**Day-3 lab manual**

**192124123**

**Indu sekhar.**

**a) Write suitable R code to compute the average of the following values.**

**12,7,3,4.2,18,2,54,-21,8,-5**

**b) Compute the mean after applying the trim option and removing 3 values from each**

**end.**

**c) Compute the mean of the following vector .**

**(12,7,3,4.2,18,2,54,-21,8,-5,NA)**

**#If there are missing values, then the mean function returns NA.**

**# Find mean dropping NA values.**

**#To drop the missing values from the calculation use na.rm = TRUE**

**a)code:**

x <- c(12,7,3,4.2,18,2,54,-21,8,-5)

mean(x)

o/p:

8.22

b)code:

x <- c(12,7,3,4.2,18,2,54,-21,8,-5)

z<-mean(x,trim = 0.3)

print(z)

o/p:5.55

3)code:

x <- c(12,7,3,4.2,18,2,54,-21,8,-5,NA)

(i) mean(x)

(ii)mean(x,na.rm = TRUE)

o/p:

(i)NA

(ii)8.22

**II.MEDIAN**

**Write suitable R code to compute the median of the following values.**

**12,7,3,4.2,18,2,54,-21,8,-5**

Code:

X<-c(12,7,3,4.2,18,2,54,-21,8,-5)

Median(X)

O/p:5.6

III. MODE

Calculate the mode for the following numeric as well as character data set in R.

(2,1,2,3,1,2,3,4,1,5,5,3,2,3) , (“0”,”it”,”the”,”it”,”it”)

Code:

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

x

charv <- c("o","it","the","it","it")

getmode(charv)

o/p:

2

It

UNIVARIATE ANALYSIS IN R - MEASURES OF DISPERSION

**i) Find the car which gives maximum city miles per gallon**

max\_city\_mpg <- mpg %>%

filter(!is.na(cty)) %>%

arrange(desc(cty)) %>%

slice(1)

car\_with\_max\_city\_mpg <- max\_city\_mpg$manufacturer[1]

**ii) Find the cars which gives minimum disp in compact and subcompact class**

compact\_and\_subcompact\_cars <- mpg %>%

filter(class %in% c("compact", "subcompact")) %>%

filter(!is.na(disp)) %>%

arrange(disp) %>%

slice(1:n())

cars\_with\_min\_disp <- compact\_and\_subcompact\_cars$manufacturer

**Exercise: 2**

**Use the same dataset as used in Exercise 1 and perform the following queries**

**i) Find the standard deviation of city miles per gallon**

cty\_sd <- mpg %>%

filter(!is.na(cty)) %>%

pull(cty) %>%

sd()

**ii) Find the variance of highway miles per gallon**

hwy\_var <- mpg %>%

filter(!is.na(hwy)) %>%

pull(hwy) %>%

var()

**Exercise 3**

**Use the same dataset and perform the following queries**

**i) Find the range of the disp in the data set mpg**

disp\_range <- range(mpg$disp, na.rm = TRUE)

**ii) Find the quartiles of the disp in the data set mpg**

disp\_quartiles <- quantile(mpg$disp, na.rm = TRUE, probs = c(0.25, 0.5, 0.75))

**iii) Find the IQR of the disp column in the data set mpg**

disp\_iqr <- IQR(mpg$disp, na.rm = TRUE)

**Exercise 4**

**#Install Library**

**library(e1071)**

# Load the e1071 library

library(e1071)

**a) Find the skewness of city miles per gallon in the data set mpg**

cty\_skewness <- skewness(mpg$cty, na.rm = TRUE)

**b) Find the kurtosis of city miles per gallon in the data set mpg**

cty\_kurtosis <- kurtosis(mpg$cty, na.rm = TRUE)

**Display the graph for the city miles per gallon column using the qplot function**

qplot(mpg$cty, geom = "histogram", main = "City Miles per Gallon Distribution", xlab = "City Miles per Gallon", ylab = "Frequency")

BIVARIATEANALYSIS IN R -COVARIANCE,CORRELATION,CROSSTAB

Reference Status Gender TestNewOrFollowUp

1 KRXH Accepted Female Test1 New

2 KRPT Accepted Male Test1 New

3 FHRA Rejected Male Test2 New

4 CZKK Accepted Female Test3 New

5 CQTN Rejected Female Test1 New

6 PZXW Accepted Female Test4 Follow-up

7 SZRZ Rejected Male Test4 New

8 RMZE Rejected Female Test2 New

9 STNX Accepted Female Test3 New

10 TMDW Accepted Female Test1 New

i) Load the dataset and Create a data frame and name it as dataframe1

ii) Load the function for crosstab

xtabs(~colname , data=Data frame name )

code:

(i)dataframe1 <- data.frame(Reference = c("KRXH", "KRPT", "FHRA", "CZKK", "CQTN", "PZXW", "SZRZ", "RMZE", "STNX", "TMDW"),

Status = c("Accepted", "Accepted", "Rejected", "Accepted", "Rejected", "Accepted", "Rejected", "Rejected", "Accepted", "Accepted"),

Gender = c("Female", "Male", "Male", "Female", "Female", "Female", "Male", "Female", "Female", "Female"),

Test = c("Test1", "Test1", "Test2", "Test3", "Test1", "Test4", "Test4", "Test2", "Test3", "Test1"),

NewOrFollowUp = c("New", "New", "New", "New", "New", "Follow-up", "New", "New", "New", "New"))

(ii)xtabs(~Status + Gender + Test + NewOrFollowUp, data = dataframe1)

o/p:

(i)Reference Status Gender Test NewOrFollowUp

1 KRXH Accepted Female Test1 New

2 KRPT Accepted Male Test1 New

3 FHRA Rejected Male Test2 New

4 CZKK Accepted Female Test3 New

5 CQTN Rejected Female Test1 New

6 PZXW Accepted Female Test4 Follow-up

7 SZRZ Rejected Male Test4 New

8 RMZE Rejected Female Test2 New

9 STNX Accepted Female Test3 New

10 TMDW Accepted Female Test1 New

(ii) , Test = Test1, NewOrFollowUp = Follow-up

Gender

Status Female Male

Accepted 0 0

Rejected 0 0

, , Test = Test2, NewOrFollowUp = Follow-up

Gender

Status Female Male

Accepted 0 0

Rejected 0 0

, , Test = Test3, NewOrFollowUp = Follow-up

Gender

Status Female Male

Accepted 0 0

Rejected 0 0

, , Test = Test4, NewOrFollowUp = Follow-up

Gender

Status Female Male

Accepted 1 0

Rejected 0 0

, , Test = Test1, NewOrFollowUp = New

Gender

Status Female Male

Accepted 2 1

Rejected 1 0

, , Test = Test2, NewOrFollowUp = New

Gender

Status Female Male

Accepted 0 0

Rejected 1 1

, , Test = Test3, NewOrFollowUp = New

Gender

Status Female Male

Accepted 2 0

Rejected 0 0

, , Test = Test4, NewOrFollowUp = New

Gender

Status Female Male

Accepted 0 0

Rejected 0 1

**Exercise: 2**

**i) Use two categorical variables and discover the relationships within a dataset**

library(ggplot2)

ggplot(dataframe1, aes(x = Reference, fill = Status)) + geom\_bar(position = "dodge") + xlab("Reference") + ylab("Count") + ggtitle("Relationship between Reference and Status")

**ii) Apply two variables from "dataframe1" to create a table delineating the relationship**

xtab <- xtabs(~Reference + Status, data = dataframe1)

**iii) Save the file in the name of dataframe2**

dataframe2 <- xtab

**Exercise: 3**

**Use the same data frame using three Categorical Variables create a Multi-Dimensional Table**

**Apply three variables from “dataframe1” to create a Multi-Dimensional Cross-Tabulation of**

**“Status“, “Gender“, and “Test“.**

xtab <- xtabs(~Status + Gender + TestNewOrFollowUp, data = dataframe1)

**Exercise: 4**

**Row Percentages**

**The R package “tigerstats” is required for the next two exercises.**

**# Load the tigerstats library**

**library(tigerstats)**

**1) Create an xtabs formula that cross-tabulates "Status" and "Test"**

xtab <- xtabs(~Status + TestNewOrFollowUp, data = dataframe1)

**2) Display row percentages for "Status" by "Test"**

rowPerc(xtab)

**Exercise 5**

**Column Percentages**

**# Load the tigerstats library**

**library(tigerstats)**

**1)Create cross-tabulation for Status and Test**

dataframe2 <- xtabs(~ Status + Test, data = dataframe1)

**2) Enclose cross-tabulation in colPerc function**

colPerc(dataframe2)

**Exercise 6**

**Covariance**

**i) Calculate the covariance between Refrence column and Status column**

cov(dataframe1$Reference, dataframe1$Status)

**ii) Display the covariance matrix**

cov(dataframe1[,c("Reference","Status")])

**Exercise 7**

**Correlation**

**Find the Correlation between gender and status. what kind of correlation does exist between the two?**

To find the correlation between two variables, we can use the cor function in R. In this case, to find the correlation between the "gender" and "status" variables in the "dataframe1" data frame, we can run the following code:

cor(dataframe1$gender, dataframe1$status)

**VISUALIZATION IN R**

**1. Write a program for creating a pie-chart in R using the input vector(21,62,10,53).**

**Provide labels for the chart as ‘London’, ‘New York’, ‘Singapore’, ‘Mumbai’. Add a title to the chart as ‘city pie-chart’ and add a legend at the top right corner of the chart.**

# Input vector

values <- c(21, 62, 10, 53)

# Labels

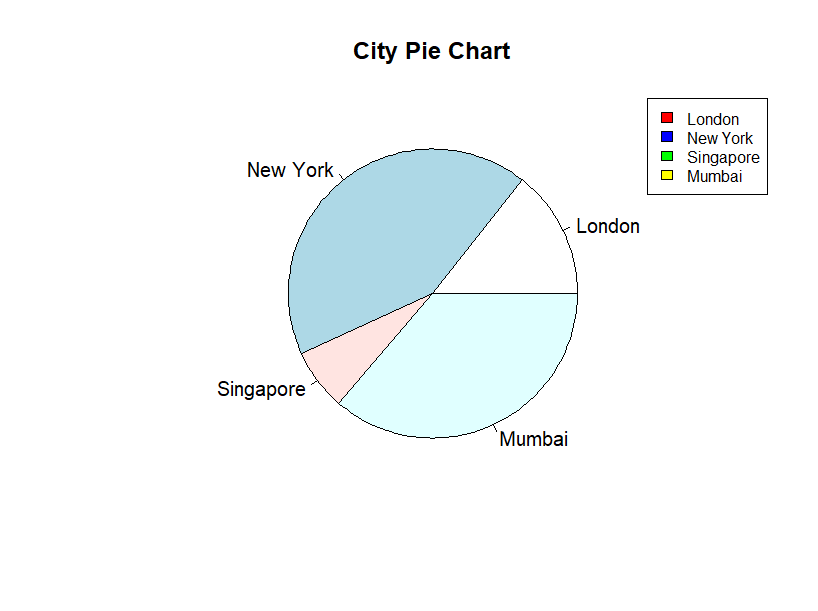
labels <- c("London", "New York", "Singapore", "Mumbai"

# Plot

pie(values, labels = labels, main = "City Pie Chart")

# Legend

legend("topright", c("London", "New York", "Singapore", "Mumbai"), cex = 0.8, fill = c("red", "blue", "green", "yellow"))



**2. Create a 3D Pie Chart for the dataset “political Knowledge” with suitable**

**labels,colours and a legend at the top right corner of the chart.**

library(plotly)

# Dataset

political\_knowledge <- c(21, 62, 10, 53)

labels <- c("London", "New York", "Singapore", "Mumbai")

colors <- c("#FF5733", "#9EE4F8", "#FFFF33", "#0066CC")

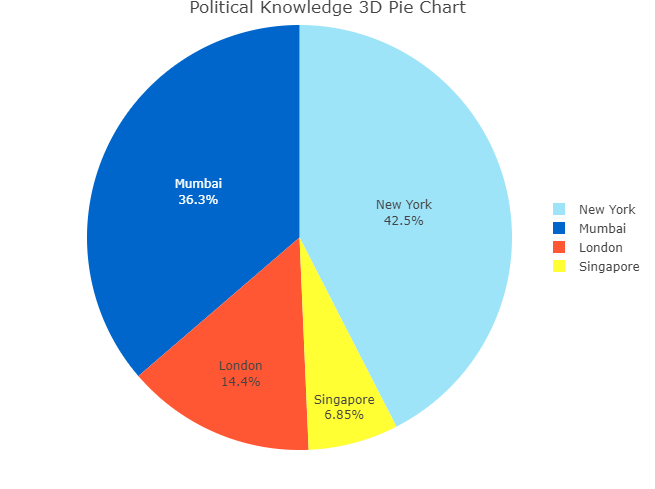
# Create pie chart

pie\_chart <- plot\_ly(type = "pie", values = political\_knowledge, labels = labels, textinfo = "label+percent", textposition = "inside", marker = list(colors = colors))

pie\_chart <- layout(pie\_chart, title = "Political Knowledge 3D Pie Chart", legend = list(x = 1, y = 0.5))

# Display pie chart

pie\_chart

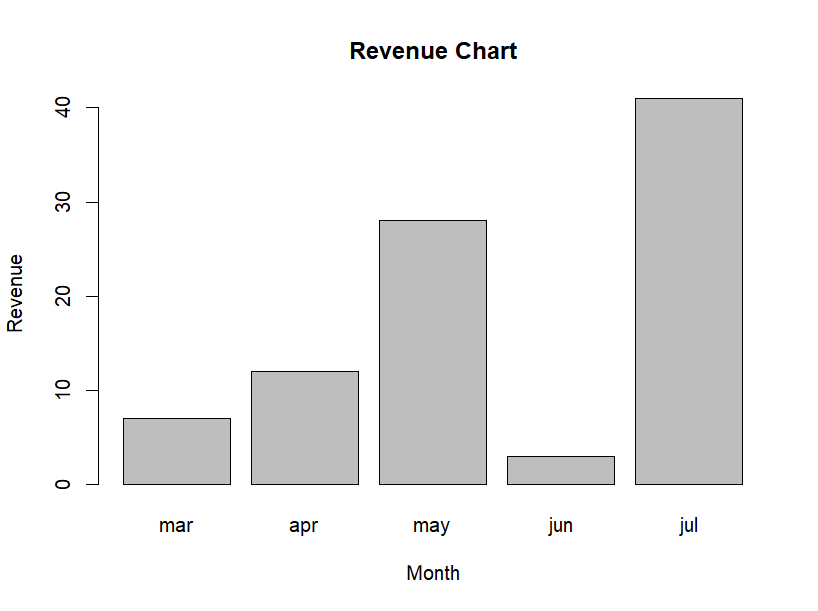


**3. Write a program for creating a bar chart using the vectors H=c(7,12,28,3,41) and M=c(“mar”, “apr”, “may”, “jun”, “jul”). Add a title to the chart as “Revenue chart”.**

H <- c(7, 12, 28, 3, 41)

M <- c("mar", "apr", "may", "jun", "jul")

barplot(H, names.arg=M, xlab="Month", ylab="Revenue", main="Revenue Chart")



**4. Make a histogram for the “AirPassengers“dataset, start at 100 on the x-axis, and from values 200 to 700, make the bins 200 wide**

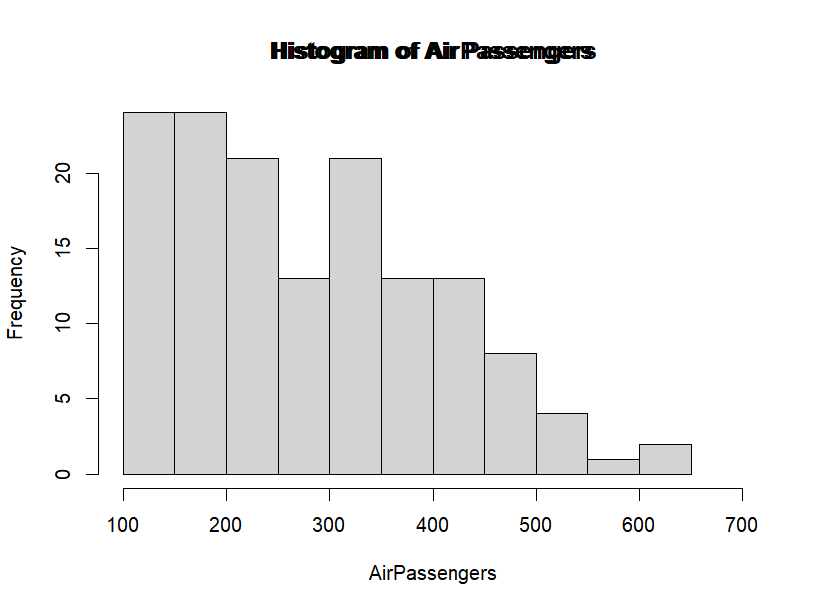
data("AirPassengers")

hist(AirPassengers, xlim = c(100, 700), breaks = seq(200, 700, 200))

title("Histogram of Air Passengers")

xlabel("Number of Air Passengers")

ylabel("Frequency")



**5. Create a Boxplot graph for the relation between “mpg”(miles per galloon) and “cyl”(number of Cylinders) for the dataset “mtcars” available in R Environment.**

# Load the data

data(mtcars)

# Create the boxplot

boxplot(mtcars$mpg ~ mtcars$cyl,

main = "Relation between mpg and cyl",

xlab = "Number of Cylinders",

ylab = "Miles per Gallon")

