predictors. The final numerical assessments are score_back= 2, and Cp = 0.2716. The SAS output of Stepwise Selection is presented in Appendix B Where we end

up with variables v23 and v12, with score_step = 2.67 and Cp = 6.1499. In terms of 'score' backward elimination performed better than stepwise in this example.

Chapter 2 - Simulation Study

In this chapter, I describe my simulation experiments and summarize the results by using tables and figures. SAS was used to carry out the simulations and to evaluate and compare the performance of backward elimination and stepwise selection. I only considered the case where $k = 3$ in (4)

2.1 Simulation Study Frame

My simulation experiment was carried out using the algorithm in (1.4) as a fully crossed, three factor balanced design and analyzed as though the treatments and replications were completely randomized. Randomization is a core principle in statistics, and it is involved in most of good experimental designs. To achieve the randomization in my study, I tried to create everything randomly, from seed selecting to data generating. Also, I randomly picked computers to run the simulation. I used 32 computers simultaneously and took 24 hours to conduct the simulation. The factors and levels used are as follows.

Four levels of sample sizes, five levels of error variance and two types of extra predictors related and unrelated.

Table 2.1: Parameter Settings

Sample Sizes					
		Small	Medium	Large	Very large
n =		15	30	70	150
Error Variance					
	Very small	Small	Medium	Large	Very large
Sigma	0.01	0.1	0.3	0.5	1
Extra Predictors					
Related			Unrelated		
Quadratic and pairwise cross products of x1, x2, x3			Independent of x1, x2, x3		

Since *score* is highly discrete and not normally distributed, I based my analysis on 1000 means of 30 replications of each of my 40 treatment combinations. Thus, each data point used in my analysis is a mean of 30 independent *scores*, yielding responses that, according to the Central Limit Theorem, are expected to be closer to being normally distributed than a response consisting of only one of these values. Due to a severe time constraint, I did not investigate the use of C_p in model building and assessment. Sample output of such means is given, in part, in Table 2.2.

Table 2.2: Partial Results of Means of 30 Responses Replicated 1000 Times

n=150, Sigma=0.01 and Extra Predictor = "Related"

Iteration number	Score_back	Score_step	R2	Cp	cp/p	True_left	Extraurb_left
1	0.433333	0.377778	0.9994	5.1207	3.03803	2.766667	0.833333
2	0.077778	0.077778	0.9985	3.2328	2.682291	3	0.233333
3	0.244444	0.1	0.9976	3.9648	2.787703	3	0.733333
4	0.233333	0.133333	0.9993	3.867	2.749465	3	0.7
5	0.2	0.2	0.9993	3.5434	2.660173	3	0.6
6	0.166667	0.122222	0.9994	4.8952	4.045753	3	0.5
7	0.255556	0.155556	0.9989	5.3452	3.975655	3	0.766667
8	0.144444	0.133333	0.9991	4.0484	3.36383	3	0.433333
9	0.188889	0.266667	0.9996	4.306	2.998899	3	0.566667
10	0.233333	0.255556	0.9996	4.2044	2.800476	3	0.7
… …							
997	0.133333	0.144444	0.9992	2.871	1.999616	3	0.4
998	0.155556	0.133333	0.9976	2.9554	2.245762	3	0.466667
999	0.155556	0.211111	0.9994	2.8806	2.110408	3	0.466667
1000	0.4	0.155556	0.9995	5.0876	3.220735	2.966667	1.133333

Notation and terms using in the above table:

Score_back: 'scores'calculated out by using *Score* = $(2\lambda+\delta)$/k, for backward elimination.

Score_step: 'score'calculated out by using *Score* = $(2\lambda+\delta)$/k, for stepwise selection.

R^2 : Coefficient of determination of the final model produced using backward elimination.

Cp: Mallow's Cp given by final model which selected by using backward elimination.

True _left: predictors left in final model which using to generate the response variable y.

Extra: Predictors in addition to the ones actually used to generate the data that appear in the final model.

2.2 Analysis of the Simulation

As noted above, in all cases the true model was generated from $k = 3$ predictors 'Score' was used as the response and the independent variables taken to be sample size, error variance and type of 'extra' predictors. A guiding principle in all of my analyses is the distinction between statistical and practical significance, especially in experiments like mine where large sample sizes can result in tests with large power that detect small effect sizes. The first step in building a model to analyze my experiment was to decide whether or not to include interaction terms in addition to the main effects. I started by putting all of the interaction terms in the model and found that all of them attained statistical significance. However, since statistical significance is not synonymous to practical significance and my goal here is to focus on the big picture, I investigated whether the interaction terms could be dropped without greatly diminishing the explanatory power of my model. To accomplish this, I fit three types of submodels of the full, three way factorial structure and compared their coefficients of determination and mean square errors. Besides of mean square errors and R-square, I also used coefficients of variation (CV) to make a meaningful basis of comparison.

In general use, the coefficient of variation is the ratio of the standard deviation of a variable to its mean. It describes the relative dispersion of the variable and does not depend on the unit of measurement. A higher CV indicates a bigger dispersion. In the regression setting used here, CV is the ratio of the root means squared error (RMSE) to the mean of the dependent variable, and it describes the model fit in terms of the relative sizes of the squared residuals and outcome values. Since it represents values of residuals relative to the predicted value of the response, a smaller CV indicates a better model fit than a large CV. Note that a CV is often presented as the given ratio multiplied by 100.

The unit free property of CV allows us to use it to compare the fit of two competing regression models it is also interesting to note the differences between a model's CV and its coefficient of variation, R-squared. Both are scale free measures that are indicative of model fit, but they define model fit in different ways: CV evaluates the

relative closeness of the predictions to the actual values; meanwhile R-squared evaluates how much of the variability in the actual values is explained by the model.

The following table shows an example of the RMSE, R-Square and CV for comparing different models based on score for backward elimination. When fitted with only main effects, RMSE = 0.308642, R-Square = 0.81497 and CV = 29.02942. When the model contained all interaction terms, RMSE = 0.25218, R-Square = 0.876571 and CV = 23.7189, with p-values for all interactions terms being less than 0.0001 and therefore highly statistically significant. However, as mentioned earlier statistical significance doesn't necessarily mean that the additional terms are meaningful in a practical sense. My decision to leave them out is supported by the relatively small difference s in RMSE, R-Square and CV.

Table 2.3: Summary of R-Square, RMSE and CV for Different Models

Model		RMSE	R-Square	Coefficient of variation
1	Main effects only	0.308642	0.814970	29.02942
2	With 2-Way Interactions	0.269280	0.859222	25.32724
3	All Interactions	0.252180	0.876571	23.71890

The pattern seen in the above table is typical of what happens when analysis of variance models based on a given data set are constructed. Specifically, raw measures of 'fit' increase as more terms are added. However, in my judgment, the increases in RMSE, R-Square and CV from the main effects model to the model containing all interactions are not large enough to justify using the two bigger models. Therefore, in view of the discussion of statistical vs practical significance given above and my search for 'the big picture', I will base the analysis of my simulation study on the simpler main effects model.

The analysis of Variance Table obtained from the analysis of the main effects model is given below in **Table 2.4**

Table 2.4: Analysis of Variance Main Effects Mode Based on Backward Score

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	8	16779.17446	2097.39681	22017.6	<.0001
Ssize	3	2090.16958	696.72319	7313.92	<.0001
Evari	4	12824.32850	3206.08213	33656.2	<.0001
Rela	1	1864.67637	1864.67637	19574.6	<.0001
Error	39991	3809.53654	0.09526		
Corrected Total	39999	20588.71100			

Assessment of Normality and Model Fit:

Before proceeding with the analysis, I carried out several checks of the assumption of normality based on the residuals obtained from fitting the main effects model based on score of backward elimination. All of the tests in Table 2.4.1.3 raise questions about the assumption of normality. However, these tests are very powerful here since sample sizes are very large and may indicate departures from normality that do not have serious consequences for the validity of the analyses of variance used here.

Table 2.5: Tests for Normality

Test	Statistic		p Value	
Kolmogorov-Smirnov	D	0.03104	Pr > D	<0.0100
Cramer-von Mises	W-Sq	7.186324	Pr > W-Sq	<0.0050

Test	Statistic		p Value	
Anderson-Darling	A-Sq	41.43853	Pr > A-Sq	<0.0050

Furthermore, the Stemplot and Normal Probability plots given below may be interpreted as indicating that lack of normality results mainly in the tails of the distributions and may be due to the presence of outliers, a conjecture supported by a plot of the Studentized Residuals presented in Figure 2. 1 and Figure 2.2

Figure 2.1: Histogram of Studentized Residuals

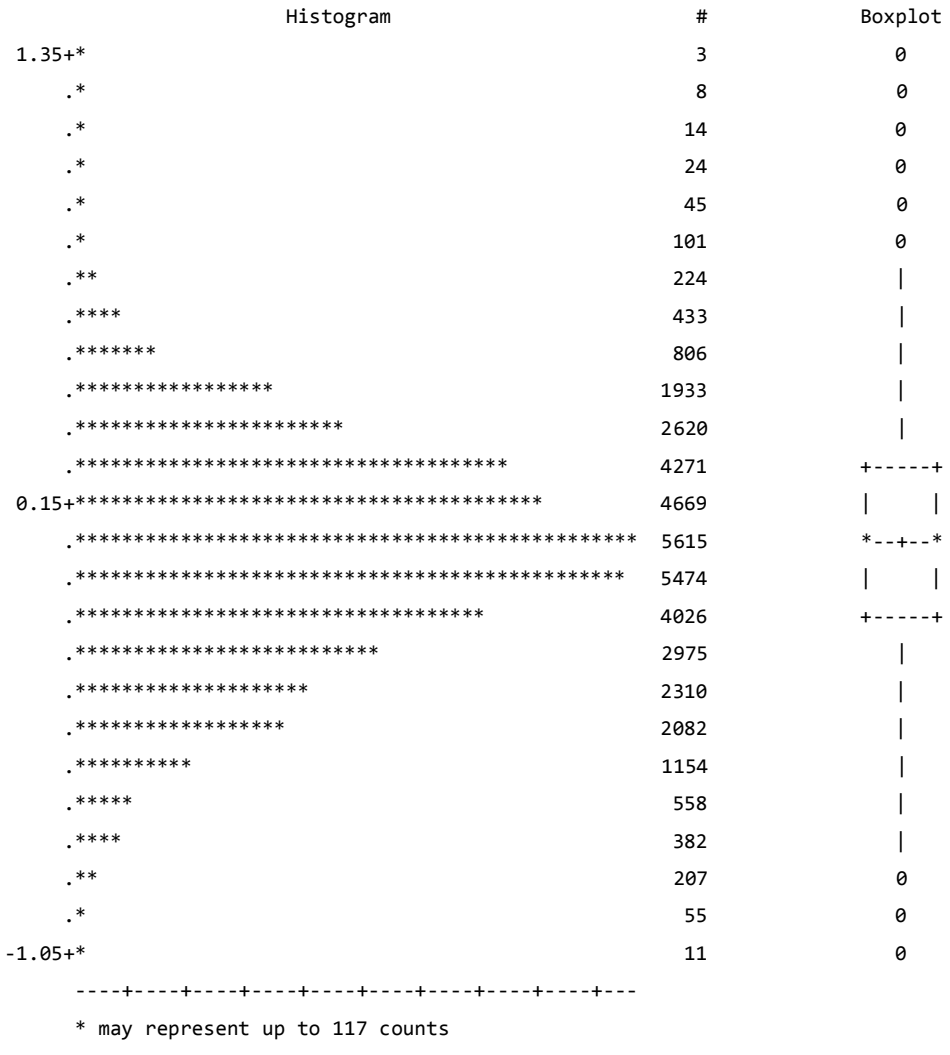


Figure 2.2: Normal Probability Plot

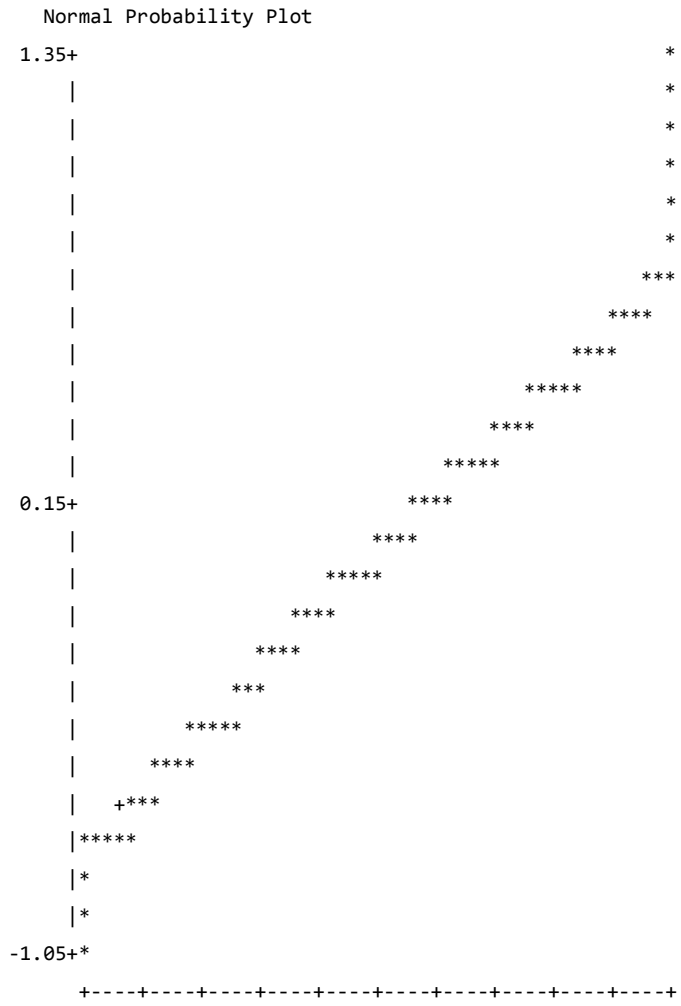
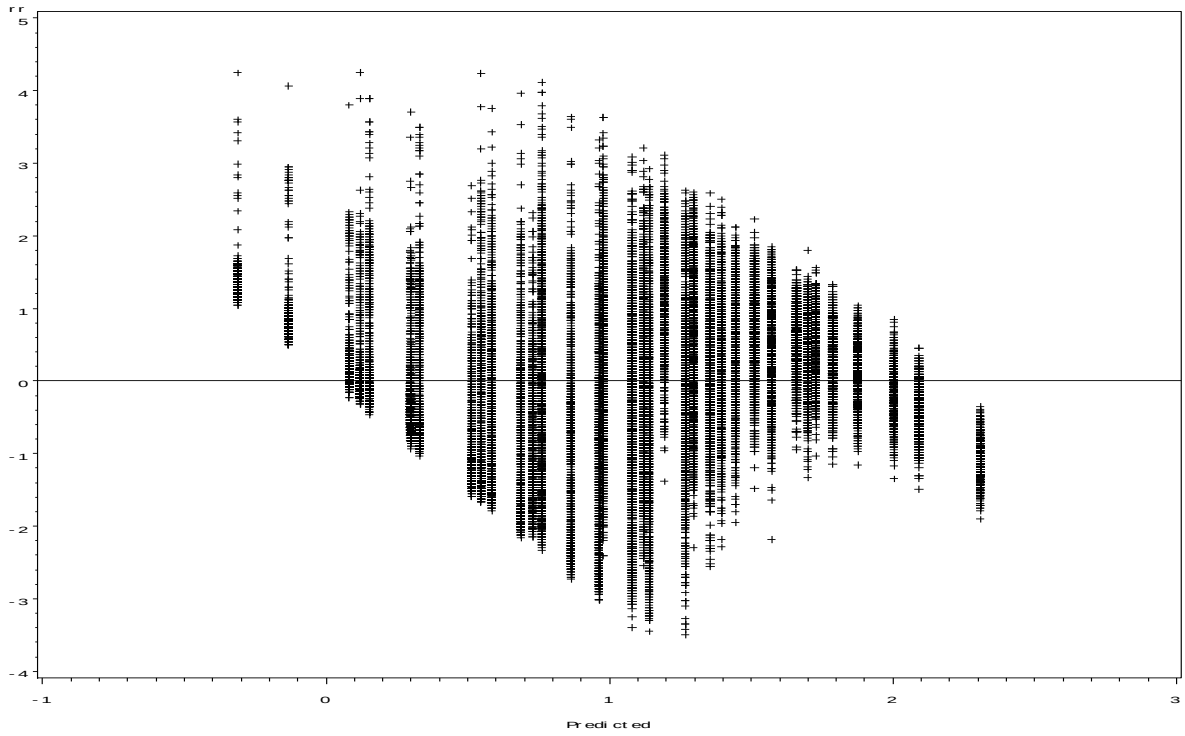


Figure 2.3: Residual Plot



The residual plot in **Figure 2.3** is very informative. It is based on 40,000 points which are scores of backward elimination, clustered in 40 vertical lines; each line corresponding to a predicted value of the response backward 'score'. Note that the residuals decrease as the predicted score increases and that there are quite a few extreme residuals at the low end. In sum, although there are some reservations about using the main effects model as a basis for analysis, given the robustness of the analysis of variance in a balanced design with respect to departures from normality and the assumption of equal variances, I feel that the main effects model is adequate for my purposes.

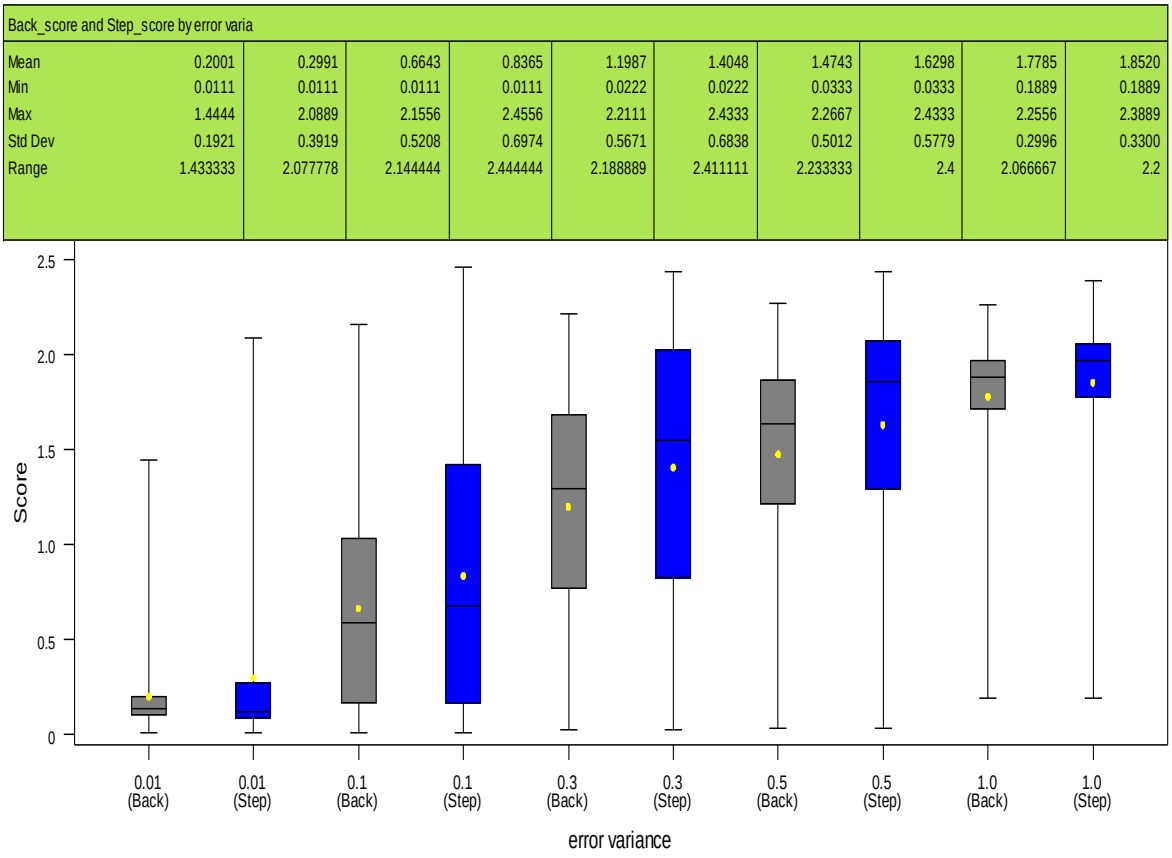
*Appendix A provides the SAS code for the plots and tables Chapter 2.3.

2.2.1 Box Plots

All of the main effects in my analysis of variance attain statistical significance. I begin my analysis by using box plots of the data categorized by each of these main effects. The box plot is a quick way of graphically examining and comparing sets of data. Box plots display features of data sets that reflect the distributions for which they were sampled without making any parametric assumptions. The spacings between the different parts of the box help indicate the degree of dispersion (spread) and skewness in the data.

Figure 2.4 below presents Box plot for Back_score by error variance and Box plot for Step_score by error variance.

Figure 2.4: Box Plot of Back_Score by Error Variance

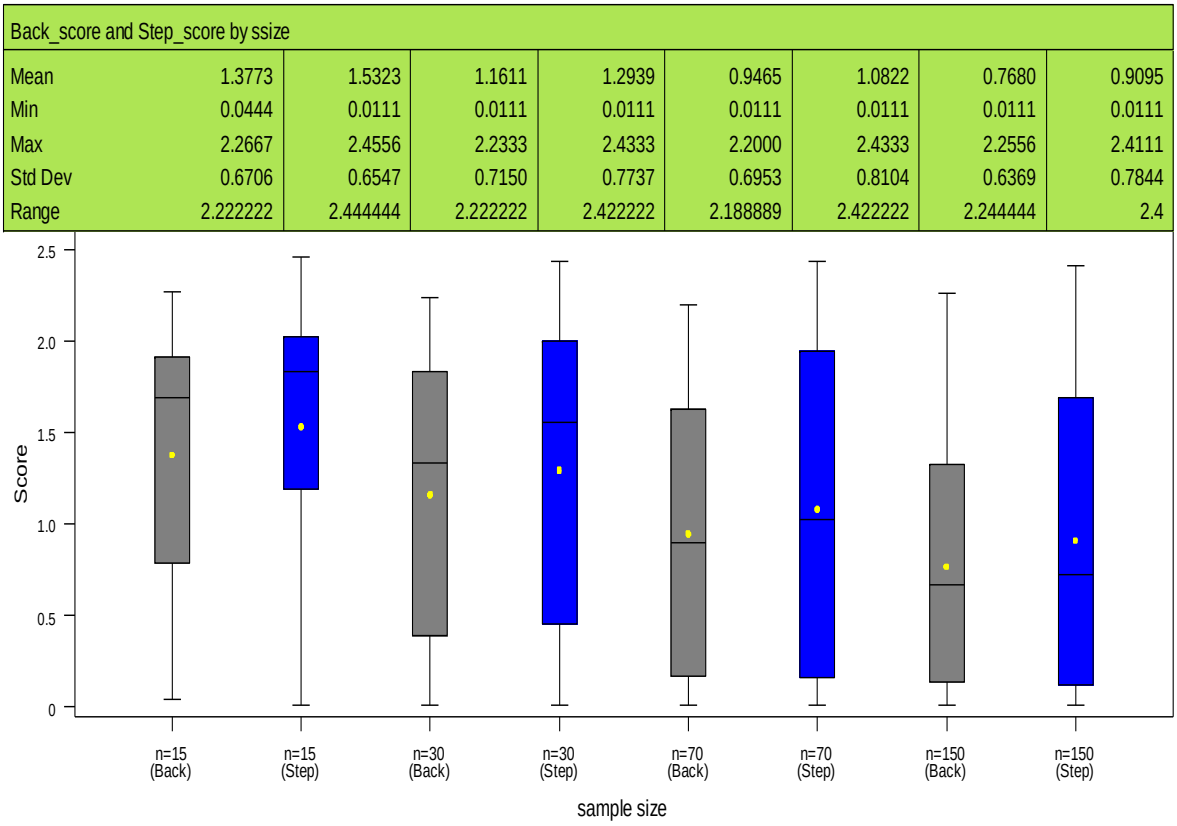


In Figure 2.4, the gray shaded boxes are for backward score, and the blue shaded boxes are for stepwise score. The main feature of **Figure 2.4** is the steady increase in score, as expected, and therefore decreases in performance as the error variance increases. In all but one case, the lower quartile of score lays above the median on the preceding box. Also, as expected, when the error variance is small, almost all the data sets come close to producing the correct model. Thus, both selection methods perform well when the error variance is small. A practical implication of this conclusion is the recommendation that researchers try to minimize the error terms of their models by carefully designing and carrying out their experiments.

A comparison of the boxes for the two methods in Figure 2.1 shows them to be very similar. Specifically, Figure 2.4 provides evidence that backward elimination and stepwise selection are two very comparable variable selection methods In terms of the effects of changes in the error variance in the regression models being studied.

Next, I present and discuss the effect of sample size on the performance of backward elimination and stepwise variable selection. **Figure 2.5** below presents Box plot for Back_score by sample size and Box plot for Step_score by sample size.

Figure 2.5: Box Plot of Back_Score and Step_Score by Sample Size



Back_score and Step_score by ssize

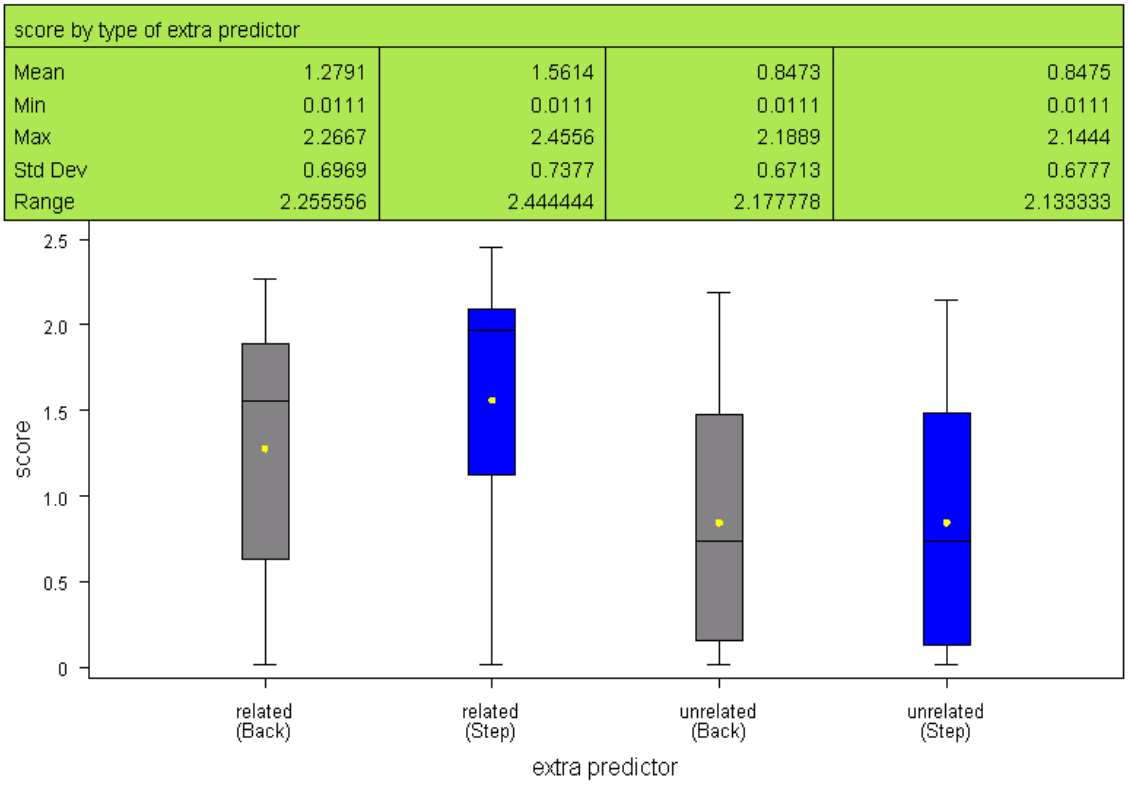
Mean	1.3773	1.5323	1.1611	1.2939	0.9465	1.0822	0.7680	0.9095
Min	0.0444	0.0111	0.0111	0.0111	0.0111	0.0111	0.0111	0.0111
Max	2.2667	2.4556	2.2333	2.4333	2.2000	2.4333	2.2556	2.4111
Std Dev	0.6706	0.6547	0.7150	0.7737	0.6953	0.8104	0.6369	0.7844
Range	2.222222	2.444444	2.222222	2.422222	2.188889	2.422222	2.244444	2.4

Again, gray shaded boxes are for Backward score, and blue shaded boxes are for Stepwise score. Although in most applications, the larger the sample size the better, in real world settings, cost and time are major limiting factors and in many cases only small sample sizes are possible. My simulation study included what I believe to be representative sample sizes, small medium and large. From **Figure 2.5** we se that performance slowly improves for both methods as sample size increases and that on the average, for each sample size, backward elimination is better than stepwise selection But,

over the range of settings of my study, the rates of improvement of the scores as sample size increases are surprisingly low, especially compared to the impact of decreasing error variance shown in Figure Again, when we look at and compare the two sets of boxplots for Backward elimination and Stepwise selection, the patterns very similar. In general, for all sample size, the mean of score is a little bit higher(worse) for Stepwise score than Backward score and the spread of score for Stepwise is bigger than spread of score for Backward.

Now, let us look at the box plots comparing the scores attained by the two types of extra predictors I used, related and unrelated. **Figure 2.6** below presents Box plot for Back_score by type of extra predictor and Box plot for Step_score by type of extra predictor.

Figure 2.6: Box plot of Back_score by Extra predictor



In **Figure 2.6,** gray shaded boxes are for Backward score, and blue shaded boxes are for Stepwise score. We notice that the reduction of score going from related to unrelated is almost 50%, for both for both Back_socre and Step_score, which is very big in my opinion and tells us that when the extra predictors are functions of the true predictors, both methods are less effective than when the extra predictors are unrelated to the true ones. This result seems reasonable to me and I am glad I included this factor in my study. Once again, when look at the box plots side by side for comparing Back_score and Step_score for the same type of extra predictor; one has a difficult time distinguishing them by simply looking at them.

*Appendix A, provides the SAS code for constructing the plots in Figures 2.4 to 2.6

2.2.2 Lsmeans and/or Means analysis

To augment the mostly visual information conveyed in the box plots discussed above, I present and analyze LSMEANS obtained from using SAS applied to my main effects model given in **Table 2.6.**

Table 2.6: LSMEANS of Backward Scores and Pairwise Comparisons of SSizes

ssize	score LSMEAN	Standard Error	Pr > |t|	LSMEAN Number
15	1.37728549	0.00308642	<.0001	1
30	1.16107824	0.00308642	<.0001	2
70	0.94649107	0.00308642	<.0001	3
150	0.76796088	0.00308642	<.0001	4

Least Squares Means for effect ssize Pr > |t| for H0: LSMean(i)=LSMean(j) Dependent Variable: score				
i/j	1	2	3	4
1		<.0001	<.0001	<.0001
2	<.0001		<.0001	<.0001
3	<.0001	<.0001		<.0001
4	<.0001	<.0001	<.0001	

The results here are consistent with what I found in the box plots. Specifically, the performance of backward elimination improves as sample size increases. As evidence,

I present **Table 2.6** which gives pair wise p-values, which shows that all sample size marginal means are statistically significantly different. Similar results, presented in Appendix B, were obtained from the LSMEANS analysis for the factors error_variance and type of extra predictors.

In Appendix B, I also provide estimated LSMEANS for all of the forty combinations of the factors sorted by descending scores based on backward scores. These means are plotted against sample size in **Figure 2.7** and against error_variance in **Figure 2.8,** although both plots indicate some level of interaction, mostly at the ends, the fairly regular decreasing pattern of score as sample sizes increases and error_varaince decreases dominate both graphs. These relationships are further explored in the next section.

Figure 2.7: Plot of SSize by Evari Interaction of Lsmean Against Error Variance

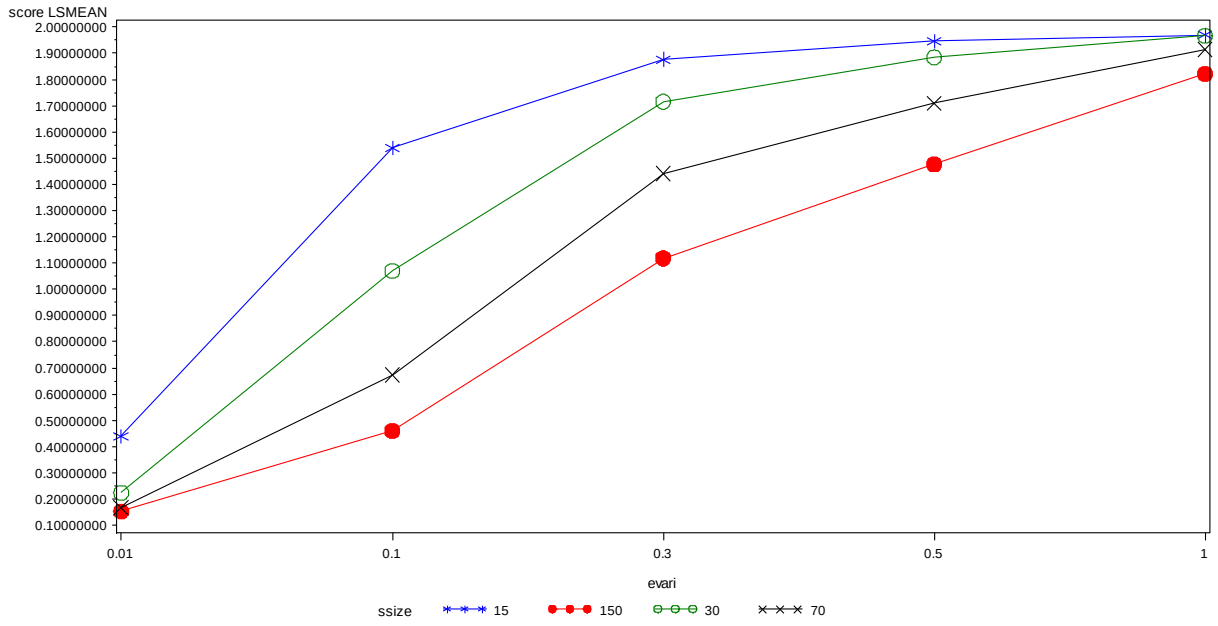
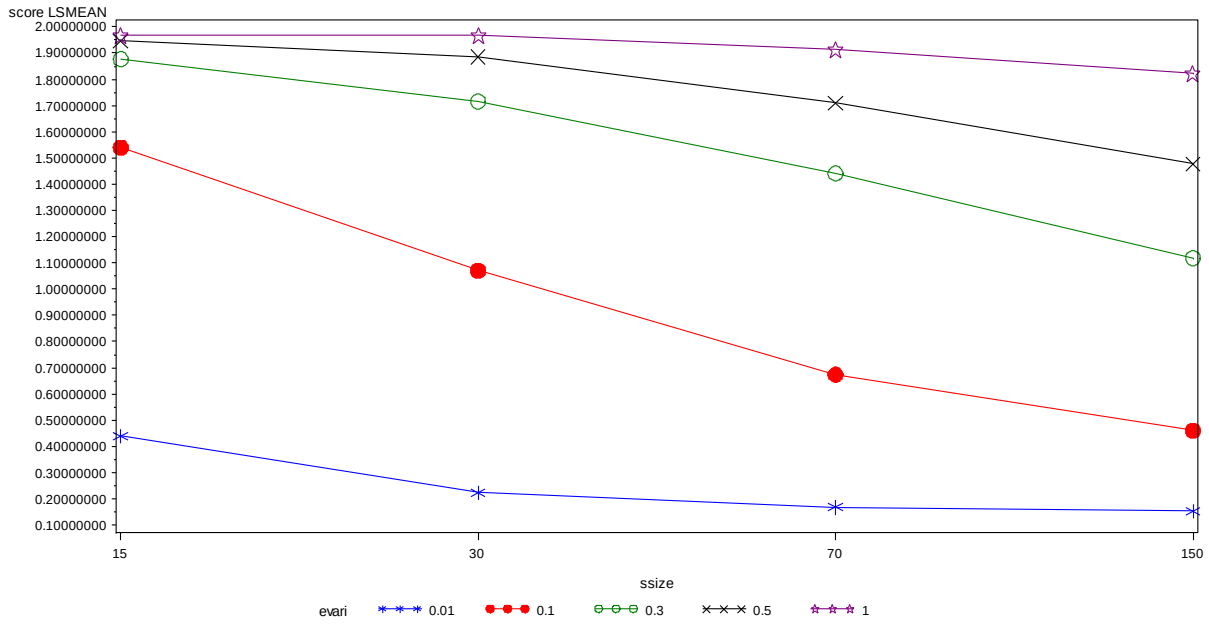


Figure 2.8: Plot of Sample Size by Evari Interaction of Lsmean Against Sample Size



Overall, the impact of these factors on mean backward score is given in Table 2.7 where it is seen that error_variance is the most important factor.

Table 2.7: Summary of Impact of Sample Size, Error Variance and Extra Predictor Based on Backward Score

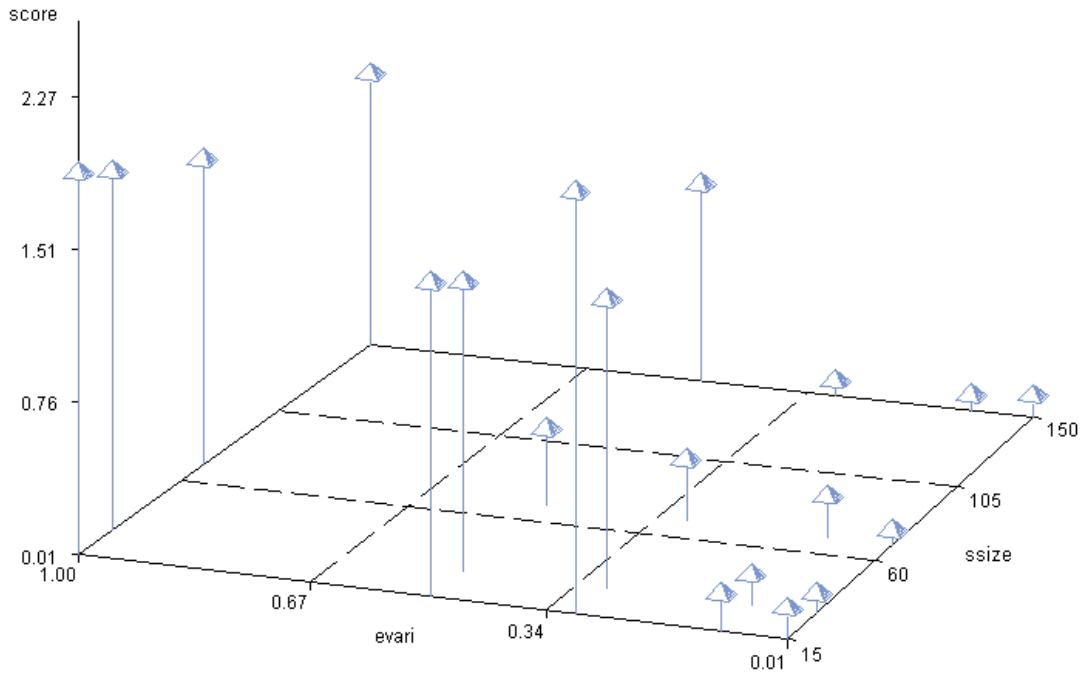
Influence Extent						
Small impact	Sample sizes	<	Relationship of Extra Predcitors	<	Error variance	Big impact

* SAS code in Appendix A.

2.2.3 Regression Analysis of Backward Elimination Score

Two of the three factors in my study, sample size and error_variance are quantitative, a structure not incorporated in the analysis carried out above. Here, I use a regression approach which yields estimates of the effect of unit increases in these predictors on mean response. This analysis can also be carried out by omitting the quantitative factors from the class statement in SAS. First I present a 3 dimensional scatter plot to visualize the relationship between the response variable score and the factors sample size and error variance, after averaging over type of extra predictor.

Figure 2.9: Three Dimensional Scatter Plot of SSize by EVariance Interaction of Backward Scores



It is apparent that the low values of score (good performance) are clustered in the region of small error variance, while the sample size differences seem not to have a big impact on the variation in scores.

The results of a regression analysis of the simulation results obtained by using backward elimination carried out by SAS with mean score as the response and sample size, error variance and type of extra variables as the independent variables are given below. Note, since the 'type of extra predictors' is a categorical predictor having only two levels, without loss of generality, we coded it '1' if the type of extra predictors is *related* and 0 otherwise.

Table 2.8: Parameter Estimates of Regression Analysis of Backward Score

Parameter Estimates						
Variable	Label	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|
Intercept	Intercept	1	1.45537	0.00521	279.43	<.0001
Ssize	Ssize	1	-0.00409	0.00003743	-109.14	<.0001
Evari	Evari	1	1.47144	0.00556	264.46	<.0001
Related		1	0.43182	0.00392	110.16	<.0001

Note that in the regression analysis I used ssize=ssize-70, and evari=evari-0.3 to facilitate interpretation of the results. For example, using this coding, the constant term represents how much higher mean score is for related than for unrelated predictors. Thus, we estimate that mean score for related is 0.43182 higher for related than for unrelated extra predictors.

The fitted surface can also be used to estimate mean score at values of the independent variables within the scope of the model. For example, for sample size equal to 70, error variance = 0.3 and for un-related extra predictors, this model estimates mean score to be the estimated intercept, which equals 1.46, and estimates mean score to be 1.89 for related extra predictors, again confirming that in cases like this backward elimination performs better for un-related than related extra predictors. Also, for a fixed error variance and type of extra predictors, this fitted model estimates for a 10 unit increase in sample size, mean score decreases by -0.0419. Similarly, for a fixed error variance and type of extra predictors, for each 1 unit increase in error variance, the model estimates that backward score increase by 1.47144, an impact more than 30 times larger than is associated with changes in sample size. Finally, conditional on fixed sample size and error variance, mean score using backward elimination is estimated to be 0.43 higher for related extra predictors than for un-related extra predictors.

I also looked at the results, presented in **Table 2.9** obtained by adding a sample_size by error_variance interaction term. Although, the p-value associated with the test for dropping this term is small, its inclusion has little practical effect on the results described above.

Table 2.9: Parameter Estimates of Regression Analysis on Backward Score with 2 Way Interaction

Parameter Estimates						
Variable	Label	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|
Intercept	Intercept	1	0.71149	0.00281	253.02	<.0001
Ssize	ssize	1	-0.00404	0.00003842	-105.16	<.0001
Evari	evari	1	1.46937	0.00558	263.50	<.0001

Parameter Estimates						
Variable	Label	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|
Inter		1	-0.00055243	0.00010622	-5.20	<.0001
Related		1	0.43182	0.00392	110.19	<.0001

* SAS code can be found in Appendix A.

2.3 A Comparison of Backward and Stepwise Variable Selection

Originally, my research only aimed on assessing the performance of backward elimination. However since stepwise selection is such a widely used competitor, I decided to compare the relative performance of the two methods. One of the few quantitative statement s about comparing variable selection methods appears in "The Little Handbook of Statistical Practice" by Gerard E. Dallal, Ph.D., He states that "among these three automatic search methods, Backwards Elimination has an advantage over forward selection and stepwise regression because it is possible for a set of variables to have considerable predictive capability even though any subset of them does not. Forward selection and stepwise regression will fail to identify them. Because the variables don't predict well individually, they will never get to enter the model to have their joint behavior noticed. Backwards elimination starts with everything in the model, so their joint predictive capability will be seen.

Since variables are chosen because they look like good predictors, estimates of anything associated with prediction can be misleading. Regression coefficients are biased away from 0, that is, their magnitudes often appear to be larger than they really are. (This is like estimating the probability of a head from the fair coin with the most heads as the value that gained it the title of "most heads.") The t statistics tend to be larger in magnitude and the standard errors smaller than what would be observed if the study were replicated. Confidence intervals tend to be too narrow. Individual P values are too small. R^2 and even adjusted R^2 is too large. The overall F ratio is too large and its P value is too small. The standard error of the estimate is too small. "

[Cited from "The Little Handbook of Statistical Practice" by Gerard E. Dallal, Ph.D]

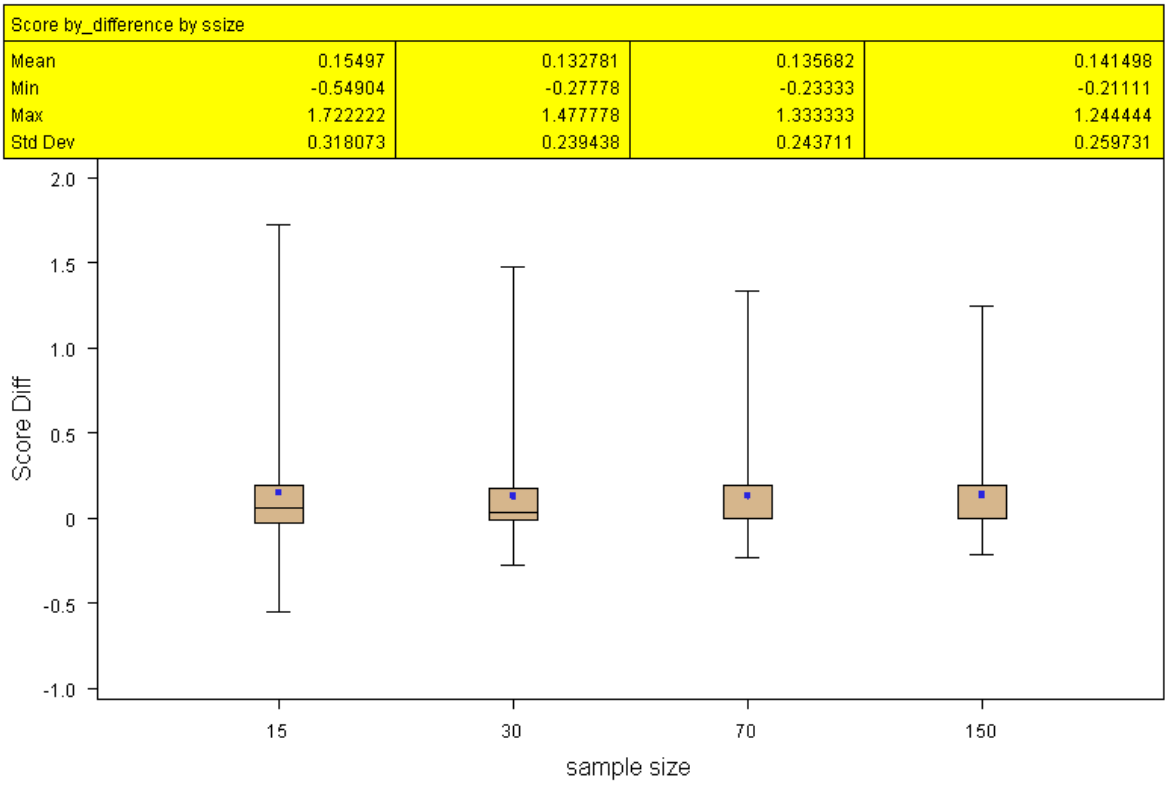
Here, in the second part of my study, I apply quantitative analyses of my simulation results to compare these two methods. As mentioned above, I expect higher (worse) scores from Stepwise than from Backward elimination. I used both methods on each of my 40000 data sets. Viewing the data sets as blocks, I carried out a paired analysis based on the difference between the scores defined by a new response variable

"Diff" equal to "Stepwise_score – Backward_score". Positive values of 'Diff' indicate a better performance of backward elimination.

2.3.1 Boxplot of Diff

Figure 2.10 below present the box plot for different sample size of Score Difference.

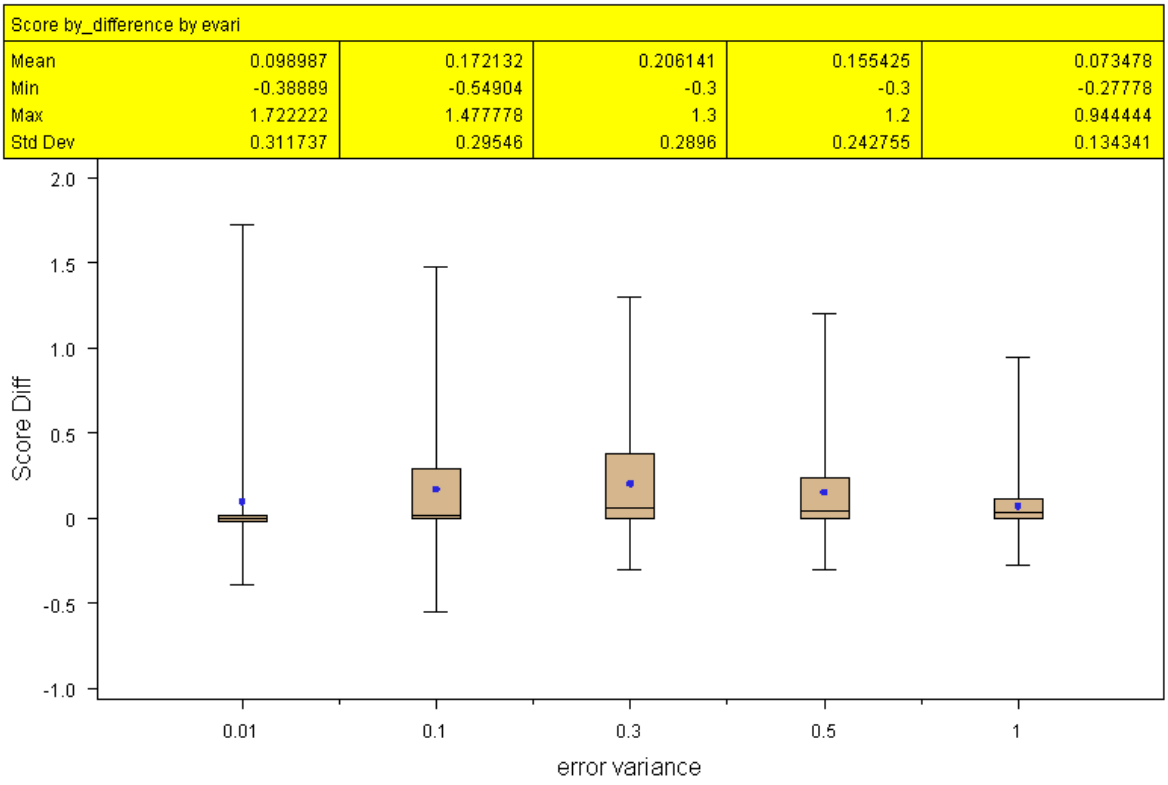
Figure 2.10: Box Plot of Score Difference by Sample Size



The above plot shows a very small spread of the Diff scores, with values having a slight tendency to be more positive than negative, favoring backward over stepwise. Note that boxes change relatively little as sample size increases, indicating that sample size has little ability to distinguish between the two methods.

Figure 2.11 below present the boxplot for different Error_Variance of Score Difference.

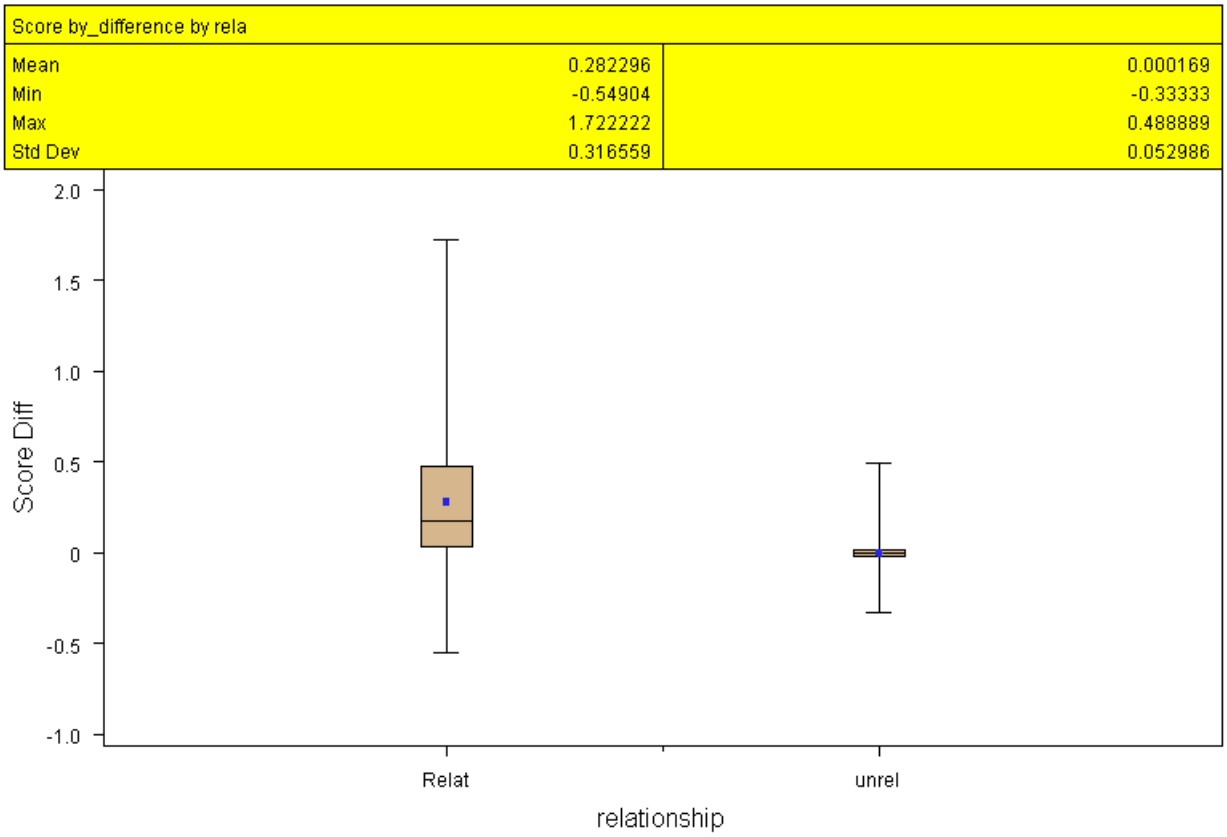
Figure 2.11: Box Plot of Score Difference by Error Variance



The evidence provided in **Figure 2.11** above indicates that the error_ variance is an important factor in distinguishing between the two methods. Note in particular that both methods behave most similarly when error_variance is at its smallest and largest settings and that most of the differences are positive, again indication the superiority of backward over stepwise.

Figure 2.12 below presents side by side box plots of Diff the two types of Extra_Predictors used in my study

Figure 2.12: Box Plot of Score Difference by Extra Predictor



In **Figure 2.12,** we see that for un-related extra predictors, the performance of the two methods is almost identical. For related extra predictors, the mean difference of the two methods is statistically significantly different from zero and large enough to be of practical interest. Once again, backward elimination seems doing a better job.

2.3.2 Lsmeans and/or Means Analysis of Diff

To follow the same pattern that we analyzed the data for Backward score, and to augment the mostly visual information conveyed in the box plots discussed above, I present and analyze LSMEANS obtained from using SAS applied to my main effects model, which is the same as we used for analyzing Backward score. The LSMEANS based on score diff are given in **Table 2.10.**

Table 2.10: Lsmean Marginal Means Diff of Sample Size

ssize	diff LSMEAN	Standard Error	Pr > |t|	LSMEAN Number
15	0.15497006	0.00221575	<.0001	1
30	0.13278065	0.00221575	<.0001	2
70	0.13568226	0.00221575	<.0001	3
150	0.14149843	0.00221575	<.0001	4

The above table list marginal means Diff for the levels of sample size used in my study. sample_size effects. Since all lower limits are above zero, we infer that backward elimination performs better on the average than stepwise for all the sample sizes involved in my study.

Table 2.11: Summary of Pairwise Comparison Lsmean Means Diff of SSize

Bonferroni Comparison Lines for Least Squares Means of ssize				
LS-means with the same letter are not significantly different.				
		diff LSMEAN	Ssize	LSMEAN Number
	A	0.15497006	15	1
	B	0.14149843	150	4

Bonferroni Comparison Lines for Least Squares Means of ssize				
LS-means with the same letter are not significantly different.				
		diff LSMEAN	Ssize	LSMEAN Number
C	B	0.13568226	70	3
C		0.13278065	30	2

The information provided by this table indicates groupings of sample size for which Diff means are statistically, significantly different. Note that despite statistical significance between several groupings in **Table 2.11,** the mean differences, from a practical standpoint, are similar across sample sizes. But, keep in mind that neither performs well for small samples.

Table 2.12: Lsmean Marginal Means Diff of Error Variance

evari	diff LSMEAN	Standard Error	Pr > |t|	LSMEAN Number
0.01	0.09898716	0.00247728	<.0001	1
0.1	0.17213197	0.00247728	<.0001	2
0.3	0.20614148	0.00247728	<.0001	3
0.5	0.15542544	0.00247728	<.0001	4
1	0.07347820	0.00247728	<.0001	5

For the error variance factor, lower confidence limits for Diff means are all positive, again favoring backward elimination, especially for error_variance = .1, .3 or .5.

Table 2.13: Summary of Pairwise Comparison LSMEANS of Diff of SSize

Bonferroni Comparison Lines for Least Squares Means of evari			
LS-means with the same letter are not significantly different.			
	Diff LSMEAN	Evari	LSMEAN Number
A	0.206141482	0.3	3
B	0.172131968	0.1	2
C	0.155425443	0.5	4
D	0.098987158	0.01	1
E	0.073478202	1	5

There is not much new information in **Table 2.13** above. We do, however note, Error_variance is an important factor in distinguishing the two methods of variable selection.

Table 2.14: LSMEAN Marginal Means Diff of Type of Extra Predictor

rela	diff LSMEAN	95% Confidence Limits	
Relat	0.282296	0.279225	0.285367
unrel	0.000169	-0.002901	0.003240

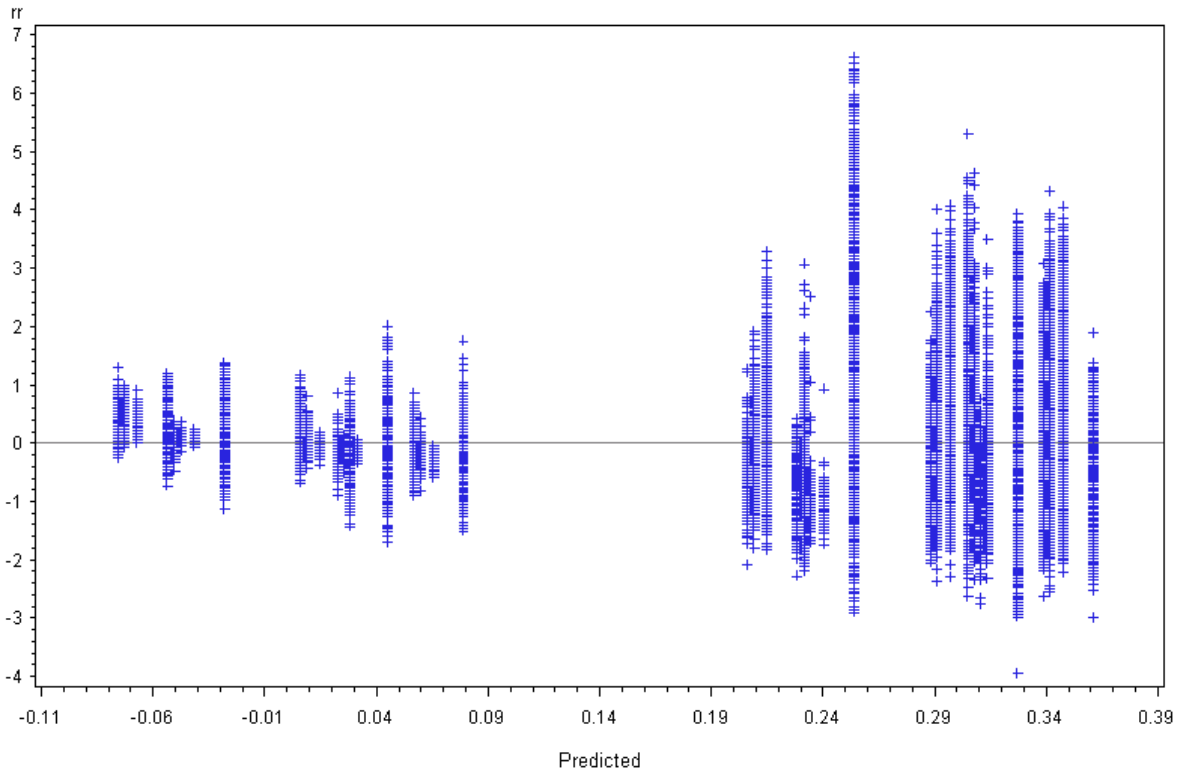
Table 2.14 confirms the box plots in **Figure 2.13**. Specifically, the mean scores are only statistically significantly different for the related extra predictors and their difference appears to be of practical importance. Another important finding here is that

mean difference score for related extra predictors is the largest among all the marginal mean differences. That is if the provisional full regression model given in (1) involves higher order terms such as quadratic form or interaction terms, backward elimination will work better than stepwise selection.

2.3.3 Regression Analysis of Diff

The purpose of this section is to quantitatively assess the effects that changes in the levels of the factors I used have on mean Diff, The studentized residuals are plotted in **Figure 2.13**

Figure 2.13: Studentized Residuals Plot of Score Difference

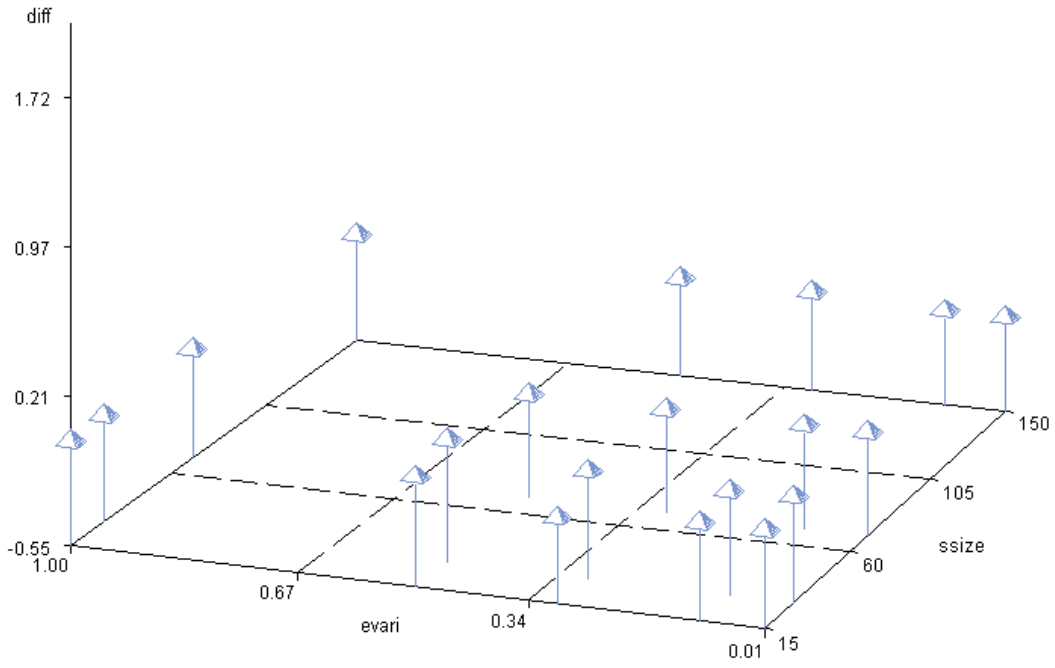


Again, each bar represents the 1000 data sets generated for each of my 40 factor combinations. Clearly, the plot indicates non-constant variance. Looking further, the 40

estimated LSMEANS of Diff cluster into two regions, above and below Zero, a pattern discussed below. The large spike in **Figure 2.13** at predicted value 0.25, where Error_Variance = 0.01, Sample_Size = 15, and the extra predictors are 'related, indicates high variation about the fitted plane at these settings. My interpretation of this behavior that when Error_Variance is very small, both backward and stepwise methods exhibit their best performances and that Diff is small for some settings of sample size and type of extra predictors and large for other settings.

Then I present the scatter plot in **Figure 2.14** (this plot averaged over the type of extra predictors), trying to get a big picture before looking at details. Clearly, mean Diff decreases as *n* increases and the error variance decreases. Next we look at a regression analysis of Diff.

Figure 2.14: Three Dimensional Scatter Plot of SSize by EVariance for Diff



I start with the model including main effects only:

Table 2.15: Parameter Estimates of Regression Analysis for Main Effects

Parameter Estimates						
Variable	Label	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|
Intercept	Intercept	1	0.00512	0.00162	3.16	0.0016
Ssize	ssize	1	-0.00003612	0.00002157	-1.67	0.0940
Evari	evari	1	-0.06208	0.00321	-19.36	<.0001
X1		1	0.28213	0.00226	124.89	<.0001

Note: To make the coefficient estimation meaningful in the regression analysis, I set ssize=ssize-70, and evari=evari-0.3. Interpretation: if sample size equal to 70, error variance = 0.3 and for related disturbing factors, the estimated mean of Diff is 0.28725. Conditional on a fixed error variance and within same category of extra predictors, we estimate that a unit increase in sample size changes mean Diff by-0.00003612, a very small but statistically significant effect, indicating that Diff changes little with sample size. For a fixed error variance and within same category of extra predictors, a unit increase in error variance corresponds to an estimated decrease in Diff of 0.06208, indicating that the advantage of backward elimination over stepwise decreases as the amount of noise increases, in which case both can perform poorly.

Chapter 3 - Conclusion and Further Study

3.1 Conclusion

(1). In this report, I conducted a simulation study to evaluate and compare the performance of backward elimination and stepwise variable selection in a regression setting. I employed ANOVA analysis and regression analysis to describe, quantify my findings. In addition I used plots and tables to summarize and present the results. LSMEANS in the analysis.

(2). To evaluate the performance of a variable selection method, I developed a *score* based on how many of the true and extraneous independent variables are included in the final model. The smaller score is, the better. Score = 0 calibrates the best possible performance.

According to the analysis I performed, the magnitude of error variance plays the most important role; the smaller the variance is the smaller the score for both methods. Another interesting finding is that the spread of the difference scores of the two methods is much smaller for large and small error variances than it is for moderate size error terms. Tends in the middle range of error variance.

Increasing sample size slowly improves the performance of both methods. More dramatically, both methods performed much better when the extra predictors were independent of the true ones than when they were functions of them.

I employed regression analysis to quantify the effects of sample size, type of extra predictors and error variance on the performance of backward elimination and stepwise variable selection. Again, type of extra predictors and magnitude of error variance play big roles in the performance of both methods. , the value of estimated parameters tell us that the error variance have a strong positive relationship with the score, while the type of extra predictors is also an important factor when using the backward elimination method.

(3). As expected, the performance of stepwise and backward elimination variable selection are very similar. Using a paired analysis based on the difference between the scores on the experimental unit, these two methods are very similar. It is very hard to tell the difference from the plots, especially for the unrelated extra predictor setting. But when we employed ANOVA and regression analysis of the differences, a pattern appears slightly favoring backward elimination over stepwise selection appears. Much can be learned when you carefully examine the scatter plot and sort the means from smallest to largest, as listed in Table 2.16 below.

Table 3.1: Summary of Means for 40 combinations of treatment effects

Obs	Ssize	evari	rela	mean
1	15	0.01	unrel	-0.04117
2	30	0.01	unrel	-0.01911
3	30	0.10	unrel	-0.01376
4	70	0.01	Relat	-0.00679
5	70	0.01	unrel	-0.00534
6	70	0.10	unrel	-0.00497
7	150	0.01	Relat	-0.00270
8	150	0.01	unrel	-0.00259
9	150	0.10	unrel	-0.00162
10	70	0.30	unrel	-0.00083
11	150	0.30	unrel	-0.00071

Obs	Ssize	evari	rela	mean
12	15	0.10	unrel	-0.00021
13	150	0.50	unrel	-0.00008
14	150	1.00	unrel	0.00398
15	15	1.00	unrel	0.00418
16	15	0.50	unrel	0.00442
17	70	0.50	unrel	0.00630
18	30	1.00	unrel	0.00826
19	70	1.00	unrel	0.00948
20	30	0.30	unrel	0.01306
21	15	0.30	unrel	0.02000
22	30	0.50	unrel	0.02410
23	15	1.00	Relat	0.05278
24	30	1.00	Relat	0.07314
25	15	0.50	Relat	0.10313
26	150	0.10	Relat	0.10534
27	30	0.01	Relat	0.10939
28	70	1.00	Relat	0.16347

Obs	Ssize	evari	rela	mean
29	30	0.50	Relat	0.19571
30	15	0.30	Relat	0.19882
31	150	1.00	Relat	0.27254
32	70	0.10	Relat	0.27868
33	30	0.30	Relat	0.37097
34	70	0.50	Relat	0.38701
35	15	0.10	Relat	0.44754
36	150	0.30	Relat	0.51801
37	150	0.50	Relat	0.52280
38	70	0.30	Relat	0.52982
39	30	0.10	Relat	0.56606
40	15	0.01	Relat	0.76020

One may observe that when things become complicated, that is when the extra predictors are quadratic or pairwise cross products of true predictors, and sample size is small backward elimination performs better than the stepwise selection. In other words, Backward elimination seems can handle more complicated situation.

3.2 Limitations and Further Study

There are some limitations in transferring the findings in this from this report and applying them in real world settings. First, normality and constant error variance were assumed. Performance with other distributions and non-constant error variance should be studied. Second, our predictors are generated from a uniform distribution on the interval (0 to 1); other models for generating the independent variables should be investigated. . Third, the score function, we developed should be viewed as an initial attempt to evaluate the performance of a variable selection method. Other scores need to be developed. Finally, I put a lot of time into constructing the SAS code for this simulation study and it take a long time to execute it. Specifically, it would have taken one computer working all day for 27 days to complete the simulation. Future effort should be invested in creating more efficient code, possibly using another language.

Bibliography

[1] Austin, Peter C. and Tu, Jack V. Automated variable selection methods for logistic regression produced unstable models for predicting acute myocardial infarction mortality. Journal of Clinical Epidemiology 57 (2004); 1138–1146.

[2] Chen, Qixuan, University of Michigan, Ann Arbor, MI； Gillespie, Brenda, University of Michigan, Ann Arbor, MI. A SAS® MACRO FOR PERFORMING BACKWARD SELECTION IN PROC SURVEYREG. Paper SD10.

[3] Derksen, S. and Keselman, H. J. 1992. Backward, forward and stepwise automated subset selection algorithms: Frequency of obtaining authentic and noise variables. British Journal of Mathematical and Statistical Psychology 45: 265-282.

[4] Draper, N.R. and Smith, H. (1998). Applied Regression Analysis, Third Edition. Wiley, NYC.

[5] Kramer, Andrew A. , Ph.D. Cerner Corporation, Vienna, VA. Using ODS to Perform Simulations on Statistics from SAS® Procedures. Paper SA05_05

[6] Orelien, Jean G. Analytical Sciences Inc., Durham, NC. A Macro for Computing a Goodness of Fit Statistic for Linear Mixed Models. Paper # ST-12

[7] SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Appendix A

Simulation code:

SAS Code: For Unrelated Disturbing Factor

```
%macro test(num,count,nn,zz); /* Simulate data and perform backward elimination and stepwise selection
*/
/* num= total runs to test one set of coefficients of predictors
count= number of predictors in the true model
nn= sample size */

proc printto log="E:\msxin.log"; ODS LISTING CLOSE;        /* turn off SAS-log, SAS-output */

data xinx; /* generate random seed to start this whole simulation from system time */
x=int(datetime());
start=int(1000000*ranuni(x));
seed9=int(1000000*ranuni(x));
call symput('start',start);
call symput('seed9',seed9); run;

%put ** Inside the macro: **; /* unmask the seeds for using in macro */
%put _user_;
%put ** In open code: **;
%put _user_;

data gam; /*Generate Beta from uniform distribution(0,1)*/
%Do i =1 %to 3;
%let gamma&i = ranuni(&seed9); %end;
gamma1=&gamma1;
gamma2=&gamma2;
gamma3=&gamma3;

%let open=&start;
%do k=1 %to &num;        *loop for n times runs of step(2)-step(5);
%let open = int(ranuni(&open)*1000000);
%Do I =1 %to 5;              *loop for generate seeds for later use;
%let seed&I = int(ranuni(&open)*1000000); %end;

data test0; set gam; seed1=&seed1; seed2=&seed2; seed3=&seed3; seed4=&seed4; seed5=&seed5;

data predictor;     /* generate a set of predictors for nn=m observations, the predictors from uniform
distribution(0,1) */
set test0;
seed=seed4;
%do _n_=1 %to &count;
SEED = mod( SEED * 397204094, 2**23-1 );
%do i = 1 %to &nn;
w=ranuni(seed); output;
%end; %end;
data ww1; set predictor; w1=w; if _N_<=&nn then output; drop w; run;
data ww2; set predictor; w2=w; if &nn<_N_ and _N_<=&nn*2 then output; drop w; run;
data ww3; set predictor; w3=w; if &nn*2<_N_ then output; drop w; run;
```

```
data test00; merge ww1 ww2 ww3;

data test001; /*generate a set of observations, with error term distributed as normal(0,sigmasquare=)*/
set test00;
y=w1*gamma1+ gamma2*w2 + gamma3*w3 + &zz*rannor(seed1);

data noise;          /* generate a set of predictors for nn=m observations, the predictors from uniform
distribution(0,1) */
set test0;
seed=seed5;
%do _n_=1 %to 6;
SEED = mod( SEED * 397204094, 2**23-1 );
%do i = 1 %to &nn;
wm=ranuni(seed); output;
%end; %end;
data ww1; set noise; w12=wm; if _N_<=&nn then output; drop wm; run;
data ww2; set noise; w13=wm; if &nn<_N_ and _N_<=&nn*2 then output; drop wm; run;
data ww3; set noise; w23=wm; if &nn*2< _N_ <=&nn*3 then output; drop wm; run;
data ww4; set noise; w11=wm; if &nn*3< _N_ <=&nn*4 then output; drop wm; run;
data ww5; set noise; w22=wm; if &nn*4< _N_ <=&nn*5 then output; drop wm; run;
data ww6; set noise; w33=wm; if &nn*5< _N_ then output; drop wm; run;

data test000&k; merge test001 ww1 ww2 ww3 ww4 ww5 ww6;

proc reg data=test000&k; /* perform backward elimination with significant level alpha= */
model y = w1 w2 w3 w12 w13 w23 w11 w22 w33/selection=backward SLS=0.05;
ods output SelectionSummary=aa&k;

proc reg data=test000&k; /* perform stepwise selection with significant level alpha= */
model y = w1 w2 w3 w12 w13 w23 w11 w22 w33/selection=stepwise sle=.05 sls=.05;
ods output SelectionSummary=aaa&k;

*model y = w1 w2 w3 w12 w13 w23 w11 w22 w33/vif;
/*
ods html;
ods graphics on;
PROC CORR DATA=sss PLOTS = MATRIX PLOTS=scatter;
  VAR y w1 w2 w3 w12 w13 w23 w11 w22 w33;
TITLE 'Correlation calculations using PROC CORR'; RUN;
*/
proc printto; /*ODS LISTING;run; /* turn on SAS-log, SAS-output */

%end; run;
%mend test;

/*
ods trace on;
proc reg data=test0002;
model y = w1 w2 w3 w12 w13 w23 w11 w22 w33/selection=backward SLS=0.05;
ods output SelectionSummary=aa; run;
ods trace off;
*/
/*Identify the variable Id for custom selection*/

%macro test1(num,nums,nn);
```

```
/* summarize the performence of backward elimination
num= total runs to test one set of coefficients of predictors */
proc printto log="E:\msxin.log"; ODS LISTING CLOSE;        /* turn off SAS-log, SAS-output */

%do k=1 %to &num;
data bb&k; set aa&k; /* pick value from final results, approach one */
if varremoved in ('w1','w2','w3') then xin=1;
if varremoved in ('w12','w13','w23','w11','w22','w33') then xin=2;
proc freq data=bb&k;TABLE xin / OUT=cc&k;run;
data dd&k; set cc&k;
keep xin score1 count_true;
if xin=1 then score1=count;else delete;
count_true=3-score1;
data ee&k; set cc&k;
keep xin score1 count_true;
if (xin^=1 and PERCENT=100) then xin=1; else delete; count=0; score1=0; count_true=3-score1;
data gg&k; set cc&k;
keep xin score2 count_noise;
if xin=2 then score2=6-count;else delete; count_noise=score2;
data ff&k; merge dd&k ee&k gg&k;
score=(2*(3-count_true)+count_noise)/3;

data hh&k; /* pick value from final results, approach two */
keep cp modelrsquare;
set aa&k;
by dependent;
if last.dependent then output;else delete;run;

data ii&k; /* combine approach one and two */
merge hh&k ff&k;
cpp=(cp /(score1+score2+1)); count=score1+score2; %end;

/* For stepwise part */
%do r=1 %to &nums;
data bbb&r; set aaa&r; /* pick value from final results, approach one */
if varentered in ('w1','w2','w3') then xin=1;
if varentered in ('w12','w13','w23','w11','w22','w33') then xin=2;
if varremoved in ('w1','w2','w3') then xin=3;
if varremoved in ('w12','w13','w23','w11','w22','w33') then xin=4;
proc freq data=bbb&r;TABLE xin / OUT=ccc&r;run;
data ccc&r; merge ccc&r qq; by xin; run;

data ddd&r; set ccc&r;
if xin=1 then score111=count;else delete;
if score111="." then score111=0;
data dddd&r; set ccc&r;
if xin=3 then score1111=count; else delete;
if score1111="." then score1111=0;
data fff&r; merge ddd&r dddd&r;
keep xin score111 score1111 count_true_s;
count_true_s=score111-score1111;

proc print data=hhh1;run;

data eee&r; set ccc&r;
```

```
if xin=2 then score222=count; else delete;
if score222="." then score222=0;
data eeee&r; set ccc&r;
if xin=4 then score2222=count; else delete;
if score2222="." then score2222=0;
data ggg&r; merge eee&r eeee&r;
keep score222 score2222 count_noise_s;
count_noise_s=score222-score2222;
run;

data hhh&r; merge fff&r ggg&r;
score_step=(2*(3-count_true_s)+count_noise_s)/3;
%end;

data jj; /* arrange the summary */
set ii1 ii2 ii3 ii4 ii5 ii6 ii7 ii8 ii9 ii10 ii11 ii12 ii13 ii14 ii15
ii16 ii17 ii18 ii19 ii20 ii21 ii22 ii23 ii24 ii25 ii26 ii27 ii28 ii29 ii30;

Time = datetime(); format time datetime16.;
data ll;
set hhh1 hhh2 hhh3 hhh4 hhh5 hhh6 hhh7 hhh8 hhh9 hhh10 hhh11 hhh12 hhh13 hhh14 hhh15
hhh16 hhh17 hhh18 hhh19 hhh20 hhh21 hhh22 hhh23 hhh24 hhh25 hhh26 hhh27 hhh28 hhh29 hhh30;

data mm;
nn=&nn;
keep count_true count_noise cpp score score_step modelrsquare cp nn;* time;
retain count_true count_noise cpp score score_step modelrsquare cp nn;* time;
merge jj ll;
proc sort data=mm; by score cpp score_step;
proc printto;/*ODS LISTING;run; /* turn on SAS-log, SAS-output */
/*proc print data=jj;run;*/

%mend test1;

proc datasets lib=work kill nolist memtype=data; quit;
/*Identify the variable Id for custom selection*/
data qq; input xin @@;
cards;
1 2 3 4
;run;
%test(30,3,150,1)
%test1(30,30,150)

ods html body='T:\msxin\xin.html';
proc means data=mm mean std p25 p50 p75 min max nolabels;
  VAR cpp score score_step modelrsquare cp count_true count_noise;
  output out=xin; run;

DATA xin; set xin; keep _STAT_ cpp score score_step modelrsquare cp count_true count_noise;
run;
ods html close;

proc print data=xin; run;

/* Combine summary from step(1) - step(6) */
```

```
libname MS 'T:\xin\library';
proc append base=ms.summary_normal data=xin;
proc print data=ms.summary_normal;run;

%macro cls();
        DM 'ODSRESULTS' CLEAR EDITOR;  ODS HTML CLOSE;
        DM 'CLEAR LOG; CLEAR OUTPUT; PGM OFF' LISTING; *EXPLORER;
%mend  cls;
%cls;
```

SAS Code: For related Disturbing Factor

```
%macro test(num,count,nn,zz); /* Simulate data and perform backward elimination and stepwise selection
*/
/* num= total runs to test one set of coefficients of predictors
count= number of predictors in the true model
nn= sample size */

proc printto log="T:\msxin.log"; ODS LISTING CLOSE;       /* turn off SAS-log, SAS-output */

data xinx; /* generate random seed to start this whole simulation from system time */
x=int(datetime());
start=int(1000000*ranuni(x));
seed9=int(1000000*ranuni(x));
call symput('start',start);
call symput('seed9',seed9); run;

%put ** Inside the macro: **; /* unmask the seeds for using in macro */
%put _user_;
%put ** In open code: **;
%put _user_;

data gam; /*Generate Beta from uniform distribution(0,1)*/
%Do i =1 %to 3;
%let gamma&i = ranuni(&seed9); %end;
gamma1=&gamma1;
gamma2=&gamma2;
gamma3=&gamma3;

%let open=&start;
%do k=1 %to &num;        *loop for n times runs of step(2)-step(5);
%let open = int(ranuni(&open)*1000000);
%Do I =1 %to 4;              *loop for generate seeds for later use;
%let seed&I = int(ranuni(&open)*1000000); %end;

data test0; set gam; seed1=&seed1; seed2=&seed2; seed3=&seed3; seed4=&seed4;

data predictor;     /* generate a set of predictors for nn=m observations, the predictors from uniform
distribution(0,1) */
set test0;
seed=seed4;
%do _n_=1 %to &count;
```

```
SEED = mod( SEED * 397204094, 2**23-1 );
%do i = 1 %to &nn;
w=ranuni(seed); output;
%end; %end;
data ww1; set predictor; w1=w; if _N_<=&nn then output; drop w; run;
data ww2; set predictor; w2=w; if &nn<_N_ and _N_<=&nn*2 then output; drop w; run;
data ww3; set predictor; w3=w; if &nn*2<_N_ then output; drop w; run;
data test00; merge ww1 ww2 ww3;

data test001; /*generate a set of observations, with error term distributed as normal(0,sigmasquare=)*/
set test00;
y=w1*gamma1+ gamma2*w2 + gamma3*w3 + &zz*rannor(seed1);

data test002; set test001; /* Create noise terms */
w12=w1*w2; w13=w1*w3; w23=w2*w3; wa=w1*w1; wb=w2*w2; wc=w3*w3;
data test003; set test002;
w12=w12; w13=w13; w23=w23; w11=wa; w22=wb; w33=wc; drop wa wb wc;
data test000&k; set test003;

proc reg data=test000&k; /* perform backward elimination with significant level alpha= */
model y = w1 w2 w3 w12 w13 w23 w11 w22 w33/selection=backward SLS=0.05;
ods output SelectionSummary=aa&k;

proc reg data=test000&k; /* perform stepwise selection with significant level alpha= */
model y = w1 w2 w3 w12 w13 w23 w11 w22 w33/selection=stepwise sle=.05 sls=.05;
ods output SelectionSummary=aaa&k;

*model y = w1 w2 w3 w12 w13 w23 w11 w22 w33/vif;
/*
ods html;
ods graphics on;
PROC CORR DATA=sss PLOTS = MATRIX PLOTS=scatter;
   VAR y w1 w2 w3 w12 w13 w23 w11 w22 w33;
TITLE 'Correlation calculations using PROC CORR'; RUN;
*/
proc printto; /*ODS LISTING;run; /* turn on SAS-log, SAS-output */

%end; run;
%mend test;

/*
ods trace on;
proc reg data=test0002;
model y = w1 w2 w3 w12 w13 w23 w11 w22 w33/selection=backward SLS=0.05;
ods output SelectionSummary=aa; run;
ods trace off;
*/

%macro test1(num,nums,nn);

/* summarize the performence of backward elimination
num= total runs to test one set of coefficients of predictors */
proc printto log="T:\msxin.log"; ODS LISTING CLOSE;        /* turn off SAS-log, SAS-output */

%do k=1 %to &num;
```

```
data bb&k; set aa&k; /* pick value from final results, approach one */
if varremoved in ('w1','w2','w3') then xin=1;
if varremoved in ('w12','w13','w23','w11','w22','w33') then xin=2;
proc freq data=bb&k;TABLE xin / OUT=cc&k;run;
data dd&k; set cc&k;
keep xin score1 count_true;
if xin=1 then score1=count;else delete;
count_true=3-score1;
data ee&k; set cc&k;
keep xin score1 count_true;
if (xin^=1 and PERCENT=100) then xin=1; else delete; count=0; score1=0; count_true=3-score1;
data gg&k; set cc&k;
keep xin score2 count_noise;
if xin=2 then score2=6-count;else delete; count_noise=score2;
data ff&k; merge dd&k ee&k gg&k;
score=(2*(3-count_true)+count_noise)/3;

data hh&k; /* pick value from final results, approach two */
keep cp modelrsquare;
set aa&k;
by dependent;
if last.dependent then output;else delete;run;

data ii&k; /* combine approach one and two */
merge hh&k ff&k;
cpp=(cp /(score1+score2+1)); count=score1+score2; %end;

/* For stepwise part */
%do r=1 %to &nums;
data bbb&r; set aaa&r; /* pick value from final results, approach one */
if varentered in ('w1','w2','w3') then xin=1;
if varentered in ('w12','w13','w23','w11','w22','w33') then xin=2;
if varremoved in ('w1','w2','w3') then xin=3;
if varremoved in ('w12','w13','w23','w11','w22','w33') then xin=4;
proc freq data=bbb&r;TABLE xin / OUT=ccc&r;run;
data ccc&r; merge ccc&r qq; by xin; run;

data ddd&r; set ccc&r;
if xin=1 then score111=count;else delete;
if score111="." then score111=0;
data dddd&r; set ccc&r;
if xin=3 then score1111=count; else delete;
if score1111="." then score1111=0;
data fff&r; merge ddd&r dddd&r;
keep xin score111 score1111 count_true_s;
count_true_s=score111-score1111;

proc print data=hhh1;run;

data eee&r; set ccc&r;
if xin=2 then score222=count; else delete;
if score222="." then score222=0;
data eeee&r; set ccc&r;
if xin=4 then score2222=count; else delete;
if score2222="." then score2222=0;
```

```
data ggg&r; merge eee&r eeee&r;
keep score222 score2222 count_noise_s;
count_noise_s=score222-score2222;
run;

data hhh&r; merge fff&r ggg&r;
score_step=(2*(3-count_true_s)+count_noise_s)/3;
%end;

data jj; /* arrange the summary */
set ii1 ii2 ii3 ii4 ii5 ii6 ii7 ii8 ii9 ii10 ii11 ii12 ii13 ii14 ii15
ii16 ii17 ii18 ii19 ii20 ii21 ii22 ii23 ii24 ii25 ii26 ii27 ii28 ii29 ii30

Time = datetime(); format time datetime16.;
data ll;
set hhh1 hhh2 hhh3 hhh4 hhh5 hhh6 hhh7 hhh8 hhh9 hhh10 hhh11 hhh12 hhh13 hhh14 hhh15
hhh16 hhh17 hhh18 hhh19 hhh20 hhh21 hhh22 hhh23 hhh24 hhh25 hhh26 hhh27 hhh28 hhh29 hhh30;

data mm;
nn=&nn;
keep count_true count_noise cpp score score_step modelrsquare cp nn;* time;
retain count_true count_noise cpp score score_step modelrsquare cp nn;* time;
merge jj ll;
proc sort data=mm; by score cpp score_step;
proc printto;/*ODS LISTING;run; /* turn on SAS-log, SAS-output */
/*proc print data=jj;run;*/

%mend test1;

proc datasets lib=work kill nolist memtype=data; quit;
/*Identify the variable Id for custom selection*/
data qq; input xin @@;
cards;
1 2 3 4
;run;
%test(30,3,150,0.1)
%test1(30,30,150)

ods html body='T:\msxin\xin.html';
proc means data=mm mean std p25 p50 p75 min max nolabels;
  VAR cpp score score_step modelrsquare cp count_true count_noise;
  output out=xin; run;

DATA xin; set xin; keep _STAT_ cpp score score_step modelrsquare cp count_true count_noise;
run;
ods html close;

proc print data=xin; run;

/* Combine summary from step(1) - step(6) */
libname MS 'T:\xin\library';
proc append base=ms.summary_normal data=xin;
proc print data=ms.summary_normal;run;

%macro cls();
```

```
        DM 'ODSRESULTS' CLEAR EDITOR;  ODS HTML CLOSE;
        DM 'CLEAR LOG; CLEAR OUTPUT; PGM OFF' LISTING; *EXPLORER;
%mend  cls;
%cls;
```

SAS Code: For Figure 2.14

```
proc G3D data=ms.diff;
SCATTER evari*ssize=diff;
run;
```

Sas Code: For Table 2.3

```
proc glm data=ms.union;
class ssize evari rela;
model  score = ssize evari rela;
run;
proc glm data=ms.union;
class ssize evari rela;
model  score = ssize|evari rela|ssize evari|rela;
run;
proc glm data=ms.union;
class ssize evari rela;
model  score = ssize|evari|rela;
run;
```

Sas Code: For Table 2.5, Figure 2.1, 2.2, 2.3

```
proc glm data=ms.union;
class ssize evari rela;
model  score = ssize evari rela;
output out=Residuals student=rr r=Residual p=Predicted stdp=stdp stdi=stdi stdr=stdr; run;
proc gplot data=Residuals;
        plot rr*score /vref=0;run;
proc gplot data=Residuals;
        plot rr*Predicted /vref=0;run;
proc univariate data=Residuals plot normal;
        var rr; run;
```

Sas Code: For Figure 2.

```
libname MS 'H:\Desktop\Master sas\';

data aa; set ms.union;
if ssize>0 then color="a";
if rela="Relat" then oo=1;
if rela="unrel" then oo=3;
keep score ssize evari rela color oo;
```

```
data bb; set ms.union;

if ssize=15 then sssize=23;
if ssize=30 then sssize=40;
if ssize=70 then sssize=80;
if ssize=150 then sssize=160;
if evari=.01 then eevari=.06;
if evari=.1 then eevari=.15;
if evari=.3 then eevari=.35;
if evari=.5 then eevari=.55;
if evari=1 then eevari=1.05;
drop ssize evari;
data cc; set bb;
ssize=sssize;
evari=eevari;
score=score_s;
if ssize>0 then color="b";
drop sssize eevari;
if rela="Relat" then oo=2;
if rela="unrel" then oo=4;
keep score ssize evari rela color oo;
data dd; set aa cc;

proc sort data=dd; by ssize;

axis1 order = (0 15 30 45 60 75 90 105 120 135) label = ( height= 1.25 'sample size')
value=(t=1 h=1 j=c "n=15" h=1 j=c "(Back)"
t=2 h=1 j=c "n=15" h=1 j=c "(Step)"
t=3 h=1 j=c "n=30" h=1 j=c "(Back)"
t=4 h=1 j=c "n=30" h=1 j=c "(Step)"
t=5 h=1 j=c "n=70" h=1 j=c "(Back)"
t=6 h=1 j=c "n=70" h=1 j=c "(Step)"
t=7 h=1 j=c "n=150" h=1 j=c "(Back)"
t=8 h=1 j=c "n=150" h=1 j=c "(Step)"
);
axis2 label = (height = 1.25 'Step score');
symbol value = dot height = .5 color=yellow;
proc boxplot data = dd;
plot score * ssize/ boxstyle=skeletal haxis=axis1 vaxis=axis2 cboxfill = (color) cboxes = BL ;
insetgroup mean(6.4) min(6.4) max(6.4) STDDEV(6.4) range/
header = 'Back_score and Step_score by ssize' height=2.5 pos = top cfill = BIGY;
run;

proc sort data=dd; by evari;

axis1 order = (0.01 0.06 0.11 0.16 0.21 0.26 0.31 0.36 0.41 0.46)
label = ( height= 1.25 'error variance')
value=(
t=1 h=1 j=c "0.01" h=1 j=c "(Back)"
t=2 h=1 j=c "0.01" h=1 j=c "(Step)"
t=3 h=1 j=c "0.1" h=1 j=c "(Back)"
t=4 h=1 j=c "0.1" h=1 j=c "(Step)"
t=5 h=1 j=c "0.3" h=1 j=c "(Back)"
t=6 h=1 j=c "0.3" h=1 j=c "(Step)"
```

```
t=7 h=1 j=c "0.5" h=1 j=c "(Back)"
t=8 h=1 j=c "0.5" h=1 j=c "(Step)"
t=9 h=1 j=c "1.0" h=1 j=c "(Back)"
t=10 h=1 j=c "1.0" h=1 j=c "(Step)"
);
axis2 label = (height = 1.25 'Step score');
symbol value = dot height = .5 color=yellow;
proc boxplot data = dd;
plot score * evari/ boxstyle=skeletal haxis=axis1 vaxis=axis2 cboxfill = (color) cboxes = BL ;
insetgroup mean(6.4) min(6.4) max(6.4) STDDEV(6.4) range/
header = 'Back_score and Step_score by ssize' height=2.5 pos = top cfill = BIGY;
run;

proc sort data=dd; by oo;

axis1 order = (1 to 4 by 1)
label = ( height= 1.25 'extra predictor')
value=(
t=1 h=1 j=c "related" h=1 j=c "(Back)"
t=2 h=1 j=c "related" h=1 j=c "(Step)"
t=3 h=1 j=c "unrelated" h=1 j=c "(Back)"
t=4 h=1 j=c "unrelated" h=1 j=c "(Step)"
);
axis2 label = (height = 1.25 'Step score');
symbol value = dot height = .5 color=yellow;
proc boxplot data = dd;
plot score * oo/ boxstyle=skeletal haxis=axis1 vaxis=axis2 cboxfill = (color) cboxes = BL ;
insetgroup mean(6.4) min(6.4) max(6.4) STDDEV(6.4) range/
header = 'score by type of extra predictor' height=2.5 pos = top cfill = BIGY;
run;
```

SAS Code: For Table 2.8, 2.9

```
libname MS 'H:\Desktop\Master sas\';
data test01;
set ms.union;
if rela="Relat" then do x1=1; x2=0; end;
if rela="unrel" then do x2=1; x1=0; end;
inter=ssize*evari;
*proc print data=test01;run;

data test01; set test01;
ssize=ssize-70;
evari=evari-0.3;
proc reg data=test01;
model  score = ssize evari x1;
run;


proc reg data=test01;
model  score = ssize evari inter x1;
run;
```

Sas code: For Figure 2.10

```
libname MS 'H:\Desktop\Master sas\';
proc glm data=ms.diff;
class ssize evari rela;
model diff = ssize evari rela;
output out=Residuals student=rr r=Residual p=Predicted stdp=stdp stdi=stdi stdr=stdr; run;
proc gplot data=Residuals;
        plot rr*diff /vref=0;run;
proc gplot data=Residuals;
        plot rr*Predicted /vref=0;run;
proc univariate data=Residuals plot normal;
        var rr; run;
```

Sas code: For Figure 2.11, 2.12, 2.13

```
/*score diff*/

axis1 order = (1 to 150 by 15) label = ( height= 1.25 'sample size');
axis2 label = (height = 1.25 'Score Diff');
symbol value = dot height = 0.5 ;
proc sort data=ms.diff; by ssize;
proc boxplot data = ms.diff;
plot diff * ssize/ haxis=axis1 vaxis=axis2 cboxfill = TAN cboxes = BL ;
insetgroup mean min max STDDEV/ header = 'Score by_difference by ssize' pos = top cfill = YELLOW ;
run;

axis3 order = (0.01 to 1 by 0.1) label = (height = 1.25 'error variance') minor = (number = 1) ;
axis2 label = (height = 1.25 'Score Diff') ;
symbol value = dot height = 0.5 ;
proc sort data=ms.diff; by evari;
proc boxplot data = ms.diff;
plot diff * evari/ vaxis=axis2 haxis=axis3 cboxfill = TAN cboxes = BL ;
insetgroup mean min max STDDEV/ header = 'Score by_difference by evari' pos = top cfill = YELLOW ;
run;

axis4 label = (height = 1.25 'relationship') minor = (number = 1) ;
axis2 label = (height = 1.25 'Score Diff') ;
symbol value = dot height = 0.5 ;
proc sort data=ms.diff; by rela;
proc boxplot data = ms.diff;
plot diff * rela/ vaxis=axis2 haxis=axis4 cboxfill = TAN cboxes = BL ;
insetgroup mean min max STDDEV/ header = 'Score by_difference by rela' pos = top cfill = YELLOW ;
run;
```

Sas code: For Table 2.15

```
libname MS 'H:\Desktop\Master sas\';
data test01;
set ms.diff;
if rela="Relat" then do x1=1; x2=0; end;
if rela="unrel" then do x2=1; x1=0; end;
*proc print data=test01;run;
```

```
data test01; set test01;
ssize=ssize-70;
evari=evari-0.3;
inter=ssize*evari;

proc reg data=test01;
model  diff = ssize evari x1;
run;
proc reg data=test01;
model  diff = ssize evari inter x1;
run;
```

Appendix B

Information for Table 3.2: General information

Class Level Information		
Class	**Levels**	**Values**
ssize	4	15 30 70 150
evari	5	0.01 0.1 0.3 0.5 1
rela	2	Relat unrel

Number of Observations Read	40000
Number of Observations Used	40000

Information for model: Main effects only

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	8	16779.17446	2097.39681	22017.6	<.0001
Error	39991	3809.53654	0.09526		
Corrected Total	39999	20588.71100			

R-Square	Coeff Var	Root MSE	score Mean
0.814970	29.02942	0.308642	1.063204

Source	DF	Type I SS	Mean Square	F Value	Pr > F
ssize	3	2090.16958	696.72319	7313.92	<.0001
evari	4	12824.32850	3206.08213	33656.2	<.0001
rela	1	1864.67637	1864.67637	19574.6	<.0001

Information for model: With 2-way interactions

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	27	17690.26878	655.19514	9035.70	<.0001
Error	39972	2898.44222	0.07251		
Corrected Total	39999	20588.71100			

R-Square	Coeff Var	Root MSE	score Mean
0.859222	25.32724	0.269280	1.063204

Source	DF	Type I SS	Mean Square	F Value	Pr > F
ssize	3	2090.16958	696.72319	9608.41	<.0001
evari	4	12824.32850	3206.08213	44214.6	<.0001
ssize*evari	12	429.14966	35.76247	493.20	<.0001
rela	1	1864.67637	1864.67637	25715.5	<.0001
ssize*rela	3	23.67957	7.89319	108.85	<.0001
evari*rela	4	458.26509	114.56627	1579.97	<.0001

Information for model: With 3-way interactions

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	39	18047.45814	462.75534	7276.61	<.0001
Error	39960	2541.25286	0.06359		
Corrected Total	39999	20588.71100			

R-Square	Coeff Var	Root MSE	score Mean
0.876571	23.71890	0.252180	1.063204

Source	DF	Type III SS	Mean Square	F Value	Pr > F
ssize	3	2090.16958	696.72319	10955.6	<.0001
evari	4	12824.32850	3206.08213	50414.1	<.0001
ssize*evari	12	429.14966	35.76247	562.35	<.0001
rela	1	1864.67637	1864.67637	29321.2	<.0001
ssize*rela	3	23.67957	7.89319	124.12	<.0001
evari*rela	4	458.26509	114.56627	1801.50	<.0001
ssize*evari*rela	12	357.18936	29.76578	468.05	<.0001

Pairwise comparison on error-variance main effect based on backward score.

evari	score LSMEAN	LSMEAN Number
0.01	0.20012587	1
0.1	0.66434164	2
0.3	1.19870574	3
0.5	1.47434817	4
1	1.77849819	5

Least Squares Means for effect evari Pr > |t| for H0: LSMean(i)=LSMean(j) Dependent Variable: score					
i/j	**1**	**2**	**3**	**4**	**5**
1		<.0001	<.0001	<.0001	<.0001

Least Squares Means for effect evari Pr > |t| for H0: LSMean(i)=LSMean(j) Dependent Variable: score					
i/j	**1**	**2**	**3**	**4**	**5**
2	<.0001		<.0001	<.0001	<.0001
3	<.0001	<.0001		<.0001	<.0001
4	<.0001	<.0001	<.0001		<.0001
5	<.0001	<.0001	<.0001	<.0001	

Pairwise comparison on related-unrelated main effect based on backward score.

rela	score LSMEAN	H0:LSMean1=LSMean2
		Pr > |t|
Relat	1.27911341	<.0001
unrel	0.84729443	

Ordered Lsmeans for 40 combinations of the three factors based on backward score.

Obs	Effect	Dependent	ssize	evari	rela	LSMean	LSMeanNumber
1	ssize_evari_rela	score	150	0.01	unrel	0.11056667	32
2	ssize_evari_rela	score	70	0.01	unrel	0.12408889	22
3	ssize_evari_rela	score	30	0.01	unrel	0.15109617	12
4	ssize_evari_rela	score	150	0.01	Relat	0.15315556	31
5	ssize_evari_rela	score	70	0.01	Relat	0.16808889	21
6	ssize_evari_rela	score	150	0.1	unrel	0.21876667	34
7	ssize_evari_rela	score	30	0.01	Relat	0.22466667	11

Obs	Effect	Dependent	ssize	evari	rela	LSMean	LSMeanNumber
8	ssize_evari_rela	score	15	0.01	unrel	0.22974576	2
9	ssize_evari_rela	score	70	0.1	unrel	0.28370000	24
10	ssize_evari_rela	score	30	0.1	unrel	0.39704215	14
11	ssize_evari_rela	score	150	0.3	unrel	0.43650000	36
12	ssize_evari_rela	score	15	0.01	Relat	0.43959836	1
13	ssize_evari_rela	score	150	0.1	Relat	0.46095556	33
14	ssize_evari_rela	score	70	0.3	unrel	0.60526667	26
15	ssize_evari_rela	score	150	0.5	unrel	0.66715326	38
16	ssize_evari_rela	score	15	0.1	unrel	0.66829833	4
17	ssize_evari_rela	score	70	0.1	Relat	0.67441111	23
18	ssize_evari_rela	score	70	0.5	unrel	0.95625556	28
19	ssize_evari_rela	score	30	0.3	unrel	0.96206667	16
20	ssize_evari_rela	score	30	0.1	Relat	1.07096667	13
21	ssize_evari_rela	score	150	0.3	Relat	1.11898889	35
22	ssize_evari_rela	score	150	1	unrel	1.21451111	40
23	ssize_evari_rela	score	30	0.5	unrel	1.40730728	18
24	ssize_evari_rela	score	15	0.3	unrel	1.43092874	6
25	ssize_evari_rela	score	70	0.3	Relat	1.44183333	25
26	ssize_evari_rela	score	150	0.5	Relat	1.47675556	37
27	ssize_evari_rela	score	15	0.1	Relat	1.54059267	3
28	ssize_evari_rela	score	70	1	unrel	1.58543333	30
29	ssize_evari_rela	score	70	0.5	Relat	1.71046667	27

Obs	Effect	Dependent	ssize	evari	rela	LSMean	LSMeanNumber
30	ssize_evari_rela	score	30	0.3	Relat	1.71634444	15
31	ssize_evari_rela	score	15	0.5	unrel	1.74421954	8
32	ssize_evari_rela	score	150	1	Relat	1.82225556	39
33	ssize_evari_rela	score	30	1	unrel	1.82834100	20
34	ssize_evari_rela	score	15	0.3	Relat	1.87771719	5
35	ssize_evari_rela	score	30	0.5	Relat	1.88575747	17
36	ssize_evari_rela	score	70	1	Relat	1.91536628	29
37	ssize_evari_rela	score	15	1	unrel	1.92460079	10
38	ssize_evari_rela	score	15	0.5	Relat	1.94687002	7
39	ssize_evari_rela	score	30	1	Relat	1.96719387	19
40	ssize_evari_rela	score	15	1	Relat	1.97028355	9

Stepwise selection example:

The REG Procedure
Model: MODEL1
Dependent Variable: y

Number of Observations Read	30
Number of Observations Used	30

Stepwise Selection: Step 1

Variable w23 Entered: R-Square = 0.7146 and C(p) = 14.1745

Analysis of Variance					
Source	**DF**	**Sum of Squares**	**Mean Square**	**F Value**	**Pr > F**

Analysis of Variance					
Source	**DF**	**Sum of Squares**	**Mean Square**	**F Value**	**Pr > F**
Model	1	0.77783	0.77783	70.12	<.0001
Error	28	0.31058	0.01109		
Corrected Total	29	1.08840			

Variable	Parameter Estimate	Standard Error	Type II SS	F Value	Pr > F
Intercept	0.38717	0.03061	1.77419	159.95	<.0001
w23	0.74023	0.08840	0.77783	70.12	<.0001

Stepwise Selection: Step 2

Variable w12 Entered: R-Square = 0.7859 and C(p) = 6.1499

Analysis of Variance					
Source	**DF**	**Sum of Squares**	**Mean Square**	**F Value**	**Pr > F**
Model	2	0.85532	0.42766	49.54	<.0001
Error	27	0.23308	0.00863		
Corrected Total	29	1.08840			

Variable	Parameter Estimate	Standard Error	Type II SS	F Value	Pr > F
Intercept	0.35057	0.02964	1.20744	139.87	<.0001
w12	0.25621	0.08551	0.07750	8.98	0.0058
w23	0.62906	0.08636	0.45806	53.06	<.0001

All variables left in the model are significant at the 0.0500 level.

No other variable met the 0.0500 significance level for entry into the model.

Summary of Stepwise Selection								
Step	**Variable Entered**	**Variable Removed**	**Number Vars In**	**Partial R-Square**	**Model R-Square**	**C(p)**	**F Value**	**Pr > F**
1	w23		1	0.7146	0.7146	14.1745	70.12	<.0001
2	w12		2	0.0712	0.7859	6.1499	8.98	0.0058