PROBABILITY AND STATISTICS (UCS410)
Experiment 1: Basics Of R Programming

Name: Jashan Arora
Roll No: 102003206
Group: 3CO9

(1) Create a vector c = [5,10,15,20,25,30] and write a program which returns the maximum and minimum of this vector.

```r
x=c(5,10,15,20,25,30)
print(paste("Maximum Value of x is ",max(x)))
print(paste("Minimum Value of x is ",min(x)))
```

```
> source("~/Documents/PROB Assignments/ASSIGN1/Q1.R")
[1] "Maximum Value of x is  30"
[1] "Minimum Value of x is  5"
```

(2) Write a program in R to find factorial of a number by taking input from user. Please print error message if the input number is negative.

```r
n=as.integer(readline(prompt = "Enter a Number: "))
fact=1
num=n
if(num < 0)
{
  print("The Input Number is Negative")
} else {
  while(num)
  {
    fact=fact*num
    num=num-1
  }
  print(paste("Factorial of ",n," is ",fact))
}
```

```
> source("~/Documents/PROB Assignments/ASSIGN1/Q2.R")
Enter a Number: 6
[1] "Factorial of  6  is  720"
```

(3) Write a program to write first n terms of a Fibonacci sequence. You may take n as an input from the user.

```r
n=as.integer(readline(prompt = "Enter a Number: "))
a=0
b=1
if(n<0)
{
  print("Number of terms can't be Negative")
} else
{
if(n==1)
{
print(paste("First",n,"terms of Fibbonacci Sequence is ",a))
} else
{
  cat(paste("First",n,"terms of Fibbonacci Sequence is ",a,b,""))
  for(i in 1:(n-2))
  {
    c=a+b
    cat(c)
    cat(" ")
    a=b
    b=c
  }
}
}
```

```
> source("~/Documents/PROB Assignments/ASSIGN1/Q3.R")
Enter a Number: 8
First 8 terms of Fibbonacci Sequence is  0 1 1 2 3 5 8 13
```

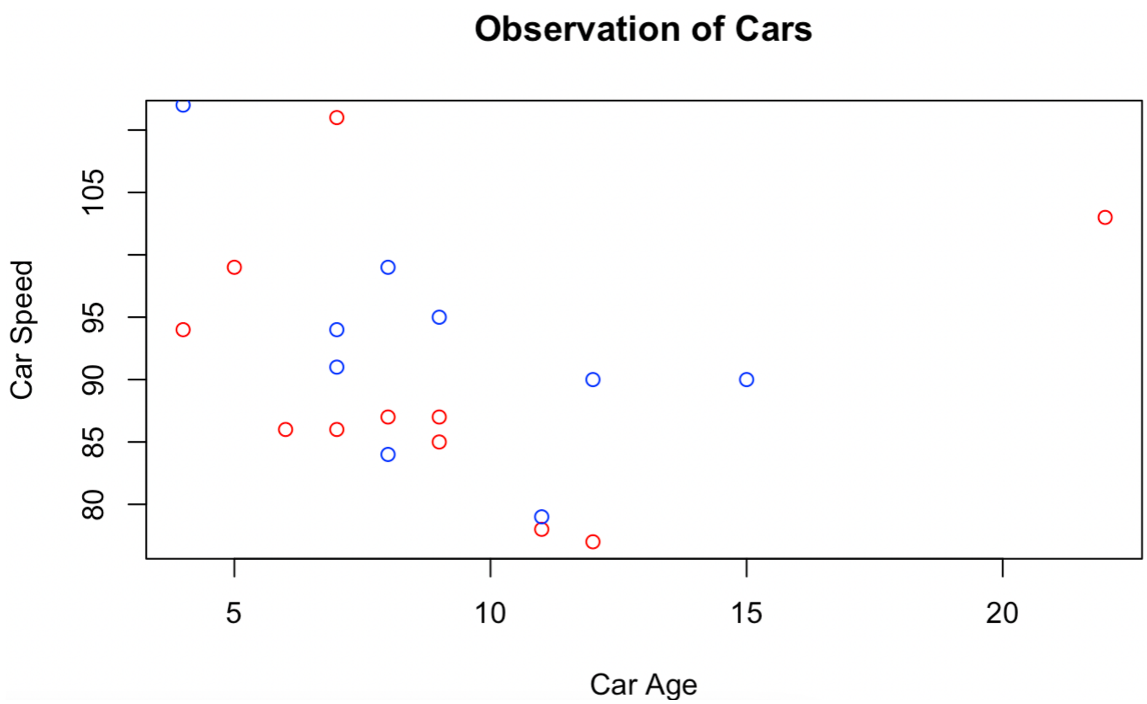
(4) Write an R program to make a simple calculator which can add, subtract, multiply and divide.

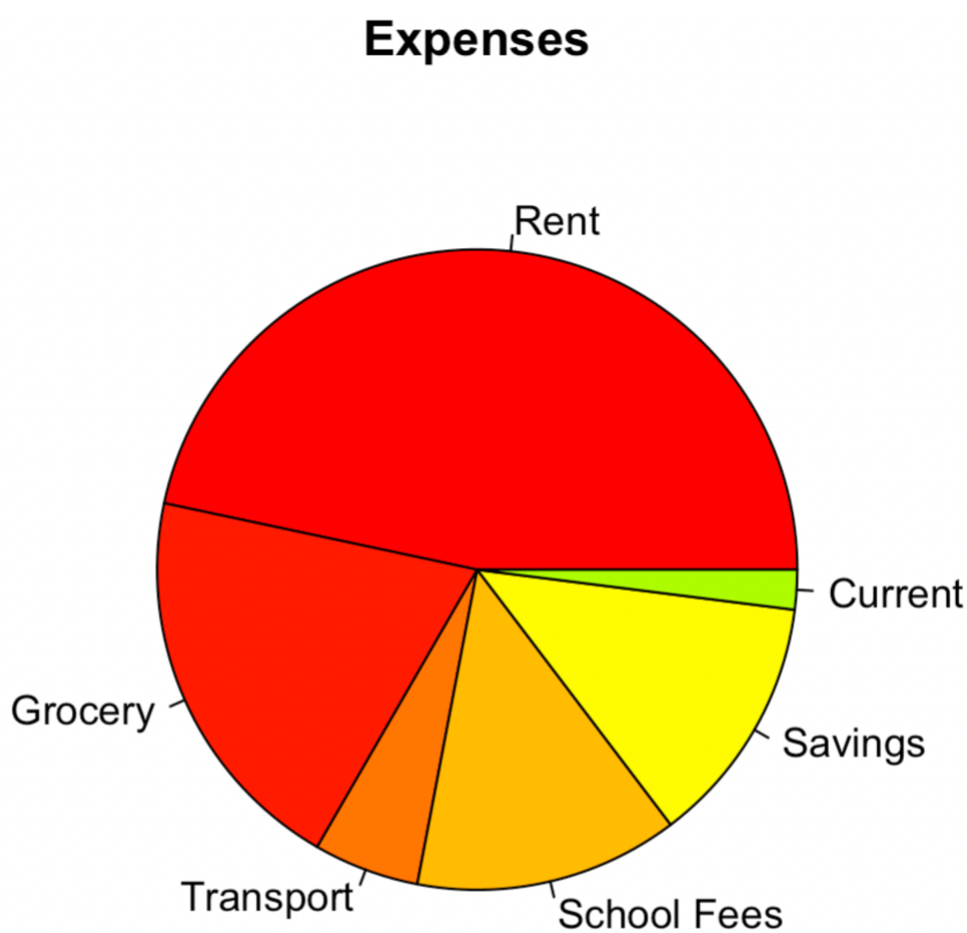
```r
add <- function(x,y){
  return (x+y)
}
sub <- function(x,y){
  return (x-y)
}
mult <- function(x,y){
  return (x*y)
}
div <- function(x,y){
  return (x/y)
}

print("Calculator....")
print("1 for Addition")
print("2 for Substraction")
print("3 for Multiplication")
print("4 for Divison")
ch=as.integer(readline(prompt = "Enter your choice : "))
a=as.integer(readline(prompt = "Enter Number 1 : "))
b=as.integer(readline(prompt = "Enter Number 2 : "))
result=switch(ch,
              "1"=add(a,b),
              "2"=sub(a,b),
              "3"=mult(a,b),
              "4"=div(a,b))
print(paste("Answer is : ",result))
```

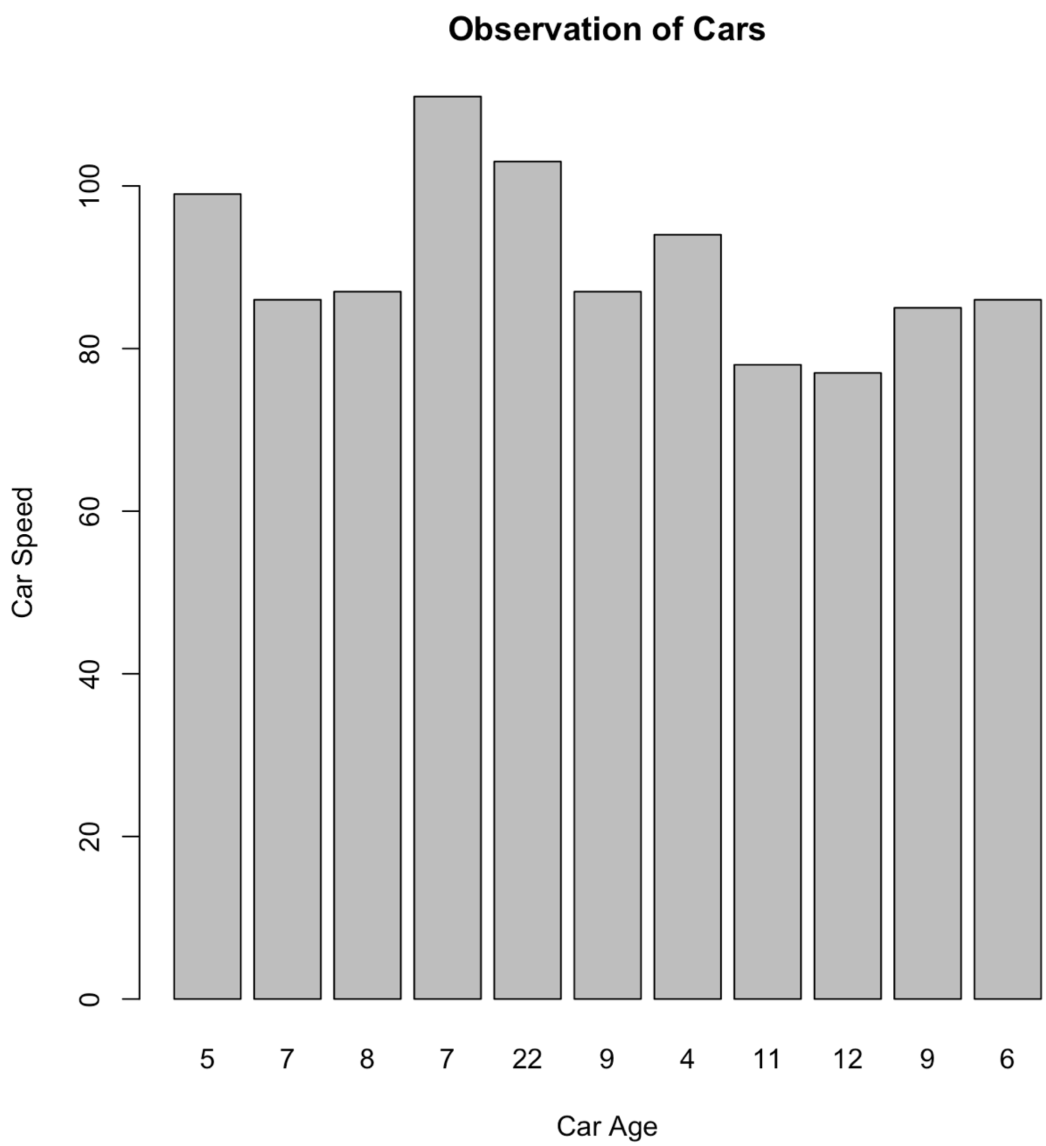
```
> source("~/Documents/PROB Assignments/ASSIGN1/Q4.R")
[1] "Calculator...."
[1] "1 for Addition"
[1] "2 for Substraction"
[1] "3 for Multiplication"
[1] "4 for Divison"
Enter your choice : 3
Enter Number 1 : 5
Enter Number 2 : 8
[1] "Answer is :  40"
```

(5) Explore plot, pie, barplot etc. (the plotting options) which are built-in functions in R.

```r
x1=c(5,7,8,7,22,9,4,11,12,9,6)
y1=c(99,86,87,111,103,87,94,78,77,85,86)
x2=c(2,2,8,1,15,8,12,9,7,3,11,4,7)
y2=c(100,105,84,105,90,99,90,95,94,100,79,112,91)

plot(x1,y1,main="Observation of Cars",xlab="Car Age",ylab="Car Speed",col="red")
points(x2,y2,col="blue")

Expenses=c(7000,3000,800,2000,1900,300)
Labels=c("Rent","Grocery","Transport","School Fees","Savings","Current")

pie(Expenses,labels = Labels,main="Expenses",col=rainbow(24))

barplot(y1,names.arg=x1,main="Observation of Cars",xlab="Car Age",ylab="Car Speed")
```


Observation of Cars
Car Speed
Car Age
80
85
90
95
105
5
10
15
20


Expenses
Rent
Current
Savings
School Fees
Transport
Grocery


Observation of Cars
Car Speed
0
20
40
60
80
100
5
7
8
7
22
9
4
11
12
9
6
Car Age

PROBABILITY AND STATISTICS (UCS410)
Experiment 2: Descriptive statistics, Sample space, definition of probability

Name: Jashan Arora
Roll No: 102003206
Group: 3CO9

(1)

(a) Suppose there is a chest of coins with 20 gold, 30 silver and 50 bronze coins. You randomly draw 10 coins from this chest. Write an R code which will give us the sample space for this experiment. (Use of sample(): an in-built function in R)

```r
chest<-c(rep("gold",20),rep("silver",30),rep("bronze",50))
print(sample(chest,10))
```

```
> source("~/Documents/PROB Assignments/ASSIGN2/Q1a.R")
 [1] "silver" "bronze" "bronze" "gold"   "gold"   "bronze" "bronze" "bronze"
 [9] "bronze" "silver"
```

(b) In a surgical procedure, the chances of success and failure are 90% and 10%respectively. Generate a sample space for the next 10 surgical procedures performed. (Use of prob(): an in-built function in R)

```r
print(sample(c("success","failure"),10,replace = T,prob=c(0.8,0.2)))
```

```
> source("~/Documents/PROB Assignments/ASSIGN2/Q1b.R")
 [1] "success" "failure" "success" "success" "success" "success" "success"
 [8] "success" "success" "success"
```

(2) A room has n people, and each has an equal chance of being born on any of the 365 days of the year. (For simplicity, we'll ignore leap years). What is the probability that two people in the room have the same birthday?

(a) Use an R simulation to estimate this for various n.

```
N=5000
n=20
sum=0
for(val in 1:N)
{
  a=as.integer(any(duplicated(sample(365,n,replace=TRUE))))
  sum=sum+a
}
prob=sum/N
print(prob)
```

```
> source("~/Documents/PROB Assignments/ASSIGN2/Q2.R")
[1] 0.4184
```

(b) Find the smallest value of n for which the probability of a match is greater than 0.5.

```
n=1
probability=0
while(probability<0.5)
{
probability = 1 - ((choose(365, n)*factorial(n))/365^n)
n=n+1
}
print(probability)
print(n-1)
```

```
> print(probability)
[1] 0.5072972
> print(n-1)
[1] 23
```

(3) Write an R function for computing conditional probability. Call this function to do the following problem:
suppose the probability of the weather being cloudy is 40%. Also suppose the probability of rain on a given day is 20% and that the probability of clouds on a rainy day is 85%. If it's cloudy outside on a given day, what is the probability that it will rain that day?

```r
Bayes <- function(Pa, Pb, PBgA)
{
  PAgB = (Pa * PBgA)/Pb
  return (PAgB)
}
```

```
> Bayes(0.2, 0.4, 0.85)
[1] 0.425
```

(4) The iris dataset is a built-in dataset in R that contains measurements on 4 different attributes (in centimetres) for 150 flowers from 3 different species. Load this dataset and do the following:

(a) Print first few rows of this dataset.

(b) Find the structure of this dataset.

(c) Find the range of the data regarding the sepal length of flowers.

(d) Find the mean of the sepal length.

(e) Find the median of the sepal length.

(f) Find the first and the third quartiles and hence the interquartile range.

(g) Find the standard deviation and variance.

(h) Try doing the above exercises for sepal.width, petal.length and petal.width.

(i) Use the built-in function summary on the dataset Iris.

```r
data <- iris
#a)
head(data)
#b)
str(data)
#c)
R <- range(data$Sepal.Length)
print(R)
#d)
mean(R)
#e)
median(R)
#f)
Q1 = quantile(data$Sepal.Length, 0.25)
Q3 = quantile(data$Sepal.Length, 0.75)
print(Q1)
print(Q3)
IQR(data$Sepal.Length)
#g)
sd(data$Sepal.Length)
var(data$Sepal.Length)
#h)
lapply(data[2:4],sd)
lapply(data[2:4],var)
#i)
summary(data)
```

```
> data <- iris
> #a)
> head(data)
  Sepal.Length Sepal.Width Petal.Length Petal.Width Species
1          5.1         3.5          1.4         0.2  setosa
2          4.9         3.0          1.4         0.2  setosa
3          4.7         3.2          1.3         0.2  setosa
4          4.6         3.1          1.5         0.2  setosa
5          5.0         3.6          1.4         0.2  setosa
6          5.4         3.9          1.7         0.4  setosa
> #b)
> str(data)
'data.frame':	150 obs. of  5 variables:
 $ Sepal.Length: num  5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...
 $ Sepal.Width : num  3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ...
 $ Petal.Length: num  1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ...
 $ Petal.Width : num  0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ...
 $ Species     : Factor w/ 3 levels "setosa","versicolor",..: 1 1 1 1 1 1 1 1 1 1 ...
> #c)
> R <- range(data$Sepal.Length)
> print(R)
[1] 4.3 7.9

> #d)
> mean(R)
[1] 6.1
> #e)
> median(R)
[1] 6.1
> #f)
> Q1 = quantile(data$Sepal.Length, 0.25)
> Q3 = quantile(data$Sepal.Length, 0.75)
> print(Q1)
25% 
5.1 
> print(Q3)
75% 
6.4 
> IQR(data$Sepal.Length)
[1] 1.3

> #g)
> sd(data$Sepal.Length)
[1] 0.8280661
> var(data$Sepal.Length)
[1] 0.6856935
> #h)
> lapply(data[2:4],sd)
$Sepal.Width
[1] 0.4358663

$Petal.Length
[1] 1.765298

$Petal.Width
[1] 0.7622377
```

```
> lapply(data[2:4],var)
$Sepal.Width
[1] 0.1899794

$Petal.Length
[1] 3.116278

$Petal.Width
[1] 0.5810063

> #i)
> summary(data)
  Sepal.Length    Sepal.Width     Petal.Length    Petal.Width          Species  
 Min.   :4.300   Min.   :2.000   Min.   :1.000   Min.   :0.100   setosa    :50  
 1st Qu.:5.100   1st Qu.:2.800   1st Qu.:1.600   1st Qu.:0.300   versicolor:50  
 Median :5.800   Median :3.000   Median :4.350   Median :1.300   virginica :50  
 Mean   :5.843   Mean   :3.057   Mean   :3.758   Mean   :1.199                  
 3rd Qu.:6.400   3rd Qu.:3.300   3rd Qu.:5.100   3rd Qu.:1.800                  
 Max.   :7.900   Max.   :4.400   Max.   :6.900   Max.   :2.500                  
```

(5) R does not have a standard in-built function to calculate mode. So, we create a user function to calculate mode of a data set in R. This function takes the vector as input and gives the mode value as output.

```
getmode<-function(v){
  uniqv<-unique(v)
  uniqv[which.max(tabulate(match(v,uniqv)))]
}
v1<-c(2,1,2,3,1,2,3,4,1,5,5,3,2,3)
v2<-c("o","it","the","it","it")
result1<-getmode(v1)
print(result1)
result2<-getmode(v2)
print(result2)
```

```
> print(result1)
[1] 2
> result2<-getmode(v2)
> print(result2)
[1] "it"
```

PROBABILITY AND STATISTICS (UCS410)
Experiment 3: Probability Distributions

Name: Jashan Arora
Roll No: 102003206
Group: 3CO9

(1) Roll 12 dice simultaneously, and let X denotes the number of 6's that appear. Calculate the probability of getting 7, 8 or 9, 6's using R. (Try using the function pbinom; If we set S = {get a 6 on one roll}, P(S) = 1/6 and the rolls constitute Bernoulli trials; thus X ~ binom(size=12, prob=1/6) and we are looking for P(7 ≤ X ≤ 9).

```r
#Q1
print(paste("Probabilty of getting 7, 8 or 9, 6's :  ",
            pbinom(9,12,1/6)-pbinom(6,12,1/6)))
```

```
> print(paste("Probabilty of getting 7, 8 or 9, 6's :  ",pbinom(9,12,1/6)-pbinom(6,12,
1/6)))
[1] "Probabilty of getting 7, 8 or 9, 6's :   0.00129175754208255"
```

(2) Assume that the test scores of a college entrance exam fits a normal distribution. Furthermore, the mean test score is 72, and the standard deviation is 15.2. What is the percentage of students scoring 84 or more in the exam?

```r
#Q2
print(paste("Percentage of Students scoring 84 or more: ",
            pnorm(84,72,15.2,lower.tail=FALSE)*100))
```

```
> print(paste("Percentage of Students scoring 84 or more: ",pnorm(84,72,15.2,lower.tail
=FALSE)*100))
[1] "Percentage of Students scoring 84 or more:  21.4917602311272"
```

(3) On the average, five cars arrive at a particular car wash every hour. Let X count the number of cars that arrive from 10AM to 11AM, then X ~Poisson(λ = 5). What is probability that no car arrives during this time. Next, suppose the car wash above is in operation from 8AM to 6PM, and we let Y be the number of customers that appear in this period. Since this period covers a total of 10 hours, we get that Y ~ Poisson(λ = 5×10 = 50). What is the probability that there are between 48 and 50 customers, inclusive?

```
#Q3
print(paste("Probability that no car arrives during 10AM to 11AM : ",
            dpois(0,5)))
print(paste("Probabilty that there are between 48 and 50 customers
            during 8AM to 6PM : ",ppois(50,50)-ppois(47,50)))
```

```
> print(paste("Probability that no car arrives during 10AM to 11AM : ",dpois(0,5)))
[1] "Probability that no car arrives during 10AM to 11AM :  0.00673794699908547"
> print(paste("Probabilty that there are between 48 and 50 customers during 8AM to 6PM
 : ",ppois(50,50)-ppois(47,50)))
[1] "Probabilty that there are between 48 and 50 customers during 8AM to 6PM :  0.16784
8518849069"
```

(4) Suppose in a certain shipment of 250 Pentium processors there are 17 defective processors. A quality control consultant randomly collects 5 processors for inspection to determine whether or not they are defective. Let X denote the number of defectives in the sample. Find the probability of exactly 3 defectives in the sample, that is, find
P(X = 3).

```
#Q4
print(paste("Probabilty of exactly 3 defectives in the sample : ",
             dhyper(3,17,233,5)))
```

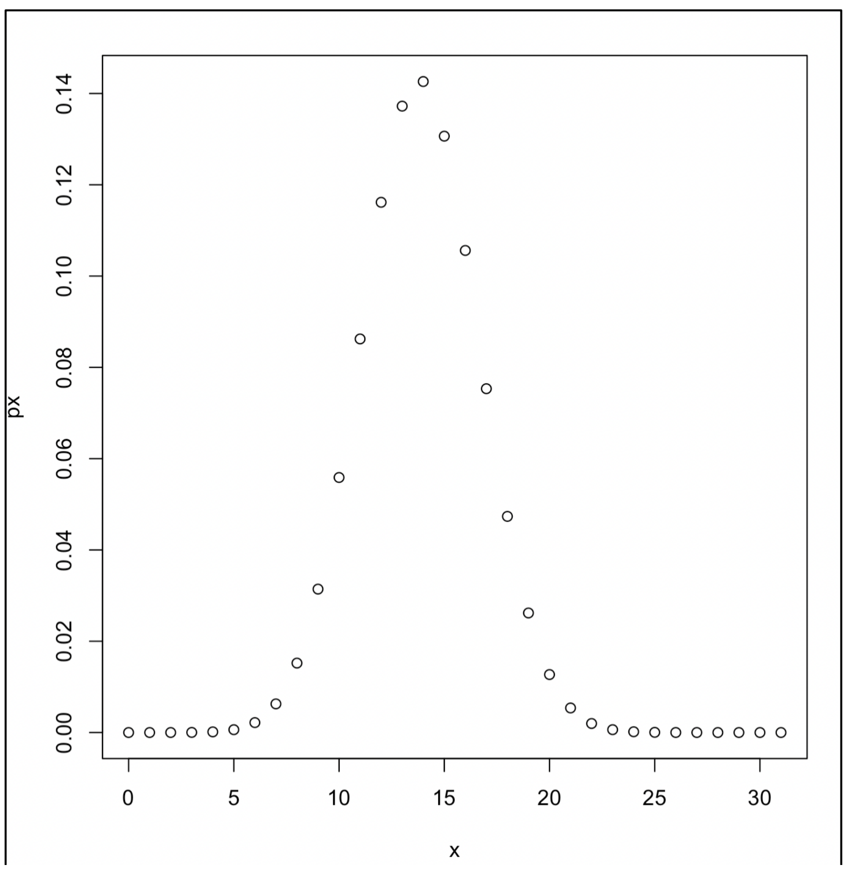
```
> print(paste("Probabilty of exactly 3 defectives in the sample : ",dhyper(3,17,233,
5)))
[1] "Probabilty of exactly 3 defectives in the sample :  0.00235115343595976"
```

(5) A recent national study showed that approximately 44.7% of college students have used Wikipedia as a source in at least one of their term papers. Let X equal the number of students in a random sample of size n = 31 who have used Wikipedia as a source.

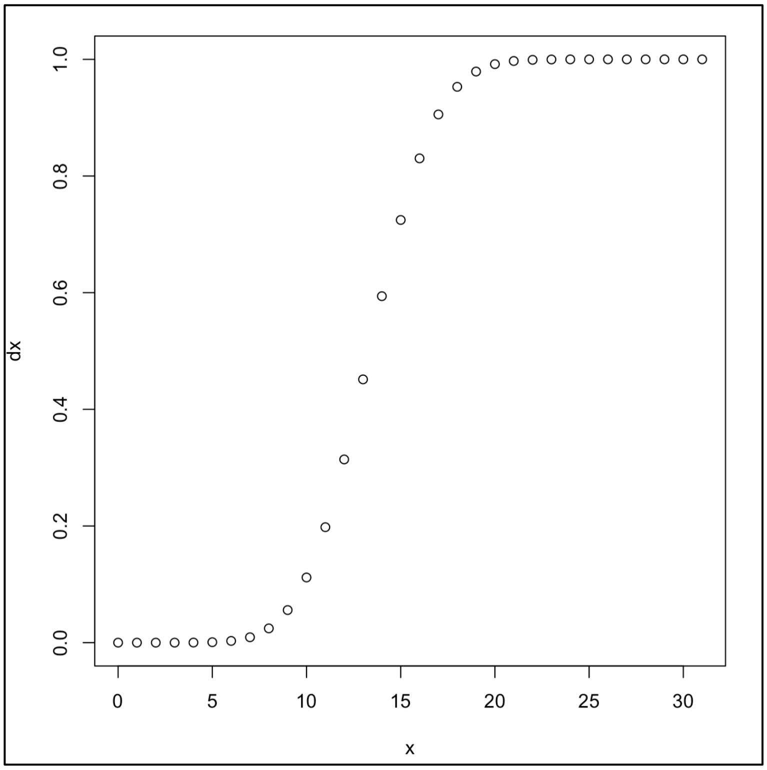
(a) How is X distributed?

```
#Q5
x<-0:31
px<-dbinom(x,31,0.447)
plot(x,px)
dx<-pbinom(x,31,0.447)
plot(x,dx)
meanx<-sum(x*px)
print(paste("Mean : ",meanx))
varx<-sum((x-meanx)^2*px)
print(paste("Variance : ",varx))
sdx<-sqrt(varx)
print(paste("Standard Deviation : ",sdx))
```

(b) Sketch the probability mass function.



(c) **Sketch the cumulative distribution function.**



(d) **Find mean, variance and standard deviation of X.**

```
> x<-0:31
> px<-dbinom(x,31,0.447)
> plot(x,px)
> dx<-pbinom(x,31,0.447)
> plot(x,dx)
> meanx<-sum(x*px)
> print(paste("Mean : ",meanx))
[1] "Mean :  13.857"
> varx<-sum((x-meanx)^2*px)
> print(paste("Variance : ",varx))
[1] "Variance :  7.66292099999999"
> sdx<-sqrt(varx)
> print(paste("Standard Deviation : ",sdx))
[1] "Standard Deviation :  2.76819815042204"
```

PROBABILITY AND STATISTICS (UCS410)

Experiment 4: Mathematical Expectation, Moments and Functions of Random Variables

Name: Jashan Arora
Roll No: 102003206
Group: 3CO9

1) The probability distribution of X, the number of imperfections per 10 meters of a synthetic fabric in continuous rolls of uniform width, is given as

x	0	1	2	3	4
p(x)	0.41	0.37	0.16	0.05	0.01

Find the average number of imperfections per 10 meters of this fabric.
(Try functions **sum( )**, **weighted.mean( )**, c(a **%*%** b) to find expected value/mean.

```r
#Q1
x=c(0,1,2,3,4)
px=c(0.41,0.37,0.16,0.05,0.01)
weighted.mean(x,px)
sum(x*px)
c(x%*%px)
```

Output:

```
> weighted.mean(x,px)
[1] 0.88
> sum(x*px)
[1] 0.88
> c(x%*%px)
[1] 0.88
```

2) The time T, in days, required for the completion of a contracted project is a random variable with probability density function $f(t) = 0.1\ e^{(-0.1t)}$ for $t > 0$ and 0 otherwise. Find the expected value of T. Use function **integrate( )** to find the expected value of continuous random variable T.

```r
#Q2
F<-function(t){t*(0.1)*exp(-0.1*t)}
Mean<-integrate(F,lower=0,upper=Inf)
print(Mean$value)
```

Output:

```
> print(Mean$value)
[1] 10
```

3) A bookstore purchases three copies of a book at \$6.00 each and sells them for \$12.00 each. Unsold copies are returned for \$2.00 each. Let X = {number of copies sold} and Y = {net revenue}. If the probability mass function of X is

x	0	1	2	3	4
p(x)	0.41	0.37	0.16	0.05	0.01

Find the expected value of Y.

```
#Q3
x<-c(0,1,2,3)
px<-c(0.1,0.2,0.2,0.5)
mean<-weighted.mean((10*x-12),px)
print(mean)
```

Output:

```
> print(mean)
[1] 9
```

4) Find the first and second moments about the origin of the random variable X with probability density function $f(x) = 0.5e^{-|x|}$, $1 < x < 10$ and 0 otherwise. Further use the results to find Mean and Variance. (kth moment = $E(X^k)$), Mean = first moment and Variance = second moment – Mean^2.

```
#Q4
F1<-function(x){x*0.5*exp(-1*abs(x))}
F2<-function(x){x*x*0.5*exp(-1*abs(x))}
F3<-function(M1,M2){return(M2-M1*M1)}
Moment1<-integrate(F1,lower=1,upper=10)
Moment2<-integrate(F2,lower=1,upper=10)
Variance<-F3(Moment1$value,Moment2$value)
Mean<-Moment1$value
print(paste("First Moment : ",Moment1$value))
print(paste("Second Moment : ",Moment2$value))
print(paste("Mean : ",Mean))
print(paste("Variance : ",Variance))
```

Output:

```
> print(paste("First Moment : ",Moment1$value))
[1] "First Moment :  0.367629741557749"
> print(paste("Second Moment : ",Moment2$value))
[1] "Second Moment :  0.916929207213094"
> print(paste("Mean : ",Mean))
[1] "Mean :  0.367629741557749"
> print(paste("Variance : ",Variance))
[1] "Variance :  0.781777580335277"
```

5) Let X be a geometric random variable with probability distribution

$$f(x) = \frac{3}{4}\left(\frac{1}{4}\right)^{x-1}, x = 1,2,3, \ldots$$

Write a function to find the probability distribution of the random variable $Y = X^2$ and find probability of Y for X = 3. Further, use it to find the expected value and variance of Y for X = 1,2,3,4,5.

```
#Q5
F<-function(x){return(3/4*(1/4)^(x-1))}
X=c(1,2,3,4,5)
E1<-sum(X^2*F(X))
E2<-sum(X^4*F(X))
Variance<-E2-E1^2
Mean<-E1
print(paste("Probability of Y for X=3 : ",F(3^2)))
print(paste("Mean : ",Mean))
print(paste("Variance : ",Variance))
```

Output:

```
> print(paste("Probability of Y for X=3 : ",F(3^2)))
[1] "Probability of Y for X=3 :  1.1444091796875e-05"
> print(paste("Mean : ",Mean))
[1] "Mean :  2.1826171875"
> print(paste("Variance : ",Variance))
[1] "Variance :  7.61411190032959"
```

PROBABILITY AND STATISTICS (UCS410)

Experiment 5: Continuous Probability Distributions

Name: Jashan Arora
Roll No: 102003206
Group: 3CO9

1) Consider that X is the time (in minutes) that a person has to wait in order to take a flight. If each flight takes off each hour X ~ U(0, 60). Find the probability that
(a) waiting time is more than 45 minutes, and
(b) waiting time lies between 20 and 30 minutes.

```
#Q1
#a
print(1-punif(45,min=0,max=60))
#b
print(punif(30,min=0,max=60)-punif(20,min=0,max=60))
```

Output:

```
> print(1-punif(45,min=0,max=60))
[1] 0.25
> print(punif(30,min=0,max=60)-punif(20,min=0,max=60))
[1] 0.1666667
```

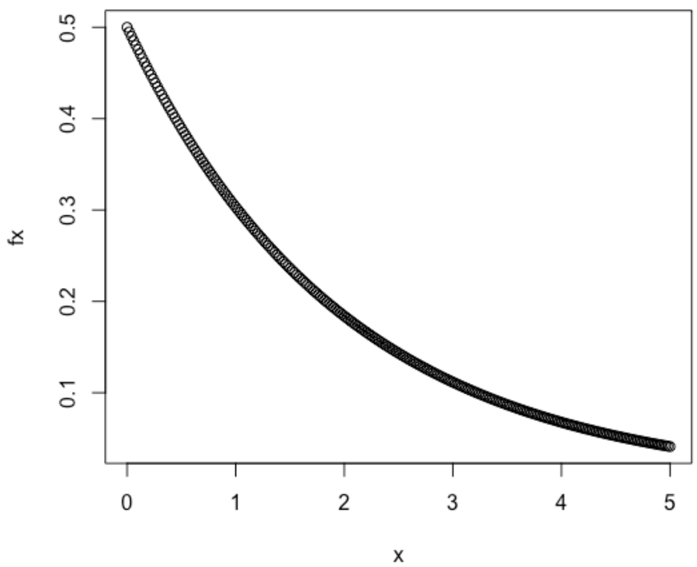
2) The time (in hours) required to repair a machine is an exponential distributed random variable with parameter $\lambda = 1/2$.
(a) Find the value of density function at x = 3.
(b) Plot the graph of exponential probability distribution for $0 \leq x \leq 5$.
(c) Find the probability that a repair time takes at most 3 hours.
(d) Plot the graph of cumulative exponential probabilities for $0 \leq x \leq 5$.
(e) Simulate 1000 exponential distributed random numbers with $\lambda = 1/2$ and plot the simulated data.

```
#Q2
#a
dexp(3,rate=1/2)
#b
x<-seq(0,5,by=0.02)
fx<-dexp(x,rate=1/2)
plot(x,fx)
#c
pexp(3,rate=1/2)
#d
x<-seq(0,5,by=0.02)
fx<-pexp(x,rate=1/2)
plot(x,fx)
#e
x_sim<-rexp(1000,rate=1/2)
plot(density(x_sim))
```

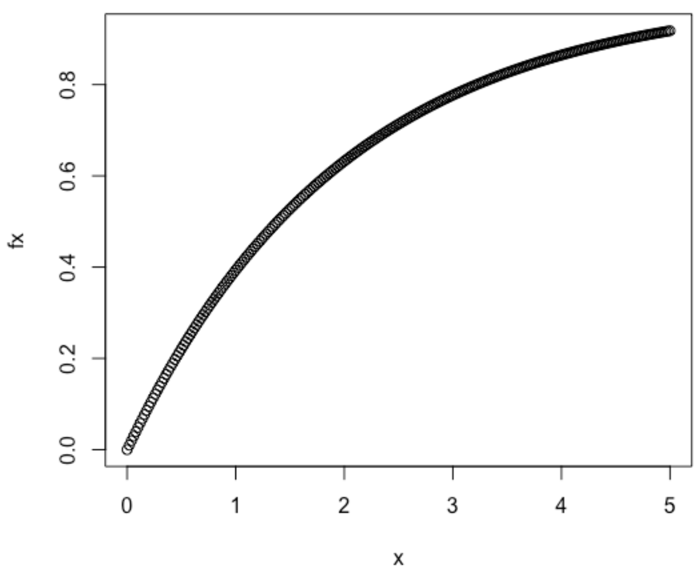
Output:

```
> #a
> dexp(3,rate=1/2)
[1] 0.1115651
> #c
> pexp(3,rate=1/2)
[1] 0.7768698
```

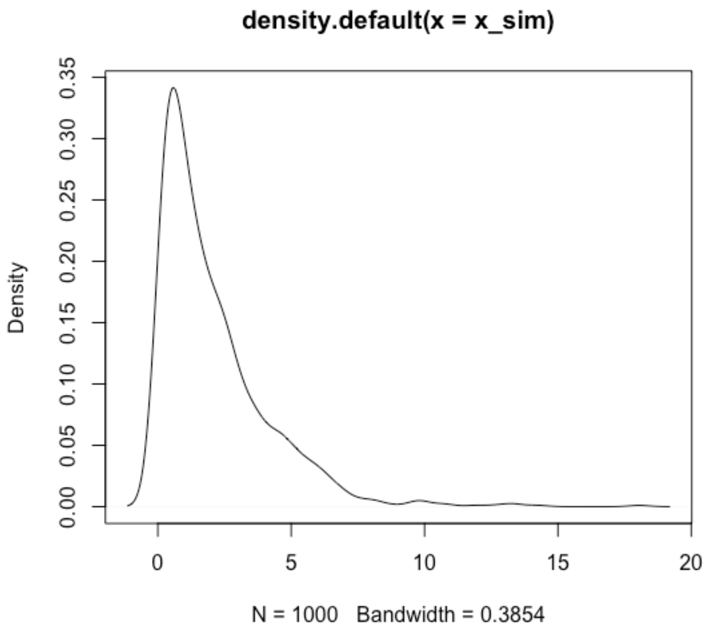
Plot (b) :



Plot (d) :



Plot (e) :



3) The lifetime of certain equipment is described by a random variable X that follows Gamma distribution with parameters $\alpha = 2$ and $\beta = 1/3$.
(a) Find the probability that the lifetime of equipment is at least 1 unit of time.
(b) What is the value of c, if $P(X \leq c) \geq 0.70$? (Hint: try quantile function qgamma())

```
#Q3
alpha<-2
beta<-1/3
#a
#i
dgamma(3,shape=alpha,scale=beta)
#ii
pgamma(1,shape=alpha,scale=beta,lower.tail=FALSE)
#b
qgamma(0.70,shape=alpha,scale=beta)
```

Output:

```
> #a
> #i
> dgamma(3,shape=alpha,scale=beta)
[1] 0.003332065
> #ii
> pgamma(1,shape=alpha,scale=beta,lower.tail=FALSE)
[1] 0.1991483
> #b
> qgamma(0.70,shape=alpha,scale=beta)
[1] 0.8130722
```

PROBABILITY AND STATISTICS (UCS410)
Experiment 6: Joint Probability Mass and Density Functions

Name: Jashan Arora
Roll No: 102003206
Group: 3CO9

(1)

The joint probability density of two random variables X and Y is

$$f(x,y) = \begin{cases} 2(2x+3y)/5; & 0 \leq x, y \leq 1 \\ 0; & elsewhere \end{cases}$$

Then write a R-code to

(i) check that it is a joint density function or not? (Use integral2())

(ii) find marginal distribution $g(x)$ at $x = 1$.

(iii) find the marginal distribution $h(y)$ at $y = 0$.

(iv) find the expected value of $g(x, y) = xy$.

(i)

```
library(pracma)

#Q1
f<-function(x,y)
{
  2*(2*x+3*y)/5
}

#i
Ans <- integral2(f,xmin=0,xmax=1,ymin=0,ymax=1)
Ans$Q
if(Ans$Q==1)
{
  print("It is a joint density function")
} else
{
  print("It is not a joint density function")
}
```

Output:

```
> #i
> Ans <- integral2(f,xmin=0,xmax=1,ymin=0,ymax=1)
> Ans$Q
[1] 1
> if(Ans$Q==1)
+ {
+   print("It is a joint density function")
+ } else
+ {
+   print("It is not a joint density function")
+ }
[1] "It is a joint density function"
```

(ii)

```
#ii
g1 <- function(y)
{
  f(1,y)
}
Ansb <- integral(g1,0,1)
Ansb
```

Output:

```
> #ii
> Ansb <- integral(g1,0,1)
> Ansb
[1] 1.4
```

(iii)

```
#iii
h0 <- function(x)
{
  f(x,0)
}
Ansc <- integral(h0,0,1)
Ansc
```

Output:

```
> #iii
> Ansc <- integral(h0,0,1)
> Ansc
[1] 0.4
```

(iv)

```
#iv
fxy <- function(x,y)
{
  x*y*f(x,y)
}
Ansd <- integral2(fxy,xmin=0,xmax=1,ymin=0,ymax=1)
Ansd$Q
```

Output:

```
> #iv
> Ansd <- integral2(fxy,xmin=0,xmax=1,ymin=0,ymax=1)
> Ansd$Q
[1] 0.3333333
```

(2)

The joint probability mass function of two random variables X and Y is

$$f(x,y) = \{(x+y)/30;\ \ x = 0,1,2,3;\ \ y = 0,1,2\}$$

Then write a R-code to

(i) display the joint mass function in rectangular (matrix) form.

(ii) check that it is joint mass function or not? (use: Sum())

(iii) find the marginal distribution $g(x)$ for $x = 0,1,2,3$. (Use:apply())

(iv) find the marginal distribution $h(y)$ for $y = 0,1,2$. (Use:apply())

(v) find the conditional probability at $x = 0$ given $y = 1$.

(vi) find $E(x), E(y), E(xy), Var(x), Var(y), Cov(x,y)$ and its correlation coefficient.

```r
#Q2

f <- function(x,y)
{
  (x+y)/30
}

X <- 0:3
Y <- 0:2

#i
M <- matrix(c(f(0,0:2),f(1,0:2),f(2,0:2),f(3,0:2)),nrow=4,ncol=3,byrow=TRUE)
M

#ii
sum(M)
if(sum(M)==1)
{
  print("It is a joint mass function")
} else
{
  print("It is not a joint mass function")
}

#iii
g <- apply(M,1,sum)
g

#iv
h <- apply(M,2,sum)
h

#v
Ansd <- M[1,2]/h[2]
Ansd

#vi
gx <- X*g
Ex <- sum(gx)
Ex

hy <- Y*h
Ey <- sum(hy)
Ey

fxy <- function(x,y)
{
  x*y*f(x,y)
}
```

```
Exy <- sum(c(fxy(0,Y),fxy(1,Y),fxy(2,Y),fxy(3,Y)))
Exy

Ex2 <- sum(gx*X)
Ey2 <- sum(hy*Y)

Vx <- Ex2-Ex^2
Vx

Vy <- Ey2-Ey^2
Vy

Covxy <- Exy-Ex*Ey
Covxy

Corrxy <- Covxy/sqrt(Vx*Vy)
Corrxy
```

Output:

(i)

```
> #i
> M <- matrix(c(f(0,0:2),f(1,0:2),f(2,0:2),f(3,0:2)),nrow=4,ncol=3,byrow=TRUE)
> M
           [,1]       [,2]       [,3]
[1,] 0.00000000 0.03333333 0.06666667
[2,] 0.03333333 0.06666667 0.10000000
[3,] 0.06666667 0.10000000 0.13333333
[4,] 0.10000000 0.13333333 0.16666667
```

(ii)

```
> #ii
> sum(M)
[1] 1
> if(sum(M)==1)
+ {
+   print("It is a joint mass function")
+ } else
+ {
+   print("It is not a joint mass function")
+ }
[1] "It is not a joint mass function"
```

(iii)

```
> #iii
> g <- apply(M,1,sum)
> g
[1] 0.1 0.2 0.3 0.4
```

(iv)

```
> #iv
> h <- apply(M,2,sum)
> h
[1] 0.2000000 0.3333333 0.4666667
```

(v)

```
> #v
> Ansd <- M[1,2]/h[2]
> Ansd
[1] 0.1
```

(vi)

```
> #vi
> gx <- X*g
> Ex <- sum(gx)
> Ex
[1] 2
>
> hy <- Y*h
> Ey <- sum(hy)
> Ey
[1] 1.266667
>
> fxy <- function(x,y)
+ {
+   x*y*f(x,y)
+ }
>
> Exy <- sum(c(fxy(0,Y),fxy(1,Y),fxy(2,Y),fxy(3,Y)))
> Exy
[1] 2.4
```

```
> Ex2 <- sum(gx*X)
> Ey2 <- sum(hy*Y)
>
> Vx <- Ex2-Ex^2
> Vx
[1] 1
>
> Vy <- Ey2-Ey^2
> Vy
[1] 0.5955556
>
> Covxy <- Exy-Ex*Ey
> Covxy
[1] -0.1333333
>
> Corrxy <- Covxy/sqrt(Vx*Vy)
> Corrxy
[1] -0.1727737
```

PROBABILITY AND STATISTICS (UCS410)

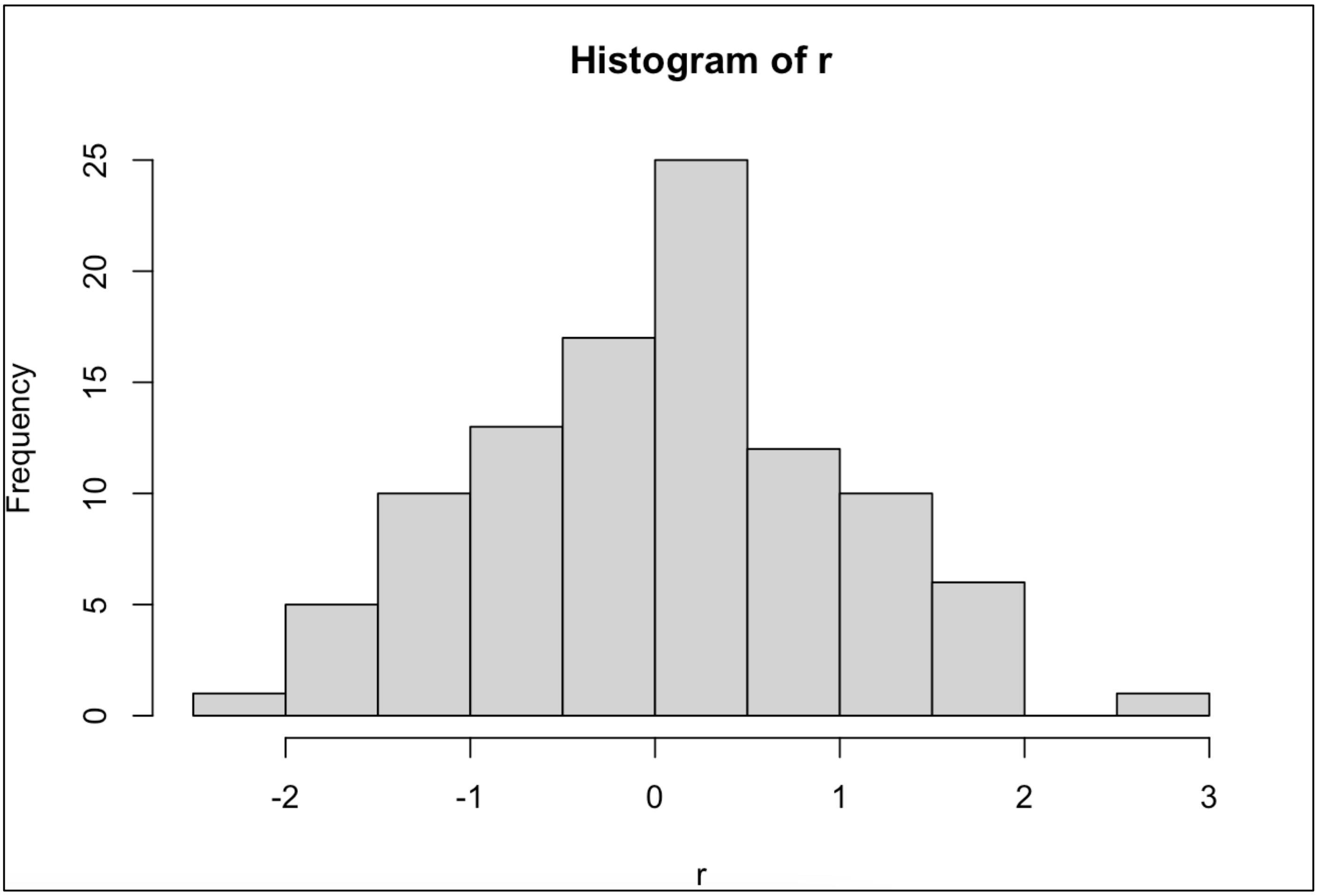
Experiment 7: Chi-square, t-distribution, F-distribution

Name: Jashan Arora
Roll No: 102003206
Group: 3CO9

(1) Use the rt(n, df) function in r to investigate the t-distribution for n = 100 and df = n − 1 and plot the histogram for the same.

```
#Q1
n=100
df=99
r=rt(n,df)
hist(r)
```

Output:



(2) Use the rchisq(n, df) function in r to investigate the chi-square distribution with n = 100 and df = 2, 10, 25.

```
#Q2
n=100
df=c(2,10,25)
rchisq(n, df[1])
rchisq(n, df[2])
rchisq(n, df[3])
```

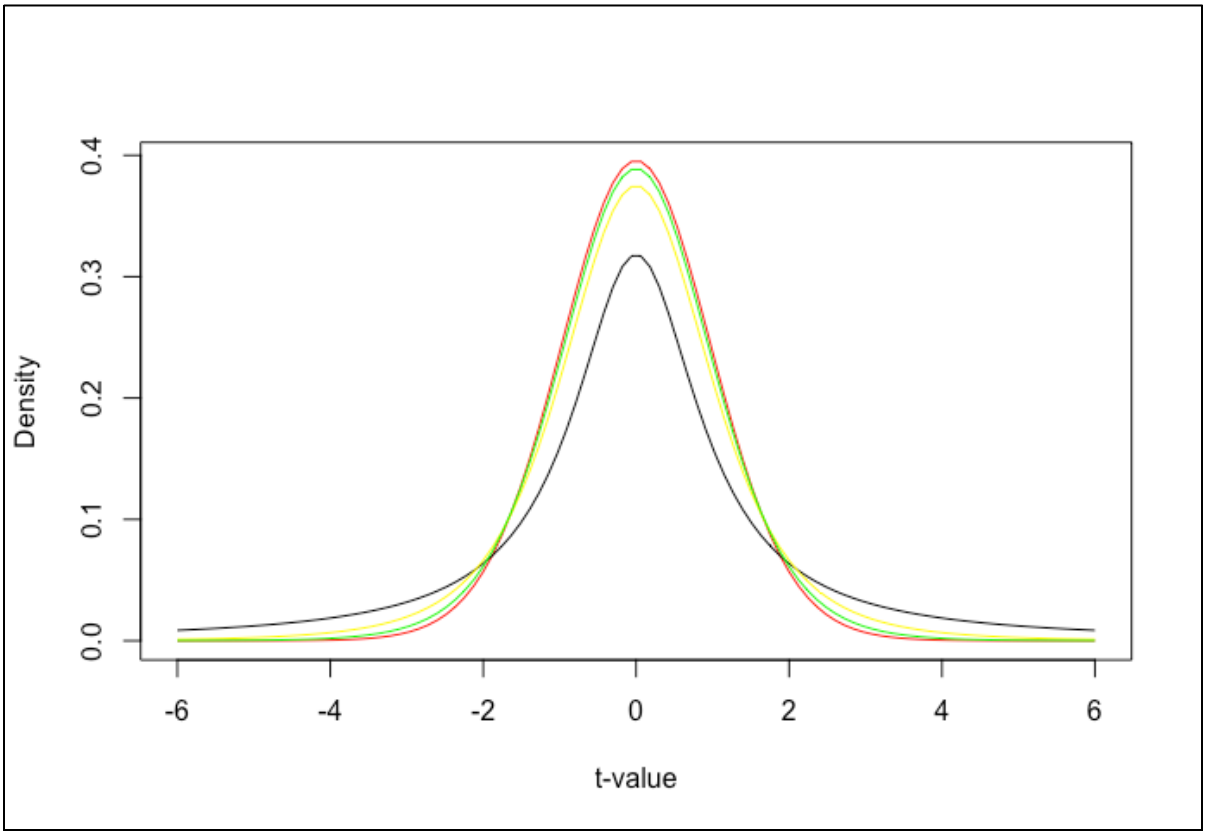
Output:

```
> n=100
> df=c(2,10,25)
> rchisq(n, df[1])
  [1]  0.75180096  3.68202705  7.87336979  1.07255888  0.54190300  2.70297382  3.49597658  1.81369234
  [9]  1.92343727  0.33879753  1.54366819  1.27891055  6.00552881  2.36028337  2.48466238  2.54930188
 [17]  0.51848047  3.34156954  7.95694141  1.61531928  1.52101905  1.98111753  0.63284876  0.91325962
 [25]  0.38254675  4.38816317 14.85923325  1.70795110  2.69625412  2.47546361  0.42037780  4.38039013
 [33]  0.30647266  2.61345938  0.20821191  2.31809899  0.01107924  1.81309270  2.52140478  0.29196598
 [41]  1.78964679  3.60379164  0.43283480  0.52412132  2.45324163  0.39618517  2.13122321  1.74948914
 [49]  0.70874481  2.64220820  1.21967797  0.31668053  1.47743712  0.90690620  0.68827996  1.08651026
 [57]  0.76602341  2.22893427  0.51843508  1.84795402  6.66548328  0.42514760  2.07008655  0.40862995
 [65]  1.76767673  1.50132770  0.47322525  0.88434489  1.07421175  1.06617590  2.55438326  0.04378265
 [73]  0.27274017  1.74258732  4.88999420  4.36691824  5.46146768  2.59520623  1.47466178  3.90446837
 [81]  4.35389358  1.02188814  3.26714372  5.83050590  1.27142782  0.13630352  0.26846773  9.35899717
 [89]  2.77720225  0.83567013  3.63687192  1.99948737  1.67492683  5.40143130  0.09216677  0.75377834
 [97]  3.10726845  0.80618692  2.50195948  1.54889695
> rchisq(n, df[2])
  [1] 11.177353 13.930449 12.573056 19.905639 11.423157  6.799807  9.712839 14.653285 12.795286 21.085650
 [11]  5.934612  8.133385  5.414152 19.178837  8.493313  2.358809  7.218367 24.042291 10.786771 13.269356
 [21] 11.921645  3.645362  7.219959 15.340736 10.424567  4.501626  8.762937  6.351566  2.788487  5.914153
 [31] 11.643256  3.614675 15.751537 11.216148  7.215360 12.372377  9.662132 14.059742  9.605214  6.957716
 [41] 13.317415  7.067045  6.661247 19.370592 12.831781  7.814507  7.602219 14.625476  8.836972 18.906832
 [51]  7.885847 12.595365  5.707353  1.461398  6.104107 16.295447 13.588398  9.856880  8.037007  5.780073
 [61] 12.898425 11.723875  7.571209  5.916638  5.111901  7.328088 14.715892 10.566750  7.185123  6.728328
 [71] 19.368125 11.462363 17.693831  6.391541 10.913646  7.467967 13.697044 10.047503  9.582684  4.774387
 [81]  5.637415  7.555075  5.862324  6.432707  2.730719 22.587361  4.714435 21.429114 10.152500 12.559542
 [91] 12.521626  7.333327  5.200023  7.343863 17.194769  9.508525 13.632705  4.952235 14.914289  7.828005
> rchisq(n, df[3])
  [1] 22.23832 13.09857 32.43228 22.25657 26.01014 13.76434 16.63859 26.91891 19.63596 22.69971 23.30508
 [12] 19.69423 27.42354 30.26376 14.60346 30.48598 17.82521 31.00044 24.87996 14.77268 40.65151 14.98268
 [23] 27.86865 28.03556 20.52871 35.40720 34.34827 18.07781 34.90062 25.36937 16.94359 23.00206 31.16876
 [34] 15.00069 22.40787 27.24155 30.80014 20.20019 33.16289 17.86001 25.79679 15.63309 17.84087 23.11543
 [45] 27.23274 27.34503 13.90737 27.61245 27.93774 21.32392 30.67258 27.93320 28.00942 21.40710 30.23849
 [56] 14.18024 14.90374 22.65711 29.77990 26.40923 11.16068 23.22052 30.24317 30.11587 18.82352 19.39854
 [67] 25.18071 16.98108 19.60585 20.11534 22.61605 38.89232 23.73208 30.10328 21.99802 26.74675 17.04011
 [78] 27.66347 16.62798 37.85068 14.95749 25.53694 25.95478 22.90247 31.80611 29.08595 30.95977 31.81041
 [89] 15.53559 19.72221 38.77281 28.59027 20.34597 16.92874 28.28467 38.12348 25.20741 17.02249 19.74088
[100] 23.30048
```

(3) Generate a vector of 100 values between -6 and 6. Use the dt() function in r to find the values of a t-distribution given a random variable x and degrees of freedom 1,4,10,30. Using these values plot the density function for students t-distribution with degrees of freedom 30. Also shows a comparison of probability density functions having different degrees of freedom (1,4,10,30).

```r
#Q3
x<-seq(-6,6,length=100)
df<-c(1,4,10,30)
dt(x,df[1])
dt(x,df[2])
dt(x,df[3])
dt(x,df[4])
colour<-c('black','yellow','green','red')
plot(x,dt(x,df[4]),type = 'l',xlab="t-value",ylab="Density",col=colour[4])
for (i in 1:3){
  lines(x,dt(x,df[i]),type = "l",col=colour[i])
}
```

Output:



(4) Write a r-code
(i) To find the 95th percentile of the F-distribution with (10, 20) degrees of freedom.
(ii) To calculate the area under the curve for the interval [0, 1.5] and the interval [1.5, +∞) of a F-curve with v1 = 10 and v2 = 20 (USE pf()).
(iii) To calculate the quantile for a given area (= probability) under the curve for a F-curve with v1 = 10 and v2 = 20 that corresponds to q = 0.25, 0.5, 0.75 and 0.999. (use the qf())
(iv) To generate 1000 random values from the F-distribution with v1 = 10 and v2 = 20 (use rf())and plot a histogram.

```r
#Q4

#i
qf(0.95,df1=10,df2=20)

#ii
pf(1.5,df1=10,df2=20)
pf(1.5,df1=10,df2=20,lower.tail = FALSE)

#iii
q=c(0.25,0.5,0.75,0.999)
qf(q[1],df1=10,df2=20)
qf(q[2],df1=10,df2=20)
qf(q[3],df1=10,df2=20)
qf(q[4],df1=10,df2=20)

#iv
r=rf(1000,df1=10,df2=20)
hist(r)
```

<u>Output:</u>

(i)

```
> #i
> qf(0.95,df1=10,df2=20)
[1] 2.347878
```

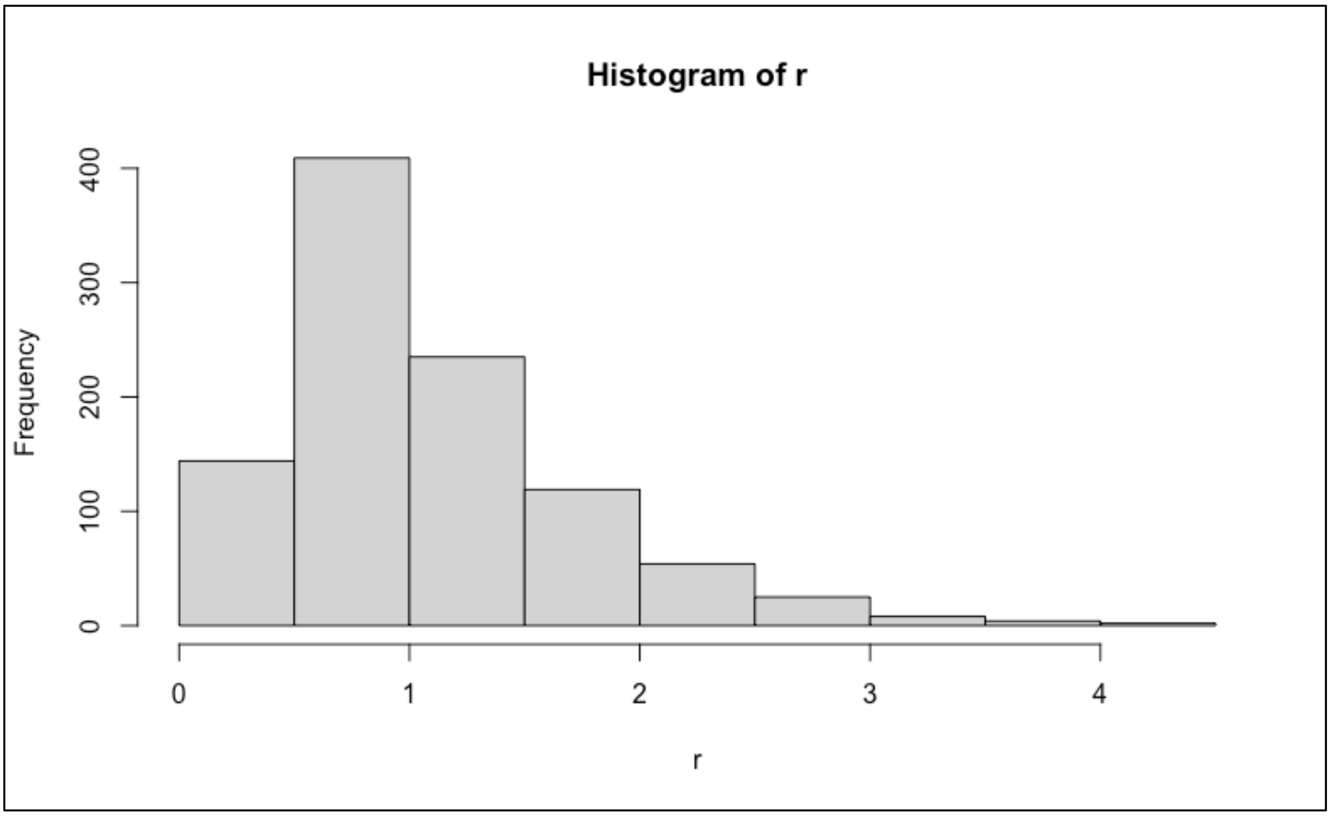
(ii)

```
> #ii
> pf(1.5,df1=10,df2=20)
[1] 0.7890535
> pf(1.5,df1=10,df2=20,lower.tail = FALSE)
[1] 0.2109465
```

(iii)

```
> #iii
> q=c(0.25,0.5,0.75,0.999)
> qf(q[1],df1=10,df2=20)
[1] 0.6563936
> qf(q[2],df1=10,df2=20)
[1] 0.9662639
> qf(q[3],df1=10,df2=20)
[1] 1.399487
> qf(q[4],df1=10,df2=20)
[1] 5.075246
```

(iv)



PROBABILITY AND STATISTICS (UCS410)

Experiment 8

Name: Jashan Arora
Roll No: 102003206
Group: 3CO9

(1)

A pipe manufacturing organization produces different kinds of pipes. We are given the monthly data of the wall thickness of certain types of pipes (data is available on LMS Clt-data.csv).

The organization has an analysis to perform and one of the basic assumption of that analysis is that the data should be normally distributed.

You have the following tasks to do:

(a) Import the csv data file in R.

(b) Validate data for correctness by counting number of rows and viewing the top ten rows of the dataset.

(c) Calculate the population mean and plot the observations by making a histogram.

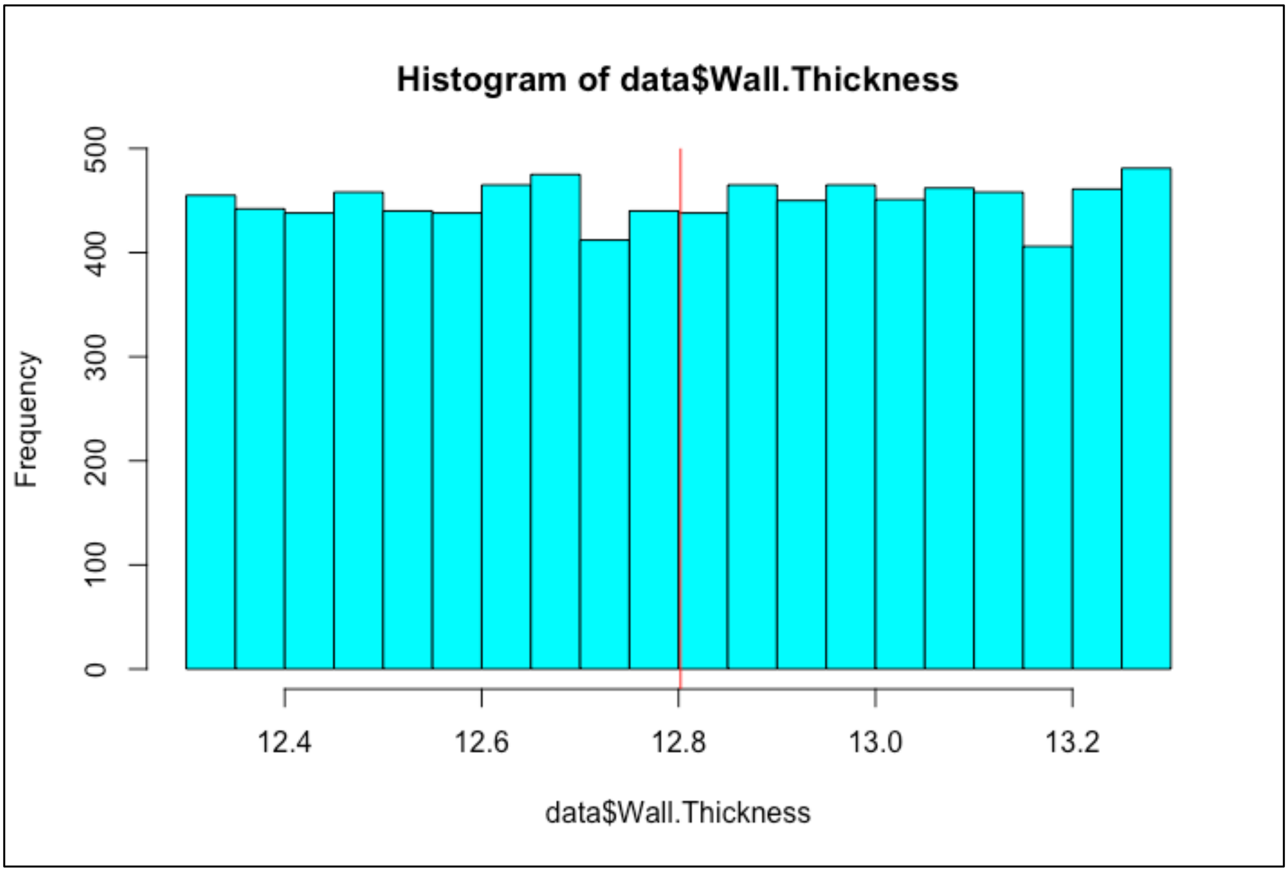
(d) Mark the mean computed in last step by using the function abline.

See the red vertical line in the histogram? That's the population mean. Comment on whether the data is normally distributed or not?

```
#Q1

#a
data = read.csv(file.choose())
#b
dim(data)
head(data,n=10)
#c
m<-mean(data$Wall.Thickness)
m
hist(data$Wall.Thickness,col="cyan")
#d
abline(v=m,col="red",lty=1)
# The data is not normally distributed
```

Output:

```
> #b
> dim(data)
[1] 9000    1
> head(data,n=10)
   Wall.Thickness
1        12.35487
2        12.61742
3        12.36972
4        13.22335
5        13.15919
6        12.67549
7        12.36131
8        12.44468
9        12.62977
10       12.90381

> #c
> m<-mean(data$Wall.Thickness)
> m
[1] 12.80205
> hist(data$Wall.Thickness,col="cyan")
```



(2)

Now perform the following tasks:

(a) Draw sufficient samples of size 10, calculate their means, and plot them in R by making histogram. Do you get a normal distribution.

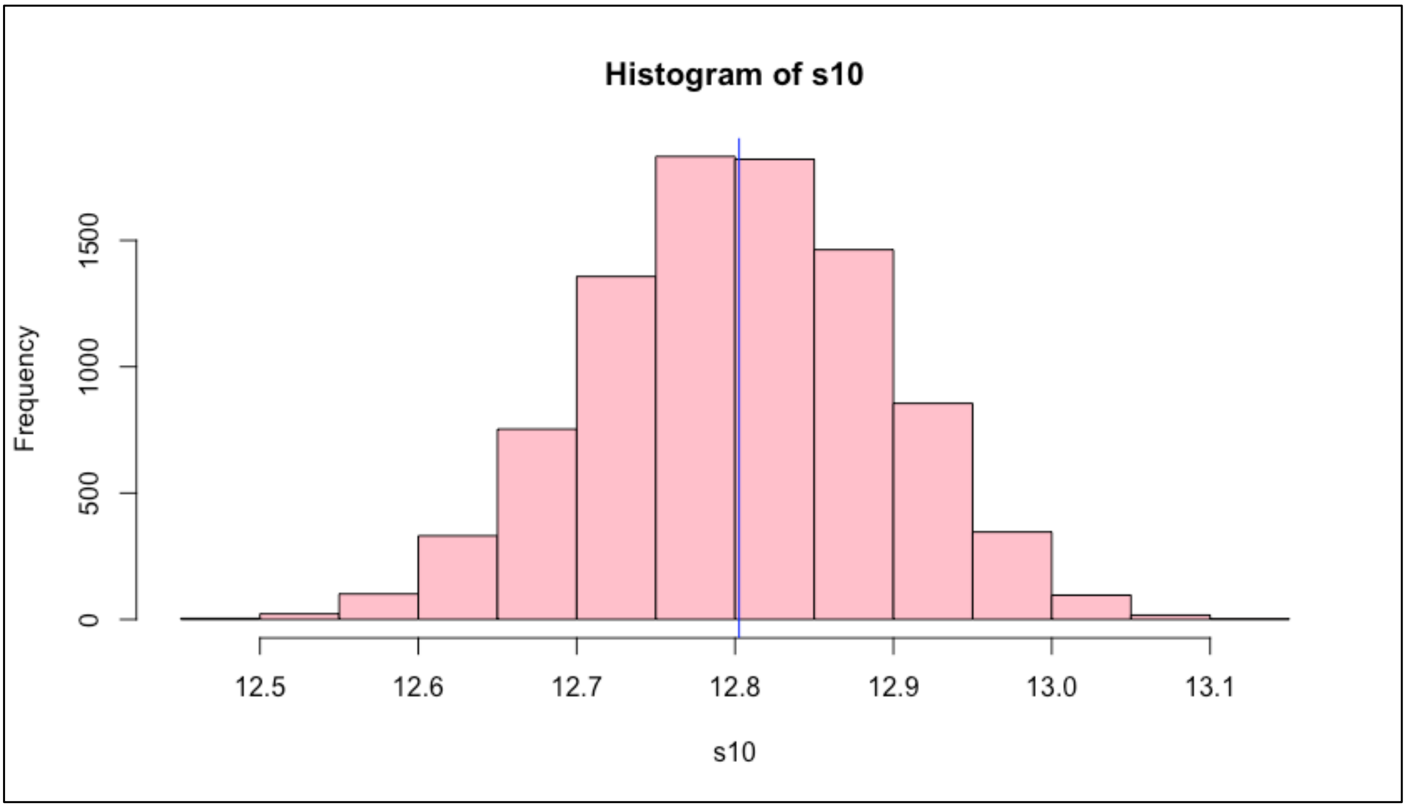
(b) Now repeat the same with sample size 50, 500 and 9000. Can you comment on what you observe.

Here, we get a good bell-shaped curve and the sampling distribution approaches normal distribution as the sample sizes increase. Therefore, we can recommend the organization to use sampling distributions of mean for further analysis.

```r
#Q2

#a
n<-9000
s10<-c()
for(i in 1:n)
{
  s10[i]<-mean(sample(data$Wall.Thickness,10,replace=TRUE))
}
hist(s10,col="pink")
abline(v=mean(s10),col="blue",lty=1)

#b
s30<-c()
s50<-c()
s500<-c()
par(mfrow=c(1,3))
for(i in 1:n)
{
  s30[i]<-mean(sample(data$Wall.Thickness,30,replace=TRUE))
  s50[i]<-mean(sample(data$Wall.Thickness,50,replace=TRUE))
  s500[i]<-mean(sample(data$Wall.Thickness,500,replace=TRUE))
}
hist(s30,col="red")
abline(v=mean(s30),col="blue",lty=1)
hist(s50,col="yellow")
abline(v=mean(s50),col="blue",lty=1)
hist(s500,col="pink")
abline(v=mean(s500),col="blue",lty=1)
```

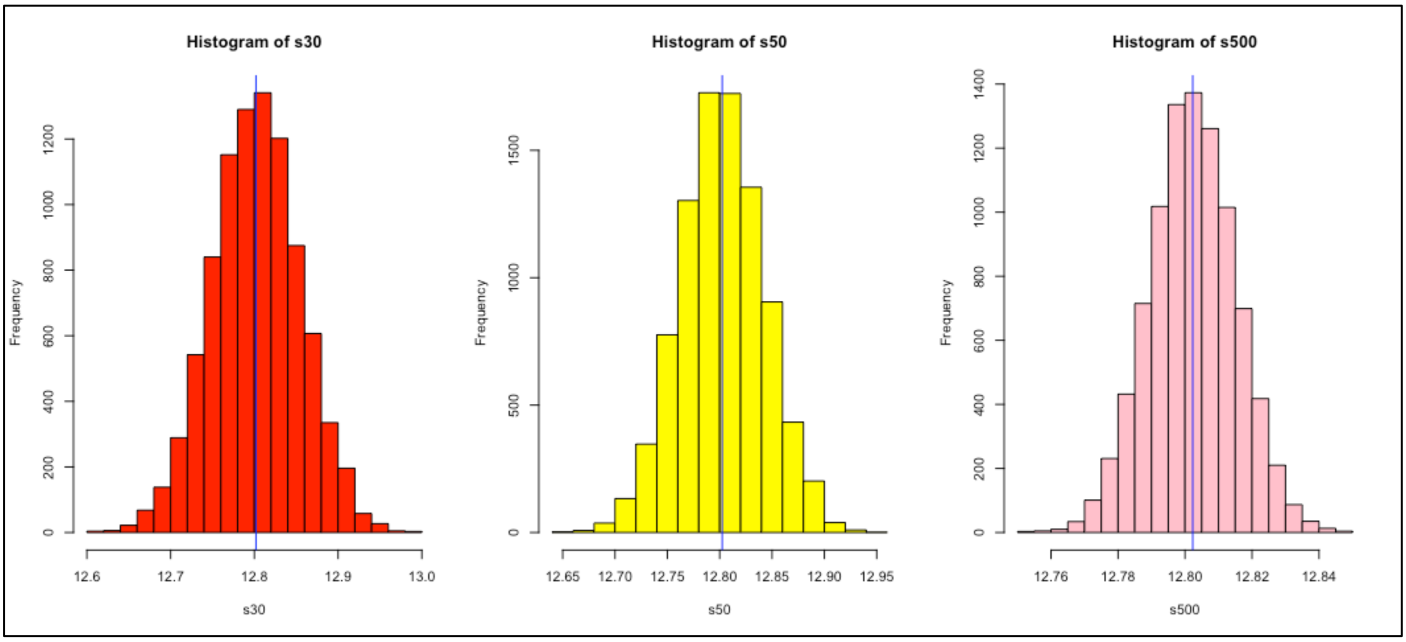
Output:

(a)



Histogram of s10
Frequency
0
500
1000
1500
12.5
12.6
12.7
12.8
12.9
13.0
13.1
s10

(b)



Histogram of s30
Histogram of s50
Histogram of s500
Frequency
s30
s50
s500