**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**Part1) Central Limit Theorem**

**Code Section 1.a:**

#Dataset

boston <- read.csv(

"https://people.bu.edu/kalathur/datasets/bostonCityEarnings.csv",

colClasses = c("character", "character", "character", "integer", "character"))

#------------------------------------------------------------------------------

#a)histogram of earnings

breaks\_hist <- seq(0,500000,by= 50000)

options(scipen = 4)

par(mar=c(5,5,2,2))

hist\_plot <- hist(boston$Earnings, breaks = breaks\_hist,xlab = "Earnings",

ylab ="Number of People(Frequency)")

#Mean Earnings

mean(boston$Earnings)

#Standard deviation of Earnings

sd(boston$Earnings)

**Console Section 1.a:**

#Dataset

> boston <- read.csv(

+ "https://people.bu.edu/kalathur/datasets/bostonCityEarnings.csv",

+ colClasses = c("character", "character", "character", "integer", "character"))

> #------------------------------------------------------------------------------

> #a)histogram of earnings

> breaks\_hist <- seq(0,500000,by= 50000)

> options(scipen = 4)

> par(mar=c(5,5,2,2))

> hist\_plot <- hist(boston$Earnings, breaks = breaks\_hist,xlab = "Earnings",

+ ylab ="Number of People(Frequency)")

> #Mean Earnings

> mean(boston$Earnings)

**[1] 108680.9**

> #Standard deviation of Earnings

> sd(boston$Earnings)

**[1] 50474.7**

**Plot section 1.a:**

A picture containing diagram, screenshot, technical drawing, line

Description automatically generated

**Inferences:**

1. The data set looks skewed (not normal).

2. Most of the people belongs to 50k to 150k earnings.

3. there is small number of people that has income beyond 300k.

4. when we do box plot, we can find there are outlines towards the upper end.

-----------------------------------------------------------------------------------------------------------------------------

**Code section 1.b:**

library(sampling)

#b)

set.seed(7356)

sample.1000 <- 1000

sample\_size <- 10

xbar.10 <- numeric(sample.1000)

for (i in 1:sample.1000) {

s10\_rows <- sample(nrow(boston),10,replace = FALSE) #using sample() method

s10\_sample <- boston[s10\_rows,] #mapping selected rows to Boston dataset

xbar.10[i] <-mean(s10\_sample$Earnings) # making 1000 samples of size 10

}

mean.1000\_10 <- mean(xbar.10)

sd.1000\_10 <- sd(xbar.10)

par(mar=(c(2,2,2,2)))

hist(xbar.10,xlab = "Earnings", ylab = "Frequency", main = "Histogram of sample means of size 10",

ylim = c(0,250))

**Console section 1.b:**

> library(sampling)

#b)

set.seed(7356)

sample.1000 <- 1000

sample\_size <- 10

xbar.10 <- numeric(sample.1000)

for (i in 1:sample.1000) {

s10\_rows <- sample(nrow(boston),10,replace = FALSE) #using sample() method

s10\_sample <- boston[s10\_rows,] #mapping selected rows to Boston dataset

xbar.10[i] <-mean(s10\_sample$Earnings) # making 1000 samples of size 10

}

mean.1000\_10 <- mean(xbar.10)

#mean of sample size 10 of 1000 samples

mean.1000\_10

sd.1000\_10 <- sd(xbar.10)

#Standard deviation of sample size 10 of 1000 samples.

sd.1000\_10

par(mar=(c(5,5,2,2)))

hist(xbar.10,xlab = "Earnings", ylab = "Frequency", main = "Histogram of sample means of size 10",

ylim = c(0,250))

**Console section 1.b:**

> set.seed(7356)

> sample.1000 <- 1000

> sample\_size <- 10

> xbar.10 <- numeric(sample.1000)

> for (i in 1:sample.1000) {

+ s10\_rows <- sample(nrow(boston),10,replace = FALSE) #using sample() method

+ s10\_sample <- boston[s10\_rows,] #mapping selected rows to Boston dataset

+

+ xbar.10[i] <-mean(s10\_sample$Earnings) # making 1000 samples of size 10

+

+ }

> mean.1000\_10 <- mean(xbar.10)

> #mean of sample size 10 of 1000 samples

> mean.1000\_10

**[1] 108216.2**

> sd.1000\_10 <- sd(xbar.10)

> #Standard deviation of sample size 10 of 1000 samples.

> sd.1000\_10

**[1] 16297.48**

> par(mar=(c(5,5,2,2)))

> hist(xbar.10,xlab = "Earnings", ylab = "Frequency", main = "Histogram of sample means of size 10",

+ ylim = c(0,250))

**Plot section 1.b:**

A picture containing diagram, plot, screenshot, line

Description automatically generated

------------------------------------------------------------------------------------------------------------------------

**Code section 1.c:**

set.seed(7356)

sample\_size\_40 <- 40 #for sample size 40

xbar.40 <- numeric(sample.1000) #initializing list of 1000 0s

for (i in 1:sample.1000) {

s40\_rows <- sample(nrow(boston),40,replace = FALSE) #getting random 40 rows out of original dataset

s40\_sample <- boston[s40\_rows,] # mapping those sample data to original data

xbar.40[i] <-mean(s40\_sample$Earnings) # replacing those

}

#mean of the sample size of 40 of 1000 samples

mean.1000\_40 <- mean(xbar.40)

mean.1000\_40

#SD of the sample size of 40 of 1000 samples

sd.1000\_40 <- sd(xbar.40)

sd.1000\_40

#histogram of sample means

par(mar=c(5,5,2,2))

hist(xbar.40,main = "Histogram of sample means of size 40",xlab = "Earnings",ylab = "Frequency")

**Console section 1.c:**

> #c)

> set.seed(7356)

> sample\_size\_40 <- 40 #for sample size 40

> xbar.40 <- numeric(sample.1000) #initializing list of 1000 0s

> for (i in 1:sample.1000) {

+ s40\_rows <- sample(nrow(boston),40,replace = FALSE) #getting random 40 rows out of original dataset

+ s40\_sample <- boston[s40\_rows,] # mapping those sample data to original data

+

+ xbar.40[i] <-mean(s40\_sample$Earnings) # replacing those

+

+ }

> #mean of the sample size of 40 of 1000 samples

> mean.1000\_40 <- mean(xbar.40)

> mean.1000\_40

[1] 108335.2

> #SD of the sample size of 40 of 1000 samples

> sd.1000\_40 <- sd(xbar.40)

> sd.1000\_40

**[1] 8013.736**

> #histogram of sample means

> par(mar=c(5,5,2,2))

> hist(xbar.40,main = "Histogram of sample means of size 40",xlab = "Earnings",ylab = "Frequency")

**Plot section 1.c:**

A picture containing diagram, line, plot, design

Description automatically generated

----------------------------------------------------------------------------------------------------------------------------

**Code section 1.d:**

#d)

# means of three type of distribution

mean\_combine <- c(Original=mean(boston$Earnings),Sample\_10=mean.1000\_10,

Sample\_40=mean.1000\_40)

mean\_combine

# it can be seen that mean of the all three type of data are almost same

sd\_combine <- c(Orignal=sd(boston$Earnings),Sample\_10=sd.1000\_10,

Sample\_40=sd.1000\_40)

sd\_combine

# however standard deviation of all three type of data are different. sd of the

# 1000 mean sample of size 10 is less diverse than original. while data with sample

# size 40 has SD way less than that of sample size of 10.

#theoretical values of sd can be calculated by using formula where sd of origincal

# sd divided by square root of sample sizes

theoritical\_sd <- sd\_combine[1]/c(sqrt(10),sqrt(40))

theoritical\_sd

**Console section 1.d:**

> #d)

> # means of three type of distribution

> mean\_combine <- c(Original=mean(boston$Earnings),Sample\_10=mean.1000\_10,

+ Sample\_40=mean.1000\_40)

> mean\_combine

Original Sample\_10 Sample\_40

108680.9 108216.2 108335.2

>

> # it can be seen that mean of the all three type of data are almost same

> sd\_combine <- c(Orignal=sd(boston$Earnings),Sample\_10=sd.1000\_10,

+ Sample\_40=sd.1000\_40)

> sd\_combine

Orignal Sample\_10 Sample\_40

50474.701 16297.481 8013.736

> # however standard deviation of all three type of data are different. sd of the

> # 1000 mean sample of size 10 is less diverse than original. while data with sample

> # size 40 has SD way less than that of sample size of 10.

>

> #theoretical values of sd can be calculated by using formula where sd of origincal

> # sd divided by square root of sample sizes

>

> theoritical\_sd <- sd\_combine[1]/c(sqrt(10),sqrt(40))

> theoritical\_sd

[1] 15961.502 7980.751

**Inferences:**

1.From above original and theoretical values of population data and samples data followed the Central Limit Theorem.

2. as the sample sizes increases, samples will have small SD which means sample means are less distributed.

3. Mean of the original data set as well as means sample of size 10 and 40 remains similar. Theoretically, this should have same values when all possible samples are drawn but we took 1000 samples only.

------------------------------------------------------------------------------------------------------------------------

**Part2) Central Limit Theorem – Negative Binomial distribution**

**Code section 2.a:**

set.seed(7356)

#a)

random.1000 <-rnbinom(1000,3,0.5)

#checking frequency of the number

Freq\_1000 <- table(random.1000)

barplot(Freq\_1000,xlab = "Numbers",ylab = "Frequency",main = "Frequency of

distinct values of distirbution")

**Console section 2.a:**

> #Part 2

> set.seed(7356)

> #a)

> random.1000 <-rnbinom(1000,3,0.5)

> #checking frequency of the number

> Freq\_1000 <- table(random.1000)

> barplot(Freq\_1000,xlab = "Numbers",ylab = "Frequency",main = "Frequency of

+ distinct values of distirbution")

**Plot section 2.a:**

A picture containing diagram, screenshot, plot, text

Description automatically generated

-----------------------------------------------------------------------------------------------------------------

Code section 2.b:

#b)

# four sample sizes

sample\_sizes <- c(10,20,30,40)

# 5000 samples of each of sample size types

xbar.5000 <- numeric(5000)

list\_mean <- c() # To store list of all four means

list\_SD <- c() # to store list of all four SD

par(mfrow = c(2,2))

for (size in sample\_sizes) {

for (i in 1:5000) {

xbar.5000[i] <- mean(sample(random.1000,size,replace = FALSE))

}

hist(t(xbar.5000),prob=TRUE,

breaks = 15,main = paste("Sample Size =", size),xlab = "Numbers")

cat("Sample Size = ",size, " Mean = ", mean(xbar.5000),

" SD = ", sd(xbar.5000), "\n")

list\_mean <- c(list\_mean,mean(xbar.5000))

list\_SD <- c(list\_SD,sd(xbar.5000))

}

**Console section 2.b:**

> # four sample sizes

> sample\_sizes <- c(10,20,30,40)

> # 5000 samples of each of sample size types

> xbar.5000 <- numeric(5000)

> list\_mean <- c() # To store list of all four means

> list\_SD <- c() # to store list of all four SD

> par(mfrow = c(2,2))

> for (size in sample\_sizes) {

+ for (i in 1:5000) {

+ xbar.5000[i] <- mean(sample(random.1000,size,replace = FALSE))

+

+ }

+ hist(t(xbar.5000),prob=TRUE,

+ breaks = 15,main = paste("Sample Size =", size),xlab = "Numbers")

+

+ cat("Sample Size = ",size, " Mean = ", mean(xbar.5000),

+ " SD = ", sd(xbar.5000), "\n")

+ list\_mean <- c(list\_mean,mean(xbar.5000))

+ list\_SD <- c(list\_SD,sd(xbar.5000))

+ }

Sample Size = 10 Mean = 2.98154 SD = 0.7476477

Sample Size = 20 Mean = 3.02617 SD = 0.5456392

Sample Size = 30 Mean = 3.001313 SD = 0.435892

Sample Size = 40 Mean = 3.007995 SD = 0.3800001

**Plot section 2.b:**

A picture containing diagram, sketch

Description automatically generated

-------------------------------------------------------------------------------------------------------------------

Code section 2.c:

#c)

#from above calculation

#mean from a

mean(random.1000)

#SD from a

sd(random.1000)

#means from b

list\_mean

#SDs from b

list\_SD

# from above, we can conclude that means of population and sample are almost similar

# While SD of sample means is lower than population. also, In sample means, the

# data variability decreases as the size of the sample increases.

#Theoretical sample sd calculation can also be done

samples.SD <- sd(random.1000)/sqrt(sample\_sizes)

samples.SD

# sample SD for different sizes is almost same as the theoretical SD we get from

**Console section 2.c:**

#c)

> #from above calculation

> #mean from a

> mean(random.1000)

[1] 3.013

> #SD from a

> sd(random.1000)

[1] 2.434082

> #means from b

> list\_mean

[1] 2.981540 3.026170 3.001313 3.007995

> #SDs from b

> list\_SD

[1] 0.7476477 0.5456392 0.4358920 0.3800001

> # from above, we can conclude that means of population and sample are almost similar

> # while SD of sample means is lower than population. also, In sample means, the

> # data variability decreases as the size of the sample increases.

> #Theoretical sample sd calculation can also be done

> samples.SD <- sd(random.1000)/sqrt(sample\_sizes)

> samples.SD

[1] 0.7697244 0.5442773 0.4444006 0.3848622

**Inferences:**

1.From above, we can conclude that means of population and sample are almost similar

2. While SD of sample means is lower than population. also, In sample means, the

3.Data variability decreases as the size of the sample increases and graphs looks more like Normal distribution plot as the sample sizes increases.

**4.**sample SD for different sizes are almost same as the theoretical SD we get from formula.

-----------------------------------------------------------------------------------------------------------------------

**Part3) Sampling**

**Code part 3.a:**

#number of employees working in each department can be done by using table

table.name <- table(boston$Department)

#now sorting the value and selecting top5 department.

top5\_depart <- sort(table.name,decreasing = TRUE)[1:5]

top5\_depart

#mapping with original data set

subset\_top5 <- subset(boston,boston$Department %in% names(top5\_depart))

#a)

library(sampling)

set.seed(7356)

sample.with.replace <- srswr(50,nrow(subset\_top5)) # using R function

row.number <- (1:nrow(subset\_top5))[sample.with.replace!=0] # mapping with top 5 dataset's rows

subset.with.replace <- subset\_top5[row.number,] # getting subset from top5 data set

#frequencies

table(subset.with.replace$Department)

# percentage with respect to sample size

table(subset.with.replace$Department)/50

#Alternatively

prop.depart.a <- prop.table(table(subset.with.replace$Department))

prop.depart.a

for (m in 1:length(prop.depart.a)) {

cat(names(prop.depart.a)[m],"will have", prop.depart.a[m]\*100,"%", "\n")

}

**Console Part 3.a:**

> #part 3

> #number of employees working in each department can be done by using table

> table.name <- table(boston$Department)

> #now sorting the value and selecting top5 department.

> top5\_depart <- sort(table.name,decreasing = TRUE)[1:5]

> top5\_depart

Boston Police Department Boston Fire Department BPS Special Education BPS Facility Management Boston Public Library

2732 1672 611 415 384

>

> #mapping with original data set

> subset\_top5 <- subset(boston,boston$Department %in% names(top5\_depart))

> #a)

> library(sampling)

> set.seed(7356)

> sample.with.replace <- srswr(50,nrow(subset\_top5)) # using R function

> row.number <- (1:nrow(subset\_top5))[sample.with.replace!=0] # mapping with top 5 dataset's rows

> subset.with.replace <- subset\_top5[row.number,] # getting subset from top5 data set

> #frequencies

> table(subset.with.replace$Department)

Boston Fire Department Boston Police Department Boston Public Library BPS Facility Management BPS Special Education

12 27 3 2 6

> # percentage with respect to sample size

> table(subset.with.replace$Department)/50

Boston Fire Department Boston Police Department Boston Public Library BPS Facility Management BPS Special Education

0.24 0.54 0.06 0.04 0.12

> #Alternatively

> prop.depart.a <- prop.table(table(subset.with.replace$Department))

> prop.depart.a

Boston Fire Department Boston Police Department Boston Public Library BPS Facility Management BPS Special Education

0.24 0.54 0.06 0.04 0.12

>

> for (m in 1:length(prop.depart.a)) {

+

+ cat(names(prop.depart.a)[m],"will have", prop.depart.a[m]\*100,"%", "\n")

+ }

Boston Fire Department will have 24 %

Boston Police Department will have 54 %

Boston Public Library will have 6 %

BPS Facility Management will have 4 %

BPS Special Education will have 12 %

-------------------------------------------------------------------------------------------------------------------

**Code section 2.b:**

#b)

set.seed(7356)

inclusion.prob <- inclusionprobabilities(subset\_top5$Earnings,50)

length(inclusion.prob)

unequal.probab <- UPsystematic(inclusion.prob)

head(unequal.probab)

new.samples.50 <- getdata(subset\_top5,unequal.probab)

head(new.samples.50)

#alternatively we can map the selected rows with subset\_top5 as follows

new.samples <- (subset\_top5)[unequal.probab !=0,]

#frequency of employee in each department can be calculated by

frequency\_depart <- table(new.samples.50$Department)

#calculating proportion

prop.depart.b <-prop.table(frequency\_depart)

for (i in 1:length(prop.depart.b)) {

cat(names(prop.depart.b)[i],"will have", prop.depart.b[i]\*100,"%", "\n")

}

**Console section2.b:**

#b)

> set.seed(7356)

> inclusion.prob <- inclusionprobabilities(subset\_top5$Earnings,50)

> length(inclusion.prob)

[1] 5814

> unequal.probab <- UPsystematic(inclusion.prob)

> head(unequal.probab)

[1] 0 0 0 0 0 0

> new.samples.50 <- getdata(subset\_top5,unequal.probab)

> head(new.samples.50)

ID\_unit NAME Department Title Earnings ZipCode

112 44 Alessandro,Dennis Charles Boston Fire Department Fire Fighter 145968 02132

412 164 Aylward,Michael Anthony Boston Fire Department Fire Fighter 137181 02118

720 290 Bent,Thomas Boston Police Department Police Officer 134682 02132

965 405 Bowen,Raymond A Boston Police Department Police Officer 176469 02136

1185 508 Brown,Nytisha D Boston Police Department Police Officer 176367 02021

1424 623 Caggiano,Joseph Albert Boston Police Department Police Officer 141363 02128

> #alternatively we can map the selected rows with subset\_top5 as follows

> new.samples <- (subset\_top5)[unequal.probab !=0,]

> #frequency of employee in each department can be calculated by

> frequency\_depart <- table(new.samples.50$Department)

> #calculating proportion

> prop.depart.b <-prop.table(frequency\_depart)

>

> for (i in 1:length(prop.depart.b)) {

+

+ cat(names(prop.depart.b)[i],"will have", prop.depart.b[i]\*100,"%", "\n")

+ }

Boston Fire Department will have 44 %

Boston Police Department will have 48 %

Boston Public Library will have 2 %

BPS Facility Management will have 2 %

BPS Special Education will have 4 %

-------------------------------------------------------------------------------------------------------------

Code section 3.c:

#c)

set.seed(7356)

#ordering the data using Department variable

order.department <- order(subset\_top5$Department)

#mapping to dataset according to ordered rows

ordered.top5 <- subset\_top5[order.department,]

#finding relative frequency of employee in each deparment

frequency.top5 <- table(ordered.top5$Department)

#finding proportions in each department based on their employee numbers.

prop.50 <- round(50\*frequency.top5/sum(frequency.top5))

sum(prop.50)

50\*frequency.top5/sum(frequency.top5)

# while using sum, it adds up to 49 only but we are supposed to have sample of size

# 50.So I need to find the department that can be added one more. here in proportion

# table Boston police department have 23.495 which would have 24 if it had 0.005 more value.

# so this is the closest department that can be used to add one more values and make it 24 instead 23.

#changing second value to 24 from 23.

prop.50[2] <- 24

#now total number of sample becomes 50

sum(prop.50)

st.d <-strata(ordered.top5,stratanames = "Department",size = prop.50,

method = "srswor",description =TRUE )

#now retrieving those 50 samples as we get from strata() method using getdata() method.

sample.d <- getdata(subset\_top5,st.d)

#checking the frequency

table(sample.d$Department)

#finding proportion of each department according to number of employee in each department

prop.table.c <- round(prop.table(prop.50),2)

prop.table.c

for (n in 1:length(prop.50)) {

cat(names(prop.50)[n],"will have", prop.table.c[n]\*100,"%", "\n")

}

**Console section 3.c:**

set.seed(7356)

> #ordering the data using Department variable

> order.department <- order(subset\_top5$Department)

> #mapping to dataset according to ordered rows

> ordered.top5 <- subset\_top5[order.department,]

> #finding relative frequency of employee in each deparment

> frequency.top5 <- table(ordered.top5$Department)

>

> #finding proportions in each department based on their employee numbers.

> prop.50 <- round(50\*frequency.top5/sum(frequency.top5))

> sum(prop.50)

[1] 49

> 50\*frequency.top5/sum(frequency.top5)

Boston Fire Department Boston Police Department Boston Public Library BPS Facility Management BPS Special Education

14.379085 23.495012 3.302374 3.568971 5.254558

> # while using sum, it adds up to 49 only but we are supposed to have sample of size

> # 50.So I need to find the department that can be added one more. here in proportion

> # table Boston police department have 23.495 which would have 24 if it had 0.005 more value.

> # so this the closest department that can be used to add one more values and make it 24 instead 23.

> #changing second value to 24 from 23.

> prop.50[2] <- 24

> #now total number of sample becomes 50

> sum(prop.50)

[1] 50

>

> st.d <-strata(ordered.top5,stratanames = "Department",size = prop.50,

+ method = "srswor",description =TRUE )

Stratum 1

Population total and number of selected units: 1672 14

Stratum 2

Population total and number of selected units: 2732 24

Stratum 3

Population total and number of selected units: 384 3

Stratum 4

Population total and number of selected units: 415 4

Stratum 5

Population total and number of selected units: 611 5

Number of strata 5

Total number of selected units 50

> #now retrieving those 50 samples as we get from strata() method using getdata() method.

> sample.d <- getdata(subset\_top5,st.d)

> #checking the frequency

> table(sample.d$Department)

Boston Fire Department Boston Police Department Boston Public Library BPS Facility Management BPS Special Education

14 24 3 4 5

>

> #finding proportion of each department according to number of employee in each department

> prop.table.c <- round(prop.table(prop.50),2)

> prop.table.c

Boston Fire Department Boston Police Department Boston Public Library BPS Facility Management BPS Special Education

0.28 0.48 0.06 0.08 0.10

>

> for (n in 1:length(prop.50)) {

+

+ cat(names(prop.50)[n],"will have", prop.table.c[n]\*100,"%", "\n")

+ }

Boston Fire Department will have 28 %

Boston Police Department will have 48 %

Boston Public Library will have 6 %

BPS Facility Management will have 8 %

BPS Special Education will have 10 %

--------------------------------------------------------------------------------------------------------------------------

**Code section 3.d:**

#d)

#list of mean of four samples

list.mean.4 <- c(mean(subset\_top5$Earnings),mean(subset.with.replace$Earnings),

mean(new.samples$Earnings),mean(sample.d$Earnings))

list.mean.4

#mean of original

mean(boston$Earnings)

**Console section 3.d:**

#list of mean of four samples

> list.mean.4 <- c(mean(subset\_top5$Earnings),mean(subset.with.replace$Earnings),

+ mean(new.samples$Earnings),mean(sample.d$Earnings))

> list.mean.4

[1] 133921.4 127668.4 158944.3 142998.9

> #mean of original

> mean(boston$Earnings)

[1] 108680.9

**Comparison/inferences:**

Here, mean of boston dataset is lower than means of fours, in which one dataset is subset of the original dataset where only top5 departments are included based on their number of employees. While other 3 are samples drawn from 3 different sampling methods. From this, it can be concluded that the employees in top5 department, which is based on number of employees working, also have higher earnings than other department that is why mean of the earnings from top 5 department higher than means of the main dataset. Here, mean of the samples drawn using simple random sampling with replacement are closer to the mean of the earnings of original boston dataset. However, if we compare among three types of sampling techniques. Stratified sampling using proportional sizes is closer to the top 5 department subset as mean earning of top 5 subset is closer than with other.

-----------------------------------------------------------------------------------------------------------------------------

**The End:**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*