COMPARISON BETWEEN RATIO ESTIMATION BASED ON SAMPLING WITH AND WITHOUT REPLACEMENT

By:

Taejoo Grand Chae

Department of Statistics

Oklahoma State University

Stillwater, Oklahoma

December 4^{th}, 2013

Submitted to the faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
Master of Science

COMPARISON BETWEEN RATIO ESTIMATION BASED ON SAMPLING WITH AND WITHOUT REPLACEMENT

Thesis Proposal Approved:

Dr. Ibrahim Ahmad

Advisor

Committee Members:

Dr. Joshua Habigar

Dr. Lan Zhu

Name-: TAEJOO GRAND CHAE

Major Field: STATISTICS

Acknowledgment

I would like to give my appreciation toward to my advisor, Dr. Ahmad. He always helped me whenever I got stuck with a certain problems, and answered my questions flawlessly.

I would like to thank to my committee members, Dr. Lan Zhu and Dr. Joshua Habiger for their help and contributions to my research.

Not just my committee members, I also like to give my gratitude to Dr. Brenda Masters, Dr. Daiho Uhm, and Dr. Mark Payton for giving me a guide for my future plan. Thanks to Dr. Mindy McCann and Dr. Carla Goad for giving me a chance to graduate. Thanks to Renee Flasch and Fran Mihura for giving me a support of cookies. Thanks to people in Statistics department giving me some helps.

I would like to thank my mother Seong Hae Park who is in Korea right now. She always supported me without any doubt. Also, Thanks to my father Seong San Chae. I would stopped my study if he did not gave me a full support.

Thanks to people who gave me a lot of help, too. Love you all.

Contents

List of Figures

Chapter 1

INTRODUCTION

The ratio estimation is a regular and easy method to compare samples from 2 distributions. It is also a way to improve estimates when there are auxiliary data available.

In this paper, populations from 3 different distributions, Normal(X), Log-Normal(Y), and Gamma(Z), were generated. Using three sampling methods, Simple Random Sampling With Replacement based on Distinct Values(SRSWR(D)) were gathered first. For Simple Random Sampling Without Replacement(SRSWOR) and Simple Random Sampling With Replacement(SRSWR), samples of sizes equal to the distinct elements count of the SRSWR(D) were gathered from those three distributions. Then, three ratio estimation methods, usual ratio estimation, Quenouille's estimation, and Chakrabarty's estimation, were performed. After repetitions of those estimation methods, mean square errors of the estimates were computed for the above mentioned three estimation methods. The simulation performed with

R program, which is a free downloaded software.

The simulation results showed that the three ratio estimation methods with SRSWR(D) reduced Mean Square Error effectively. However, SRSWOR method also shown to effectively reduce the MSE compared with SRSWR(D) but slightly worse. Moreover, the simulation proved that SRSWR(D)'s MSE is closer to be normally distributed than the other two sampling methods' MSE. Further and more detailed conclusions offered in the conclusion chapter.

Chapter 2

REVIEW OF LITERATURE

2.1 Usual Ratio Estimation

To obtain Ratio Estimates, first step is to figure out what would be the population. There exists $Y_1, Y_2, ..., Y_N$, which are population of independent and identically distributed random variables from distribution with density function $f(y)$. Attached to Y_is, there is auxiliary variables, $X_1, X_2, ..., X_N$, where $\bar{X} = \frac{\sum_{i=1}^{N} X_i}{N}$ would usually be known. However, since the populations of the Y_is are not known, $\bar{Y}$ cannot be estimated from the population.

Because of that, samples are gathered from each Y_is and X_is. From the Y_is, a sample $y_1, y_2, ..., y_n$ is gathered and corresponding sample units, $x_1, x_2, ..., x_n$ are gathered, too. Then, the usual ratio estimate is given by $\frac{\bar{y}}{\bar{x}} = \hat{r}$, where $\bar{y} = \frac{\sum_{i=1}^{n} y_i}{n}$ and $\bar{x} = \frac{\sum_{i=1}^{n} x_i}{n}$, for the population ratio $R = \frac{\bar{Y}}{\bar{X}}$, according to

Cochran[3].

2.2 Sampling Methods

There are three sampling methods that we consider in this report: Simple Random Sampling With Replacement with Only Distinct Units(SRSWR(D)), Simple Random Sampling Without Replacement(SRSWOR), and Simple Random Sampling With Replacement(SRSWR). These methods of choosing y_is and x_is give different effects to the goodness of the estimate $\bar{r}$.

SRSRWR and SRSWOR are the most common sampling methods and they were well developed in the sampling textbooks. Examples of textbooks would be Cochran[3], Chaudhuri and Stenga[4], Fuller[5] and Sukhatme, et. al[10].

On the other hand, SRSWR(D) was receiving attention from several authors. Among them, Basu[2] and Raj and Khamis[9] developed basics of SRSWR(D) sampling method while Pathak[8], Korwar and Serfling[6] and Asok[1] did further research about it, discussing nice properties and gave exact formulas for the variance of mean estimate.

2.2.1 SRSWR(D)

To perform SRSWR(D), a SRSWR is performed and those are $y_1, ..., y_n$. Let there be auxiliary samples data $x_1, ..., x_n$ as well. From these samples, duplicate sample units were deleted and removed. Samples of distinct response would be $y_1^*, ..., y_{v_1}^*$ and corresponding auxiliary samples would be $x_1^*, ..., x_{v_2}^*$. Then, let

$\bar{y}_{v_1} = \frac{1}{v_1}\sum_{i=1}^{v_1} y_i^*$, where $y_1^* \neq y_2^* \neq ... \neq y_{v_1}^*$

and

$\bar{x}_{v_2} = \frac{1}{v_2}\sum_{i=1}^{v_2} x_i^*$, where $x_1^* \neq x_2^* \neq ... \neq x_{v_2}^*$.

Note that $v_1 \leq n$, and $v_2 \leq n$, and v_1 and v_2 are integer-valued random variables having same moments.

Using these facts, R can be estimated by $\hat{r}(v_1, v_2) = \frac{\bar{y}_{v_1}}{\bar{x}_{v_2}}$.

Also, from Basu[2], Raj and Khamis[9], Korwar and Serfling[6], Pathak[8] and Asok[1]. we have

$E(\bar{y}_{v_1}) = \bar{Y}$, $V(\bar{y}_{v_1}) = (E(\frac{1}{v_1} - \frac{1}{N}))S_Y^2$

and

$E(\bar{x}_{v_2}) = \bar{X}$, $V(\bar{x}_{v_2}) = (E(\frac{1}{v_2} - \frac{1}{N}))S_X^2$,

where

$S_Y^2 = \frac{1}{N-1}\sum_{i=1}^{N}(Y_i - \bar{Y})^2$

$S_X^2 = \frac{1}{N-1}\sum_{i=1}^{N}(X_i - \bar{X})^2$

$E(\frac{1}{v_1}) = E(\frac{1}{v_2}) = \frac{1}{N^n}\sum_{i=1}^{N} i^{(n-1)}$.

Note that $E(\frac{1}{v_1}) = E(\frac{1}{v_2})$ since the starting sample size are equal (n).

Approximate MSE of Ratio Estimate Based on SRSWR(D)

Theorem 2.2.1. *The MSE of $\hat{r}(v_1, v_2)$ is given by:*

$$E(\hat{r}(v_1, v_2) - R)^2 \simeq V(\bar{y}_{v_1}) + \frac{V(\bar{x}_{v_2})}{R} + (\frac{\bar{X}}{R})^2 V(\bar{y}_{v_1})V(\bar{x}_{v_2})$$

Proof: *We find it easier to work with:*

$$\delta \bar{y}_{v_1} = \frac{\bar{y}_{v_1} - \bar{Y}}{\bar{Y}}, \delta \bar{x}_{v_2} = \frac{\bar{x}_{v_2} - \bar{X}}{\bar{X}}$$

Thus, set

$$\bar{y}_{\hat{r}(v_1,v_2)} = \bar{X}\frac{\bar{y}_{v_1}}{\bar{x}_{v_2}}$$

Then,

$$\begin{aligned}(\bar{y}_{\hat{r}(v_1,v_2)} - \bar{Y})^2 \quad &\simeq \bar{Y}^2[(\delta\bar{y}_{v_1} - \delta\bar{x}_{v_2}) + (\delta\bar{x}_{v_2})^2(\delta\bar{y}_{v_1} - \delta\bar{x}_{v_2})^2] \\ &= \bar{Y}^2[(\delta\bar{y}_{v_1})^2 + (\delta\bar{x}_{v_2})^2 + (\delta\bar{y}_{v_1})^2(\delta\bar{x}_{v_2})^2]\end{aligned}$$

Hence,

$$(\hat{r}(v_1, v_2) - R)^2 \simeq (\delta\bar{y}_{v_1})^2 + (\delta\bar{x}_{v_2})^2 + (\delta\bar{y}_{v_1})^2(\delta\bar{x}_{v_2})^2$$

Taking expectation we get

$$E((\hat{r}(v_1, v_2) - R)^2) \simeq (E(\frac{1}{v}) - \frac{1}{N})(\frac{S_Y^2}{\bar{Y}^2} + \frac{S_X^2}{\bar{X}^2} + \frac{S_Y^2 S_X^2}{\bar{Y}^2\bar{X}^2})$$

This completes the proof.

2.2.2 SRSWOR

According to the Sukhatme et. al[10], the expected value of $\bar{y}$ of SRSWOR is:

$$E(\bar{y}_R) = \bar{X}E(\hat{R}) = \bar{X}E(\frac{\bar{y}}{\bar{x}})$$

and

$$\bar{y}_R = \bar{X}\frac{\bar{y}}{\bar{x}}$$

Then,

$$E(\frac{\bar{y}}{\bar{x}}) = \frac{1}{n\binom{N}{n}}\sum_{i=1}^{N} Y_i(\sum_{s\in S}\frac{1}{\bar{x}_s}) = \sum_{i=1}^{N} Y_i T_i^{(1)}$$

Using this fact,

$$MSE(\frac{\bar{y}}{\bar{x}}) = \sum_{i=1}^{N} Y_i^2 T_i^{(2)} + \sum\sum_{i\neq j}^{N} Y_i Y_j T_{ij}^{(2)} - 2\bar{Y}\sum_{i=1}^{N} Y_i T_i^{(1)} + \bar{Y}^2$$

where

$$T_i^{(1)} = \frac{\bar{X}}{n\binom{N}{n}}\sum_{s\in S, i\in S}\frac{1}{\bar{x}_s}$$

$$T_i^{(2)} = \frac{\bar{X}^2}{n^2\binom{N}{n}}\sum_{s\in S, i\in S}\frac{1}{\bar{x}_s^2}$$

$$T_{ij}^{(2)} = \frac{\bar{X}^2}{n^2\binom{N}{n}}\sum_{s\in S, i,j\in S}\frac{1}{\bar{x}_s^2}$$

For $s \in S$, S is number of population, and s is number of observed sample.

However, there is a newer version of equation that approximates MSE of SRSWOR.

Approximate MSE for ratio estimation in SRSWOR

The bias of $\hat{r} = \frac{\bar{y}}{\bar{x}}$ would be

$$Bias(\hat{r}) = R\frac{(N-n)}{(N-1)}\frac{1}{n}(S_X^2 - S_{XY})$$

where

$$S_{XY} = \frac{1}{N-1}\sum_{i=1}^{N}(X_i - \bar{X})(Y_i - \bar{Y})$$

and the MSE of $\hat{r}$ would be

$$MSE(\hat{r}) = R^2(\frac{N-n}{N-1})(\frac{1}{n})(S_X^2 - 2S_{XY} + S_Y^2)$$

2.3 Problem

All theories of approximating MSE showed a good results and my advisor Dr.Ahmad showed some new interesting theoretical results. However, solving MSE with these theories would be complicated especially when comparing sampling methods. For example, the MSE of SRSWR(D)'s equation is known, but the $E(\frac{1}{v})$ in MSE could not be figured out yet. Since population would not be known, thus he variance of X and variance of Y would not be known, too.

Other problems would be the theories for solving MSE need to have large samples, and SRSWR(D) has different sample sizes during repetitions of the process because of removal of repeated values from gathered samples.

The simulation of estimating MSE, therefore, would be necessary.

2.4 Types of Ratio Estimates used

Mussa[7] tested several ratio estimation methods. Quenouille's estimate and Chakrabarty's estimate are among those ratio estimation methods, and these estimation methods are picked with usual ratio estimate.

2.4.1 Usual ratio estimate

Cochran[3] mentioned that the usual ratio estimate of the population ratio $\frac{Y}{X}$ would be

$$\hat{r} = \frac{\bar{y}}{\bar{x}} = \frac{\frac{\sum_{i=1}^{n_1} y_i}{n_1}}{\frac{\sum_{i=1}^{n_2} x_i}{n_2}}.$$

2.4.2 Quenouille's ratio estimate

The Quenouille's ratio estimate is t_Q, which can be expressed as,

$$t_Q = 2\hat{r} - \frac{1}{2}(\hat{r}_1 + \hat{r}_2)$$

where $r = \frac{\bar{y}}{\bar{x}}$, and $\hat{r}_1$ and $\hat{r}_2$ would be ratios from two random halves of sample data, which denoted by $\hat{r}_i = \frac{\bar{y}_i}{\bar{x}_i}, i = 1, 2$

Note that we can instead define $\hat{r}_{-i} = \hat{r}$ based on all y's except y_i and all x's except x_i, and define

$$t_Q = n\hat{r} - \frac{1}{n}\sum_{i=1}^{n} \hat{r}_{-i}$$

which is generalized Quenouille's ratio estimate.

2.4.3 Chakrabarty ratio estimate

The Chakrabarty's estimators has two ratio estimators, $\bar{y}_{c1}$, which is simpler, and $\bar{y}_{c2}$, which has least bias compare with $\bar{y}_{c1}$. Those estimators can be expressed as:

$$\bar{y}_{c1} = (1 - W)\bar{y} + W\bar{y}_r = (1 - W)\bar{y} + W\frac{\bar{y}}{\bar{x}}\bar{X}$$

and

$$\bar{y}_{c2} = (1 - W)\bar{y} + Wt_Q\bar{X} = (1 - W)\bar{y} + W[2\hat{r} - \frac{1}{2}(\hat{r}_1 + \hat{r}_2)]\bar{X},$$

where $0 \leq W \leq 1, \hat{r} = \frac{\bar{y}}{\bar{x}}$, and $\hat{r}_1$ and $\hat{r}_2$ would be ratios from two random halves of sample data, which denoted by $\hat{r}_i = \frac{\bar{y}_i}{\bar{x}_i}, i = 1, 2$. Then the Chakrabarty ratio estimates are given by:

$$\hat{r}_{c1} = \frac{\bar{y}_{c1}}{\bar{x}_{c1}}$$

and

$$\hat{r}_{c2} = \frac{\bar{y}_{c2}}{\bar{x}_{c2}}.$$

Chapter 3

METHODOLOGY

The simulation was performed by the software called R. It can be downloaded from http://cran.r-project.org. The simulation code for R is in the appendix.

The first step of the simulation was generating data of population size N from 3 distributions based on mean 10 and variance 200. Those three distributions were Normal(X), LogNormal(Y), and Gamma(Z) population distributions. After the generation of populations was done, selection of the pair of those populations was necessary.

When the pair of the populations was selected, then performing SRSWR(D) was necessary by first taking SRSWR of size n from a selected populations. Then, duplicates of sampling units from SRSWR will be removed, and SRSWR changed to SRSWR(D). Since there were a removal of duplicated sample units, sample size changed from n to v_1 and v_2. Using samples with distinct units, ratio estimate of any two of the 3 populations would be calculated.

The SRSWR would be performed with sample size of v_1 and v_2, and SRSWOR would be performed with sample size of v_1 and v_2, then ratio estimate for those samples were performed also.

After the calculation of the ratio estimate, those steps were repeated to get the number of replications of the ratio estimate. Then, when the replications of the ratio estimate were gathered enough, MSE calculation with the following equation is given.

$$s_j^2 = \frac{1}{reps - 1} \sum_{i=1}^{reps} (\hat{r}_i - \bar{r})^2,$$

where

$$\bar{r} = \frac{1}{reps} \sum_{i=1}^{reps} \hat{r}_i,$$

where j is repitition number j.

The whole process would be repeated for the remaining two other pair choices.

Constant W = 0.1, 0.2, and 0.5 were used to show difference in simulation while performing Chakrabarty ratio estimate, and histogram was used to compare MSEs for each sampling methods with different estimation methods.

Chapter 4

FINDINGS

4.1 Sample Size Difference

When the sample size was low, histograms of MSE were widespread and shifted away from 0. This means when ratio estimate is performed with small sample size, there are quite large variations and not much of stability.

As the sample sizes were increased, the histogram bars were located near 0, or shifted closely to 0, and the variations were getting smaller. The mode of the histogram for large sample were having smaller value compare with small sample histograms' mode, and the range of the histogram was getting smaller.

With these results, when performing ratio estimate, using large sample gives smaller MSE than when performing ratio estimate with small samples.

The examples are shown in Figure A.1 through A.12 in Appendix A.

4.2 Sampling Methods Difference

According to the output of R, the histograms for MSE of SRSWR(D), SRSWOR, SRSWR were going close to 0, and the variation was narrow. The curve for the graph was curved symmetrically, which gave normally distributed curve.

When samples were increased, SRSWR(D), SRSWOR, SRSWR with three estimation methods' were close to normal distribution and they tend to be more stable, and the variances were getting narrower. However, the compared to SRSWR(D) and SRSWOR, MSE of SRSWR were a bit worse.

The graph for MSE of SRSWOR was somewhat similar to the output of SRSWR(D). Slight edge is noted in favor of SRSWR(D) over SRSWOR. The histograms were close to 0 and the variances were narrow also. According to Figures in Appendix A, The curve for the graph was somewhat symmetrical, which indicate a normally distributed curve.

The SRSWR was having the worst result among the three methods. Compared with SRSWR(D) and SRSWOR, it stayed away from 0. Though most of simulation results proved that the variation of MSE were close to other sampling methods, sometimes SRSWR had more wider graph which reflect more variation than SRSWR(D) or SRSWOR.

4.3 Estimation Methods Difference

Both usual ratio estimation and Quenouille's estimation methods were showing a good MSE histograms and curves. Chakrabarty also showed good result, but the results depended on the choices of the W values.

The W for Chakrabarty estimator gave variation effect to Mean Square Error. If the W gets close to 0, the histogram for MSE gets less variation. If W gets close to 1, there was more variation.

When using Chakrabarty, W had to be small in order to get appropriate Chakrabarty's estimation method. If the W was big, the estimates were totally inefficient and we decided not to include estimates with big W here. In order to gain acceptable Chakrabarty's estimate, W below 0.2 were efficient.

Chapter 5

CONCLUSION

Based on the results presented, when comparing sampling mehods, SRSWR(D) is an outstanding method to reduce the Mean Square Error. The SRSWR(D) also gives the fact that it is the stable sampling method. Plus, since SRSWOR does not have independent observations while SRSWR(D) does, SRSWR(D) is better to use and its properties are worth further both theoretical and empirical investigation. SRSWR, however, is vicinal to SRSWR(D) with less capability and it provides a close second methodology.

In the project we assumed independence between the measured observation y's and the auxiliary x's. But this assumption may not be feasible since it is violated. Thus SRSWR(D) for ratios in this case must be investigated and studied. This could be a future topic for Ph.D. dissertation as the development in expected to be intensely more complicated.

Bibliography

[1] Asok, C., A Note on the comparison between simple mean and mean based on distinct units in sampling with replacement, *The American Statistician* **34** 3 (1980) 158.

[2] Basu, D., On sampling with and without replacement, *Sankhya: The Indian Journal of Statistics*, **20** 3 (1958), 287-294.

[3] Cochran, W. G., Sampling techniques 3^{rd} ed., New York, NY: Wiley and Sons, 1977

[4] Chaudhuri, A., Stenger H., Survey sampling: Theory and methods., 2^{nd} ed., New York, NY: Chapman and Hoff, 2005

[5] Fuller, W.A., Sampling statistics., New York, NY: Wiley and Sons, 2009

[6] Korwar, R. M. and Serfling, R. J., On averaging over distinct units in sampling with replacement, *The Annals of Mathematical Statistics*, **41** 6 (1970), 2132-2134.

[7] Mussa, A. S., A new bias-reducing modification of the finite population ratio estimator and a comparison among proposed alternatives, *Journal of Official Statistics*, **15** 1 (1999), 25-38.

[8] Pathak, P. K., On the evaluation of moments of distinct units in a sample, *Sankhya: The Indian Journal of Statistics*, **23** 4 (1961) 415-420.

[9] Raj, D., Khamis, S. H., Some remarks on sampling with replacement, *Ann. Math. Statist.*, **29** 2 (1958) 550-557.

[10] Sukhatme, P. V., Sukhatme, B. V., Sampling theory of surveys applications., Ames, IA: Iowa State University Press, 198

Appendices

Appendix A

Histograms of MSE

-Normal(X), LogNormal(Y), Gamma(Z)

-Blue histogram: YX ratio estimate

-Green histogram: XZ ratio estimate

-Red histogram : YZ ratio estimate

N = 2000, n = 100, 300, 500

R \ Pop	XY	XZ	YZ	Figure number
Usual	$\odot$			A.1 - A.3
Quenouille			$\odot$	A.10 - A.12
C(1)		$\odot$		A.4 - A.6
C(2)		$\odot$		A.7 - A.9

Figure A.1: histogram of MSE of usual ratio estimate method, LogNormal(Y), Normal(X) ratio estimation, 2000 population, 100 samples, 1000 MSEs, , 1000 replications. SRSWR(D)(Left), SRSWOR(Middle), SRSWR(Right)

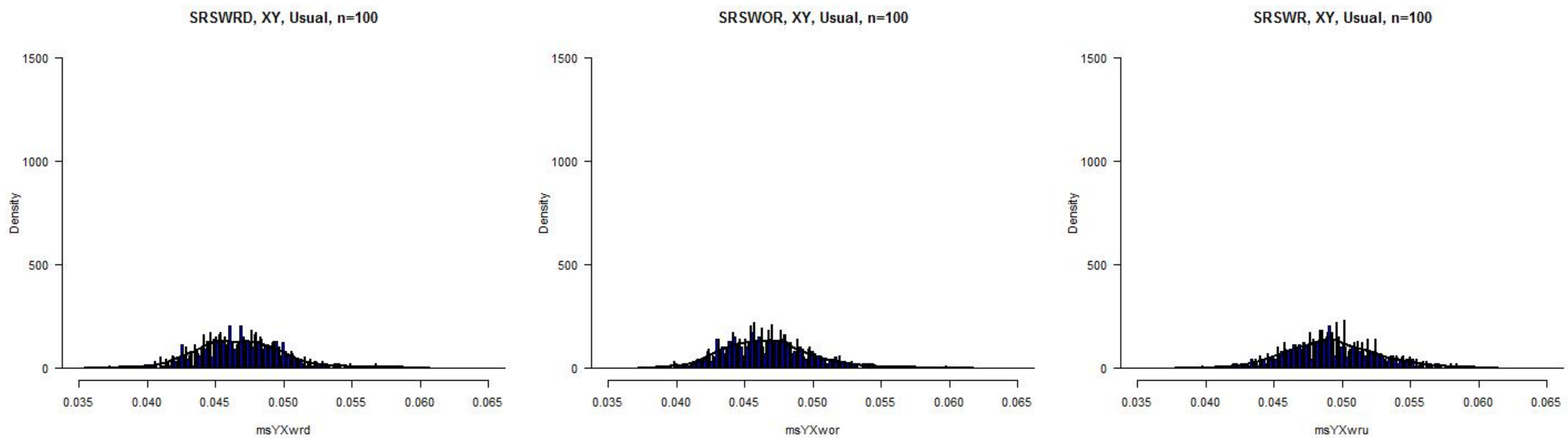


Figure A.2: histogram of MSE of usual ratio estimate method, LogNormal(Y), Normal(X) ratio estimation, 2000 population, 300 samples, 1000 MSEs, , 1000 replications. SRSWR(D)(Left), SRSWOR(Middle), SRSWR(Right)

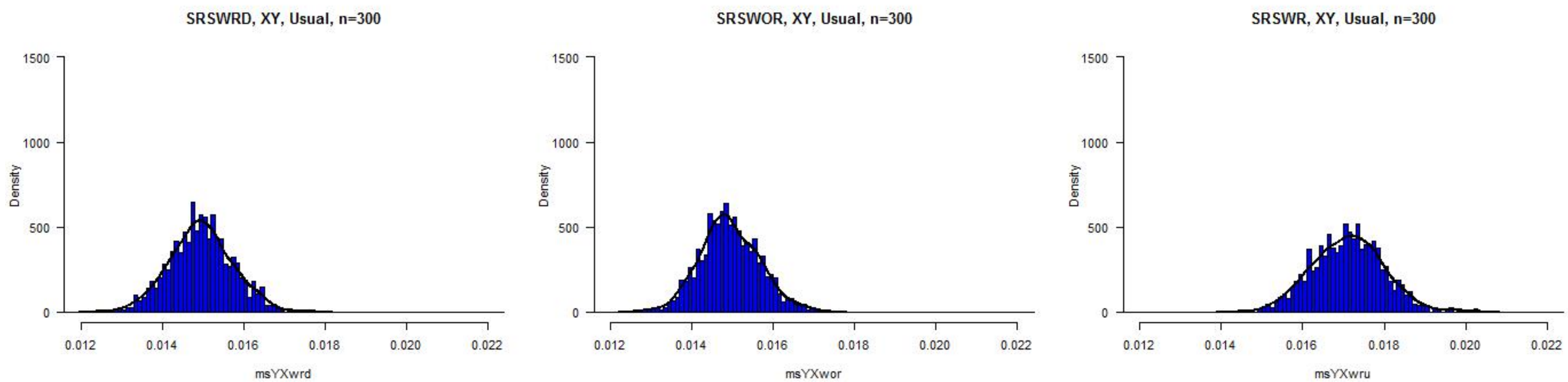


Figure A.3: histogram of MSE of usual ratio estimate method, LogNormal(Y), Normal(X) ratio estimation, 2000 population, 500 samples, 1000 MSEs, , 1000 replications. SRSWR(D)(Left), SRSWOR(Middle), SRSWR(Right)

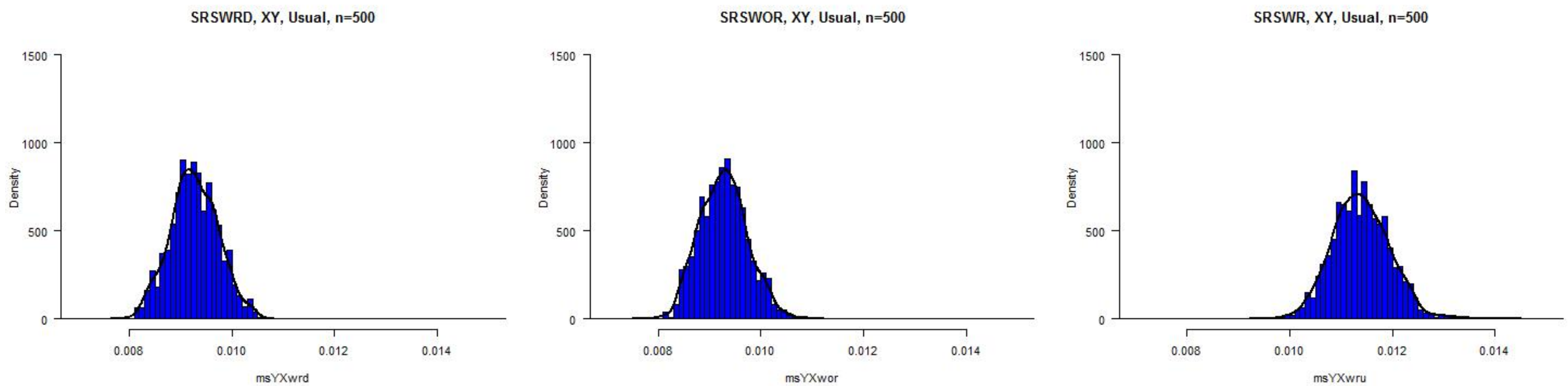


Figure A.4: histogram of MSE of Chakrabarty(1) ratio estimate method, Normal(X), Gammal(Z) ratio estimation, 2000 population, 100 samples, 1000 MSEs, , 1000 replications. SRSWR(D)(Left), SRSWOR(Middle), SRSWR(Right)

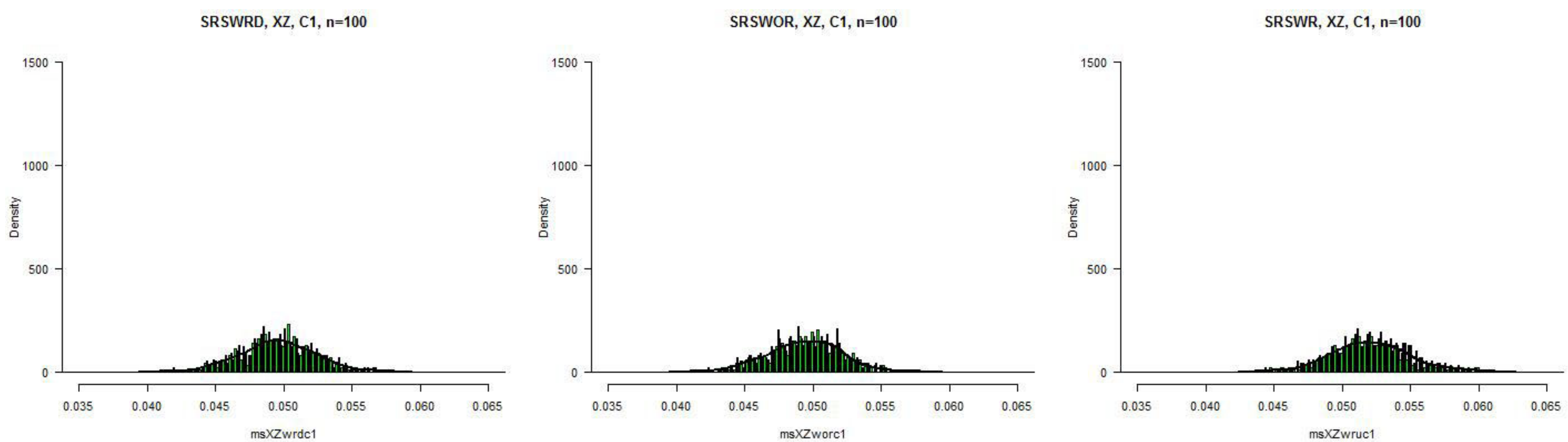


Figure A.5: histogram of MSE of Chakrabarty(1) ratio estimate method, Normal(X), Gammal(Z) ratio estimation, 2000 population, 300 samples, 1000 MSEs, , 1000 replications. SRSWR(D)(Left), SRSWOR(Middle), SRSWR(Right)

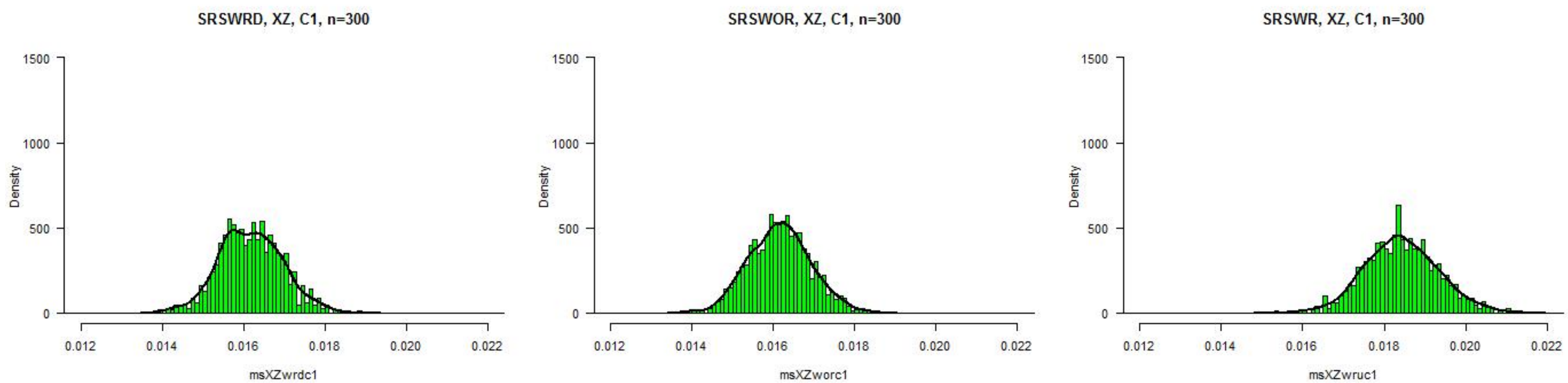


Figure A.6: histogram of MSE of Chakrabarty(1) ratio estimate method, Normal(X), Gammal(Z) ratio estimation, 2000 population, 500 samples, 1000 MSEs, , 1000 replications. SRSWR(D)(Left), SRSWOR(Middle), SRSWR(Right)

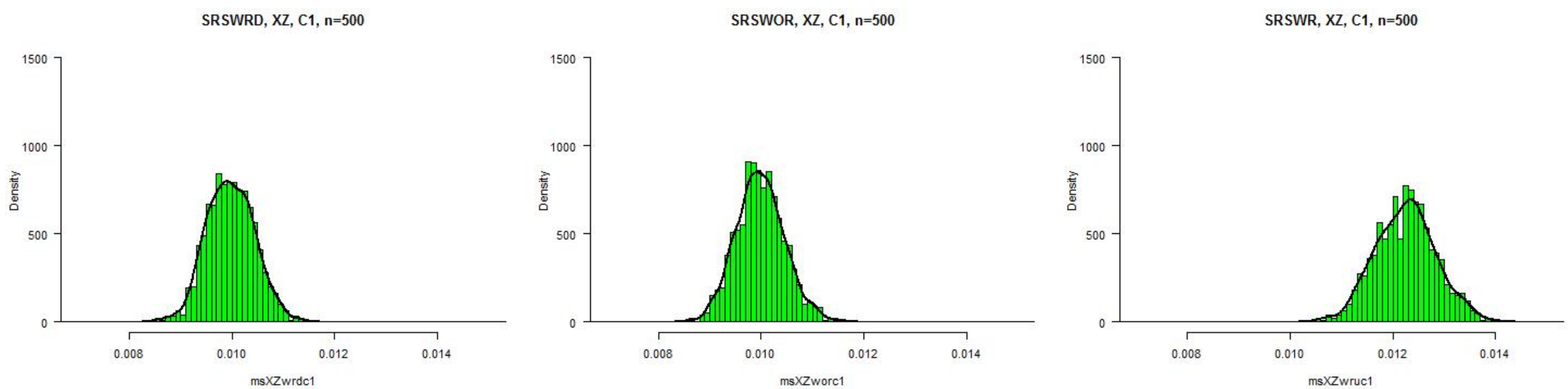


Figure A.7: histogram of MSE of Chakrabarty(2) ratio estimate method, Normal(X), Gammal(Z) ratio estimation, 2000 population, 100 samples, 1000 MSEs, , 1000 replications. SRSWR(D)(Left), SRSWOR(Middle), SRSWR(Right)

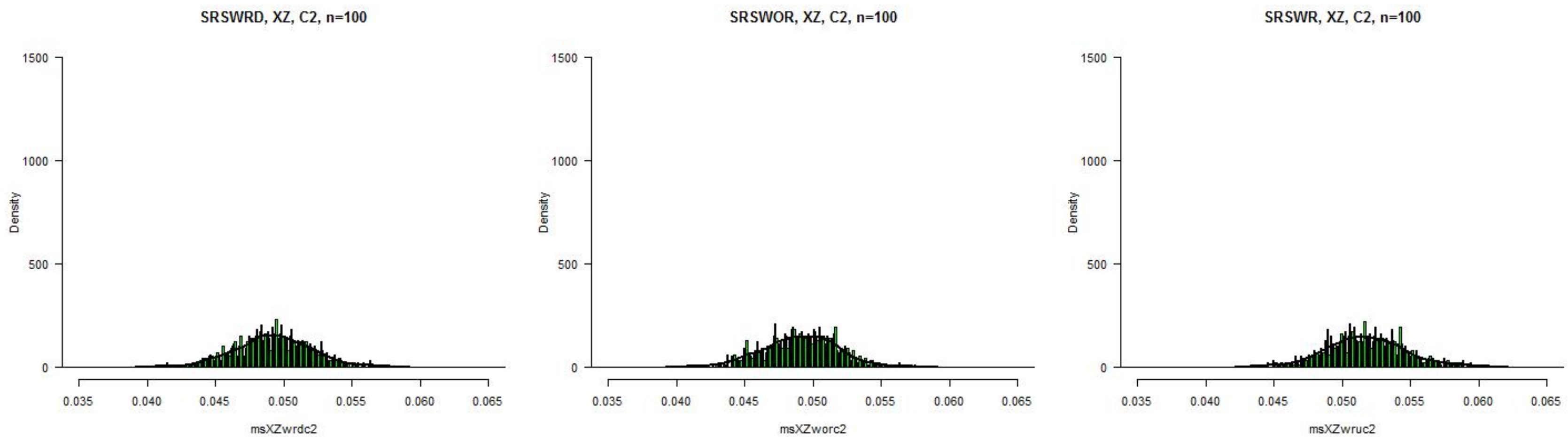


Figure A.8: histogram of MSE of Chakrabarty(2) ratio estimate method, Normal(X), Gammal(Z) ratio estimation, 2000 population, 300 samples, 1000 MSEs, , 1000 replications. SRSWR(D)(Left), SRSWOR(Middle), SRSWR(Right)

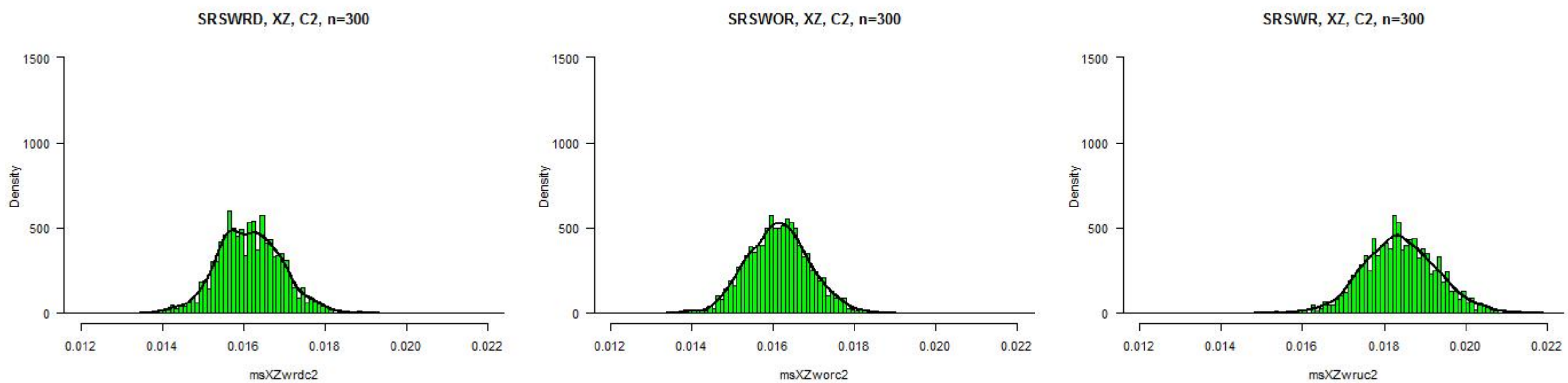


Figure A.9: histogram of MSE of Chakrabarty(2) ratio estimate method, Normal(X), Gammal(Z) ratio estimation, 2000 population, 500 samples, 1000 MSEs, , 1000 replications. SRSWR(D)(Left), SRSWOR(Middle), SRSWR(Right)

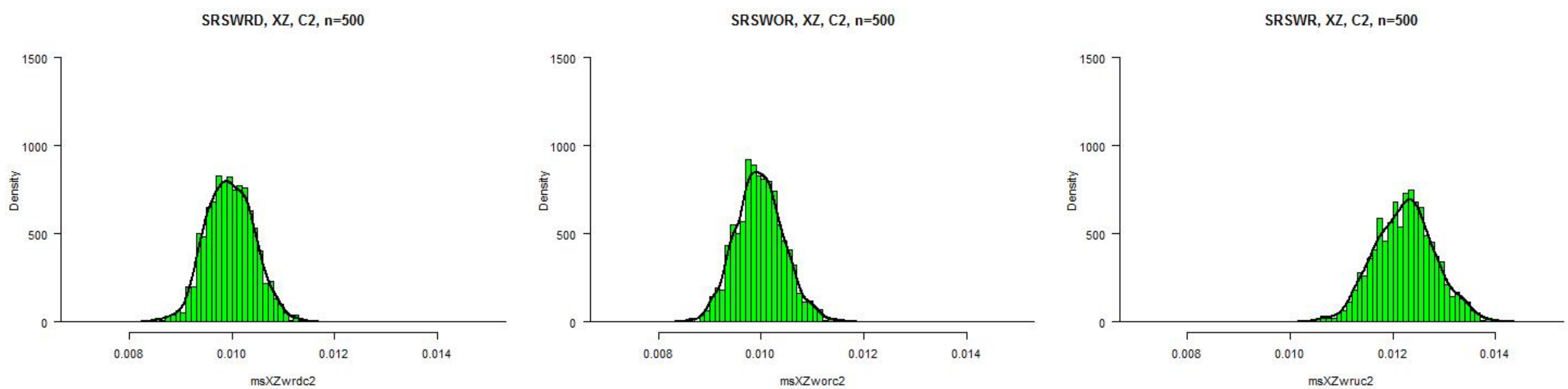


Figure A.10: histogram of MSE of Quenouille ratio estimate method, LogNormal(Y), Gammal(Z) ratio estimation, 2000 population, 100 samples, 1000 MSEs, , 1000 replications. SRSWR(D)(Left), SRSWOR(Middle), SRSWR(Right)

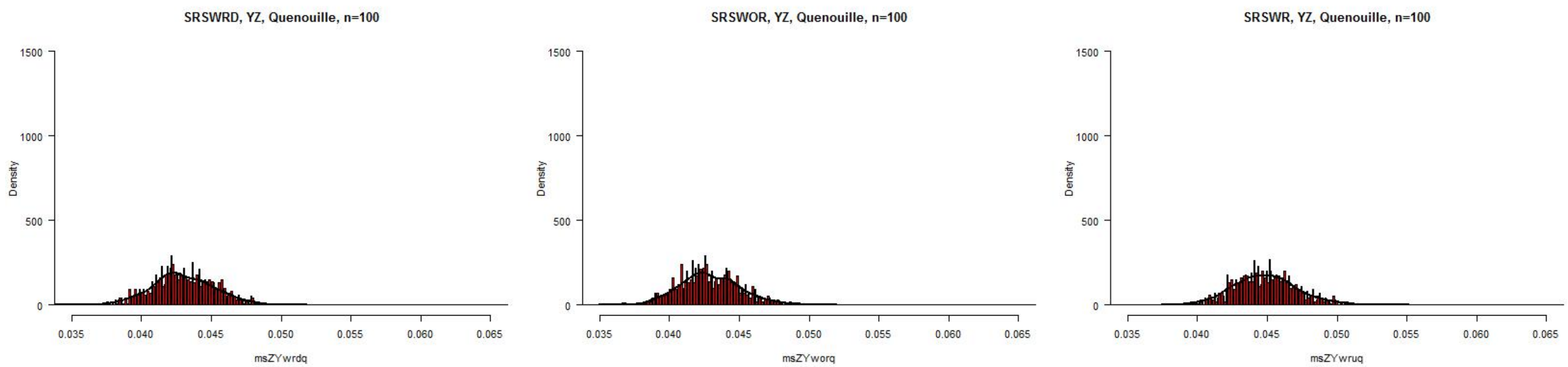


Figure A.11: histogram of MSE of Quenouille ratio estimate method, LogNormal(Y), Gammal(Z) ratio estimation, 2000 population, 300 samples, 1000 MSEs, , 1000 replications. SRSWR(D)(Left), SRSWOR(Middle), SRSWR(Right)

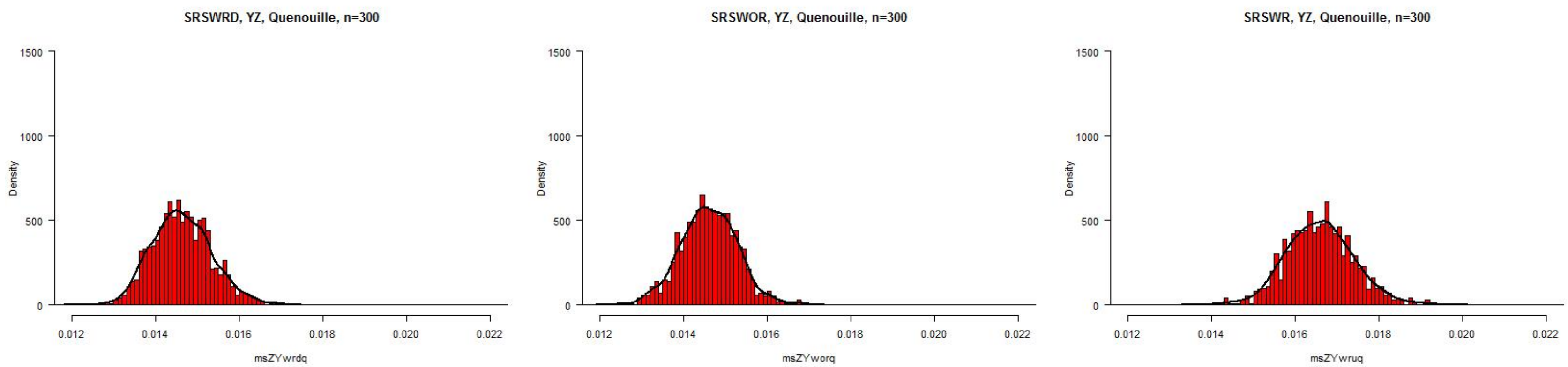
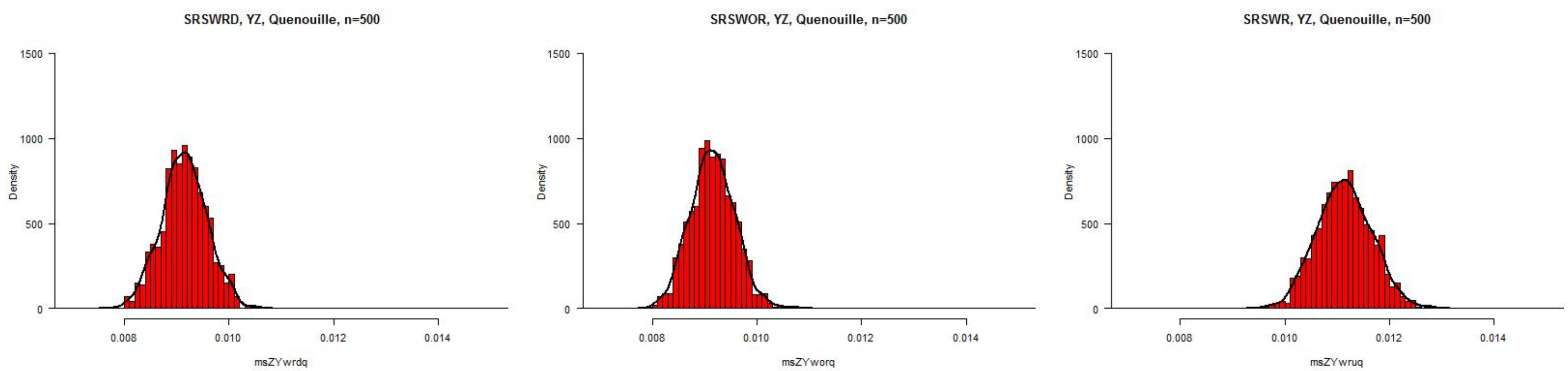


Figure A.12: histogram of MSE of Quenouille ratio estimate method,LogNormal(Y), Gammal(Z) ratio estimation, 2000 population, 500 samples, 1000 MSEs, , 1000 replications. SRSWR(D)(Left), SRSWOR(Middle), SRSWR(Right)



Appendix B

Simulation R Codes

```
#Load package semTools before starting
library(semTools)


# || or
# & and
################
#Prepare values#
################

rm(list = ls()) # remove workspace data
rm(list=ls(all=TRUE))


Xpop = 0
Ypop = 0
Zpop = 0

XpopM = 0
YpopM = 0
ZpopM = 0
XpopV = 0
YpopV = 0
ZpopV = 0

Xwr = 0
Ywr = 0
Zwr = 0
Xwor = 0
Ywor = 0
Zwor = 0
Xwrd = 0
Ywrd = 0
Zwrd = 0
```

```
Xwru = 0
Ywru = 0
Zwru = 0

XworM = 0
YworM = 0
ZworM = 0
XwrM = 0
YwrM = 0
ZwrM = 0
XwrdM = 0
YwrdM = 0
ZwrdM = 0
XwruM = 0
YwruM = 0
ZwruM = 0


ctX = 0
ctY = 0
ctZ = 0

lengthX=0
lengthY=0
lengthZ=0

msYXwor=0
msZYwor=0
msXZwor=0

msYXwr=0
msZYwr=0
msXZwr=0

msYXwrd=0
msZYwrd=0
msXZwrd=0

msYXwru = 0
msZYwru = 0
msXZwru = 0

RwrYXM = 0
RwrZYM = 0
RwrXZM = 0

RwrdYXM = 0
RwrdZYM = 0
RwrdXZM = 0

RworYXM = 0
RworZYM = 0
RworXZM = 0

RwruYXM = 0
RwruZYM = 0
RwruXZM = 0

RwrYX = 0
RwrZY = 0
```

RwrXZ = 0
RwrdYX = 0
RwrdZY = 0
RwrdXZ = 0
RworYX = 0
RworZY = 0
RworXZ = 0
RwruYX = 0
RwruZY = 0
RwruXZ = 0

#Quenouille

XwrMq1 = 0
XwrMq2 = 0
YwrMq1 = 0
YwrMq2 = 0
ZwrMq1 = 0
ZwrMq2 = 0

XwrdMq1 = 0
XwrdMq2 = 0
YwrdMq1 = 0
YwrdMq2 = 0
ZwrdMq1 = 0
ZwrdMq2 = 0

XworMq1 = 0
XworMq2 = 0
YworMq1 = 0
YworMq2 = 0
ZworMq1 = 0
ZworMq2 = 0

XwruMq1 = 0
XwruMq2 = 0
YwruMq1 = 0
YwruMq2 = 0
ZwruMq1 = 0
ZwruMq2 = 0

RwrqYX = 0
RwrqZY = 0
RwrqXZ = 0

RwrdqYX = 0
RwrdqZY = 0
RwrdqXZ = 0

RworqYX = 0
RworqZY = 0
RworqXZ = 0

RwruqYX = 0
RwruqZY = 0
RwruqXZ = 0

RwrqYXM = 0
RwrqZYM = 0
RwrqXZM = 0

RwrdqYXM = 0
RwrdqZYM = 0
RwrdqXZM = 0

RworqYXM = 0
RworqZYM = 0
RworqXZM = 0

RwruqYXM = 0
RwruqZYM = 0
RwruqXZM = 0

msYXwrq = 0
msZYwrq = 0
msXZwrq = 0
msYXwrdq = 0
msZYwrdq = 0
msXZwrdq = 0
msYXworq = 0
msZYworq = 0
msXZworq = 0
msYXwruq = 0
msZYwruq = 0
msXZwruq = 0

#Chakrabarty
Rwrc1YX = 0
Rwrc2YX = 0
Rwrc1ZY = 0
Rwrc2ZY = 0
Rwrc1XZ = 0
Rwrc2XZ = 0

Rwrc1YXM = 0
Rwrc2YXM = 0
Rwrc1ZYM = 0
Rwrc2ZYM = 0
Rwrc1XZM = 0
Rwrc2XZM = 0

msYXwrc1 = 0
msYXwrc2 = 0
msZYwrc1 = 0
msZYwrc2 = 0
msXZwrc1 = 0
msXZwrc2 = 0

Rwrdc1YX = 0
Rwrdc2YX = 0
Rwrdc1ZY = 0

Rwrdc2ZY = 0
Rwrdc1XZ = 0
Rwrdc2XZ = 0

Rwrdc1YXM = 0
Rwrdc2YXM = 0
Rwrdc1ZYM = 0
Rwrdc2ZYM = 0
Rwrdc1XZM = 0
Rwrdc2XZM = 0

msYXwrdc1 = 0
msYXwrdc2 = 0
msZYwrdc1 = 0
msZYwrdc2 = 0
msXZwrdc1 = 0
msXZwrdc2 = 0

Rworc1YX = 0
Rworc2YX = 0
Rworc1ZY = 0
Rworc2ZY = 0
Rworc1XZ = 0
Rworc2XZ = 0

Rworc1YXM = 0
Rworc2YXM = 0
Rworc1ZYM = 0
Rworc2ZYM = 0
Rworc1XZM = 0
Rworc2XZM = 0

msYXworc1 = 0
msYXworc2 = 0
msZYworc1 = 0
msZYworc2 = 0
msXZworc1 = 0
msXZworc2 = 0

Rwruc1YX = 0
Rwruc2YX = 0
Rwruc1ZY = 0
Rwruc2ZY = 0
Rwruc1XZ = 0
Rwruc2XZ = 0

Rwruc1YXM = 0
Rwruc2YXM = 0
Rwruc1ZYM = 0
Rwruc2ZYM = 0
Rwruc1XZM = 0
Rwruc2XZM = 0

msYXwruc1 = 0
msYXwruc2 = 0
msZYwruc1 = 0
msZYwruc2 = 0
msXZwruc1 = 0

```
msXZwruc2 = 0


u = 10
sigmasq = 200

#Beginning of the#
##################
######
#logN#
######
varlg = log(sigmasq/((u)^2)+1) ## Parameter B w/ sqrs
stlg = sqrt(log(sigmasq/((u)^2)+1))
meanlg = log(u) - varlg/2  ## Parameter A

#######
#Gamma#
#######
thetaGam = sigmasq / u #scale
kGam = u^2 / sigmasq #shape

#ct base = 1000
#N = 2000
#n = 500
#rep = 1000

ct = 500
W = 0.6
N = 2000

for(try in 1:ct)
{
  for(rep in 1:1000)
  {

  n = 500

#######
#Basic#
#######

  Xpop = rnorm(N, u, sqrt(sigmasq))
  Ypop = rlnorm(N, meanlog = meanlg, sdlog = sqrt(varlg))
  Zpop = rgamma(N, shape = kGam, scale = thetaGam)

  XpopM[rep] = mean(Xpop)
  YpopM[rep] = mean(Ypop)
  ZpopM[rep] = mean(Zpop)

# SRSWR w/ sample n
  Xwr = sample(Xpop, n, replace = TRUE, prob = NULL)
  Ywr = sample(Ypop, n, replace = TRUE, prob = NULL)
  Zwr = sample(Zpop, n, replace = TRUE, prob = NULL)

# each mean w/ SRSWR
  XwrM[rep] = mean(Xwr)
  YwrM[rep] = mean(Ywr)
  ZwrM[rep] = mean(Zwr)

#Quenouille
```

```
  Xwrq = split(Xwr, 1:2, drop = FALSE)
  Ywrq = split(Ywr, 1:2, drop = FALSE)
  Zwrq = split(Zwr, 1:2, drop = FALSE)

  XwrMq1[rep] = mean(Xwrq[[1]])
  XwrMq2[rep] = mean(Xwrq[[2]])
  YwrMq1[rep] = mean(Ywrq[[1]])
  YwrMq2[rep] = mean(Ywrq[[2]])
  ZwrMq1[rep] = mean(Zwrq[[1]])
  ZwrMq2[rep] = mean(Zwrq[[2]])
#Quenouille end

#SRSWR END


#SRSWRD
#OMIT repeated values
  Xwrd = unique(Xwr)
  Ywrd = unique(Ywr)
  Zwrd = unique(Zwr)
#OMIT end

  lengthX[rep] = length(Xwrd)
  lengthY[rep] = length(Ywrd)
  lengthZ[rep] = length(Zwrd)

#each mean w/ SRSWRD
  XwrdM[rep] = mean(Xwrd)
  YwrdM[rep] = mean(Ywrd)
  ZwrdM[rep] = mean(Zwrd)

#Quenouille
  Xwrdq = split(Xwrd, 1:2, drop = FALSE)
  Ywrdq = split(Ywrd, 1:2, drop = FALSE)
  Zwrdq = split(Zwrd, 1:2, drop = FALSE)

  XwrdMq1[rep] = mean(Xwrdq[[1]])
  XwrdMq2[rep] = mean(Xwrdq[[2]])
  YwrdMq1[rep] = mean(Ywrdq[[1]])
  YwrdMq2[rep] = mean(Ywrdq[[2]])
  ZwrdMq1[rep] = mean(Zwrdq[[1]])
  ZwrdMq2[rep] = mean(Zwrdq[[2]])
#Quenouille end

#SRSWRD END


#SRSWOR
  Xwor = sample(Xpop, lengthX[rep], replace = FALSE, prob = NULL)
  Ywor = sample(Ypop, lengthY[rep], replace = FALSE, prob = NULL)
  Zwor = sample(Zpop, lengthZ[rep], replace = FALSE, prob = NULL)

#each mean w/ SRSWOR
  XworM[rep] = mean(Xwor)
  YworM[rep] = mean(Ywor)
  ZworM[rep] = mean(Zwor)

#Quenouille
  Xworq = split(Xwor, 1:2, drop = FALSE)
  Yworq = split(Ywor, 1:2, drop = FALSE)
```

```
  Zworq = split(Zwor, 1:2, drop = FALSE)

  XworMq1[rep] = mean(Xworq[[1]])
  XworMq2[rep] = mean(Xworq[[2]])
  YworMq1[rep] = mean(Yworq[[1]])
  YworMq2[rep] = mean(Yworq[[2]])
  ZworMq1[rep] = mean(Zworq[[1]])
  ZworMq2[rep] = mean(Zworq[[2]])
#Quenouille end

#SRSWOR END

#SRSWR with UNIQUE values
  Xwru = sample(Xpop, lengthX[rep], replace = TRUE, prob = NULL)
  Ywru = sample(Ypop, lengthY[rep], replace = TRUE, prob = NULL)
  Zwru = sample(Zpop, lengthZ[rep], replace = TRUE, prob = NULL)

  XwruM[rep] = mean(Xwru)
  YwruM[rep] = mean(Ywru)
  ZwruM[rep] = mean(Zwru)

#Quenouille
  Xwruq = split(Xwru, 1:2, drop = FALSE)
  Ywruq = split(Ywru, 1:2, drop = FALSE)
  Zwruq = split(Zwru, 1:2, drop = FALSE)

  XwruMq1[rep] = mean(Xwruq[[1]])
  XwruMq2[rep] = mean(Xwruq[[2]])
  YwruMq1[rep] = mean(Ywruq[[1]])
  YwruMq2[rep] = mean(Ywruq[[2]])
  ZwruMq1[rep] = mean(Zwruq[[1]])
  ZwruMq2[rep] = mean(Zwruq[[2]])
#Quenouille end


###########
#Basic End#
###########
  }
#rep end



#############
#Ratio begin#
#############

RwrYX = YwrM / XwrM
RwrYXM[try] = mean(RwrYX)
RwrZY = ZwrM / YwrM
RwrZYM[try] = mean(RwrZY)
RwrXZ = XwrM / ZwrM
RwrXZM[try] = mean(RwrXZ)

RwrdYX = YwrdM / XwrdM
RwrdYXM[try] = mean(RwrdYX)
RwrdZY = ZwrdM / YwrdM
RwrdZYM[try] = mean(RwrdZY)
RwrdXZ = XwrdM / ZwrdM
RwrdXZM[try] = mean(RwrdXZ)
```

```
RworYX = YworM / XworM
RworYXM[try] = mean(RworYX)
RworZY = ZworM / YworM
RworZYM[try] = mean(RworZY)
RworXZ = XworM / ZworM
RworXZM[try] = mean(RworXZ)

RwruYX = YwruM / XwruM
RwruYXM[try] = mean(RwruYX)
RwruZY = ZwruM / YwruM
RwruZYM[try] = mean(RwruZY)
RwruXZ = XwruM / ZwruM
RwruXZM[try] = mean(RwruXZ)

RwrdYX
RwrdZY
RwrdXZ

RwrYX
RwrZY
RwrXZ

RworYX
RworZY
RworXZ

RwruZY
RwruZY
RwruXZ

msYXwor[try] = sum((RworYX - RworYXM[try])^2) / (rep-1)
msZYwor[try] = sum((RworZY - RworZYM[try])^2) / (rep-1)
msXZwor[try] = sum((RworXZ - RworXZM[try])^2) / (rep-1)

msYXwr[try] = sum((RwrYX - RwrYXM[try])^2) / (rep-1)
msZYwr[try] = sum((RwrZY - RwrZYM[try])^2) / (rep-1)
msXZwr[try] = sum((RwrXZ - RwrXZM[try])^2) / (rep-1)

msYXwrd[try] = sum((RwrdYX - RwrdYXM[try])^2) / (rep-1)
msZYwrd[try] = sum((RwrdZY - RwrdZYM[try])^2) / (rep-1)
msXZwrd[try] = sum((RwrdXZ - RwrdXZM[try])^2) / (rep-1)

msYXwru[try] = sum((RwruYX - RwruYXM[try])^2) / (rep-1)
msZYwru[try] = sum((RwruZY - RwruZYM[try])^2) / (rep-1)
msXZwru[try] = sum((RwruXZ - RwruXZM[try])^2) / (rep-1)


RwrdYXM
RwrdZYM
RwrdXZM

RworYXM
RworZYM
RworXZM

RwrYXM
RwrZYM
RwrXZM
```

```
RwruYXM
RwruZYM
RwruXZM

###########
#Ratio end#
###########

##########################
#Quenouille's R estimation#
##########################

RwrqYX = 2*RwrYX-(1/2)*((YwrMq1/XwrMq1)+(YwrMq2/XwrMq2))
RwrdqYX = 2*RwrdYX-(1/2)*((YwrdMq1/XwrdMq1)+(YwrdMq2/XwrdMq2))
RworqYX = 2*RworYX-(1/2)*((YworMq1/XworMq1)+(YworMq2/XworMq2))
RwruqYX = 2*RwruYX-(1/2)*((YwruMq1/XwruMq1)+(YwruMq2/XwruMq2))

RwrqXZ = 2*RwrXZ-(1/2)*((XwrMq1/ZwrMq1)+(XwrMq2/ZwrMq2))
RwrdqXZ = 2*RwrdXZ-(1/2)*((XwrdMq1/ZwrdMq1)+(XwrdMq2/ZwrdMq2))
RworqXZ = 2*RworXZ-(1/2)*((XworMq1/ZworMq1)+(XworMq2/ZworMq2))
RwruqXZ = 2*RwruXZ-(1/2)*((XwruMq1/ZwruMq1)+(XwruMq2/ZwruMq2))

RwrqZY = 2*RwrZY-(1/2)*((ZwrMq1/YwrMq1)+(ZwrMq2/YwrMq2))
RwrdqZY = 2*RwrdZY-(1/2)*((ZwrdMq1/YwrdMq1)+(ZwrdMq2/YwrdMq2))
RworqZY = 2*RworZY-(1/2)*((ZworMq1/YworMq1)+(ZworMq2/YworMq2))
RwruqZY = 2*RwruZY-(1/2)*((ZwruMq1/YwruMq1)+(ZwruMq2/YwruMq2))


RwruqYXM[try] = mean(RwruqYX)
RwruqZYM[try] = mean(RwruqZY)
RwruqXZM[try] = mean(RwruqXZ)

RwrqYXM[try] = mean(RwrqYX)
RwrqZYM[try] = mean(RwrqZY)
RwrqXZM[try] = mean(RwrqXZ)

RwrdqYXM[try] = mean(RwrqYX)
RwrdqZYM[try] = mean(RwrqZY)
RwrdqXZM[try] = mean(RwrqXZ)

RworqYXM[try] = mean(RworqYX)
RworqZYM[try] = mean(RworqZY)
RworqXZM[try] = mean(RworqXZ)

msYXwrq[try] = sum((RwrqYX-RwrqYXM[try])^2)/(rep-1)
msZYwrq[try] = sum((RwrqZY-RwrqZYM[try])^2)/(rep-1)
msXZwrq[try] = sum((RwrqXZ-RwrqXZM[try])^2)/(rep-1)

msYXwrdq[try] = sum((RwrdqYX-RwrdqYXM[try])^2)/(rep-1)
msZYwrdq[try] = sum((RwrdqZY-RwrdqZYM[try])^2)/(rep-1)
msXZwrdq[try] = sum((RwrdqXZ-RwrdqXZM[try])^2)/(rep-1)


msYXworq[try] = sum((RworqYX-RworqYXM[try])^2)/(rep-1)
msZYworq[try] = sum((RworqZY-RworqZYM[try])^2)/(rep-1)
msXZworq[try] = sum((RworqXZ-RworqXZM[try])^2)/(rep-1)

msYXwruq[try] = sum((RwruqYX-RwruqYXM[try])^2)/(rep-1)
msZYwruq[try] = sum((RwruqZY-RwruqZYM[try])^2)/(rep-1)
msXZwruq[try] = sum((RwruqXZ-RwruqXZM[try])^2)/(rep-1)
```

```
#############
#Chakrabarty#
#############

Rwrc1YX = (1-W)*RwrYX+W*RwrYX*(XpopM/XwrM)
Rwrc2YX = (1-W)*RwrYX+W*RwrqYX*(XpopM/XwrM)
Rwrc1ZY = (1-W)*RwrZY+W*RwrZY*(YpopM/YwrM)
Rwrc2ZY = (1-W)*RwrZY+W*RwrqZY*(YpopM/YwrM)
Rwrc1XZ = (1-W)*RwrXZ+W*RwrXZ*(ZpopM/ZwrM)
Rwrc2XZ = (1-W)*RwrXZ+W*RwrqXZ*(ZpopM/ZwrM)

Rwrc1YXM[try] = mean(Rwrc1YX)
Rwrc2YXM[try] = mean(Rwrc2YX)
Rwrc1ZYM[try] = mean(Rwrc1ZY)
Rwrc2ZYM[try] = mean(Rwrc2ZY)
Rwrc1XZM[try] = mean(Rwrc1XZ)
Rwrc2XZM[try] = mean(Rwrc2XZ)

msYXwrc1[try] = sum((Rwrc1YX-Rwrc1YXM[try])^2)/(rep-1)
msYXwrc2[try] = sum((Rwrc2YX-Rwrc2YXM[try])^2)/(rep-1)
msZYwrc1[try] = sum((Rwrc1ZY-Rwrc1ZYM[try])^2)/(rep-1)
msZYwrc2[try] = sum((Rwrc2ZY-Rwrc2ZYM[try])^2)/(rep-1)
msXZwrc1[try] = sum((Rwrc1XZ-Rwrc1XZM[try])^2)/(rep-1)
msXZwrc2[try] = sum((Rwrc2XZ-Rwrc2XZM[try])^2)/(rep-1)



Rwrdc1YX = (1-W)*RwrdYX+W*RwrdYX*(XpopM/XwrdM)
Rwrdc2YX = (1-W)*RwrdYX+W*RwrdqYX*(XpopM/XwrdM)
Rwrdc1ZY = (1-W)*RwrdZY+W*RwrdZY*(YpopM/YwrdM)
Rwrdc2ZY = (1-W)*RwrdZY+W*RwrdqZY*(YpopM/YwrdM)
Rwrdc1XZ = (1-W)*RwrdXZ+W*RwrdXZ*(ZpopM/ZwrdM)
Rwrdc2XZ = (1-W)*RwrdXZ+W*RwrdqXZ*(ZpopM/ZwrdM)

Rwrdc1YXM[try] = mean(Rwrdc1YX)
Rwrdc2YXM[try] = mean(Rwrdc2YX)
Rwrdc1ZYM[try] = mean(Rwrdc1ZY)
Rwrdc2ZYM[try] = mean(Rwrdc2ZY)
Rwrdc1XZM[try] = mean(Rwrdc1XZ)
Rwrdc2XZM[try] = mean(Rwrdc2XZ)

msYXwrdc1[try] = sum((Rwrdc1YX-Rwrdc1YXM[try])^2)/(rep-1)
msYXwrdc2[try] = sum((Rwrdc2YX-Rwrdc2YXM[try])^2)/(rep-1)
msZYwrdc1[try] = sum((Rwrdc1ZY-Rwrdc1ZYM[try])^2)/(rep-1)
msZYwrdc2[try] = sum((Rwrdc2ZY-Rwrdc2ZYM[try])^2)/(rep-1)
msXZwrdc1[try] = sum((Rwrdc1XZ-Rwrdc1XZM[try])^2)/(rep-1)
msXZwrdc2[try] = sum((Rwrdc2XZ-Rwrdc2XZM[try])^2)/(rep-1)



Rworc1YX = (1-W)*RworYX+W*RworYX*(XpopM/XworM)
Rworc2YX = (1-W)*RworYX+W*RworqYX*(XpopM/XworM)
Rworc1ZY = (1-W)*RworZY+W*RworZY*(YpopM/YworM)
Rworc2ZY = (1-W)*RworZY+W*RworqZY*(YpopM/YworM)
Rworc1XZ = (1-W)*RworXZ+W*RworXZ*(ZpopM/ZworM)
Rworc2XZ = (1-W)*RworXZ+W*RworqXZ*(ZpopM/ZworM)
```

```
Rworc1YXM[try] = mean(Rworc1YX)
Rworc2YXM[try] = mean(Rwrdc2YX)
Rworc1ZYM[try] = mean(Rworc1ZY)
Rworc2ZYM[try] = mean(Rworc2ZY)
Rworc1XZM[try] = mean(Rworc1XZ)
Rworc2XZM[try] = mean(Rworc2XZ)

msYXworc1[try] = sum((Rworc1YX-Rworc1YXM[try])^2)/(rep-1)
msYXworc2[try] = sum((Rworc2YX-Rworc2YXM[try])^2)/(rep-1)
msZYworc1[try] = sum((Rworc1ZY-Rworc1ZYM[try])^2)/(rep-1)
msZYworc2[try] = sum((Rworc2ZY-Rworc2ZYM[try])^2)/(rep-1)
msXZworc1[try] = sum((Rworc1XZ-Rworc1XZM[try])^2)/(rep-1)
msXZworc2[try] = sum((Rworc2XZ-Rworc2XZM[try])^2)/(rep-1)



Rwruc1YX = (1-W)*RwruYX+W*RwruYX*(XpopM/XwruM)
Rwruc2YX = (1-W)*RwruYX+W*RwruqYX*(XpopM/XwruM)
Rwruc1ZY = (1-W)*RwruZY+W*RwruZY*(YpopM/YwruM)
Rwruc2ZY = (1-W)*RwruZY+W*RwruqZY*(YpopM/YwruM)
Rwruc1XZ = (1-W)*RwruXZ+W*RwruXZ*(ZpopM/ZwruM)
Rwruc2XZ = (1-W)*RwruXZ+W*RwruqXZ*(ZpopM/ZwruM)

Rwruc1YXM[try] = mean(Rwruc1YX)
Rwruc2YXM[try] = mean(Rwruc2YX)
Rwruc1ZYM[try] = mean(Rwruc1ZY)
Rwruc2ZYM[try] = mean(Rwruc2ZY)
Rwruc1XZM[try] = mean(Rwruc1XZ)
Rwruc2XZM[try] = mean(Rwruc2XZ)

msYXwruc1[try] = sum((Rwruc1YX-Rwruc1YXM[try])^2)/(rep-1)
msYXwruc2[try] = sum((Rwruc2YX-Rwruc2YXM[try])^2)/(rep-1)
msZYwruc1[try] = sum((Rwruc1ZY-Rwruc1ZYM[try])^2)/(rep-1)
msZYwruc2[try] = sum((Rwruc2ZY-Rwruc2ZYM[try])^2)/(rep-1)
msXZwruc1[try] = sum((Rwruc1XZ-Rwruc1XZM[try])^2)/(rep-1)
msXZwruc2[try] = sum((Rwruc2XZ-Rwruc2XZM[try])^2)/(rep-1)
}

#try end


#Quenoille#
###########
#RwrqXY
#RwrqYZ
#RwrqXZ

#RwrdqXY
#RwrdqYZ
#RwrdqXZ

#RworqXY
#RworqYZ
#RworqXZ

msYXwrq
msYXwrdq
msYXworq
msYXwruq
```

msZYwrq
msZYwrdq
msZYworq
msZYwruq

msXZwrq
msXZwrdq
msXZworq
msXZwruq

#Chakrabarty#
#############

#Rwrc1XY
#Rwrc2XY
#Rwrc1YZ
#Rwrc2YZ
#Rwrc1XZ
#Rwrc2XZ

#Rwrdc1XY
#Rwrdc2XY
#Rwrdc1YZ
#Rwrdc2YZ
#Rwrdc1XZ
#Rwrdc2XZ

#Rworc1XY
#Rworc2XY
#Rworc1YZ
#Rworc2YZ
#Rworc1XZ
#Rworc2XZ

msYXwrc1
msYXwrc2
msYXwrdc1
msYXwrdc2
msYXworc1
msYXworc2
msYXwruc1
msYXwruc2

msZYwrc1
msZYwrc2
msZYwrdc1
msZYwrdc2
msZYworc1
msZYworc2
msZYwruc1
msZYwruc2

msXZwrc1
msXZwrc2
msXZwrdc1
msXZwrdc2
msXZworc1
msXZworc2
msXZwruc1

```
msXZwruc2


#Ratio MSE#
###########

msYXwrd
msYXwor
msYXwr
msYXwru

msZYwor
msZYwrd
msZYwr
msZYwru

msXZwor
msXZwrd
msXZwr
msXZwru




##########################
#use when n = 500 N = 2000, W = 0.6#
##########################

############
#Quenouille#
############
par(mfrow=c(1,3))

LowerYXq = min(min(msYXwrdq), min(msYXworq), min(msYXwruq))
UpperYXq = max(max(msYXwrdq), max(msYXworq), max(msYXwruq))
binYXq = seq(LowerYXq-0.01, UpperYXq+0.01, by = 0.0001)

hist(msYXwrdq, las=1, prob=TRUE, xlim=c(LowerYXq, UpperYXq),
 breaks = binYXq, col="red", ylim = c(0,1500))
lines(density(msYXwrdq),lwd=2)
hist(msYXworq, las=1, prob=TRUE, xlim=c(LowerYXq, UpperYXq),
breaks = binYXq, col="red", ylim = c(0,1500))
lines(density(msYXworq),lwd=2)
hist(msYXwruq, las=1, prob=TRUE, xlim=c(LowerYXq, UpperYXq),
breaks = binYXq, col="red", ylim = c(0,1500))
lines(density(msYXwruq),lwd=2)


LowerZYq = min(min(msZYwrdq), min(msZYworq), min(msZYwruq))
UpperZYq = max(max(msZYwrdq), max(msZYworq), max(msZYwruq))

binZYq = seq(LowerZYq-0.01, UpperZYq+0.01, by = 0.0001)

hist(msZYwrdq, las=1, prob=TRUE, xlim=c(LowerZYq, UpperZYq),
 breaks = binZYq, col="red", ylim = c(0,1500))
lines(density(msZYwrdq),lwd=2)
hist(msZYworq, las=1, prob=TRUE, xlim=c(LowerZYq, UpperZYq),
 breaks = binZYq, col="red", ylim = c(0,1500))
lines(density(msZYworq),lwd=2)
hist(msZYwruq, las=1, prob=TRUE, xlim=c(LowerZYq, UpperZYq),
```

```
breaks = binZYq, col="red", ylim = c(0,1500))

lines(density(msZYwruq),lwd=2)


LowerXZq =min(min(msXZwrdq), min(msXZworq), min(msXZwruq))
UpperXZq =max(max(msXZwrdq), max(msXZworq), max(msXZwruq))

binXZq = seq(LowerYXq-0.01, UpperXZq+0.01, by = 0.0001)


hist(msXZwrdq, las=1, prob=TRUE, xlim=c(LowerXZq, UpperXZq),
 breaks = binXZq, col="red", ylim = c(0,1500))
lines(density(msXZwrdq),lwd=2)
hist(msXZworq, las=1, prob=TRUE, xlim=c(LowerXZq, UpperXZq),
 breaks = binXZq, col="red", ylim = c(0,1500))
lines(density(msXZworq),lwd=2)
hist(msXZwruq, las=1, prob=TRUE, xlim=c(LowerXZq, UpperXZq),
 breaks = binXZq, col="red", ylim = c(0,1500))
lines(density(msXZwruq),lwd=2)





#############
#Chakrabarty#
#############
###############
#n=500, N=2000#
###############
par(mfrow=c(1,3))

LowerYXc1=min(min(msYXwrdc1),min(msYXworc1), min(msYXwruc1))
UpperYXc1=max(max(msYXwrdc1), max(msYXworc1), max(msYXwruc1))
LowerYXc2=min(min(msYXwrdc2),min(msYXworc2), min(msYXwruc2))
UpperYXc2=max(max(msYXwrdc2), max(msYXworc2), max(msYXwruc2))

binYXc1 = seq(LowerYXc1-0.01, LowerYXc1+0.01, by = 0.0001)
binYXc2 = seq(LowerYXc2-0.01, UpperYXc2+0.01, by = 0.0001)


hist(msYXwrdc1, las=1, prob=TRUE, xlim=c(LowerYXc1, UpperYXc1),
 breaks = binYXc1, col="green", ylim = c(0,1500))
lines(density(msYXwrdc1),lwd=2)
hist(msYXworc1, las=1, prob=TRUE, xlim=c(LowerYXc1, UpperYXc1),
 breaks = binYXc1, col="green", ylim = c(0,1500))
lines(density(msYXworc1),lwd=2)
hist(msYXwruc1, las=1, prob=TRUE, xlim=c(LowerYXc1, UpperYXc1),
breaks = binYXc1, col="green", ylim = c(0,1500))
lines(density(msYXwruc1),lwd=2)
hist(msYXwrdc2, las=1, prob=TRUE, xlim=c(LowerYXc2, UpperYXc2),
breaks = binYXc2 , col="green", ylim = c(0,1500))
lines(density(msYXwrdc2),lwd=2)
hist(msYXworc2, las=1, prob=TRUE, xlim=c(LowerYXc2, UpperYXc2),
breaks = binYXc2 , col="green", ylim = c(0,1500))
lines(density(msYXworc2),lwd=2)
hist(msYXwruc2, las=1, prob=TRUE, xlim=c(LowerYXc2, UpperYXc2),
 breaks = binYXc2 , col="green", ylim = c(0,1500))
```

```
lines(density(msYXwruc2),lwd=2)


LowerZYc1=min(min(msZYwrdc1),min(msZYworc1), min(msZYwruc1))
UpperZYc1=max(max(msZYwrdc1), max(msZYworc1), max(msZYwruc1))
LowerZYc2=min(min(msZYwrdc2),min(msZYworc2), min(msZYwruc2))
UpperZYc2=max(max(msZYwrdc2), max(msZYworc2), max(msZYwruc2))

binZYc1 = seq(LowerZYc1-0.01, UpperZYc1+0.01, by = 0.0001)
binZYc2 = seq(LowerZYc2-0.01, UpperZYc2+0.01, by = 0.0001)


hist(msZYwrdc1, las=1, prob=TRUE, xlim=c(LowerZYc1, UpperZYc1),
breaks = binZYc1, col="green", ylim = c(0,1500))
lines(density(msZYwrdc1),lwd=2)
hist(msZYworc1, las=1, prob=TRUE, xlim=c(LowerZYc1, UpperZYc1),
 breaks = binZYc1, col="green", ylim = c(0,1500))
lines(density(msZYworc1),lwd=2)
hist(msZYwruc1, las=1, prob=TRUE, xlim=c(LowerZYc1, UpperZYc1),
 breaks = binZYc1, col="green", ylim = c(0,1500))
lines(density(msZYwruc1),lwd=2)
hist(msZYwrdc2, las=1, prob=TRUE, xlim=c(LowerZYc2, UpperZYc2),
 breaks = binZYc2 , col="green", ylim = c(0,1500))
lines(density(msZYwrdc2),lwd=2)
hist(msZYworc2, las=1, prob=TRUE, xlim=c(LowerZYc2, UpperZYc2),
breaks = binZYc2 , col="green", ylim = c(0,1500))
lines(density(msZYworc2),lwd=2)
hist(msZYwruc2, las=1, prob=TRUE, xlim=c(LowerZYc2, UpperZYc2),
breaks = binZYc2 , col="green", ylim = c(0,1500))
lines(density(msZYwruc2),lwd=2)


LowerXZc1=min(min(msXZwrdc1),min(msXZworc1), min(msXZwruc1))
UpperXZc1=max(max(msXZwrdc1), max(msXZworc1), max(msXZwruc1))
LowerXZc2=min(min(msXZwrdc2),min(msXZworc2), min(msXZwruc2))
UpperXZc2=max(max(msXZwrdc2), max(msXZworc2), max(msXZwruc2))

binXZc1 = seq(LowerXZc1-0.01, UpperXZc1+0.01, by = 0.0001)
binXZc2 = seq(LowerXZc2-0.01, UpperXZc2+0.01, by = 0.0001)


hist(msXZwrdc1, las=1, prob=TRUE, xlim=c(LowerXZc1, UpperXZc1),
 breaks = binXZc1, col="green", ylim = c(0,1500))
lines(density(msXZwrdc1),lwd=2)
hist(msXZworc1, las=1, prob=TRUE, xlim=c(LowerXZc1, UpperXZc1),
breaks = binXZc1, col="green", ylim = c(0,1500))
lines(density(msXZworc1),lwd=2)
hist(msXZwruc1, las=1, prob=TRUE, xlim=c(LowerXZc1, UpperXZc1),
breaks = binXZc1, col="green", ylim = c(0,1500))
lines(density(msXZwruc1),lwd=2)
hist(msXZwrdc2, las=1, prob=TRUE, xlim=c(LowerXZc2, UpperXZc2),
breaks = binXZc2 , col="green", ylim = c(0,1500))
lines(density(msXZwrdc2),lwd=2)
hist(msXZworc2, las=1, prob=TRUE, xlim=c(LowerXZc2, UpperXZc2),
 breaks = binXZc2 , col="green", ylim = c(0,1500))
lines(density(msXZworc2),lwd=2)
hist(msXZwruc2, las=1, prob=TRUE, xlim=c(LowerXZc2, UpperXZc2),
 breaks = binXZc2 , col="green", ylim = c(0,1500))
lines(density(msXZwruc2),lwd=2)
```

```
##############################
#FOR RATIO ESTIMATOR HIST MSE#
##############################


par(mfrow=c(1,3))

LowerYX = min(min(msYXwrd), min(msYXwor), min(msYXwru))
UpperYX = max(max(msYXwrd), max(msYXwor), max(msYXwru))

binYX = seq(LowerYX-0.01, UpperYX+0.01, by = 0.0001)

hist(msYXwrd, las=1, prob=TRUE, xlim=c(LowerYX, UpperYX), breaks = binYX,
 col="blue", ylim = c(0,1500))
lines(density(msYXwrd),lwd=2)
hist(msYXwor, las=1, prob=TRUE, xlim=c(LowerYX, UpperYX), breaks = binYX,
 col="blue", ylim = c(0,1500))
lines(density(msYXwor),lwd=2)
hist(msYXwru, las=1, prob=TRUE, xlim=c(LowerYX, UpperYX), breaks = binYX,
 col="blue", ylim = c(0,1500))
lines(density(msYXwru),lwd=2)


LowerZY=min(min(msZYwrd), min(msZYwor), min(msZYwru))
UpperZY=max(max(msZYwrd), max(msZYwor), max(msZYwru))
binZY = seq(LowerZY-0.01, UpperZY+0.01, by = 0.0001)

hist(msZYwrd, las=1, prob=TRUE, xlim=c(LowerZY, UpperZY), breaks = binZY,
col="blue", ylim = c(0,1500))
lines(density(msZYwrdq),lwd=2)
hist(msZYwor, las=1, prob=TRUE, xlim=c(LowerZY, UpperZY), breaks = binZY,
 col="blue", ylim = c(0,1500))
lines(density(msZYworq),lwd=2)
hist(msZYwru, las=1, prob=TRUE, xlim=c(LowerZY, UpperZY), breaks = binZY,
col="blue", ylim = c(0,1500))
lines(density(msZYwruq),lwd=2)



LowerXZ = min(min(msXZwrd), min(msXZwor), min(msXZwru))
UpperXZ = max(max(msXZwrd), max(msXZwor), max(msXZwru))
binXZ = seq(LowerXZ-0.01, UpperXZ+0.01, by = 0.0001)

hist(msXZwrd, las=1, prob=TRUE, xlim=c(LowerXZ, UpperXZ), breaks = binXZ,
 col="blue", ylim = c(0,1500))
lines(density(msXZwrdq),lwd=2)
hist(msXZwor, las=1, prob=TRUE, xlim=c(LowerXZ, UpperXZ), breaks = binXZ,
col="blue", ylim = c(0,1500))
lines(density(msXZworq),lwd=2)
hist(msXZwru, las=1, prob=TRUE, xlim=c(LowerXZ, UpperXZ), breaks = binXZ,
col="blue", ylim = c(0,1500))
lines(density(msXZwruq),lwd=2)


#test Values#
QLowBnd= min(LowerYXq, LowerZYq, LowerXZq)
QUpBnd= max(UpperYXq, UpperZYq, UpperXZq)
```

```
Qbin = seq(QLowBnd-0.01, QUpBnd+0.01, by = 0.0001)

hist(msYXwrdq, las=1, prob=TRUE, xlim=c(QLowBnd, QUpBnd), breaks = Qbin,
col="red", ylim = c(0,1500))
lines(density(msYXwrdq),lwd=2)
hist(msYXworq, las=1, prob=TRUE, xlim=c(QLowBnd, QUpBnd), breaks = Qbin,
 col="red", ylim = c(0,1500))
lines(density(msYXworq),lwd=2)
hist(msYXwruq, las=1, prob=TRUE, xlim=c(QLowBnd, QUpBnd), breaks = Qbin,
 col="red", ylim = c(0,1500))
lines(density(msYXwruq),lwd=2)

hist(msZYwrdq, las=1, prob=TRUE, xlim=c(QLowBnd, QUpBnd), breaks = Qbin,
 col="red", ylim = c(0,1500))
lines(density(msZYwrdq),lwd=2)
hist(msZYworq, las=1, prob=TRUE, xlim=c(QLowBnd, QUpBnd), breaks = Qbin,
col="red", ylim = c(0,1500))
lines(density(msZYworq),lwd=2)
hist(msZYwruq, las=1, prob=TRUE, xlim=c(QLowBnd, QUpBnd), breaks = Qbin,
col="red", ylim = c(0,1500))
lines(density(msZYwruq),lwd=2)

hist(msXZwrdq, las=1, prob=TRUE, xlim=c(QLowBnd, QUpBnd), breaks = Qbin,
col="red", ylim = c(0,1500))
lines(density(msXZwrdq),lwd=2)
hist(msXZworq, las=1, prob=TRUE, xlim=c(QLowBnd, QUpBnd), breaks = Qbin,
 col="red", ylim = c(0,1500))
lines(density(msXZworq),lwd=2)
hist(msXZwruq, las=1, prob=TRUE, xlim=c(QLowBnd, QUpBnd), breaks = Qbin,
col="red", ylim = c(0,1500))
lines(density(msXZwruq),lwd=2)
```

```
#TestingValues#
ChakLrBnd=min(LowerYXc1,LowerYXc2,LowerZYc1,LowerZYc2,LowerXZc1,LowerXZc2)
ChakUpBnd=max(UpperYXc1,UpperYXc2,UpperZYc1,UpperZYc2,UpperXZc1,UpperXZc2)
ChakBin = seq(ChakLrBnd-0.01, ChakUpBnd+0.01, by = 0.0001)

hist(msYXwrdc1, las=1, prob=TRUE, xlim=c(ChakLrBnd, ChakUpBnd), breaks = ChakBin ,
col="green", ylim = c(0,1500))
lines(density(msYXwrdc1),lwd=2)
hist(msYXworc1, las=1, prob=TRUE, xlim=c(ChakLrBnd, ChakUpBnd), breaks = ChakBin ,
 col="green", ylim = c(0,1500))
lines(density(msYXworc1),lwd=2)
hist(msYXwruc1, las=1, prob=TRUE, xlim=c(ChakLrBnd, ChakUpBnd), breaks = ChakBin ,
col="green", ylim = c(0,1500))
lines(density(msYXwruc1),lwd=2)
hist(msYXwrdc2, las=1, prob=TRUE, xlim=c(ChakLrBnd, ChakUpBnd), breaks = ChakBin ,
 col="green", ylim = c(0,1500))
lines(density(msYXwrdc2),lwd=2)
hist(msYXworc2, las=1, prob=TRUE, xlim=c(ChakLrBnd, ChakUpBnd), breaks = ChakBin ,
col="green", ylim = c(0,1500))
lines(density(msYXworc2),lwd=2)
```

```
hist(msYXwruc2, las=1, prob=TRUE, xlim=c(ChakLrBnd, ChakUpBnd), breaks = ChakBin ,
col="green", ylim = c(0,1500))
lines(density(msYXwruc2),lwd=2)

hist(msZYwrdc1, las=1, prob=TRUE, xlim=c(ChakLrBnd, ChakUpBnd), breaks = ChakBin ,
 col="green", ylim = c(0,1500))
lines(density(msZYwrdc1),lwd=2)
hist(msZYworc1, las=1, prob=TRUE, xlim=c(ChakLrBnd, ChakUpBnd), breaks = ChakBin ,
 col="green", ylim = c(0,1500))
lines(density(msZYworc1),lwd=2)
hist(msZYwruc1, las=1, prob=TRUE, xlim=c(ChakLrBnd, ChakUpBnd), breaks = ChakBin ,
 col="green", ylim = c(0,1500))
lines(density(msZYwruc1),lwd=2)
hist(msZYwrdc2, las=1, prob=TRUE, xlim=c(ChakLrBnd, ChakUpBnd), breaks = ChakBin ,
 col="green", ylim = c(0,1500))
lines(density(msZYwrdc2),lwd=2)
hist(msZYworc2, las=1, prob=TRUE, xlim=c(ChakLrBnd, ChakUpBnd), breaks = ChakBin ,
 col="green", ylim = c(0,1500))
lines(density(msZYworc2),lwd=2)
hist(msZYwruc2, las=1, prob=TRUE, xlim=c(ChakLrBnd, ChakUpBnd), breaks = ChakBin ,
 col="green", ylim = c(0,1500))
lines(density(msZYwruc2),lwd=2)

hist(msXZwrdc1, las=1, prob=TRUE, xlim=c(ChakLrBnd, ChakUpBnd), breaks = ChakBin ,
 col="green", ylim = c(0,1500))
lines(density(msXZwrdc1),lwd=2)
hist(msXZworc1, las=1, prob=TRUE, xlim=c(ChakLrBnd, ChakUpBnd), breaks = ChakBin ,
 col="green", ylim = c(0,1500))
lines(density(msXZworc1),lwd=2)
hist(msXZwruc1, las=1, prob=TRUE, xlim=c(ChakLrBnd, ChakUpBnd), breaks = ChakBin ,
 col="green", ylim = c(0,1500))
lines(density(msXZwruc1),lwd=2)
hist(msXZwrdc2, las=1, prob=TRUE, xlim=c(ChakLrBnd, ChakUpBnd), breaks = ChakBin,
 col="green", ylim = c(0,1500))
lines(density(msXZwrdc2),lwd=2)
hist(msXZworc2, las=1, prob=TRUE, xlim=c(ChakLrBnd, ChakUpBnd), breaks = ChakBin ,
col="green", ylim = c(0,1500))
lines(density(msXZworc2),lwd=2)
hist(msXZwruc2, las=1, prob=TRUE, xlim=c(ChakLrBnd, ChakUpBnd), breaks = ChakBin ,
 col="green", ylim = c(0,1500))
lines(density(msXZwruc2),lwd=2)


#TestingValues#

RatioLrBnd = min(LowerYX, LowerZY, LowerXZ)
RatioUpBnd = max(UpperYX, UpperZY, UpperZY)
RatioBin = seq(RatioLrBnd-0.01, RatioUpBnd+0.01, by = 0.0001)

hist(msYXwrd, las=1, prob=TRUE, xlim=c(RatioLrBnd, RatioUpBnd), breaks = RatioBin,
 col="blue", ylim = c(0,1500))
lines(density(msYXwrd),lwd=2)
hist(msYXwor, las=1, prob=TRUE, xlim=c(RatioLrBnd, RatioUpBnd), breaks = RatioBin,
 col="blue", ylim = c(0,1500))
lines(density(msYXwor),lwd=2)
hist(msYXwru, las=1, prob=TRUE, xlim=c(RatioLrBnd, RatioUpBnd), breaks = RatioBin,
col="blue", ylim = c(0,1500))
lines(density(msYXwru),lwd=2)
hist(msZYwrd, las=1, prob=TRUE, xlim=c(RatioLrBnd, RatioUpBnd), breaks = RatioBin,
 col="blue", ylim = c(0,1500))
```

```
lines(density(msZYwrdq),lwd=2)
hist(msZYwor, las=1, prob=TRUE, xlim=c(RatioLrBnd, RatioUpBnd), breaks = RatioBin,
 col="blue", ylim = c(0,1500))
lines(density(msZYworq),lwd=2)
hist(msZYwru, las=1, prob=TRUE, xlim=c(RatioLrBnd, RatioUpBnd), breaks = RatioBin,
col="blue", ylim = c(0,1500))
lines(density(msZYwruq),lwd=2)
hist(msXZwrd, las=1, prob=TRUE, xlim=c(RatioLrBnd, RatioUpBnd), breaks = RatioBin,
col="blue", ylim = c(0,1500))
lines(density(msXZwrdq),lwd=2)
hist(msXZwor, las=1, prob=TRUE, xlim=c(RatioLrBnd, RatioUpBnd), breaks = RatioBin,
 col="blue", ylim = c(0,1500))
lines(density(msXZworq),lwd=2)
hist(msXZwru, las=1, prob=TRUE, xlim=c(RatioLrBnd, RatioUpBnd), breaks = RatioBin,
col="blue", ylim = c(0,1500))
lines(density(msXZwruq),lwd=2)

###############
#Total testing#
###############
LowerBnd =min(QLowBnd, ChakLrBnd, RatioLrBnd)
UpperBnd =max(QUpBnd, ChakUpBnd, RatioUpBnd)
TotalBin = seq(LowerBnd -0.01, UpperBnd +0.01, by = 0.0001)
hist(msYXwrdq, las=1, prob=TRUE, xlim=c(LowerBnd , UpperBnd ), breaks = TotalBin ,
col="red", ylim = c(0,1500))
lines(density(msYXwrdq),lwd=2)
hist(msYXworq, las=1, prob=TRUE, xlim=c(LowerBnd , UpperBnd ), breaks = TotalBin ,
 col="red", ylim = c(0,1500))
lines(density(msYXworq),lwd=2)
hist(msYXwruq, las=1, prob=TRUE, xlim=c(LowerBnd , UpperBnd ), breaks = TotalBin ,
col="red", ylim = c(0,1500))
lines(density(msYXwruq),lwd=2)

hist(msZYwrdq, las=1, prob=TRUE, xlim=c(LowerBnd , UpperBnd ), breaks = TotalBin ,
 col="red", ylim = c(0,1500))
lines(density(msZYwrdq),lwd=2)
hist(msZYworq, las=1, prob=TRUE, xlim=c(LowerBnd , UpperBnd ), breaks = TotalBin ,
 col="red", ylim = c(0,1500))
lines(density(msZYworq),lwd=2)
hist(msZYwruq, las=1, prob=TRUE, xlim=c(LowerBnd , UpperBnd ), breaks = TotalBin ,
col="red", ylim = c(0,1500))
lines(density(msZYwruq),lwd=2)

hist(msXZwrdq, las=1, prob=TRUE, xlim=c(LowerBnd , UpperBnd ), breaks = TotalBin ,
col="red", ylim = c(0,1500))
lines(density(msXZwrdq),lwd=2)
hist(msXZworq, las=1, prob=TRUE, xlim=c(LowerBnd , UpperBnd ), breaks = TotalBin ,
col="red", ylim = c(0,1500))
lines(density(msXZworq),lwd=2)
hist(msXZwruq, las=1, prob=TRUE, xlim=c(LowerBnd , UpperBnd ), breaks = TotalBin ,
 col="red", ylim = c(0,1500))
lines(density(msXZwruq),lwd=2)

hist(msYXwrdc1, las=1, prob=TRUE, xlim=c(LowerBnd , UpperBnd ), breaks = TotalBin ,
col="green", ylim = c(0,1500))
lines(density(msYXwrdc1),lwd=2)
hist(msYXworc1, las=1, prob=TRUE, xlim=c(LowerBnd , UpperBnd ), breaks = TotalBin ,
col="green", ylim = c(0,1500))
lines(density(msYXworc1),lwd=2)
hist(msYXwruc1, las=1, prob=TRUE, xlim=c(LowerBnd , UpperBnd ), breaks = TotalBin ,
```

```
col="green", ylim = c(0,1500))
lines(density(msYXwruc1),lwd=2)
hist(msYXwrdc2, las=1, prob=TRUE, xlim=c(LowerBnd , UpperBnd ), breaks = TotalBin ,
 col="green", ylim = c(0,1500))
lines(density(msYXwrdc2),lwd=2)
hist(msYXworc2, las=1, prob=TRUE, xlim=c(LowerBnd , UpperBnd ), breaks = TotalBin ,
col="green", ylim = c(0,1500))
lines(density(msYXworc2),lwd=2)
hist(msYXwruc2, las=1, prob=TRUE, xlim=c(LowerBnd , UpperBnd ), breaks = TotalBin ,
col="green", ylim = c(0,1500))
lines(density(msYXwruc2),lwd=2)

hist(msZYwrdc1, las=1, prob=TRUE, xlim=c(LowerBnd , UpperBnd ), breaks = TotalBin ,
 col="green", ylim = c(0,1500))
lines(density(msZYwrdc1),lwd=2)
hist(msZYworc1, las=1, prob=TRUE, xlim=c(LowerBnd , UpperBnd ), breaks = TotalBin ,
col="green", ylim = c(0,1500))
lines(density(msZYworc1),lwd=2)
hist(msZYwruc1, las=1, prob=TRUE, xlim=c(LowerBnd , UpperBnd ), breaks = TotalBin ,
 col="green", ylim = c(0,1500))
lines(density(msZYwruc1),lwd=2)
hist(msZYwrdc2, las=1, prob=TRUE, xlim=c(LowerBnd , UpperBnd ), breaks = TotalBin ,
 col="green", ylim = c(0,1500))
lines(density(msZYwrdc2),lwd=2)
hist(msZYworc2, las=1, prob=TRUE, xlim=c(LowerBnd , UpperBnd ), breaks = TotalBin ,
col="green", ylim = c(0,1500))
lines(density(msZYworc2),lwd=2)
hist(msZYwruc2, las=1, prob=TRUE, xlim=c(LowerBnd , UpperBnd ), breaks = TotalBin ,
 col="green", ylim = c(0,1500))
lines(density(msZYwruc2),lwd=2)

hist(msXZwrdc1, las=1, prob=TRUE, xlim=c(LowerBnd , UpperBnd ), breaks = TotalBin ,
col="green", ylim = c(0,1500))
lines(density(msXZwrdc1),lwd=2)
hist(msXZworc1, las=1, prob=TRUE, xlim=c(LowerBnd , UpperBnd ), breaks = TotalBin ,
 col="green", ylim = c(0,1500))
lines(density(msXZworc1),lwd=2)
hist(msXZwruc1, las=1, prob=TRUE, xlim=c(LowerBnd , UpperBnd ), breaks = TotalBin ,
col="green", ylim = c(0,1500))
lines(density(msXZwruc1),lwd=2)
hist(msXZwrdc2, las=1, prob=TRUE, xlim=c(LowerBnd , UpperBnd ), breaks = TotalBin ,
col="green", ylim = c(0,1500))
lines(density(msXZwrdc2),lwd=2)
hist(msXZworc2, las=1, prob=TRUE, xlim=c(LowerBnd , UpperBnd ), breaks = TotalBin ,
col="green", ylim = c(0,1500))
lines(density(msXZworc2),lwd=2)
hist(msXZwruc2, las=1, prob=TRUE, xlim=c(LowerBnd , UpperBnd ), breaks = TotalBin ,
 col="green", ylim = c(0,1500))
lines(density(msXZwruc2),lwd=2)

hist(msYXwrd, las=1, prob=TRUE, xlim=c(LowerBnd , UpperBnd ), breaks = TotalBin ,
 col="blue", ylim = c(0,1500))
lines(density(msYXwrd),lwd=2)
hist(msYXwor, las=1, prob=TRUE, xlim=c(LowerBnd , UpperBnd ), breaks = TotalBin ,
 col="blue", ylim = c(0,1500))
lines(density(msYXwor),lwd=2)
hist(msYXwru, las=1, prob=TRUE, xlim=c(LowerBnd , UpperBnd ), breaks = TotalBin ,
 col="blue", ylim = c(0,1500))
lines(density(msYXwru),lwd=2)
hist(msZYwrd, las=1, prob=TRUE, xlim=c(LowerBnd , UpperBnd ), breaks = TotalBin ,
```

```
 col="blue", ylim = c(0,1500))
lines(density(msZYwrdq),lwd=2)
hist(msZYwor, las=1, prob=TRUE, xlim=c(LowerBnd , UpperBnd ), breaks = TotalBin ,
col="blue", ylim = c(0,1500))
lines(density(msZYworq),lwd=2)
hist(msZYwru, las=1, prob=TRUE, xlim=c(LowerBnd , UpperBnd ), breaks = TotalBin ,
col="blue", ylim = c(0,1500))
lines(density(msZYwruq),lwd=2)
hist(msXZwrd, las=1, prob=TRUE, xlim=c(LowerBnd , UpperBnd ), breaks = TotalBin ,
col="blue", ylim = c(0,1500))
lines(density(msXZwrdq),lwd=2)
hist(msXZwor, las=1, prob=TRUE, xlim=c(LowerBnd , UpperBnd ), breaks = TotalBin ,
col="blue", ylim = c(0,1500))
lines(density(msXZworq),lwd=2)
hist(msXZwru, las=1, prob=TRUE, xlim=c(LowerBnd , UpperBnd ), breaks = TotalBin ,
 col="blue", ylim = c(0,1500))
lines(density(msXZwruq),lwd=2)
}
```