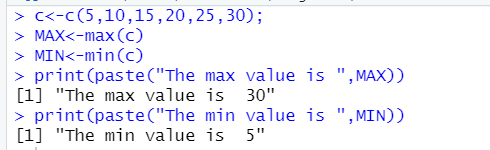
**PROBABILITY AND STATISTICS**

**EXPERIMENT-1**

1. c<-c(5,10,15,20,25,30);

MAX<-max(c)

MIN<-min(c)

print(paste("The max value is ",MAX))

print(paste("The min value is ",MIN))

1. n=as.integer(readline(prompt = "enter a number:"))

fact=1

if(n<0){

print(paste("The number is negative"))

}else if(n==0){

print("The factorial of 0 is 1")

}else{

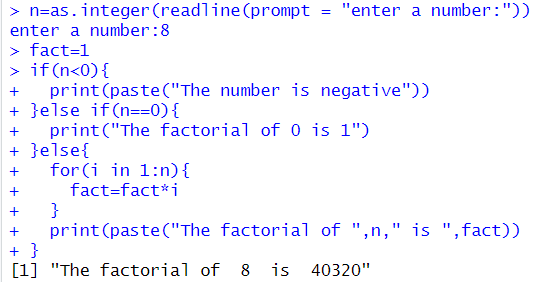
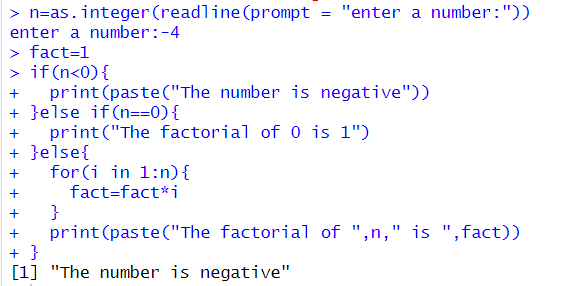
for(i in 1:n){

fact=fact\*i

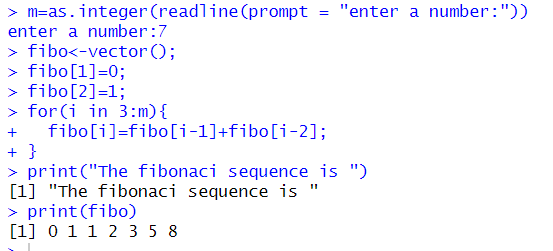
}

print(paste("The factorial of ",n," is ",fact))

}

1. Positive Number as input: b) Negative Number as input:
2. m=as.integer(readline(prompt = "enter a number:"))

fibo<-vector();

fibo[1]=0;

fibo[2]=1;

for(i in 3:m){

fibo[i]=fibo[i-1]+fibo[i-2];

}

print("The fibonaci sequence is ")

print(fibo)

1. 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)

}

num1=as.integer(readline(prompt = "enter a number:"))

num2=as.integer(readline(prompt = "enter a number:"))

print(" Enter Choice")

print("1. Add")

print("2. Subtraction")

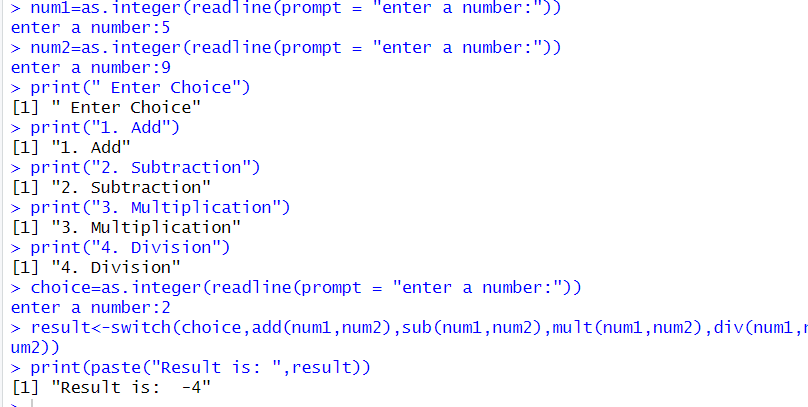
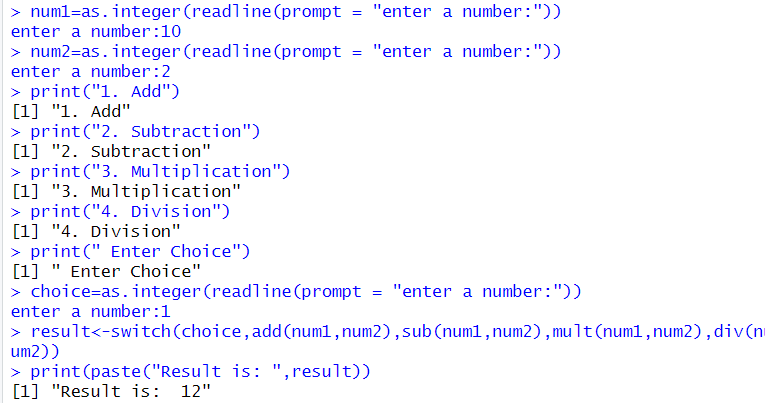
print("3. Multiplication")

print("4. Division")

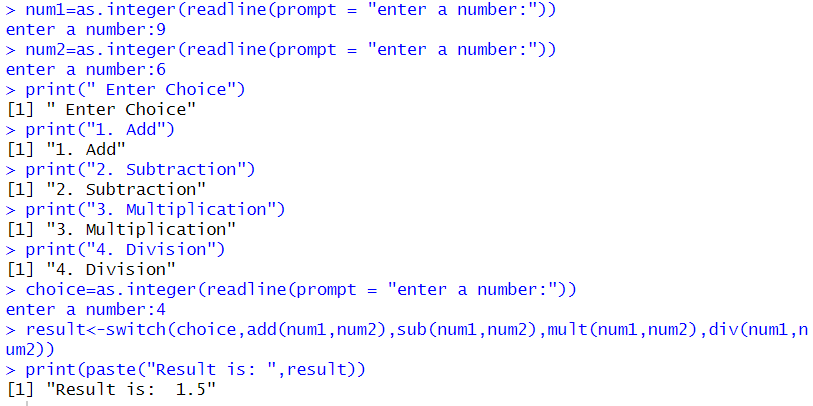
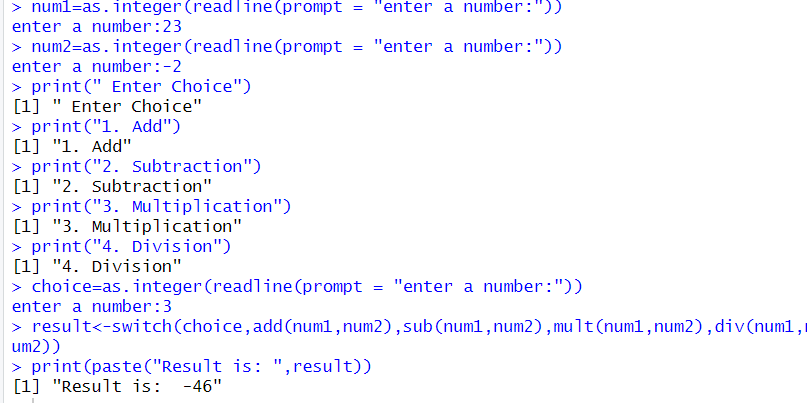
choice=as.integer(readline(prompt = "enter a number:"))

result<-switch(choice,add(num1,num2),sub(num1,num2),mult(num1,num2),div(num1,num2))

print(paste("Result is: ",result))

1. Addition: b) Subtraction:

c)Multiplication: d) Division:



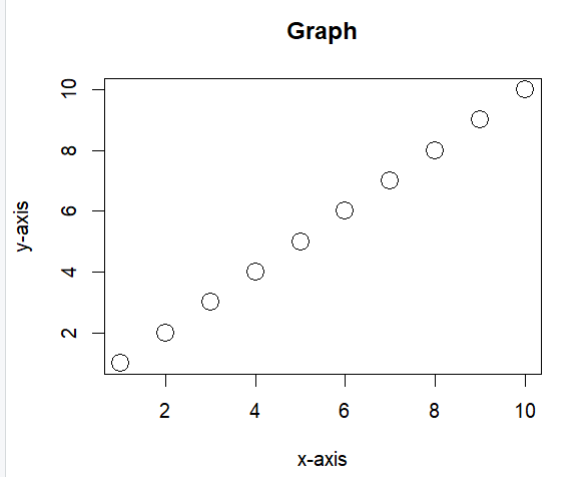
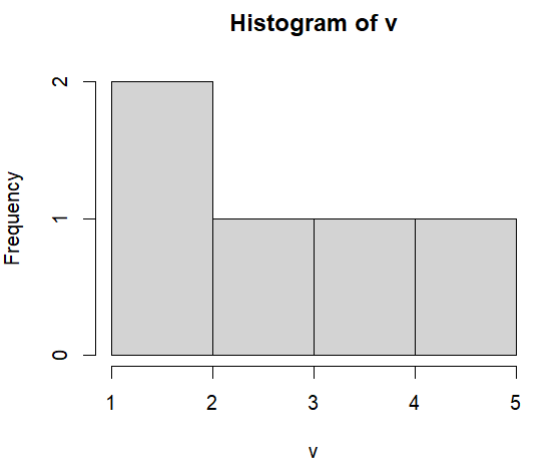
1. #Histogram

v<-c(1,2,3,4,5)

hist(v)

#plot()

plot(1:10,main="Graph",xlab="x-axis",ylab="y-axis",cex=2)



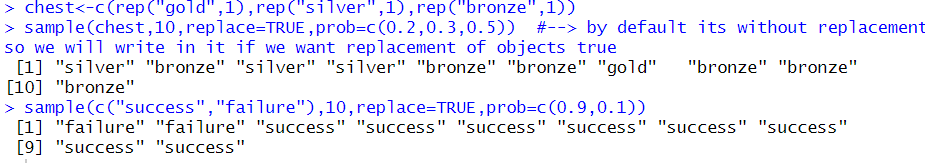
**PROBABILITY AND STATISTICS**

**EXPERIMENT-2**

1. chest<-c(rep("gold",1),rep("silver",1),rep("bronze",1))

sample(chest,10,replace=TRUE,prob=c(0.2,0.3,0.5)) #--> by default its without replacement #so we will write in it if we want replacement of objects true

sample(c("success","failure"),10,replace=TRUE,prob=c(0.9,0.1))



1. n<-2

while(n>=2){

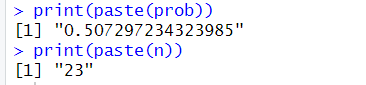
prod<-1

for(i in 1:(n-1)){

prod=prod\*(1-(i/365))

}

prob<-1-prod

if(prob>0.5){

break

}

n=n+1

}

print(paste(prob))

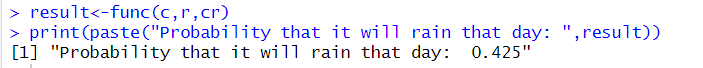
print(paste(n))

1. func<-function(c,r,cr){

rc<-r\*cr/c

return(rc)

}

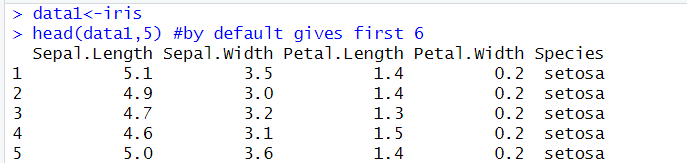
c<-0.4

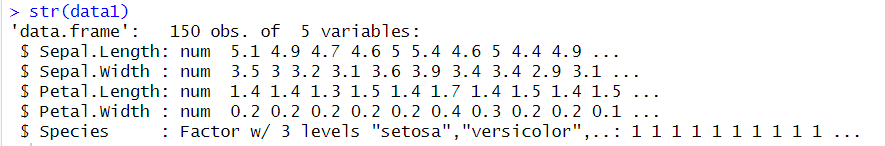
r<-0.2

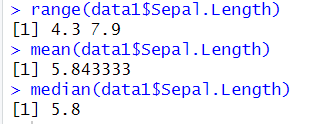
cr<-0.85

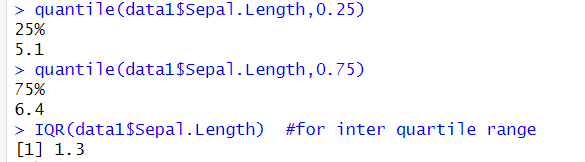
result<-func(c,r,cr)

print(paste("Probability that it will rain that day: ",result))



1. data1<-iris
2. head(data1,5)
3. str(data1)

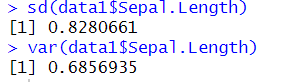


1. range(data1$Sepal.Length)
2. mean(data1$Sepal.Length)
3. median(data1$Sepal.Length)
4. quantile(data1$Sepal.Length,0.25)

quantile(data1$Sepal.Length,0.75)

IQR(data1$Sepal.Length) #for inter quartile range

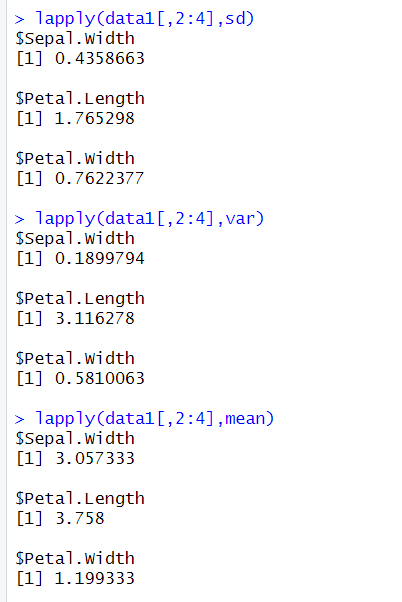
1. sd(data1$Sepal.Length)

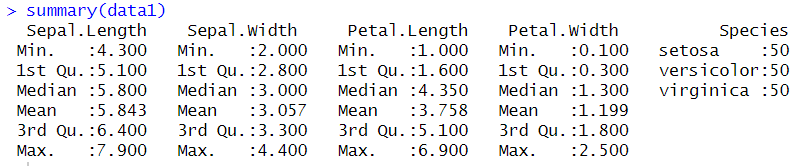
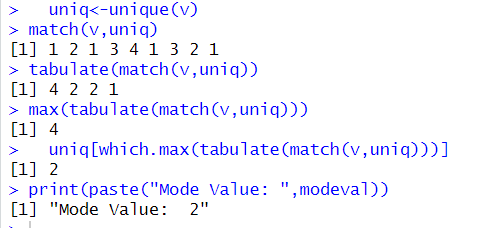
var(data1$Sepal.Length)

1. lapply(data1[,2:4],sd)

lapply(data1[,2:4],var)

lapply(data1[,2:4],mean)



1. summary(data1)
2. mode<-function(v){

uniq<-unique(v)

uniq[which.max(tabulate(match(v,uniq)))]

}

v<-c(2,1,2,3,4,2,3,1,2)

modeval<-mode(v)

print(paste("Mode Value: ",modeval))

**PROBABILITY AND STATISTICS**

**EXPERIMENT-3**

1. n<-12

p<-1/6

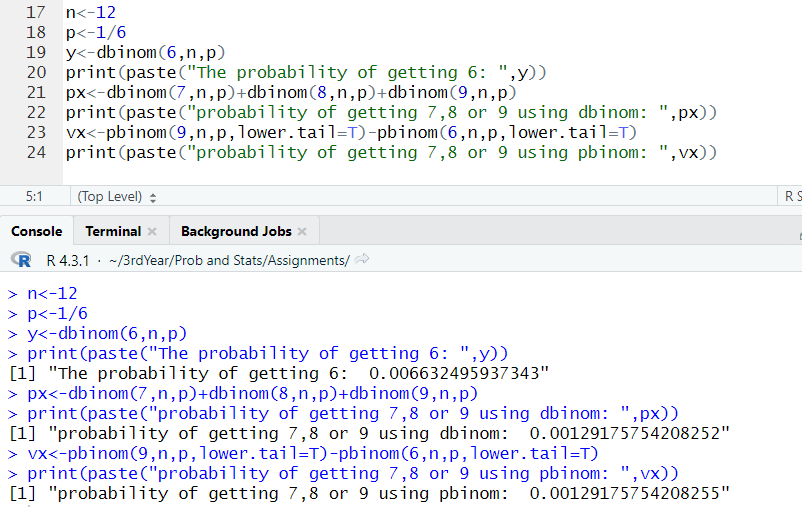
y<-dbinom(6,n,p)

print(paste("The probability of getting 6: ",y))

px<-dbinom(7,n,p)+dbinom(8,n,p)+dbinom(9,n,p)

print(paste("probability of getting 7,8 or 9 using dbinom: ",px))

vx<-pbinom(9,n,p,lower.tail=T)-pbinom(6,n,p,lower.tail=T)

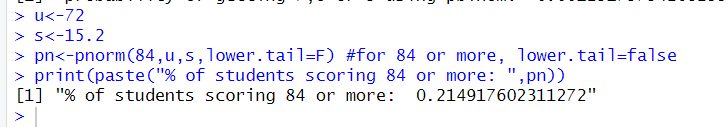
 print(paste("probability of getting 7,8 or 9 using pbinom: ",vx))

1. u<-72

s<-15.2

pn<-pnorm(84,u,s,lower.tail=F) #for 84 or more, lower.tail=false

print(paste("% of students scoring 84 or more: ",pn))



1. l<-5

q<-dpois(0,l)

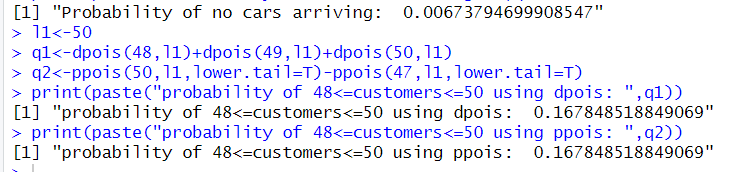
print(paste("Probability of no cars arriving: ",q))

l1<-50

q1<-dpois(48,l1)+dpois(49,l1)+dpois(50,l1)

q2<-ppois(50,l1,lower.tail=T)-ppois(47,l1,lower.tail=T)

print(paste("probability of 48<=customers<=50 using dpois: ",q1))

print(paste("probability of 48<=customers<=50 using ppois: ",q2))

1. x<-3

M<-17

N<-250

m<-5

h<-dhyper(x,M,N-M,m)

print(paste("Probability of exactly 3 defectives: ",h))

1. prob<-0.447

n1<-31

x<-seq(0,31,1)

pmf<-c()

for(i in 1:length(x)){

pmf[i]=dbinom(x[i],n1,prob)

}

plot(x,pmf)

cmf<-c()

for(i in 1:length(x)){

cmf[i]=pbinom(x[i],n1,prob)

}

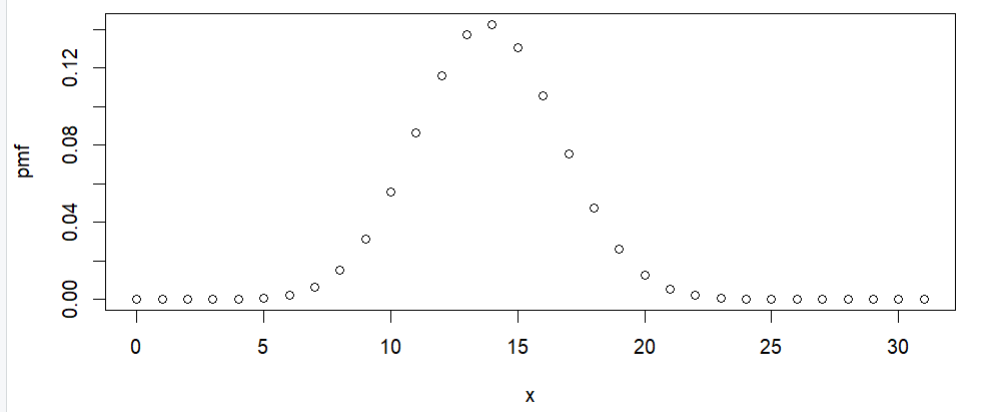
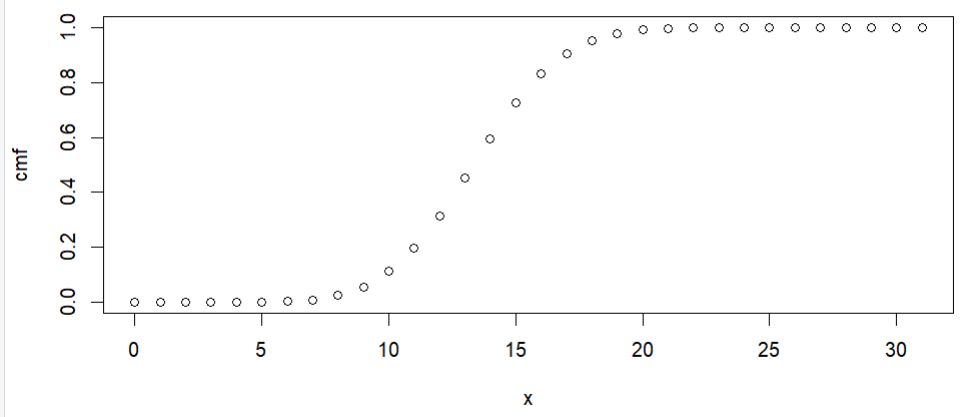
plot(x,cmf)

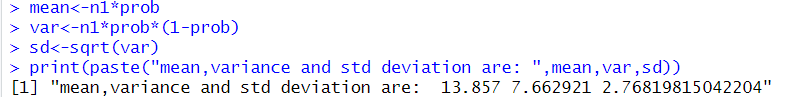
mean<-n1\*prob

var<-n1\*prob\*(1-prob)

sd<-sqrt(var)

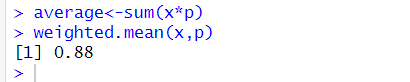
print(paste("mean,variance and std deviation are: ",mean,var,sd))

1. Binomial distribution.
2. Probability Mass Function:
3. Cumulative Distribution Function:
4. Mean, variance and standard deviation of X:



**PROBABILITY AND STATISTICS**

**EXPERIMENT-4**

1. x=c(0,1,2,3,4)

p=c(0.41,0.37,0.16,0.05,0.01)

average<-sum(x\*p)

weighted.mean(x,p)

1. g<-function(t){

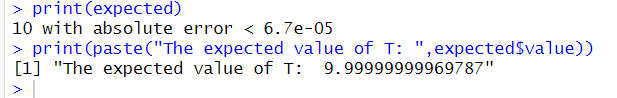
t\*0.1\*exp(-0.1\*t)

}

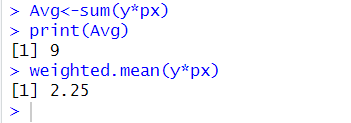
expected=integrate(g,lower=0,upper=Inf)

print(expected)

print(paste("The expected value of T: ",expected$value))



1. x1=c(0,1,2,3)

px=c(0.1,0.2,0.2,0.5)

y<-(12\*x1)+2\*(3-x1)-18

Avg<-sum(y\*px)

print(Avg)

weighted.mean(y\*px)

1. g1=function(x){

x\*0.5\*exp(-abs(x)) }

M1=integrate(g1,lower=1,upper=10)

print(M1)

M3=(M1$value)

print(paste("First Moment or Mean: ",M3))

g2=function(x){

x\*x\*0.5\*exp(-abs(x))

}

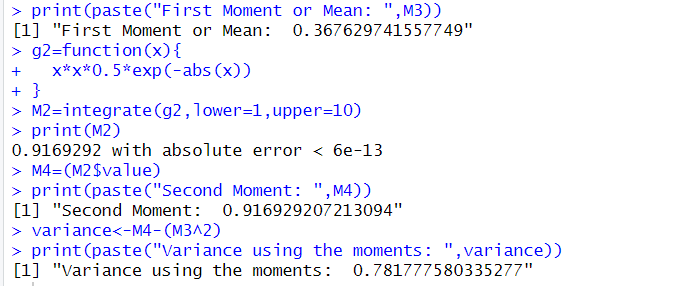
M2=integrate(g2,lower=1,upper=10)

print(M2)

M4=(M2$value)

print(paste("Second Moment: ",M4))

variance<-M4-(M3^2)

print(paste("Variance using the moments: ",variance))

1. f<-function(y){ (3/4)\*((1/4)^(sqrt(y)-1)) }

ans<-f(3\*3)

print(ans)

x<-c(1,2,3,4,5)

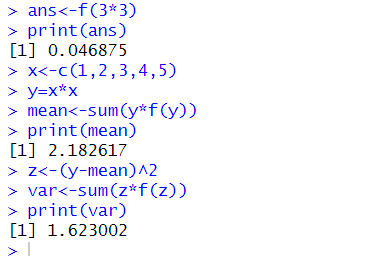
y=x\*x

mean<-sum(y\*f(y))

print(mean)

z<-(y-mean)^2

var<-sum(z\*f(z))

print(var)