1)Harmonic Mean

dist <- c(25, 25, 25)

speed <- c(25, 50, 75)

arithmetic\_mean\_speed <- mean(speed)

harmonic\_mean\_speed <- length(speed) / sum(1 / speed)

cat("Arithmetic Mean:", arithmetic\_mean\_speed, "mi/h\n")

cat("Harmonic Mean:", harmonic\_mean\_speed, "mi/h\n")

2) Mean

library(data.table)

salarydata<-data.frame(

salaries\_low=c(25000,26000,27000,28000,29000,30000,31000),

salaries\_high=c(25999,26999,27999,28999,29999,30999,32999),

Numbers=c(8,10,16,14,10,5,2)

)

View(salarydata)

cal\_data <- salarydata

# center value of group

cal\_data$center\_value = (cal\_data$salaries\_high+cal\_data$salaries\_low)/2

attach(cal\_data)

calculation\_data <- rep(center\_value,Numbers)

calculation\_data

# To find mean of salary

mean\_salary <- mean(calculation\_data)

mean\_salary

# To find Standard deviation of salary

sd\_salary <- sd(calculation\_data)

sd\_salary

# To find Variance of Salary

var\_salary <- var(calculation\_data)

var\_salary

3) Median

library(data.table)

salarydata<-data.frame(

salaries\_low=c(25000,26000,27000,28000,29000,30000,31000),

salaries\_high=c(25999,26999,27999,28999,29999,30999,32999),

Numbers=c(8,10,16,14,10,5,2)

)

print(salarydata)

salarydata <- as.data.table(salarydata)

salarydata <-salarydata[,cumNumbers:=cumsum(Numbers)]

salarydata

# Compute total count (N)

total\_count <- sum(salarydata$Numbers)

# Find the median group using data.table syntax

median\_group <- salarydata[

(cumNumbers - Numbers) <= (total\_count / 2) &

cumNumbers >= (total\_count / 2)

]

# Print or use these variables as needed

print(median\_group)

# Extract relevant values

l1 <- median\_group$salaries\_low

l2 <- median\_group$salaries\_high

f <- median\_group$Numbers

pcf <- median\_group$cumNumbers - median\_group$Numbers

n <- sum(salarydata$Numbers)

median<-l1+(((n/2)-pcf)/f\*(l2-l1))

median

4) Mode

library(data.table)

salarydata <- data.frame(

salaries\_low=c(25000,26000,27000,28000,29000,30000,31000),

salaries\_high=c(25999,26999,27999,28999,29999,30999,32999),

Numbers=c(8,10,16,14,10,5,2)

)

print(salarydata)

salarydata <- as.data.table(salarydata)

salarydata[ , prevNumbers := shift(Numbers,1)]

salarydata[ , nextNumbers := shift(Numbers,-1)]

salarydata

#identifying mode group

modegroup <- salarydata[Numbers == max(Numbers)]

modegroup

#creating the variables needed to calculate mode

l1 = modegroup[,salaries\_low]

l2 = modegroup[,salaries\_high]

f1 = modegroup[,Numbers]

f0 = modegroup[,prevNumbers]

f2 = modegroup[,nextNumbers]

#calculating mode

groupmode <- l1 + ((f1-f0)/(f1-f0+f1-f2)\*(l2-l1))

groupmode

5) SD

input<-data.frame(

X1=c(12, 6, 7, 3, 15, 10, 18, 5),

X2=c(9, 3, 8, 8, 9, 8, 9, 18)

)

print("Standard Deviaton of X1 and X2 are : ")

sd(input$X1)

sd(input$X2)

print("Variance of X1 and X2 are : ")

var(input$X1)

var(input$X2)

6) Pearson’s

input<-data.frame(

X=c(0,1,2,3,4),

F1=c(10,5,2,2,1),

F2=c(1,2,14,2,1),

F3=c(1,2,5,5,10)

)

getmode <- function(v) {

uniqv <- unique(v)

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

}

mean1<-mean(rep(input$X,input$F1))

median1<-median(rep(input$X,input$F1))

mode1<-getmode(rep(input$X,input$F1))

sd1<-sd(rep(input$X,input$F1))

Pearsonfirst<-((mean1-mode1)/sd1)

Pearsonsecond<-(3\*(mean1-median1)/sd1)

print("Pearson's first coefficients of skewness with x and f1 is : ")

print(Pearsonfirst)

print("Pearson's second coefficients of skewness with x and f1 is : ")

print(Pearsonsecond)

mean1<-mean(rep(input$X,input$F2))

median1<-median(rep(input$X,input$F2))

mode1<-getmode(rep(input$X,input$F2))

sd1<-sd(rep(input$X,input$F2))

Pearsonfirst<-((mean1-mode1)/sd1)

Pearsonsecond<-(3\*(mean1-median1)/sd1)

print("Pearson's first coefficients of skewness with x and f2 is : ")

print(Pearsonfirst)

print("Pearson's second coefficients of skewness with x and f2 is : ")

print(Pearsonsecond)

mean1<-mean(rep(input$X,input$F3))

median1<-median(rep(input$X,input$F3))

mode1<-getmode(rep(input$X,input$F3))

sd1<-sd(rep(input$X,input$F3))

Pearsonfirst<-((mean1-mode1)/sd1)

Pearsonsecond<-(3\*(mean1-median1)/sd1)

print("Pearson's first coefficients of skewness with x and f3 is : ")

print(Pearsonfirst)

print("Pearson's second coefficients of skewness with x and f3 is : ")

print(Pearsonsecond)

7) # Practical - 6

data <- data.frame(

spade = 402,

diamond = 358,

club = 273,

heart = 467

)

View(data)

result <- chisq.test(data)

result

if (result$p.value > 0.05) {

print("Null hypothesis is Accepted")

} else {

print("Null hypothesis is Rejected")

}

# Practical - 7

data <- data.frame(

finance = c(12,7),

sales = c(38,19),

hr = c(5,3),

technology = c(8,1)

)

View(data)

result <- chisq.test(data)

result

if (result$p.value > 0.05) {

print("Null hypothesis is Accepted")

} else {

print("Null hypothesis is Rejected")

}

8) Probability

#Suppose the number of games in which major league baseball players play during

#their careers is normally distributed with mean equal to 1500 games and standard

#deviation equal to 350 games. Use R to solve the following problems.

#(a) What percentage play in fewer than 750 games?

#(b) What percentage play in more than 2000 games?

#(c) Find the 90th percentile for the number of games played during a career.

print("What percentage play in fewer than 750 games")

pa<-pnorm(750, mean = 1500, sd = 350)

Percenta <- pa\*100

print(Percenta)

print("What percentage play in more than 2000 games")

pb<-pnorm(2000, mean = 1500, sd = 350, lower.tail = FALSE)

Percentb <- pb\*100

print(Percentb)

print("the 90th percentile for the number of games played during a career")

p05<-round(qnorm(0.05,mean = 1500, sd = 350),0)

p95<-round(qnorm(0.95,mean = 1500, sd = 350),0)

cat("Range for 90 Percentile is : ",p05,"-",p95)

9) 9-12 queation from 123pdf

# Question 9

height <- c(70,63,72,60,66,70,74,65,62,67,65,68)

weight <- c(155,150,180,135,156,168,178,160,132,145,139,152)

# A) H is independent Variable

# linear regression

line\_reg1 <- lm(weight ~ height)

line\_reg1

print(paste("weight = ",line\_reg1$coefficients[1],"+",

line\_reg1$coefficients[2],"\* Height"))

# B) H is dependent Variable

line\_reg2 <- lm(height ~ weight)

line\_reg2

print(paste("Height = ",line\_reg2$coefficients[1],"+",

line\_reg2$coefficients[2],"\* Weight"))

# Question 10 or Practical 12

year <- c(2000:2005)

Total\_value <- c(51246, 53659,53115,59364, 61383, 62958)

reg\_line <- lm(Total\_value ~ year)

reg\_line

plot(year,Total\_value,

main = "Total Agriculture export",

xlab = "Year",

ylab = "Total value",

cex = 1.3,

pch = 16,

abline(reg\_line, col = "red"),

xlim = c(1999,2006),

ylim = c(50000,65000),

col = "blue")

predict(reg\_line,data.frame(year = c(2006)))

# Practical 11

# Question 9

x <- c(6,5,8,8,7,6,10,4,9,7)

y <- c(8,7,7,10,5,10,8,6,8,6)

# A) Y on X

# linear regression

line\_reg1 <- lm(y ~ x)

line\_reg1

print(paste("weight = ",line\_reg1$coefficients[1],"+",

line\_reg1$coefficients[2],"\* Height"))

# B) x on Y

line\_reg2 <- lm(x ~ y)

line\_reg2

print(paste("Height = ",line\_reg2$coefficients[1],"+",

line\_reg2$coefficients[2],"\* Weight"))

10) Matrix a

a <- matrix(c(2, 4, 3, 1, 5, 7, 6, 8, 9), nrow = 3, byrow = TRUE)

b <- matrix(c(1, 6, 9, 2, 5, 8, 3, 4, 7), nrow = 3, byrow = TRUE)

print("Sum of two matrix")

result <- a + b

result

print("Product of two matrix")

result <- a \* b

result

# A inverse

if(det(a)!=0){

inverse\_A <- solve(a)

print("Inverse of A")

inverse\_A

}else{

print("Matrix A does not have inverse ")

}

# Note:- B inverse is not possible

if(det(b)!=0){

inverse\_B <- solve(b)

print("Inverse of B")

round(inverse\_B)

}else{

print("Matrix B does not have inverse ")

}

#Transpose

transpose\_B <- t(b)

print("Transpose of B is:")

print(transpose\_B)