DOKUZ EYLÜL UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

REGRESSION ESTIMATORS IN MEDIAN RANKED SET SAMPLING

by
Dilek BALIN

February, 2017
İZMİR

REGRESSION ESTIMATORS IN MEDIAN RANKED SET SAMPLING

A Thesis Submitted to the
Graduate School of Natural and Applied Sciences of Dokuz Eylül University
In Partial Fulfillment of the Requirements for the Master of Science in
Statistics, Statistics Program

by

Dilek BALIN

February, 2017
İZMİR

M.Sc THESIS EXAMINATION RESULT FORM

We have read the thesis entitled **"REGRESSION ESTIMATORS IN MEDIAN RANKED SET SAMPLING"** completed by **DİLEK BALIN** under supervision of **ASSOC. PROF. DR. NESLİHAN DEMİREL** and we certify that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Assoc. Prof. Dr. Neslihan DEMİREL

Supervisor

Asst. Prof. Dr. Senem VAHAPLAR

(Jury Member)

Asst. Prof. Dr. Banu ÖZGÜVEN

(Jury Member)

Prof.Dr. Emine İlknur CÖCEN
Director
Graduate School of Natural and Applied Sciences

ACKNOWLEDGEMENTS

I would like to express my deepest appreciation to my supervisor Assoc. Prof. Dr. Neslihan DEMİREL for her useful comments, support, patience. I am honored to be a student of her.

I would like to thank all of academic staff in Dokuz Eylül University Department of Statistics for their everlasting support during my studies. I would like to thank to Büşra SEVİNÇ and Yusuf Can SEVİL for their contributions.

Finally, I want to thank to my dear family for their endless support and patience during my studies. Besides I would like to thank my friends who gave me motiviation and supporting me in every way when I need.

Dilek BALIN

REGRESSION ESTIMATORS IN MEDIAN RANKED SET SAMPLING

ABSTRACT

Simple random sample is the most popular sampling method in literature. Ranked set sampling is proposed an alternative to random sampling for setting in which measurements are difficult or costly. Ranked set sampling is more efficient than simple random sampling for estimating the population mean. Median ranked set sampling is an alternative sampling method which reduces the errors in ranking. In this study, mean estimator and regression estimators are obtained by median ranked set sampling, simple random sampling and ranked set sampling methods. These estimators of biases and relative efficiencies are investigated for different set and cycle sizes with some distributions. This study shows that in some situations median ranked set sampling gives better results than its competitor designs.

Keywords: Ranked set sampling, median ranked set sampling, concomitant variable, errors in ranking, regression estimator, relative efficiency.

MEDYAN SIRA SAYILI KÜME ÖRNEKLEMESİNDE REGRESYON KESTİRİCİLERİ

ÖZ

Basit rasgele örnekleme yöntemi literatürde en çok bilinen örnekleme yöntemidir. Sıralı küme örneklemesi, ilgilenilen değişkenin ölçümünün zaman alıcı ve maliyetli olması durumunda, basit rasgele örnekleme yöntemine alternatif bir yöntemdir. Sıralı küme örneklemesi, kitle ortalamasının kestiriminde basit rasgele örneklemeden daha etkindir. Medyan sıralı küme örneklemesi, sıralı küme örneklemesinde sıralama yapılırken oluşabilecek hataları azaltabilmek amacı ile alternatif bir yöntem olarak geliştirilmiştir. Bu çalışmada basit rasgele örnekleme, sıralı küme örneklemesi ve medyan sıralı küme örneklemesi yöntemleri için ortalama kestiricisi ve regresyon kestiricileri elde edilmiştir. Bu kestiricilerin yansızlık ve göreli etkinlikleri, farklı küme ve döngü sayıları bakımından çeşitli dağılımlar üzerinde incelenmiştir. İlgilenilen kestiriciler için medyan sıralı küme örneklemesinin bazı durumlarda diğer yöntemlere göre daha iyi sonuçlar verdiği gösterilmiştir.

Anahtar kelimeler: Sıralı küme örneklemesi, medyan sıra sayılı küme örneklemesi, yardımcı değişken, sıralama hataları, regresyon kestiricileri, göreli etkinlik.

CONTENTS

Page

THESIS EXAMINATION RESULT FORM ii
ACKNOWLEDGEMENTS iii
ABSTRACT iv
ÖZ v
LIST OF FIGURES vii
LIST OF TABLES viii

CHAPTER ONE - INTRODUCTION 1

CHAPTER TWO – SIMPLE RANDOM SAMPLING, RANKED SET SAMPLING AND MEDIAN RANKED SET SAMPLING 3

2.1 Simple Random Sampling 3
2.2 Ranked Set Sampling 4
2.3 Median Ranked Set Sampling 8

CHAPTER THREE – SIMULATION STUDY 12

CHAPTER FOUR - CONCLUSIONS 19

REFERENCES 20

APPENDICES 22

LIST OF FIGURES

Page

Figure 2.1 Changing of ranked to units the selected from population 4
Figure 2.2 How RSS can provide better information about the parameter 5
Figure 3.1 Histogram of distributions which are used in the simulation study 12

LIST OF TABLES

Page

Table 2.1 RSS procedure with k=3 and m=3 7
Table 2.3 MRSS procedure for *k*=3 and *m*=1 9
Table 2.4 MRSS procedure for *k*=4 and *m*=3. 10
Table 3.1 Used values of set sizes, cycle sizes, sample size and ρ values in simulation study. 13
Table 3.2 Relative efficiencies of regression model mean square error (MSE) 16
Table 3.3 Relative efficiencies of the sample mean estimator ($\overline{Y}$) 17
Table 3.4 The relative efficiency of the regression coefficient $\hat{\beta}_1$ 18

CHAPTER ONE
INTRODUCTION

Sample is a subset containing the characteristics of a larger population that researchers to estimate inference about the population. Many environmental, ecological studies and data collection procedure can be improved by using a special sampling design in statistics analysis. The sample should be a representation of the general population. Consequently, there are several sampling techniques accessible in literature.

The most common method of sampling is known as simple random sampling (SRS). In this method, each member of the population has an equal chance of being selected as subject. If this method is used in practical application, sometimes sampling does not represent the population and measurement of the related variable can be time-consuming and cost is high.Therefore, a sampling scheme known as ranked set sampling (RSS) can be applied in such situation to reduce cost and to increase efficiency (Chen, 2000).

Ranked set sampling (RSS) was first suggested by McIntyre (1952) to solve in the problem of estimating yield per acre of pasture in Australia. Halls and Dell (1966) examined forage yield area for the estimating the average herbage. Takahasi and Wakimoto (1968) developed the necessary mathematical theory upon McIntyre's suggestion. These studies show the variance of the mean of the simple random sample has a higher variance of the mean of the ranked set sample and RSS estimator is an unbiased estimator with the same sample size. RSS is more efficient than SRS. Dell and Clutter (1972) assumed that errors in ranking reduce the efficiency of the RSS mean relative to the simple random sampling (SRS) mean. Stokes (1977) taken into account this case, when the variable of interest is difficult to measure and order, but there is a concomitant variable which is correlated with the variable of interest that can be used to judge the order of the variable of interest.

After the development of the theory of RSS, applications increased in this direction. Stokes (1976) studied the estimation of the population variance using of a concomitant variable for ranking. Stokes (1980a, 1980b) estimated the variance using judgment ordered ranked set samples set samples in bivariate normal distribution.

Patil et al. (1993) showed the estimator of the population mean using RSS is considerably more efficient than the SRS regression estimator unless the correlation between the variable of interest and the concomitant variable is more than 0.85. Johnson et al. (1993) considered the variance estimation, the effect of ranking error when ranking is based on concomitant variables and other issues (Muttlak,1998). For a review and more bibliography on the RSS, see Patil et al. (1993).

In the literature, there are many studies modification in the process of RSS. Muttlak (1997) suggested using the median ranked set sampling (MRSS) method for estimating the population mean.MRSS is used to estimate the population mean of a variable of interest when ranking is based on a concomitant variable. The regression estimator uses an auxiliary variable to estimate the population mean of the variable of interest (Muttlak,1998). This method is proven to reduce errors in ranking and to increase the efficiency of the estimator.

The aim of this thesis is to shown the sampling methods such as simple random sampling, ranked set sampling, median ranked set sampling and compare the estimation of mean, mean square error of model and regression coefficient of $\hat{\beta}_1$ Biases and relative efficiency of these estimators are investigated for different set and cycle sizes with some distributions. This study shows that in some situations MRSS gives better results than its competitor designs.This study is arranged as follows. In second chapter involves introduction of sampling methods such as simple random sampling, ranked set sampling and median ranked set sampling descriptions and procedures. The simulation results are presented in third chapter. These results are shown that the comparison of SRS, RSS with MRSS. Finally, the last chapter presents the conclusion remarks.

CHAPTER TWO

SIMPLE RANDOM SAMPLING, RANKED SET SAMPLING AND MEDIAN RANKED SET SAMPLING

2.1 Simple Random Sampling

Simple random sampling (SRS) design is one the basic sampling method. SRS is the most popular probability sampling method. SRS is a subset of a population in which each member of the subset has an equal probability of being chosen. The most important characteristic of simple random sampling is simple to use and it does not have any selection bias. On the other hand, SRS design may not be desirable in settings where a full measurement of a unit is the consuming or expensive while ranking a few set of units is easy and cheap.

One of the great advantages of simple random sampling method is that it needs only a minimum knowledge of the study group of population in advance. One of the major disadvantages of simple random sampling method is that it cannot be employed where the units of the population are heterogeneous in nature. Although simple random samplings are a common research method, they are expensive to use, extremely time consuming and difficult to organize.

The estimator of population mean is calculated as in Equation 2.1.

$$\overline{X}_{SRS} = \frac{\sum_{i=1}^{n} X_i}{n} \tag{2.1}$$

The variance of the sample mean estimator is calculated as in Equation 2.2.

$$V\left(\overline{X}_{SRS}\right) = \frac{\sigma^2}{n} \tag{2.2}$$

2.2 Ranked Set Sampling

Ranked set sampling (RSS) is an efficient and preferred method for estimating the mean of population. Especially RSS is in the case of the measurement of the related variable time-consuming and cost is high, considering the response to the measure of a variable to another variable is a method of ranking obtained by the estimator.

RSS was first suggested by McIntyre in 1952 to estimate the yield of pasture in Australia. RSS is more efficient than commonly used simple random sampling (SRS) method. This method provides unbiased and reliable results. In fact for most practical applications, it is not easy to rank the units without errors in ranking. There will be a loss in efficiency RSS will give a larger variance due to the errors in ranking the units.

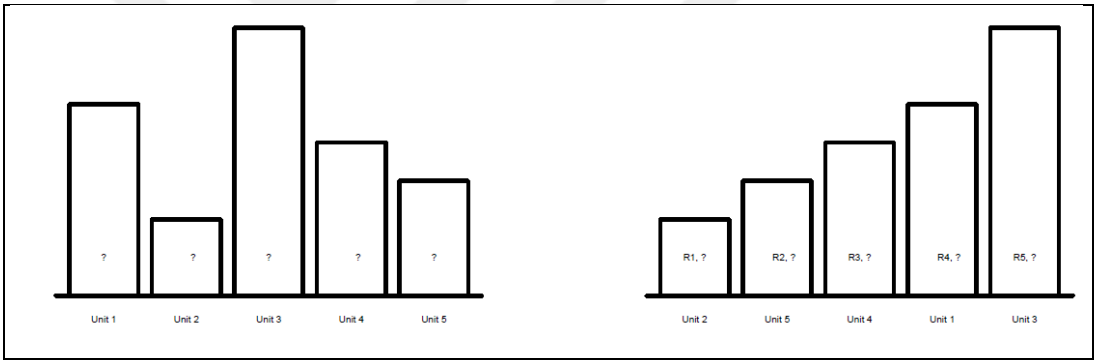


Figure 2.1 Changing of ranked to units the selected from population

The aim of RSS is illustrated in Figure 2.1. RSS method is cost effective in most situations than commonly used SRS method. In SRS method, there may be no information about the part of interest in the population. So that this method is proved to choose from ranked each set a unit.

RSS is illustrated in Figure 2.2. This figure showed that ranked set sampling method's effective to estimate population estimator from taken ranked each set a unit upon sampling process. As we can see in Figure 2.2, the selection of observations by RSS from a population is more likely to span the full range of values in the population than observations obtained by simple random sampling. In this manner,

the estimator of a parameter can provide better information about the parameter (Wolfe, 2012).

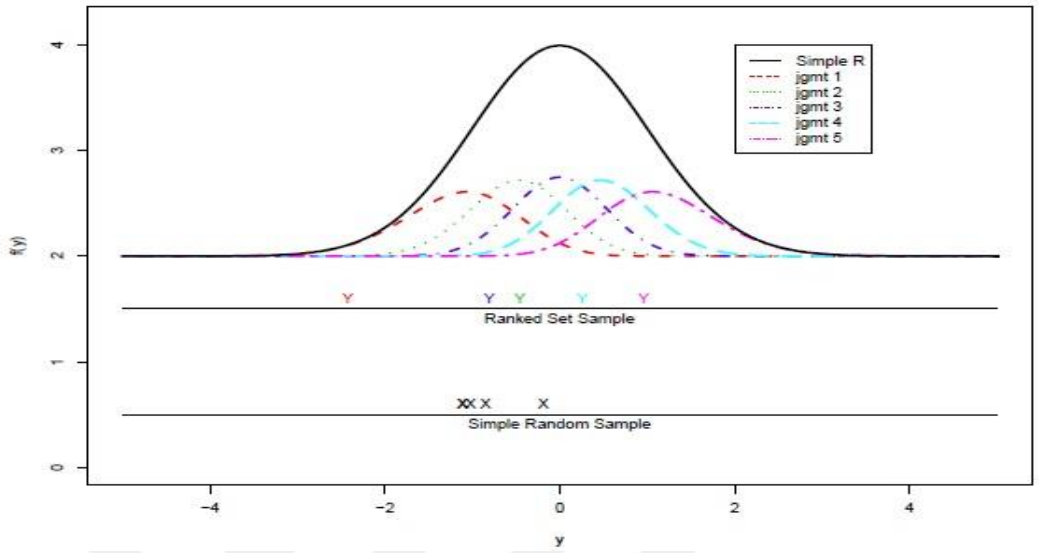


Figure 2.2 How RSS can provide better information abaout the parameter (Wolfe, 2012).

RSS method's advantages are wanted to summary. We can say this situations as below:

- Efficient
- Unbiased
- Cost effective
- Time effective
- Easy to apply

Let *k* is defined which denotes set size and *m* is defined which denotes cycle size. RSS method's steps are defined below:

Step 1: First,randomly select k^2 units randomly from the population.

Step 2: Allocate the k^2 selected units as randomly into *k* sets.

Step 3: Rank the units with in each set with related to variable of interest. This ranking process may be judgment or a concomitant variable correlated with the

variable of interest. This measurement of concomitant variable must be cheaper and must not be take time-consuming.

Step 4: Select a sample by containing the smallest ranked unit from the first set,then second smallest ranked unit from the second set.The procedure is developing until the largest ranked unit from the last set.

Step 5: Repeat the former steps for *m* cycles to obtain a sample of size $n=kxm$ observations.

The process of RSS for set size $k= 3$, cycle size $m=3$ is denoted in Table 2.1. Let randomly select nine units from the population and allocate nine units as randomly into three sets. We rank the units with in each set with related to variable of interest. Then we select a sample by containing the smallest ranked unit from the first set,then second smallest ranked unit from the second set. The procedure is developing until the third ranked unit from the last set. This process can be repeated for three cycles to obtain a sample size nine observations.

Table 2.1 RSS procedure with k=3 and m=3

Cycle 1
$\boldsymbol{X_{[1]1}} \leq X_{[2]1} \leq X_{[3]1} \rightarrow \boldsymbol{X_{[1]1}}$
$X_{[1]1} \leq \boldsymbol{X_{[2]1}} \leq X_{[3]1} \rightarrow \boldsymbol{X_{[2]1}}$
$X_{[1]1} \leq X_{[2]1} \leq \boldsymbol{X_{[3]1}} \rightarrow \boldsymbol{X_{[3]1}}$
Cycle 2
$\boldsymbol{X_{[1]2}} \leq X_{[2]2} \leq X_{[3]2} \rightarrow \boldsymbol{X_{[1]2}}$
$X_{[1]2} \leq \boldsymbol{X_{[2]2}} \leq X_{[3]2} \rightarrow \boldsymbol{X_{[2]2}}$
$X_{[1]2} \leq X_{[2]2} \leq \boldsymbol{X_{[3]2}} \rightarrow \boldsymbol{X_{[3]2}}$
Cycle 3
$\boldsymbol{X_{[1]3}} \leq X_{[2]3} \leq X_{[3]3} \rightarrow \boldsymbol{X_{[1]3}}$
$X_{[1]3} \leq \boldsymbol{X_{[2]3}} \leq X_{[3]3} \rightarrow \boldsymbol{X_{[2]3}}$
$X_{[1]3} \leq X_{[2]3} \leq \boldsymbol{X_{[3]3}} \rightarrow \boldsymbol{X_{[3]3}}$

Consequently, there is a significant saving on the measurement costs. Thus, based on the measured ranked set sample, we can obtain unbiased estimators of population parameters, such as the mean, and for more than one cycle, the population variance (Al-Omari and Bouza, 2014). Additionally, the ranked set sampling is defined as ordering mechanism. This sampling scheme is suitable in situations where the ranking of observations can easily be done based on an auxiliary variable correlated with the variable of interest or any inexpensive method (Al-Omari and Zamanzade, 2014).

The unbiased estimator of population mean is defined as Equation in 2.3. For more information, see Takahasi and Wakimoto (1968). They proposed the population mean and developed mathematically theory of ranked set sample.

$$\overline{X}_{RSS} = \frac{1}{km}\sum_{i=1}^{k}\sum_{j=1}^{m} X_{[i]j} \tag{2.3}$$

The variance of RSS estimator can shown as below in Equation 2.4

$$V(\bar{X}_{RSS}) = \frac{1}{km^2}\sum_{i=1}^{k}\sum_{j=1}^{m} V\left(X_{[i]j}\right). \tag{2.4}$$

The variance of RSS estimator is equal or smaller than SRS's variance estimator. This situation can be shown as in Equation 2.5

$$V(\bar{X}_{SRS}) = V(\bar{X}_{RSS}) + \frac{1}{k^2}\sum_{i=1}^{k}\left(\mu_{[i]} - \mu\right)^2. \tag{2.5}$$

The estimator of variance SRS and RSS's efficiency defined as in Equation 2.6

$$eff\left(V(\bar{X}_{SRS}), V(\bar{X}_{RSS})\right) = \frac{V(\bar{X}_{SRS})}{V(\bar{X}_{RSS})} = \frac{V(\bar{X}_{RSS}) + \frac{1}{k^2}\sum_{i=1}^{k}\left(\mu_{[i]} - \mu\right)^2}{V(\bar{X}_{RSS})} > 1 \tag{2.6}$$

It is clear that the mean estimator of RSS is cost effective in some situations and more efficient than commonly used SRS. Ranked set sample is more effective than simple random sample on representing the population.

2.3 Median Ranked Set Sampling

Many researchers have developed new sampling methods by making changes on RSS method. One of the modified methods is median ranked set sampling (MRSS).

Muttlak (1997) suggested median ranked set sampling method which reduces the errors in ranking. MRSS is used to estimate the population mean of the variable of interest when ranking is based on a concomitant variable. The difference of the MRSS is to select the middle unit of the sample and measure it. MRSS method can be easily employed in the field and save some time in performing the ranking of the units corresponding respect to the variable of interest (Hajighorbani and Saba, 2012). MRSS method provides to increase the efficiency of the estimator over RSS with perfect ranking for some probability distributions.

The steps of MRSS are given below:

Step 1: First, randomly select *k* random sets each of size *k*.

Step 2: Rank the units in each set by judgment.

Step 3: For odd set size, the median value is selected from each set as in $\left(\frac{k+1}{2}\right)^{th}$.

For even set size, $\left(\frac{k}{2}\right)^{th}$ smallest unit is selected from first $\left(\frac{k}{2}\right)$sets and $\left(\frac{k+2}{2}\right)^{th}$ smallest unit from last $\left(\frac{k}{2}\right)$sets.

Step 4: Repeat the first three steps for *m* times to get a sample of size *n=kxm* units.

As an illustration in Table 2.2, consider the set size *k*=3 and cycle size *m*=1. The other illustration in Table 2.3, consider the set size *k*=4 and cycle size *m*=3. Let $X_{[i]m}$ denote a median ranked set sample unit in the m^{th} cycle on rank *i*, where each row denotes a judgment-ordered sample, and the units selected for quantitative analysis are bolded.

Table 2.2 MRSS procedure for *k*=3 and *m*=1

Cycle 1
$X_{[1]1} \leq \boldsymbol{X_{[2]1}} \leq X_{[3]1} \rightarrow \boldsymbol{X_{[1]1}}$
$X_{[1]1} \leq \boldsymbol{X_{[2]1}} \leq X_{[3]1} \rightarrow \boldsymbol{X_{[2]1}}$
$X_{[1]1} \leq \boldsymbol{X_{[2]1}} \leq X_{[3]1} \rightarrow \boldsymbol{X_{[3]1}}$

Table 2.3 MRSS procedure for k=4 and m=3

Cycle 1
$X_{[1]1} \leq \boldsymbol{X_{[2]1}} \leq X_{[3]1} \leq X_{[4]1} \rightarrow \boldsymbol{X_{[1]1}}$
$X_{[1]1} \leq \boldsymbol{X_{[2]1}} \leq X_{[3]1} \leq X_{[4]1} \rightarrow \boldsymbol{X_{[2]1}}$
$X_{[1]1} \leq X_{[2]1} \leq \boldsymbol{X_{[3]1}} \leq X_{[4]1} \rightarrow \boldsymbol{X_{[3]1}}$
$X_{[1]1} \leq X_{[2]1} \leq \boldsymbol{X_{[3]1}} \leq X_{[4]1} \rightarrow \boldsymbol{X_{[4]1}}$
Cycle 2
$X_{[1]2} \leq \boldsymbol{X_{[2]2}} \leq X_{[3]2} \leq X_{[4]2} \rightarrow \boldsymbol{X_{[1]2}}$
$X_{[1]2} \leq \boldsymbol{X_{[2]2}} \leq X_{[3]2} \leq X_{[4]2} \rightarrow \boldsymbol{X_{[2]2}}$
$X_{[1]2} \leq X_{[2]2} \leq \boldsymbol{X_{[3]2}} \leq X_{[4]2} \rightarrow \boldsymbol{X_{[3]2}}$
$X_{[1]2} \leq X_{[2]2} \leq \boldsymbol{X_{[3]2}} \leq X_{[4]2} \rightarrow \boldsymbol{X_{[4]2}}$
Cycle 3
$X_{[1]3} \leq \boldsymbol{X_{[2]3}} \leq X_{[3]3} \leq X_{[4]3} \rightarrow \boldsymbol{X_{[1]3}}$
$X_{[1]3} \leq \boldsymbol{X_{[2]3}} \leq X_{[3]3} \leq X_{[4]3} \rightarrow \boldsymbol{X_{[2]3}}$
$X_{[1]3} \leq X_{[2]3} \leq \boldsymbol{X_{[3]3}} \leq X_{[4]3} \rightarrow \boldsymbol{X_{[3]3}}$
$X_{[1]3} \leq X_{[2]3} \leq \boldsymbol{X_{[3]3}} \leq X_{[4]3} \rightarrow \boldsymbol{X_{[4]3}}$

The mean and the variance estimators of MRSS for odd set size are denoted by $\overline{X}_{MRSS1}$ and $V\left(\overline{X}_{MRSS1}\right)$,respectively. Estimators with one cycle are given below:

$$\overline{X}_{MRSS1} = \frac{1}{k}\sum_{i=1}^{k} X_{i((k+1)/2)} \tag{2.7}$$

$$V\left(\overline{X}_{MRSS1}\right) = \frac{1}{k^2}\sum_{i=1}^{k} V\left(X_{i((k+1)/2)}\right) \tag{2.8}$$

where k can be any positive odd integer.

The mean and the variance estimators of MRSS for even set size are denoted by $\overline{X}_{MRSS2}$ and $V\left(\overline{X}_{MRSS2}\right)$, respectively. Estimators with one cycle are given below:

$$\overline{X}_{MRSS2} = \frac{1}{k}\left(\sum_{i=1}^{k/2} X_{i(k/2)} + \sum_{i=(k/2+1)}^{k} X_{i((k+2)/2)}\right) \tag{2.9}$$

$$V(\overline{X}_{MRSS2}) = \frac{1}{k^2}\left(\sum_{i=1}^{k/2} V\left(X_{i(k/2)}\right) + \sum_{i=(k/2)+1}^{k} V\left(X_{i((k+2)/2)}\right)\right) \tag{2.10}$$

where k can be any positive even integer.

The procedure which is mentioned above can be repeated m times to obtain a sample of size kxm for estimating the population mean. It was shown that each of the variances of these estimators, $V\left(\overline{X}_{MRSS1}\right)$ and $V\left(\overline{X}_{MRSS2}\right)$is less than the variance for the simple random sample mean, $V(\overline{X}_{SRS})$, if the underlying distribution is symmetric about the population mean μ. For more detail, especially asymmetric distributions see Shaibu(1999).

CHAPTER THREE
SIMULATION STUDY

In this chapter, we show a simulation study with different distribution, set size, cycle size and correlation coefficient (ρ) values. Then we discuss and compare relative efficiencies of mean square errors of the regression models, relative efficiencies of sample mean estimators and relative efficiencies of the regression coefficient $\hat{\beta}_1$ for SRS, RSS and MRSS methods.

This study presents mean estimator and regression estimators are obtained by median ranked set sampling (MRSS), simple random sampling (SRS) and ranked set sampling (RSS) methods. Biases and relative efficiency of these estimators are investigated for different set and cycle sizes with some distributions.

Such as distribution functions are considered: normal(0,1), exponential(1), uniform(0,1) and gamma (5,1).

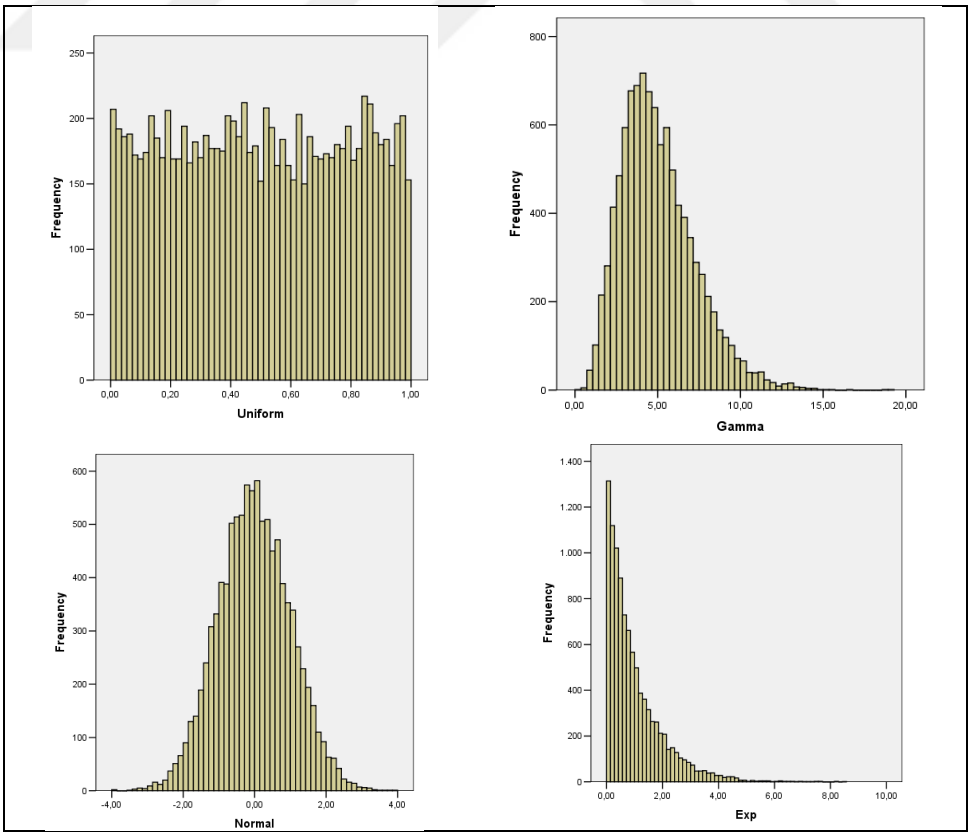


Figure 3.1 The histograms of distributions which are used in the simulation study

The simulation study is designed with R Project (the codes are given in Appendix) concluding the process given below:

1. Generate 10,000 observations which come from different four distributions and various parameters with different set size, cycle size and ρ values. This distributions: normal(0,1), exponential(1), uniform(0,1) and gamma(5,1).
2. Generate random error term (ε) from normal distribution with parameters N(0, σ_ε^2), and σ_ε^2 is calculated by using the equation of Dell&Clutter (1972)

$$\rho = \sqrt{1 - \frac{\sigma_e^2}{\sigma_y^2}} \tag{3.1}$$

   where correlation coefficient (ρ) =0.95 and 0.75.

3. We are interested in two different set sizes and cycle sizes. Set size *k*=3, 4 and cycle size *m*=3, 5.
4. Y_i is calculated by simple linear regression model where β_0 and β_1 equals 1.

$$Y_i = \beta_0 + \beta_1 X_i + \varepsilon_i \qquad (i=1,\ldots N) \tag{3.2}$$

5. Run the simulation 10,000 times for simple random sampling, ranked set sampling and median ranked set sampling

Table 3.1 Used values of set sizes, cycle sizes, sample size and ρ values in simulation study.

Set size	**Cycle size**	**ρ value**	**Sample size**
3	3	0.95	9
		0.75	9
	5	0.95	15
		0.75	15
4	3	0.95	12
		0.75	12
	5	0.95	20
		0.75	20

6. Biases and relative efficiency of these estimators are investigated for different set and cycle sizes with some distributions and compared SRS, RSS with MRSS methods.
7. Biases value of regression coefficient are calculated given below:

$$Bias = \beta_i - \hat{\beta}_i \qquad i = 0,1 \tag{3.3}$$

8. The relative efficiency of population mean estimator using MRSS is calculated with respect variable to corresponding estimator in SRS and RSS methods for relative efficiencies MSE of the regression model, the relative efficiency of the sample mean estimator and the relative efficiency of the regression coefficient $\hat{\beta}_1$ by equations are given below:

$$RE_1 = \frac{MSE_{SRS}}{MSE_{RSS}} \qquad RE_2 = \frac{MSE_{SRS}}{MSE_{MRSS}} \qquad RE_3 = \frac{MSE_{RSS}}{MSE_{MRSS}}$$

$$RE_4 = \frac{MSE_{\bar{Y}_{SRS}}}{MSE_{\bar{Y}_{RSS}}} \qquad RE_5 = \frac{MSE_{\bar{Y}_{SRS}}}{MSE_{\bar{Y}_{MRSS}}} \qquad RE_6 = \frac{MSE_{\bar{Y}_{RSS}}}{MSE_{\bar{Y}_{MRSS}}}$$

$$RE_7 = \frac{MSE_{\hat{\beta}_{1_{SRS}}}}{MSE_{\hat{\beta}_{1_{RSS}}}} \qquad RE_8 = \frac{MSE_{\hat{\beta}_{1_{SRS}}}}{MSE_{\hat{\beta}_{1_{MRSS}}}} \qquad RE_9 = \frac{MSE_{\hat{\beta}_{1_{RSS}}}}{MSE_{\hat{\beta}_{1_{MRSS}}}}$$

In this part, we shown that the results of simulation study according to SRS, RSS and MRSS methods. The tables of simulation study are given below:

1. The relative efficiencies MSE of the regression model

2. The relative efficiencies of the sample mean estimator

3. The relative efficiencies of the regression coefficient $\hat{\beta}_1$.

The simulation of results are compared for MRSS, RSS and SRS methods and analyzed this results in Table 3.2-3.4.the relative efficiencies MSE of the regression model, the relative efficiencies of the sample mean estimator and the relative efficiencies of the regression coefficient $\hat{\beta}_1$ tables are shown for different set size,

cycle size, ρ values and distributions different parameters such as normal (0,1),exponential(1),uniform(0,1) and gamma(5,1).

Table 3.2 shows that the relative efficiencies MSE of the regression models under four distributions for SRS, RSS and MRSS designs. As you can see in 3.2 Table, when ρ values decrease relative efficiency of MSE decrease especially symmetric distributions. MRSS gives better results than SRS and RSS for relative efficiencies of mean square errors of regression models especially under asymmetric distributions. MRSS gives worse results under symmetric distributions but results close to 1.

The relative efficiency of the sample mean estimator of the simulation study results are shown in Table 3.3. According the Table 3.3, it can be seen MRSS gives better results than SRS and RSS for relative efficiency of estimator sample mean under normal distribution. Same time, MRSS gives better results than SRS for relative efficiency of estimator sample mean under exponential, uniform and gamma distributions. When ρ values decrease relative efficiency of the sample mean estimator decrease especially symmetric distributions. RSS is more efficient than MRSS for relative efficiency of estimator sample mean. In addition MRSS with normal distribution is observed more effective than other distributions. Especially sample mean estimator of MRSS is more effective than sample mean estimator SRS.

The relative efficiency of the regression coefficient $\hat{\beta}_1$ results are shown in Table 3.4. As you can see in table, MRSS isn't efficient than SRS and RSS for regression coefficient $\hat{\beta}_1$. The results are close to 1 for different distributions. So that we can say between the methods are not differences because of that values of relative efficiency are close to 1 for regression coefficient $\hat{\beta}_1$.

Table 3.2 Relative efficiencies of regression model mean square error (MSE)

Set size	Cycle size	ρ	N(0,1)			U(0,1)			Exp (1)			Gamma(5,1)		
			RE_1	RE_2	RE_3	RE_1	RE_2	RE_3	RE_1	RE_2	RE_3	RE_1	RE_2	RE_3
3	3	0.95	0.989	1.013	1.024	1.000	0.967	0.967	0.986	1.041	1.056	0.990	0.995	1.005
		0.75	1.000	0.991	0.991	1.000	0.914	0.914	0.976	1.141	1.169	0.990	1.006	1.017
	5	0.95	0.998	1.008	1.009	1.000	0.961	0.961	0.992	1.044	1.053	0.999	1.006	1.006
		0.75	0.995	0.997	1.003	1.000	0.898	0.898	0.983	1.143	1.163	0.998	1.044	1.047
4	3	0.95	0.996	0.994	0.998	1.000	0.963	0.963	0.992	1.046	1.055	1.000	1.009	1.008
		0.75	1.006	0.984	0.978	1.000	0.894	0.894	0.990	1.158	1.171	0.999	1.031	1.033
	5	0.95	1.006	1.003	0.998	1.000	0.969	0.969	0.981	1.048	1.069	0.997	1.001	1.004
		0.75	1.004	1.007	1.002	1.000	0.903	0.903	0.997	1.152	1.156	0.998	1.026	1.028

Table 3.3 Relative efficiencies of the sample mean estimator ($\bar{Y}$)

Set size	Cycle size	ρ	N(0,1)			U(0,1)			Exp (1)			Gamma(5,1)		
			RE_4	RE_5	RE_6	RE_4	RE_5	RE_6	RE_4	RE_5	RE_6	RE_4	RE_5	RE_6
3	3	0.95	1.743	1.975	1.133	1.876	1.522	0.811	1.537	1.478	0.962	1.720	1.825	1.061
		0.75	1.302	1.404	1.078	1.563	1.255	0.803	1.364	1.374	1.007	1.467	1.562	1.064
	5	0.95	1.805	1.974	1.094	1.822	1.567	0.860	1.521	1.118	0.735	1.796	1.711	0.953
		0.75	1.357	1.419	1.045	1.552	1.285	0.828	1.356	1.141	0.842	1.430	1.390	0.972
4	3	0.95	2.090	2.353	1.126	2.210	1.798	0.813	1.769	1.455	0.823	2.029	2.034	1.002
		0.75	1.464	1.504	1.027	1.733	1.488	0.859	1.492	1.356	0.909	1.609	1.576	0.979
	5	0.95	2.046	2.408	1.177	2.200	1.828	0.831	1.810	1.049	0.580	2.092	1.857	0.888
		0.75	1.495	1.572	1.051	1.718	1.403	0.817	1.487	1.011	0.680	1.593	1.341	0.842

Table 3.4 Relative efficiencies of the regression coefficient ($\hat{\beta}_1$)

Set size	Cycle size	ρ	N(0,1)			U(0,1)			Exp (1)			Gamma(5,1)		
			RE_7	RE_8	RE_9	RE_7	RE_8	RE_9	RE_7	RE_8	RE_9	RE_7	RE_8	RE_9
3	3	0.95	1.009	0.976	0.968	1.003	0.895	0.892	0.989	0.963	0.974	0.998	0.969	0.971
		0.75	1.015	0.872	0.859	1.004	0.760	0.757	0.990	1.013	1.023	1.007	0.951	0.944
	5	0.95	1.004	0.993	0.989	0.999	0.900	0.901	0.994	0.992	0.998	0.998	0.974	0.977
		0.75	1.005	0.944	0.940	1.005	0.794	0.791	0.994	1.128	1.135	0.995	0.976	0.981
4	3	0.95	1.000	0.982	0.982	0.997	0.894	0.896	1.003	0.982	0.979	0.997	0.988	0.991
		0.75	1.008	0.922	0.915	1.015	0.785	0.774	1.002	1.113	1.111	1.005	0.961	0.957
	5	0.95	0.999	0.993	0.994	0.996	0.888	0.892	1.004	1.003	0.999	1.000	0.989	0.989
		0.75	1.002	0.950	0.948	1.009	0.798	0.790	0.998	1.162	1.164	0.995	0.975	0.981

CHAPTER FOUR

CONCLUSIONS

In this thesis, we are compared simple random sampling, ranked set sampling, median ranked set sampling with the estimation of mean, mean square error of model and regression coefficient of $\hat{\beta}_1$ Biases and relative efficiency of these estimators are investigated for different set and cycle sizes with some distributions. This thesis shows that in some situations MRSS gives better results than its competitor designs. In the simulation study, 10,000 samples are compared for some symmetric and asymmetric distributions such as normal, uniform, exponential and gamma. For all methods in comparison, we study with fix sample size. conclusions are given below.

Relative efficiencies of mean square errors of regression models are compared for SRS, RSS and MRSS. We can say that MRSS gives better results than SRS and RSS for relative efficiencies of mean square errors of regression models especially under asymmetric distributions. MRSS gives worse results under symmetric distributions but results close to 1.

Relative efficiency of estimator sample mean are examined for all sampling methods. MRSS gives better results than SRS and RSS for relative efficiency of estimator sample mean under normal distribution. Same time, MRSS gives better results than SRS for relative efficiency of estimator sample mean under exponential, uniform and gamma distributions. RSS is more efficient than MRSS for relative efficiency of the estimator of sample mean.

The relative efficiency of the regression coefficient $\hat{\beta}_1$ results are shown that MRSS is not efficient than SRS and RSS for regression coefficient $\hat{\beta}_1$. The results are close to 1 for different distributions. So that we can say there are not differences between sampling methods.MRSS is not preferable for the regression coefficient $\hat{\beta}_1$.

REFERENCES

Al-Omari, A.I., & Bouza, C.N. (2014). Review of ranked set sampling: modifications and applications. *Revista Investigación Operacional*, *3,* 215-240.

Al-Omari, A.I., & Zamanzade, E. (2014). *New ranked set sampling for estimating the population mean and variance*. DOI:10.15672/HJMS.20159213166

Chen, Z. (2000). On ranked-set sample quantiles and their applications. *Journal of Statistical Planning and Inference*, *83*(1), 125-135.

Dell, T.R., & Clutter, J.L. (1972). Ranked set sampling theory with order statistics background. *Biometrics*, *28,* 545-555.

Hajighorbani, S., & Saba, R.A. (2012). Stratified median ranked set sampling: optimum and proportional allocations. *Journal of Statistical Research of Iran*, *9*(1), 87-102.

Halls, L.K., & Dell, T.R. (1966). Trial of ranked-set sampling for forage yields. *Forest Science*, *12*(1), 22-26.

Johnson, G.D., Patil, G.P., & Sinha, A.K. (1992). Ranked set sampling for vegetation research. *Abstracta Botanica*, *17,* 87-102.

McIntyre, G.A. (1952). A method for unbiased selective sampling using ranked sets. *Crop and Pasture Science*, *3*(4), 385-390.

Muttlak, H.A. (1997). Median ranked set sampling. *Journal of Applied Statistical Science*, *6*(4), 245-255.

Muttlak, H.A. (1998). Median ranked set sampling with concomitant variables and a comparison with ranked set sampling and regression estimators. *Environmetrics, 9*, 255-267.

Patil, G.P., Sinha, A.K., & Taille, C. (1993). Relative precision of ranked set sampling: A comparison with the regression estimator. *Environmetrics*. *4*(4), 399-412.

Shaibu, A. (1999). *Parameter estimation using median ranked set sampling*. P.hD Thesis, University of Saudi Arabia, Dhahran.

Stokes, S.L. (1976). *An investigation of the consequences of ranked set sampling*. P.hD Thesis, University of North Carolina, Chapel Hill, NC.

Stokes, S.L. (1977). Ranked set sampling with concomitant variables. *Communications in Statistics - Theory and Methods*, *6*(12), 1207–1211.

Stokes, S.L. (1980a). Inferences on the correlation coefficient in bivariate normal populations from ranked set samples. *Journal of the American Statistical Association*, *75*(372), 989-995.

Stokes, S.L. (1980b). Estimation of variance using judgment ordered ranked set samples. *Biometrics, 36*(1), 35-42.

Takahasi, K., & Wakimoto, K. (1968). On unbiased estimates of the population mean based on the sample stratified by means of ordering. *Annals of the Institute of Statistical Mathematics*, *20*(1), 1-31.

Wolfe, D.A. (2012). Ranked set sampling: its relevance and impact on statistical inference. *International Scholarly Research Network*. Retrieved May 12, 2015. from http://www.hindawi.com/journals/isrn/2012/568385/.

APPENDICES

R Project code;

For Normal Distribution:

```
library(MASS)
kv<-3            # kv=4
mv<-3            # mv=5
rov<-c(0.95,0.75)
simn=10000
N=10000
mx=0
my=0
vx=1
vy=1
s=1
m=1
t=1
mu<-c(mx,my)
var<-c(vx,vy)

SRS<- matrix(0, nrow=kv*mv, ncol=2)
colnames(SRS) <- c("x","y")

DataSRS=matrix(0,nrow=length(rov),ncol=16)
```

```
colnames(DataSRS)=c("kv","mv","ro","mean(Ysrs)","var(Ysrs)","b0_kepsrs",
"b1_kepsrs","var_b0_kepsrs","var_b1_kepsrs","MSE_SRS","biasy","biasb0",
"biasb1","biasykare","biasb0kare","biasb1kare")

finalsrs=matrix(0,length(rov),ncol=10)
colnames(finalsrs)=c("kv","mv","ro","meanysrsp","varysrsp","popsrsbe0",
"popsrsbe1","popvar_b0_kepsrs","popvar_b1_kepsrs","popMSE_SRS")

RSS<- matrix(0, nrow=kv, ncol=kv*2)
colnames(RSS)<-rep(c("x","y"),kv)

DataRSS=matrix(0,nrow=length(rov),ncol=16)
colnames(DataRSS)=c("kv","mv","ro","mean(Yrss)","var(Yrss)","b0_keprss",
"b1_keprss","var_b0_keprss","var_b1_keprss","MSE_RSS","biasy","biasb0",
"biasb1","biasykare","biasb0kare","biasb1kare")

finalrss=matrix(0,length(rov),ncol=10)
colnames(finalrss)=c("kv","mv","ro","meanyrssp","varyrssp","poprssbe0",
"poprssbe1","popvar_b0_keprss","popvar_b1_keprss","popMSE_RSS")

DataMRSS=matrix(0,nrow=length(rov),ncol=16)
colnames(DataMRSS)=c("kv","mv","ro","mean(Yrm)","var(Yrm)","b0_kepmrss",
"b1_kepmrss","var_b0_kepmrss","var_b1_kepmrss","MSE_MRSS","biasy",
"biasb0","biasb1","biasykare","biasb0kare","biasb1kare")
finalmrss=matrix(0,length(rov),ncol=10)
colnames(finalmrss)=c("kv","mv","ro","meanymrssp","varymrssp","popmrssbe0",
```

```
"popmrssbe1","popvar_b0_kepmrss","popvar_b1_kepmrss","popMSE_MRSS")

for( ro in rov) {
Datasrs=matrix(0,nrow=simn,ncol=16)
colnames(Datasrs)=c("kv","mv","ro","mean(Ysrs)","var(Ysrs)","b0_kepsrs",
"b1_kepsrs","var_b0_kepsrs","var_b1_kepsrs","MSE_SRS","biasy","biasb0",
"biasb1","biasykare","biasb0kare","biasb1kare")
Datarss=matrix(0,nrow=simn,ncol=16)
colnames(Datarss)=c("kv","mv","ro","mean(Yrrss)","var(Yrrss)","b0_keprss",
"b1_keprss","var_b0_keprss","var_b1_keprss","MSE_RSS","biasy","biasb0",
"biasb1","biasykare","biasb0kare","biasb1kare")
Datamrss=matrix(0,nrow=simn,ncol=16)
colnames(Datamrss)=c("kv","mv","ro","mean(Yrm)","var(Yrm)","b0_kepmrss",
"b1_kepmrss","var_b0_kepmrss","var_b1_kepmrss","MSE_MRSS","biasy",
"biasb0","biasb1","biasykare","biasb0kare","biasb1kare")

covxy<- ro*sqrt(vx)*sqrt(vy)
sigma<-matrix(c(vx, covxy,covxy ,vy),nrow=2,ncol=2,byrow=T)

data<-mvrnorm(N,mu,sigma)
colnames(data)<-c("x","y")
x=data[,1]
y=data[,2]
model<-lm(y~x)
ep=model$residuals
xms<-cbind(1,x)
```

```
for (sim in 1:simn) {
for (ss in (1:(kv*mv))){
vsrs<-data[sample(nrow(data),size=1,replace=TRUE),]
SRS[ss,]<-vsrs
}
n=kv*mv
Xsrs<-cbind(1,SRS[,1])
Ysrs<-as.matrix(SRS[,2])
Ysrsmean<-colMeans(Ysrs)
betasrs<-solve(t(Xsrs)%*%Xsrs )%*%t(Xsrs)%*%Ysrs
Ykepsrs<-Xsrs%*%betasrs
Ykepsrsmean<-colMeans(Ykepsrs)
residsrs<-Ysrs-Ykepsrs
varesrs<-var(residsrs)
msesrs<-as.numeric((t(residsrs)%*%residsrs)/(n-2))
Skarebetasrs<-as.numeric(msesrs) * solve(t(Xsrs)%*%Xsrs)

popbetas<-matrix(0,length(rov),2)
colnames(popbetas)<-c("popsrsbe0","popsrsbe1")
meanysrspop<-matrix(0,length(rov),2)
colnames(meanysrspop)<-c("meanysrsp","varysrsp")

srsbe<-solve(t(xms)%*%xms )%*%t(xms)%*% y
Ykeppop<-xms%*%srsbe
varepsrs<-var(ep)
mseps<-as.numeric((t(ep)%*%ep)/(N-2))
```

```
Skarebetaps<-mseps*solve(t(xms)%*%xms)
#Sbetaps<-sqrt(Skarebetaps)
#sbetasp0= Sbetaps[1,1]
#sbetasp1= Sbetaps[2,2]
popbetas[s,]<-cbind(srsbe[1,],srsbe[2,])
##print(popbetas)
meanysrsp<-mean(y)
varysrsp<-var(y)
meanysrspop[s,]<-cbind(meanysrsp,varysrsp)
##print(meanysrspop)
v=c(kv,mv,ro,meanysrsp,varysrsp,srsbe[1,],srsbe[2,],Skarebetaps[1,1],
Skarebetaps[2,2],mseps)
finalsrs[s,]=v
srsby=mean(Ysrs)-mean(data[,2])
srsbb0=betasrs[1,1]-popbetas[1,1]
srsbb1=betasrs[2,1]-popbetas[2,1]
srsbykare=srsby^2
srsbb0kare=srsbb0^2
srsbb1kare=srsbb1^2
db=c(kv,mv,ro,mean(Ysrs),var(Ysrs),betasrs[1,1],betasrs[2,1],Skarebetasrs[1,1],
Skarebetasrs[2,2],msesrs,srsby,srsbb0,srsbb1,srsbykare,srsbb0kare,srsbb1kare)
Datasrs[sim,]=db
DataSRS[s,]=colMeans(Datasrs)
colnames(DataSRS)=c("kv","mv","ro",
"mean(Ysrs)","var(Ysrs)","b0_kepsrs","b1_kepsrs","var_b0_kepsrs",
"var_b1_kepsrs","MSE_SRS","biasy","biasb0","biasb1","biasykare","biasb0kare",
```

```
"biasb1kare")
}
poprss<-cbind(x,y)
xmr<-cbind(1,x)
for (sim in 1:simn) {
for (j in 1:mv) {
for (rs in 1:kv) {
A<-matrix(0,nrow=kv,ncol=2)
for(aa in (1:kv)) {
vrss<-poprss[sample(nrow(poprss),size=1,replace=TRUE),]
A[aa,]<-vrss
}
As<-A[sort.list(A[,1]), ]
RSS[rs,] <- as.vector(t(As))
}
Xrss<-matrix(0,kv,1)
zsü<-(-1)
for (rs in 1:kv) {
zsü<-zsü+2
Xrss[rs,]=RSS[rs,zsü]
}
Yrss<-matrix(0,kv,1)
zsü<-(0)
for (rs in 1:kv){
zsü<-zsü+2
Yrss[rs,]=RSS[rs,zsü]
```

```
}
Rrss<-cbind(Xrss,Yrss)
namerss<-paste("Rrss",j,sep="")
assign(namerss,Rrss)
}
if(mv==3){Rrss<-rbind(Rrss1,Rrss2,Rrss3)}
if(mv==5){ Rrss<-rbind(Rrss1,Rrss2,Rrss3,Rrss4,Rrss5)}
if(mv==10){Rrss<- rbind(Rrss1,Rrss2,Rrss3,Rrss4,Rrss5,Rrss6,Rrss7,Rrss8,
Rrss9,Rrss10)}
Xrrss<-cbind(1,Rrss[,1])
Yrrss<-as.matrix(Rrss[,2])
Yrrssmean<-colMeans(Yrrss)
betarss<-solve(t(Xrrss)%*%Xrrss )%*%t(Xrrss)%*%Yrrss
Ykeprss<-Xrrss%*%betarss
Ykeprssmean<-colMeans(Ykeprss)
residrss<-Yrrss-Ykeprss
hatrss<-Xrrss%*%solve(t(Xrrss)%*%Xrrss)%*%t(Xrrss)
mserss<-as.numeric((t(residrss)%*%residrss)/(n-2))
Skarebetarss<-as.numeric(mserss) * solve(t(Xrrss)%*%Xrrss)
popbetar<-matrix(0,length(rov),2)
colnames(popbetar)<-c("poprssbe0","poprssbe1")
meanyrsspop<-matrix(0,length(rov),2)
colnames(meanyrsspop)<-c("meanyrssp","varyrssp")
rssbe<-solve(t(xmr)%*%xmr )%*%t(xmr)%*%y
Ykeppop<-xmr%*%rssbe
vareprss<-var(ep)
```

```
msepr<-as.numeric((t(ep)%*%ep)/(N-2))
#Sbetapr<-sqrt(Skarebetapr)
#sbetapr0= Sbetapr[1,1]
#sbetapr1= Sbetapr[2,2]
popbetar[m,]<-cbind(rssbe[1,],rssbe[2,])
##print(popbetar)
meanyrssp<-mean(y)
varyrssp<-var(y)
meanyrsspop[m,]<-cbind(meanyrssp,varyrssp)
##print(meanyrsspop)
a=c(kv,mv,ro,meanyrssp,varyrssp,rssbe[1,],rssbe[2,],Skarebetapr[1,1],
Skarebetapr[2,2],msepr)
finalrss[m,]=a
rssby=mean(Yrrss)-mean(poprss[,2])
rssbb0=betarss[1,1]-popbetar[1,1]
rssbb1=betarss[2,1]-popbetar[2,1]
rssbykare=rssby^2
rssbb0kare=rssbb0^2
rssbb1kare=rssbb1^2
d=c(kv,mv,ro,mean(Yrrss),var(Yrrss),betarss[1,1],betarss[2,1],Skarebetarss[1,1],
Skarebetarss[2,2],mserss,rssby,rssbb0,rssbb1,rssbykare,rssbb0kare,rssbb1kare)
Datarss[sim,]=d
DataRSS[m,]=colMeans(Datarss)
colnames(DataRSS)=c("kv","mv","ro",
"mean(Yrrss)","var(Yrrss)","b0_keprss","b1_keprss","var_b0_keprss",
"var_b1_keprss","MSE_RSS","biasy","biasb0","biasb1","biasykare","biasb0kare",
```

```
"biasb1kare")
}
popmrss<-cbind(x,y)
xmm<-cbind(1,x)
for (sim in 1:simn) {
for (p in 1:mv) {
Sm<-matrix(0,kv,2)
colnames(Sm)<-c("x","y")
Zm<-matrix(0,kv,kv*2)
colnames(Zm)<-rep(c("x","y"),kv)
for (zsa in (1:kv)){
for (ssa in (1:kv)) {
vm<-popmrss[sample(nrow(popmrss),size=1,replace=TRUE),]
Sm[ssa,]<-vm
}
SSm<-Sm[sort.list(Sm[,1]), ]
Zm[zsa,] <- as.vector(t(SSm))
}
Px<-matrix(0,kv,kv)
zp=1
for(px in (1:kv)) {
Px[,px]<-Zm[,zp]
zp=zp+2
}
Py<-matrix(0,kv,kv)
zp=2
```

```
for(py in (1:kv)) {
Py[,py]<-Zm[,zp]
zp=zp+2
}
if (kv%%2==1) {
X<-matrix(0,kv,1)
ad=1
for (ab in 1:(kv)){
X[ab,]<-Px[ad,((kv+1)/2)]
ad=ad+1
} }
else {
X3<-matrix(0,kv/2,1)
ab=1
for (ad in 1:(kv/2)){
X3[ab,]<-Px[ad,(kv/2)]
ab=ab+1
}
X4<-matrix(0,kv/2,1)
bb=1
for (bd in ((kv/2)+1):kv){
X4[bb,]<-Px[bd,((kv+2)/2)]
bb=bb+1
X<-rbind(X3,X4)
}
}
```

```
if (kv%%2==1) {
Y<-matrix(0,kv,1)
ad=1
for (ab in 1:(kv)){
Y[ab,]<-Py[ad,((kv+1)/2)]
ad=ad+1
} }
else {
Y3<-matrix(0,kv/2,1)
ab=1
for (ad in 1:(kv/2)){
Y3[ab,]<-Py[ad,(kv/2)]
ab=ab+1
}
Y4<-matrix(0,kv/2,1)
bb=1
for (bd in ((kv/2)+1):kv){
Y4[bb,]<-Py[bd,((kv+2)/2)]
bb=bb+1
Y<-rbind(Y3,Y4)
}
}
Mrss<-cbind(X,Y)

name<-paste("Mrss",p,sep="")
assign(name,Mrss)
```

```
}
if(mv==3){Mrss<-rbind(Mrss1,Mrss2,Mrss3)}
if(mv==5){Mrss<-rbind(Mrss1,Mrss2,Mrss3,Mrss4,Mrss5)}
if(mv==10){Mrss<-rbind(Mrss1,Mrss2,Mrss3,Mrss4,Mrss5,Mrss6,Mrss7,Mrss8,
Mrss9,Mrss10)}
Xrm<-cbind(1,Mrss[,1])
Yrm<-as.matrix(Mrss[,2])
Yrmmean<-colMeans(Yrm)
betam<-solve(t(Xrm)%*%Xrm )%*%t(Xrm)%*%Yrm
Ykepm<-Xrm%*%betam
Ykepmean<-colMeans(Ykepm)
resm<-Yrm-Ykepm
hatm<-Xrm%*%solve(t(Xrm)%*%Xrm)%*%t(Xrm)
msem<-as.numeric((t(resm)%*%resm)/(n-2))
Skarebetam<-msem* solve(t(Xrm)%*%Xrm)
popbetam<-matrix(0,length(rov),2)
colnames(popbetam)<-c("popmrssbe0","popmrssbe1")
meanymrsspop<-matrix(0,length(rov),2)
colnames(meanymrsspop)<-c("meanymrssp","varymrssp")
bem<-solve(t(xmm)%*%xmm )%*%t(xmm)%*%y
Ykeppopm<-xmm%*%bem
varepmrss<-var(ep)
msepm<-as.numeric((t(ep)%*%ep)/(N-2))
Skarebetapm<-msepm * solve(t(xmm)%*%xmm)
#Sbetapm<-sqrt(Skarebetapm)
#sbetamp0= Sbetapm[1,1]
```

```
#sbetamp1= Sbetapm[2,2]

popbetam[t,]<-cbind(bem[1,],bem[2,])

##print(popbetam)

meanymrssp<-mean(y)

varymrssp<-var(y)

meanymrsspop[t,]<-cbind(meanymrssp,varymrssp)

##print(meanymrsspop)

b=c(kv,mv,ro,meanymrssp,varymrssp,bem[1,],bem[2,],Skarebetapm[1,1],
Skarebetapm[2,2],msepm)

finalmrss[t,]=b

mrssby=mean(Yrm)-mean(popmrss[,2])

mrssbb0=betam[1,1]-popbetam[1,1]

mrssbb1=betam[2,1]-popbetam[2,1]

mrssbykare=mrssby^2

mrssbb0kare=mrssbb0^2

mrssbb1kare=mrssbb1^2

f=c(kv,mv,ro,mean(Yrm),var(Yrm),betam[1,1],betam[2,1],Skarebetam[1,1],

Skarbetam[2,2],msem,mrssby,mrssbb0,mrssbb1,mrssbykare,mrssbb0kare,

mrssbb1kare)

Datamrss[sim,]=f

DataMRSS[t,]=colMeans(Datamrss)

colnames(DataMRSS)=c("kv","mv","ro","mean(Yrm)","var(Yrm)","b0_kepmrss",

"b1_kepmrss","var_b0_kepmrss","var_b1_kepmrss","MSE_MRSS","biasy","biasb0"
,"biasb1","biasykare","biasb0kare","biasb1kare")

}

s=s+1

m=m+1
```

```
t=t+1
}
re1<-DataSRS[,c(""MSE_SRS")]/DataRSS[,c(""MSE_MRSS")]
re2<-DataSRS[,c(""MSE_SRS")]/DataMRSS[,c(""MSE_MRSS")]
re3<-DataRSS[,c(""MSE_RSS")]/DataMRSS[,c(""MSE_MRSS")]
re4<-DataSRS[,c("biasykare")]/DataRSS[,c("biasykare")]
re5<-DataSRS[,c("biasykare")]/DataMRSS[,c("biasykare")]
re6<-DataRSS[,c("biasykare")]/DataMRSS[,c("biasykare")]
re7<-DataSRS[,c("biasb1kare")]/DataRSS[,c("biasb1kare")]
re8<-DataSRS[,c("biasb1kare")]/DataMRSS[,c("biasb1kare")]
re9<-DataRSS[,c("biasb1kare")]/DataMRSS[,c("biasb1kare")]
re_norm<-cbind(re1,re2,re3,re4,re5,re6,re7,re8,re9)
colnames(re_norm)<c("srs/rss","srs/mrss","rss/mrss","srsy/rssy","srsy/mrssy","rssy/
mrssy","srsb1/rssb1","srsb1/mrssb1","rssb1/mrssb1")
rownames(re_norm)<-c("ro=0.95","ro=0.85","ro=0.75")

#write.table(DataSRS,file="D:\\DataSRSnorm.csv",row.names=T,col.names=T,sep=
";",dec=",")
#write.table(DataRSS,file="D:\\DataRSSnorm.csv",row.names=T,col.names=T,sep=
";",dec=",")
#write.table(DataMRSS,file="D:\\DataMRSSnorm.csv",row.names=T,col.names=T,
sep=";",dec=",")

write.table(DataSRS,file="D:\\DataSRSnorm.csv",row.names=T,col.names=F,sep=";
",dec=",",append=T)
write.table(DataRSS,file="D:\\DataRSSnorm.csv",row.names=T,col.names=F,sep=";
",dec=",",append=T)
write.table(DataMRSS,file="D:\\DataMRSSnorm.csv",row.names=T,col.names=F,se
p=";",dec=",",append=T)
```

```
write.table(re_norm,file="D://re_norm.csv",row.names=T,col.names=F,sep=";",dec=
",",append=T)
```

For Uniform Distribution:

```
y<-runif(10000,0,1)

vary<-var(y)

varev=(1-((ro)^2))*vary

ep<-rnorm(10000,0,sd=sqrt(varev))

x<-y-1-ep

data<-cbind(x,y)

xms<-cbind(1,x)
```

For Exponential Distribution:

```
y<-rexp(10000,1)

vary<-var(y)

varev=(1-((ro)^2))*vary

ep<-rnorm(10000,0,sd=sqrt(varev))

x<-y-1-ep

data<-cbind(x,y)

xms<-cbind(1,x)
```

For Gamma Distribution:

```
y<-rexp(10000,5,1)

vary<-var(y)

varev=(1-((ro)^2))*vary

ep<-rnorm(10000,0,sd=sqrt(varev))
```

```
x<-y-1-ep

data<-cbind(x,y)

xms<-cbind(1,x)
```