R PROGRAMMING

1)

intervals <- c(1, 5, 15, 20, 50, 80, 110)

frequencies <- c(200, 450, 300, 1500, 700, 44)

cumulative <- cumsum(frequencies)

total\_frequency <- sum(frequencies)

median\_interval <- intervals[which.max(cumulative >= total\_frequency / 2)]

lower\_bound <- ifelse(median\_interval == intervals[1], 0, intervals[which(intervals == median\_interval) - 1])

frequency\_below\_median <- ifelse(median\_interval == intervals[1], 0, cumulative[which(intervals == median\_interval) - 1])

frequency\_at\_median\_interval <- frequencies[which(intervals == median\_interval)]

interval\_width <- median\_interval - lower\_bound

approx\_median <- lower\_bound + ((total\_frequency / 2 - frequency\_below\_median) / frequency\_at\_median\_interval) \* interval\_width

print(paste("Approximate median value:", approx\_median))

2)

age <- c(13, 15, 16, 16, 19, 20, 20, 21, 22, 22, 25, 25, 25, 25, 30, 33, 33, 35, 35, 35, 35, 36, 40, 45, 46, 52, 70)

mean\_age <- mean(age)

median\_age <- median(age)

print(paste("Mean of the data:", mean\_age))

print(paste("Median of the data:", median\_age))

mode\_age <- names(table(age))[which.max(table(age))]

modality <- length(unique(age)) - length(table(age))

print(paste("Mode of the data:", mode\_age))

print(paste("Modality of the data:", ifelse(modality == 0, "Unimodal", ifelse(modality == 1, "Bimodal", "Multimodal"))))

midrange\_age <- (min(age) + max(age)) / 2

print(paste("Midrange of the data:", midrange\_age))

q1 <- quantile(age, 0.25)

q3 <- quantile(age, 0.75)

print(paste("First quartile (Q1) of the data:", q1))

print(paste("Third quartile (Q3) of the data:", q3))

3)

data <- c(200, 300, 400, 600, 1000)

min\_max\_normalized <- (data - min(data)) / (max(data) - min(data))

print("Min-Max Normalized Data:")

print(min\_max\_normalized)

z\_score\_normalized <- (data - mean(data)) / sd(data)

print("Z-Score Normalized Data:")

print(z\_score\_normalized)

4)

data <- c(11, 13, 13, 15, 15, 16, 19, 20, 20, 20, 21, 21, 22, 23, 24, 30, 40, 45, 45, 45, 71, 72, 73, 75)

bin\_size <- 3

smoothed\_mean <- filter(data, rep(1/bin\_size, bin\_size), sides = 2)

smoothed\_median <- filter(data, rep(1/bin\_size, bin\_size), sides = 2, circular = TRUE)

smoothed\_boundaries <- filter(data, c(0.5, rep(1, bin\_size - 1), 0.5), sides = 2)

cat("Smoothing by bin mean:\n")

print(smoothed\_mean)

cat("\nSmoothing by bin median:\n")

print(smoothed\_median)

cat("\nSmoothing by bin boundaries:\n")

print(smoothed\_boundaries)

5)

classA=c(76,35,47,64,95,66,89,36,84,76,35,47,64,95,66,89,36,84)

classB=c(51,56,84,60,59,70,63,66,50,51,56,84,60,59,70,63,66,50)

boxplot(classA,classB)

scatter.smooth(classA,classB)

qqplot(classA,classB)

6)

age <- 35

min\_max\_normalized <- (age - min\_age) / (max\_age - min\_age)

min\_age <- 0

max\_age <- 100 # Assuming a maximum age of 100 for illustration purposes

print(paste("Min-Max Normalized Age:", min\_max\_normalized))

mean\_age <- 0 # Assuming mean age is 0 for illustration purposes

sd\_age <- 12.94 # Given standard deviation of age

z\_score\_normalized <- (age - mean\_age) / sd\_age

print(paste("Z-Score Normalized Age:", z\_score\_normalized))

decimal\_scaled <- age / 10^ceiling(log10(age))

print(paste("Decimal Scaled Age:", decimal\_scaled))

7)

pencil\_counts <- c(9, 25, 23, 12, 11, 6, 7, 8, 9, 10)

mean\_pencils <- mean(pencil\_counts)

print(paste("Mean number of pencils:", mean\_pencils))

median\_pencils <- median(pencil\_counts)

print(paste("Median number of pencils:", median\_pencils))

mode\_pencils <- names(sort(table(pencil\_counts), decreasing = TRUE))[1]

print(paste("Mode number of pencils:", mode\_pencils))

8)

x\_values <- c(4, 1, 5, 7, 10, 2, 50, 25, 90, 36)

y\_values <- c(12, 5, 13, 19, 31, 7, 153, 72, 275, 110)

plot(x\_values, y\_values, xlab = "Number of Mobile Phones Sold", ylab = "Money",main = "Scatter Plot of Mobile Phones Sold vs Money")

9)

# Define the marks scored by the student

marks <- c(55, 60, 71, 63, 55, 65, 50, 55, 58, 59, 61, 63, 65, 67, 71, 72, 75)

# Plot the data points using a histogram

par(mfrow = c(2, 1)) # Set up a 2x1 grid for plots

# (a) Equal-frequency (equi-depth) partitioning

hist(marks, breaks = "FD", main = "Histogram with Equal-Frequency Partitioning",

xlab = "Marks", ylab = "Frequency")

# (b) Equal-width partitioning

hist(marks, breaks = "Sturges", main = "Histogram with Equal-Width Partitioning",

xlab = "Marks", ylab="Frequency")

10)

speed <- c(78.3, 81.8, 82, 74.2, 83.4, 84.5, 82.9, 77.5, 80.9, 70.6)

# Calculate interquartile range (IQR)

iqr\_value <- IQR(speed)

# Calculate standard deviation

sd\_value <- sd(speed)

# Print the results

print(paste("Interquartile Range (IQR):", iqr\_value))

print(paste("Standard Deviation:",sd\_value))

11)

age <- c(13, 15, 16, 16, 19, 20, 20, 21, 22, 22, 25, 25, 25, 25, 30, 33, 33, 35, 35, 35, 35, 36, 40, 45, 46, 52, 70)

# Calculate the first quartile (Q1)

q1 <- quantile(age, 0.25)

# Calculate the third quartile (Q3)

q3 <- quantile(age, 0.75)

# Print the results

print(paste("Roughly the first quartile (Q1):", q1))

print(paste("Roughly the third quartile (Q3):", q3))

day -02):

01)

age <- c("5-6", "7-8", "9-10")

photo\_A <- c(18, 2, 20)

photo\_B <- c(22, 28, 10)

photo\_C <- c(20, 40, 40)

cov\_BC <- cov(photo\_B, photo\_C)

pref\_matrix <- cov(cbind(photo\_A, photo\_B, photo\_C))

cor\_BC <- cor(photo\_B, photo\_C)

cor\_matrix <- cor(cbind(photo\_A, photo\_B, photo\_C))

print("Covariance between B and C:")

print(cov\_BC)

print("Covariance matrix for preferences:")

print(pref\_matrix)

print("Correlation between B and C:")

print(cor\_BC)

print("Correlation matrix for preferences:")

print(cor\_matrix)

02)

data <- c(1, 1, 5, 5, 5, 5, 5, 8, 8, 10, 10, 10, 10, 12, 14, 14, 14, 15, 15, 15, 15, 15, 15, 18, 18, 18, 18, 18, 20, 20, 20, 20, 20, 20, 20, 21, 21, 21, 21, 25, 25, 25, 25, 25, 28, 28, 30, 30, 30)

bins <- cut(data, breaks=3, labels=FALSE)

print(bins)

bin\_means <- tapply(data, bins, mean)

bin\_boundaries <- tapply(data, bins, range)

print(bin\_means)

print(bin\_boundaries)

hist(data, breaks=3, main="Histogram with Equal-Frequency Partitioning", xlab="Value", ylab="Frequency")

03)

class\_A <- c(76, 35, 47, 64, 95, 66, 89, 36, 84)

class\_B <- c(51, 56, 84, 60, 59, 70, 63, 66, 50)

mean\_A <- mean(class\_A)

mean\_B <- mean(class\_B)

median\_A <- median(class\_A)

median\_B <- median(class\_B)

range\_A <- max(class\_A) - min(class\_A)

range\_B <- max(class\_B) - min(class\_B)

cat("Class A:\n")

cat("Mean:", mean\_A, "\n")

cat("Median:", median\_A, "\n")

cat("Range:", range\_A, "\n")

cat("\nClass B:\n")

cat("Mean:", mean\_B, "\n")

cat("Median:", median\_B, "\n")

cat("Range:", range\_B, "\n")

boxplot(class\_A, class\_B, names = c("Class A", "Class B"), col = c("blue", "green"),main = "Comparison of Exam Scores",xlab = "Class", ylab = "Scores")

04)

data <- c(200, 300, 400, 600, 1000)

normalized\_minmax <- (data - 50000) / (100000 - 50000)

normalized\_zscore <- (data - mean(data)) / sd(data)

cat("Min-Max Normalization:\n", normalized\_minmax, "\n")

cat("Z-Score Normalization:\n", normalized\_zscore, "\n")

05)

data("AirPassengers")

hist(AirPassengers, xlim=c(100, 700), breaks=seq(200, 700, by=150), main="Histogram of AirPassengers Dataset", xlab="Passenger Count", ylab="Frequency")

06)

data(mtcars)

plot(mpg ~ qsec, data=mtcars, type="l", col="blue", xlab="Qsec", ylab="Miles per gallon", main="Line Chart of MPG vs Qsec")

07)

data("water")

plot(water$hardness, water$mortality, xlab="Hardness", ylab="Mortality", main="Scatterplot of Mortality vs. Hardness")

model <- lm(mortality ~ hardness, data=water)

abline(model, col="red")

predicted\_mortality <- predict(model, newdata=data.frame(hardness=88))

print(predicted\_mortality)

08)

data(mtcars)

boxplot(mpg ~ cyl, data=mtcars, xlab="Number of Cylinders", ylab="Miles per Gallon", main="Boxplot of MPG vs. Cylinders")

09)

points <- c(10, 15, 20, 25, 30, 35, 40, 45, 50, 200)

boxplot(points, main="Boxplot of Points Scored", ylab="Points")

title(main="Boxplot of Points Scored", ylab="Points")

10)

diabetes <- read.csv("diabetes.csv")

plot(diabetes$Age, diabetes$BloodPressure,

xlab="Age", ylab="Blood Pressure",

main="Blood Pressure vs Age",

col="blue", pch=19)

age\_groups <- cut(diabetes$Age, breaks=c(0, 20, 40, 60, 80, 100))

bar\_data <- aggregate(BloodPressure ~ age\_groups, data=diabetes, mean)

barplot(bar\_data$BloodPressure,

names.arg=levels(age\_groups),

xlab="Age Group", ylab="Average Blood Pressure",

main="Average Blood Pressure by Age Group",

col="green")

# weka

1.Covariance and correlation

Children of three ages are asked to indicate their preference for three photographs of adults. Do the data suggest that there is a significant relationship between age and photograph preference? What is wrong with this study?

**Photograph:**

**Age of child** A B C

5-6 years: 18 22 20

7-8 years: 2 28 40

9-10 years: 20 10 40

(i)Use cov() to calculate the sample covariance between B and C.

(ii)Use another call to cov() to calculate the sample covariance matrix for the preferences.

(iii)Use cor() to calculate the sample correlation between B and C.

(iv)Use another call to cor() to calculate the sample correlation matrix for the preferences.

CODE:

(i)b<-c(22, 28, 10)

c<-c(20, 40, 40)

cov(b,c)

(ii)a<-c(18, 2, 20)

b<-c(22, 28, 10)

c<-c(20, 40, 40)

pre<-cbind(a,b,c)

cov(pre)

(iii).b<-c(22, 28, 10)

c<-c(20, 40, 40)

cor(b,c)

(iv)a<-c(18, 2, 20)

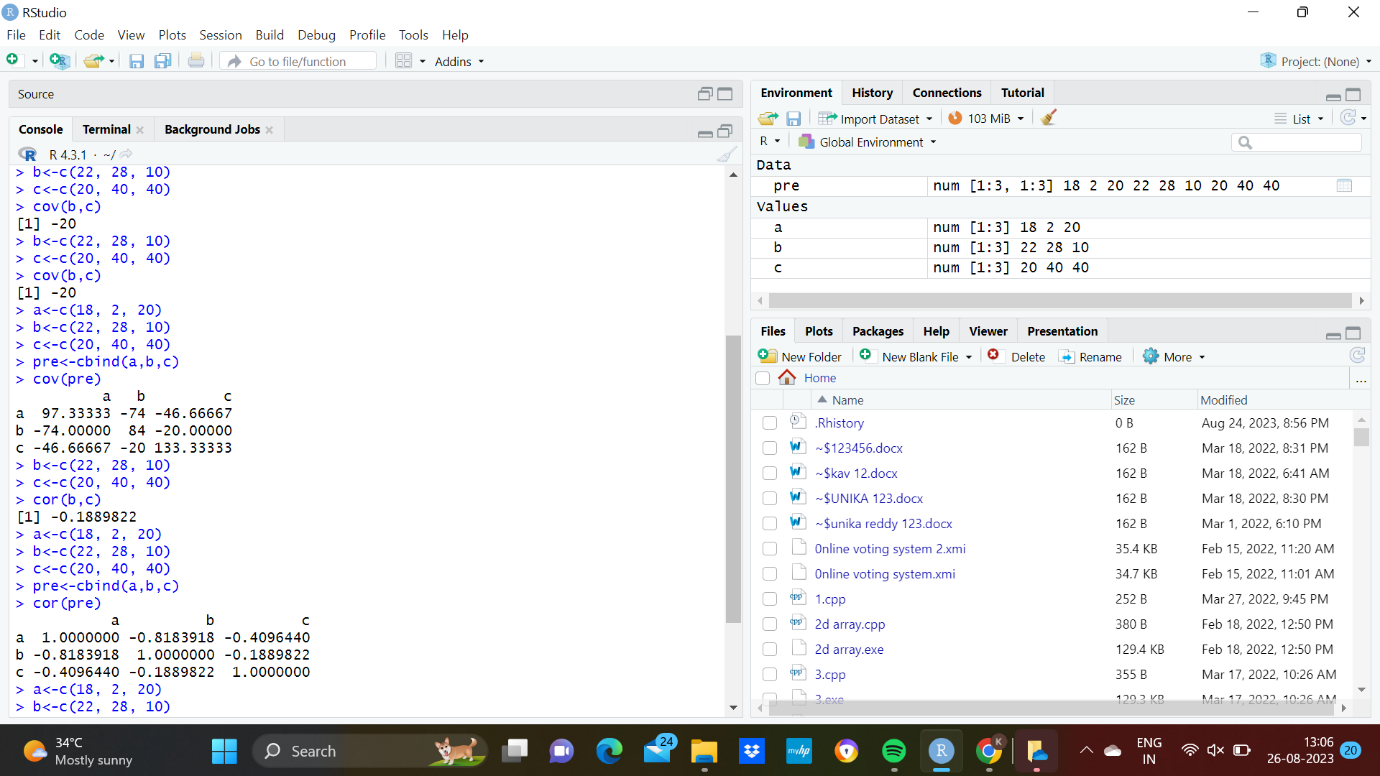
b<-c(22, 28, 10)

c<-c(20, 40, 40)

pre<-cbind(a,b,c)

cor(pre)

Output



2.Imagine that you have selected data from the All Electronics data warehouse for analysis. The data set will be huge! The following data are a list of All Electronics prices for commonly sold items (rounded to the nearest dollar). The numbers have been sorted: 1, 1, 5, 5, 5, 5, 5, 8, 8, 10, 10, 10, 10, 12, 14, 14, 14, 15, 15, 15, 15, 15, 15, 18, 18, 18, 18, 18, 8, 8, 10, 10, 10, 10, 12, 14, 14, 14, 15, 15, 15, 15, 15, 15, 18, 18, 18, 18, 18,

|  |
| --- |
| 18, 18, 18, 20, 20, 20, 20, 20, 20, 20, 21, 21, 21, 21, 25, 25, 25, 25, 25, 28, 28, 30, |
| 30, 30.  (i) Partition the dataset using an equal-frequency partitioning method with bin equal to 3 (ii) apply data smoothing using bin means and bin boundary. (iii) Plot Histogram for the above frequency division |

CODE:

data<-c(1, 1, 5, 5, 5, 5, 5, 8, 8, 10, 10, 10, 10, 12, 14, 14, 14, 15, 15, 15, 15, 15, 15, 18, 18, 18, 18, 18, 18, 18, 18, 20, 20, 20, 20, 20, 20, 20, 21, 21, 21, 21, 25, 25, 25, 25, 25, 28, 28, 30, 30, 30)

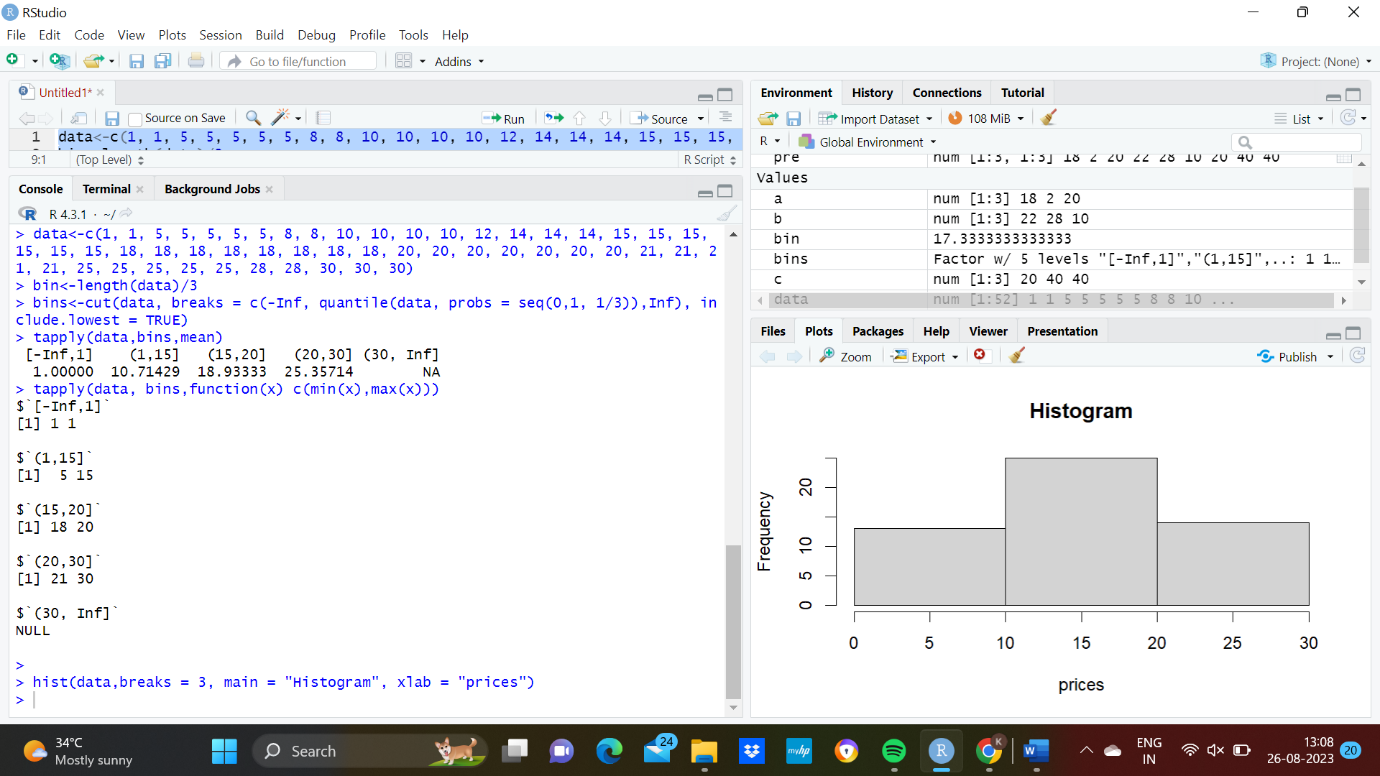
bin<-length(data)/3

bins<-cut(data, breaks = c(-Inf, quantile(data, probs = seq(0,1, 1/3)),Inf), include.lowest = TRUE)

tapply(data,bins,mean)

tapply(data, bins,function(x) c(min(x),max(x)))

hist(data,breaks = 3, main = "Histogram", xlab = "prices")



3.Two Maths teachers are comparing how their Year 9 classes performed in the end of year exams. Their results are as follows:  
Class A: 76, 35, 47, 64, 95, 66, 89, 36, 8476,35,47,64,95,66,89,36,84

Class B: 51, 56, 84, 60, 59, 70, 63, 66, 5051,56,84,60,59,70,63,66,50

(i) Find which class had scored higher mean, median and range.  
(ii) Plot above in boxplot and give the inferences

Class B: 51, 56, 84, 60, 59, 70, 63, 66, 5051,56,84,60,59,70,63,66,50

CODE:

A <- c(76, 35, 47, 64, 95, 66, 89, 36, 84)

B <- c(51, 56, 84, 60, 59, 70, 63, 66, 50)

mean\_A <- mean(A)

median\_A <- median(A)

range\_A <- max(A) - min(A)

mean\_B <- mean(B)

median\_B <- median(B)

range\_B <- max(B) - min(B)

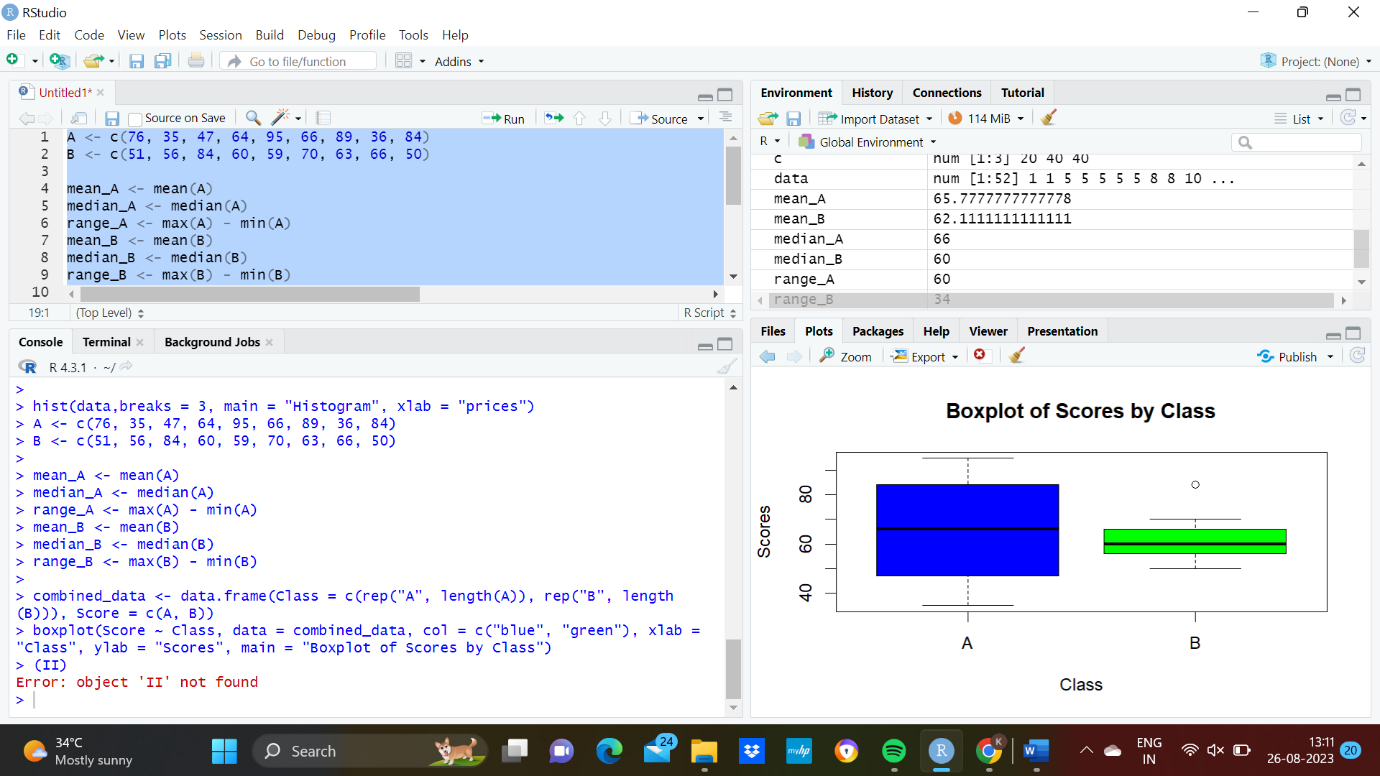
combined\_data <- data.frame(Class = c(rep("A", length(A)), rep("B", length(B))), Score = c(A, B))

boxplot(Score ~ Class, data = combined\_data, col = c("blue", "green"), xlab = "Class", ylab = "Scores", main = "Boxplot of Scores by Class")

(II)

combined\_data <- data.frame(Class = c(rep("A", length(A)), rep("B", length(B))), Score = c(A, B))

boxplot(Score ~ Class, data = combined\_data, col = c("blue", "green"), xlab = "Class", ylab = "Scores", main = "Boxplot of Scores by Class")



4.Let us consider one example to make the calculation method clear. Assume that the minimum and maximum values for the feature F are $50,000 and $100,000 correspondingly. It needs to range F from 0 to 1. In accordance with min-max normalization, v = $80,

b) Use the two methods below to normalize the following group of data: 200, 300, 400, 600, 1000

(a) min-max normalization by setting min = 0 and max = 1

(b) z-score normalization

CODE:

data <- c(200, 300, 400, 600, 1000)

min\_value <- 50000

max\_value <- 100000

v <- 80

min\_max\_normalized <- (v - min\_value) / (max\_value - min\_value)

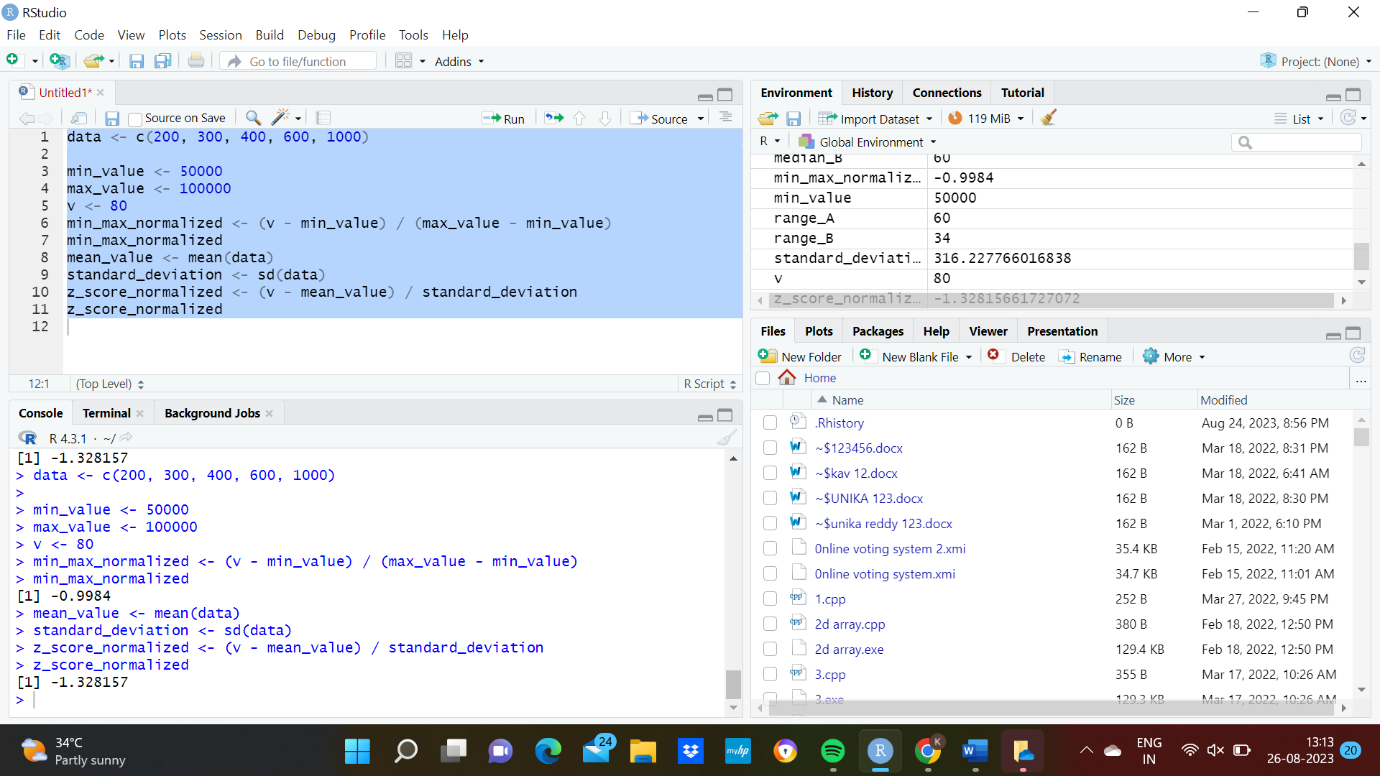
min\_max\_normalized

mean\_value <- mean(data)

standard\_deviation <- sd(data)

z\_score\_normalized <- (v - mean\_value) / standard\_deviation

z\_score\_normalized

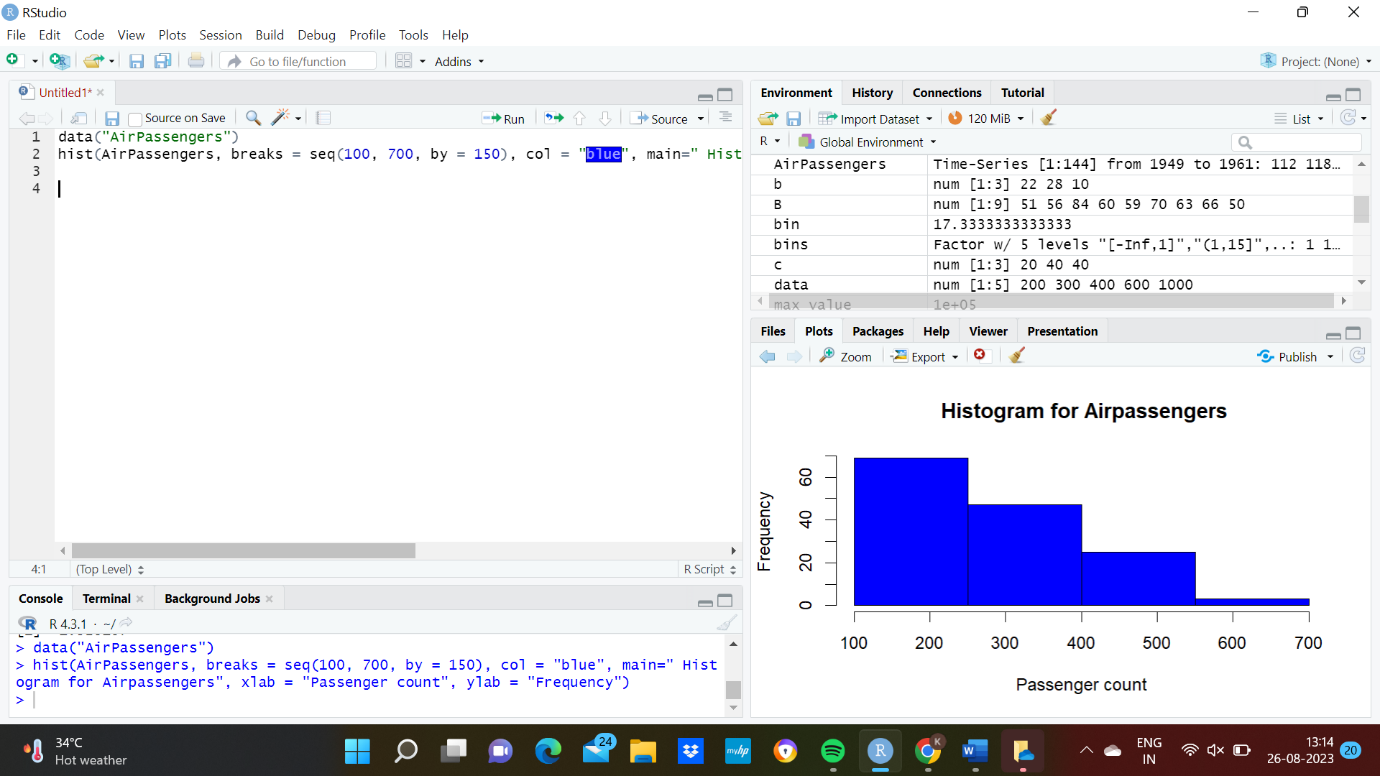


5.Make a histogram for the “AirPassengers “dataset, start at 100 on the x-axis, and from values 200 to 700, make the bins 150 wide

CODE:

data("AirPassengers")

hist(AirPassengers, breaks = seq(100, 700, by = 150), col = "blue", main=" Histogram for Airpassengers", xlab = "Passenger count", ylab = "Frequency")



6.Obtain Multiple Lines in Line Chart using a single Plot Function in R.Use attributes“mpg”and“qsec”of the dataset “mtcars”

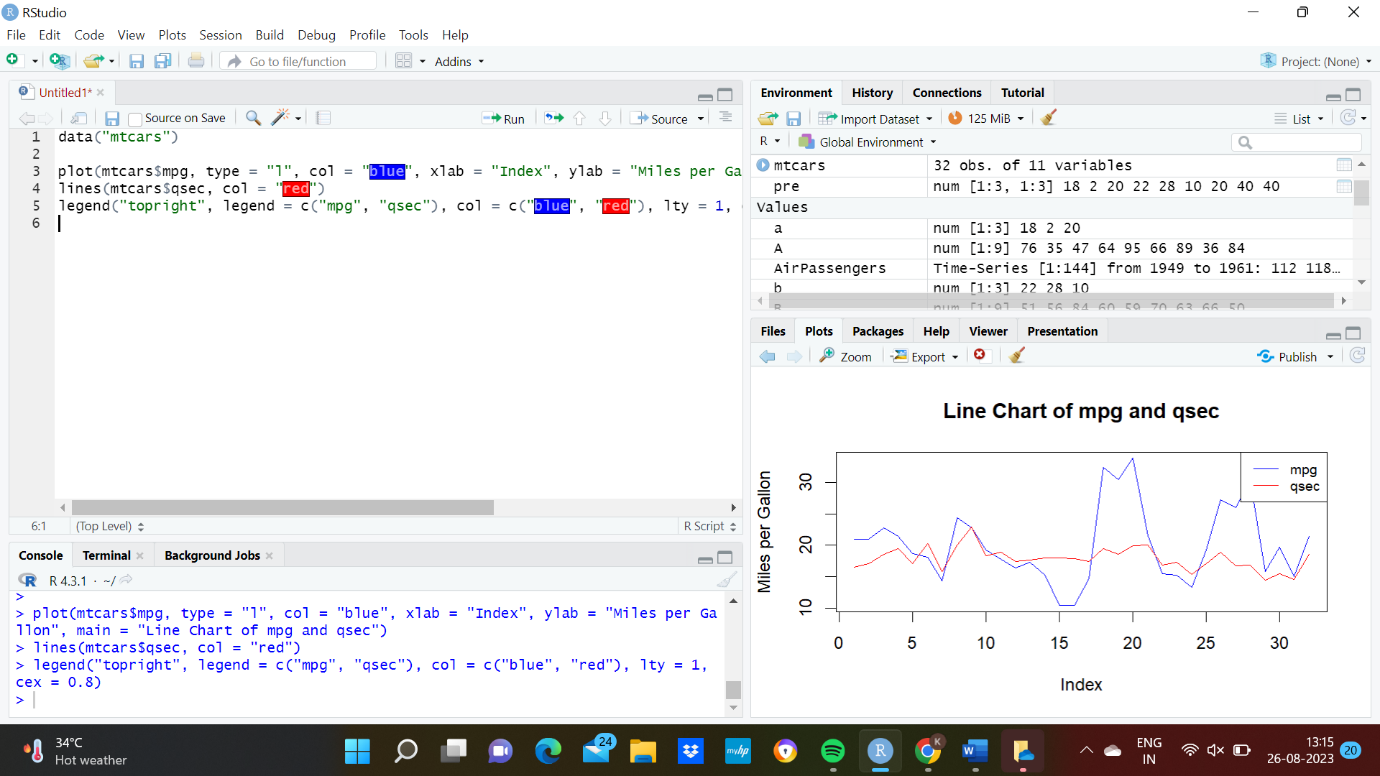
CODE:

data("mtcars")

plot(mtcars$mpg, type = "l", col = "blue", xlab = "Index", ylab = "Miles per Gallon", main = "Line Chart of mpg and qsec")

lines(mtcars$qsec, col = "red")

legend("topright", legend = c("mpg", "qsec"), col = c("blue", "red"), lty = 1, cex = 0.8)



7.Download the Dataset "water" From R dataset Link.Find out whether there is a linear relation between attributes"mortality" and"hardness" by plot function.Fit the Data into the Linear Regression model.Predict the mortality for the hardness=88.

CODE:

data("iris")

str(iris)

plot(iris$Sepal.Length, iris$Petal.Length, main = "Scatter plot of Sepal.Length vs. Petal.Length",xlab = "Sepal.Length", ylab = "Petal.Length", col = "blue", pch = 16)

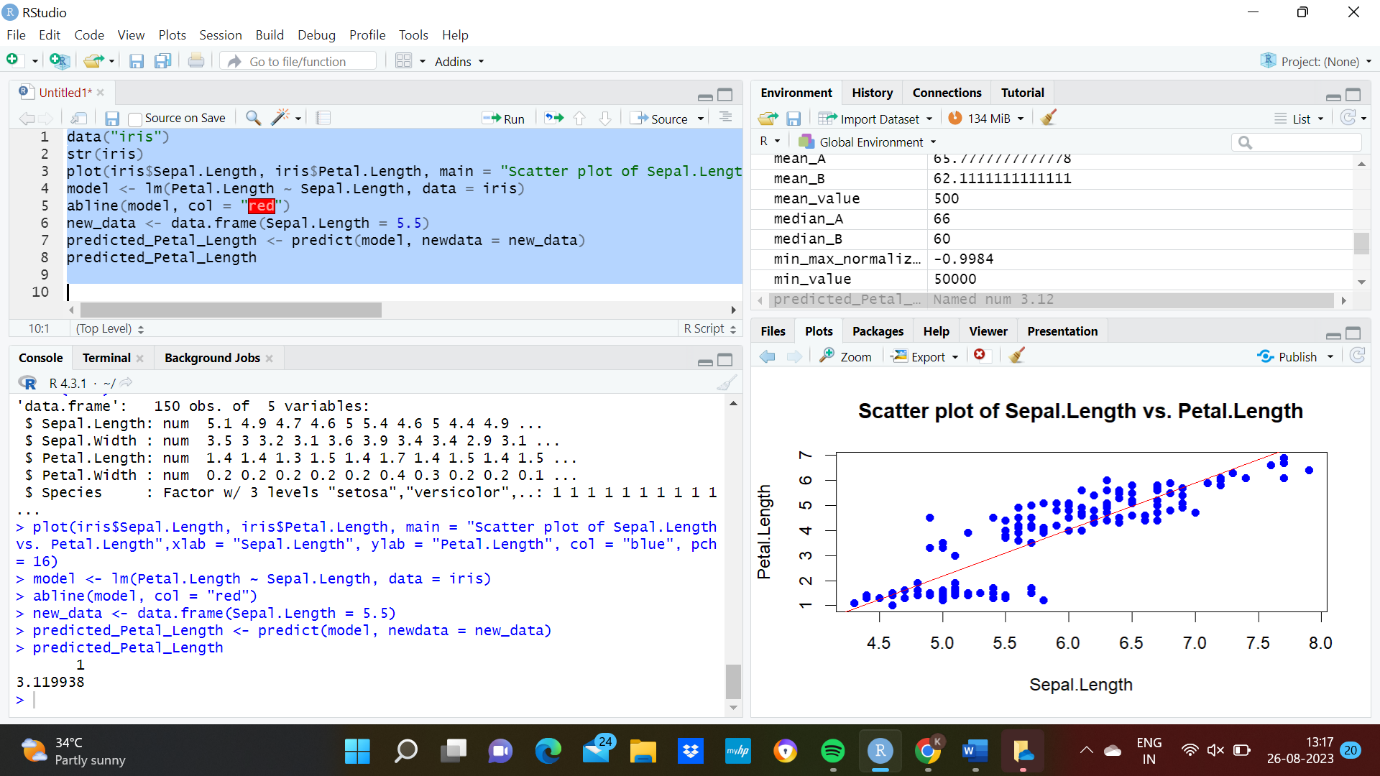
model <- lm(Petal.Length ~ Sepal.Length, data = iris)

abline(model, col = "red")

new\_data <- data.frame(Sepal.Length = 5.5)

predicted\_Petal\_Length <- predict(model, newdata = new\_data)

predicted\_Petal\_Length

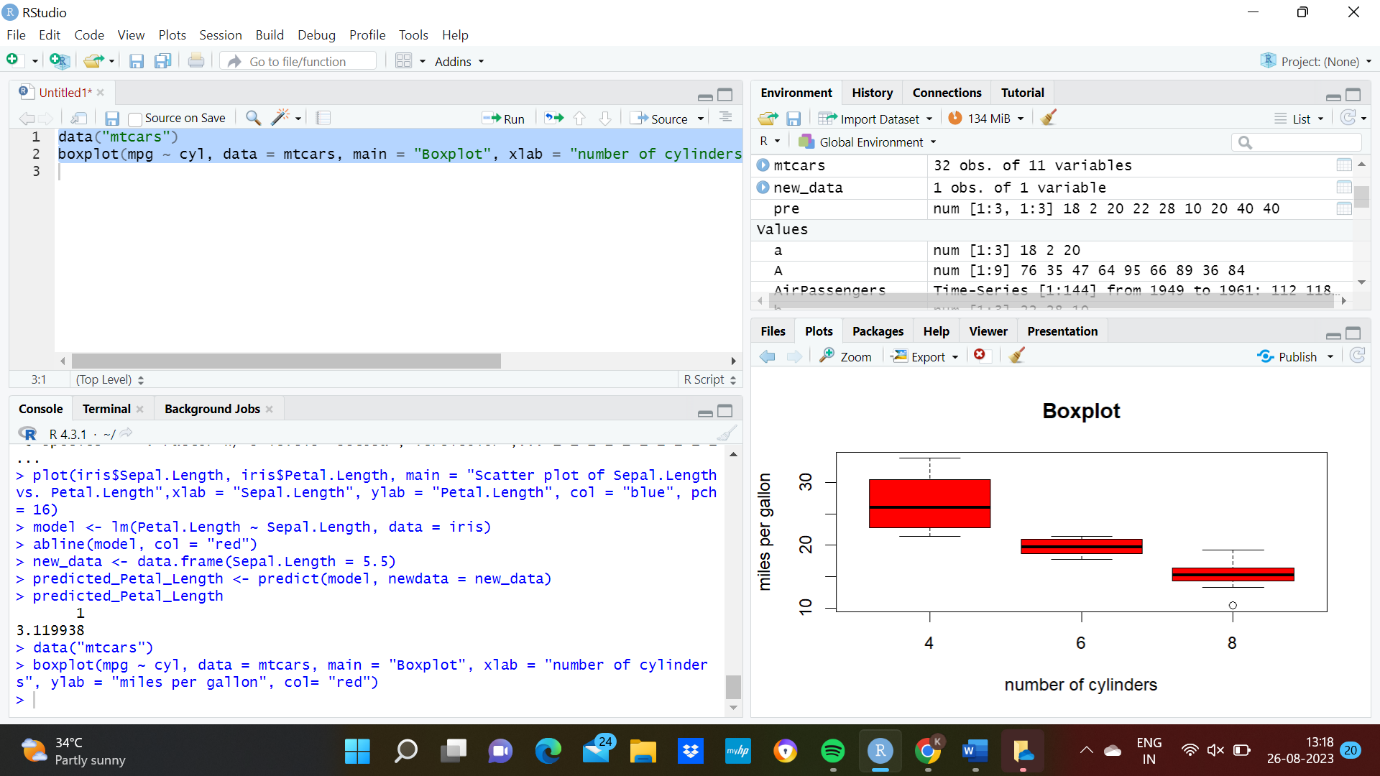


8.Create a Boxplot graph for the relation between "mpg"(miles per galloon) and "cyl"(number of Cylinders) for the dataset "mtcars" available in R Environment.

CODE:

data("mtcars")

boxplot(mpg ~ cyl, data = mtcars, main = "Boxplot", xlab = "number of cylinders", ylab = "miles per gallon", col= "red")



9. Assume the Tennis coach wants to determine if any of his team players are scoring

outliers. To visualize the distribution of points scored by his players, then how can he

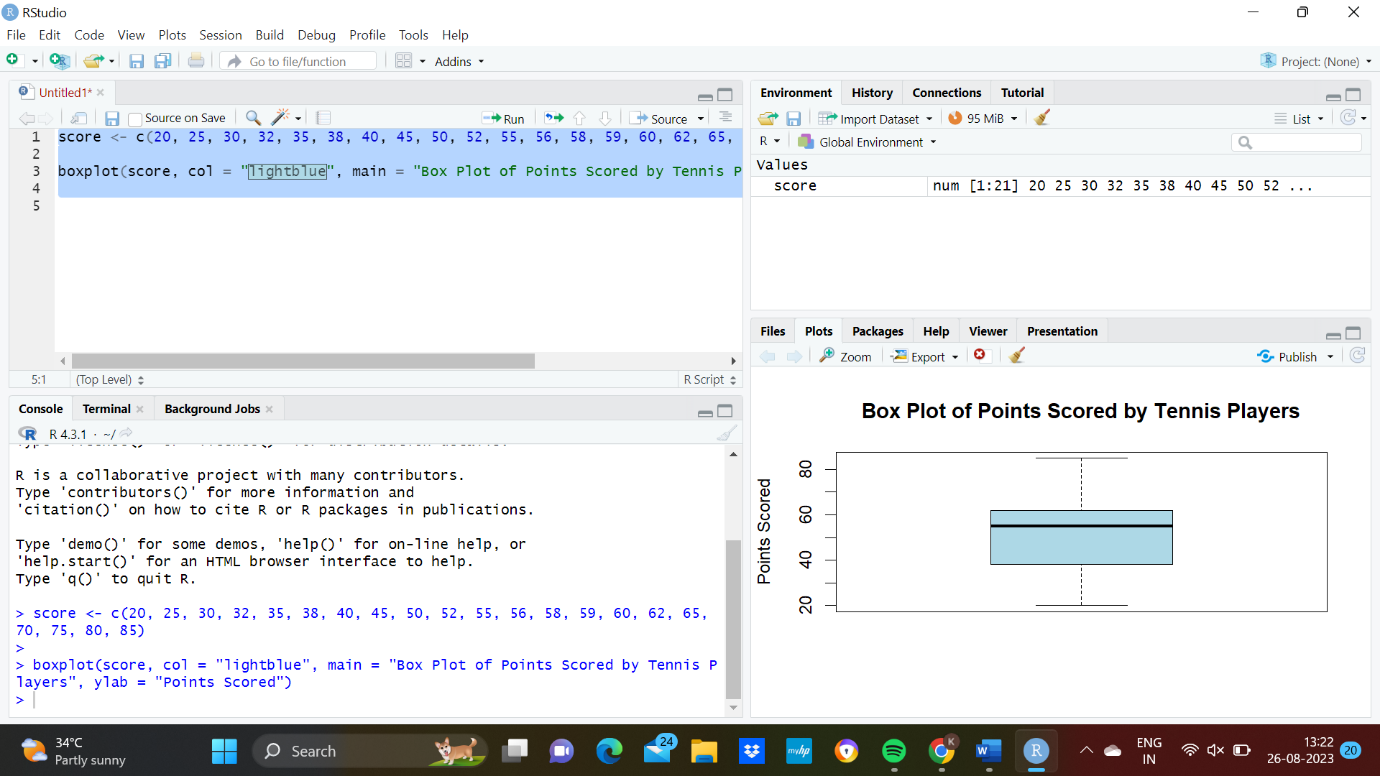
decide to develop the box plot? Give suitable example using Boxplot visualization

technique.

CODE:

score <- c(20, 25, 30, 32, 35, 38, 40, 45, 50, 52, 55, 56, 58, 59, 60, 62, 65, 70, 75, 80, 85)

boxplot(score, col = "lightblue", main = "Box Plot of Points Scored by Tennis Players", ylab = "Points Scored")



10. Implement using R language in which age group of people are affected by blood pressure based on the diabetes dataset show it using scatterplot and bar chart (that is Blood Pressure vs Age using dataset “diabetes.csv”)

CODE:

dia<-read.csv("C://Users//FLORENCIA ABEL//OneDrive//Documents//diabetes.csv")

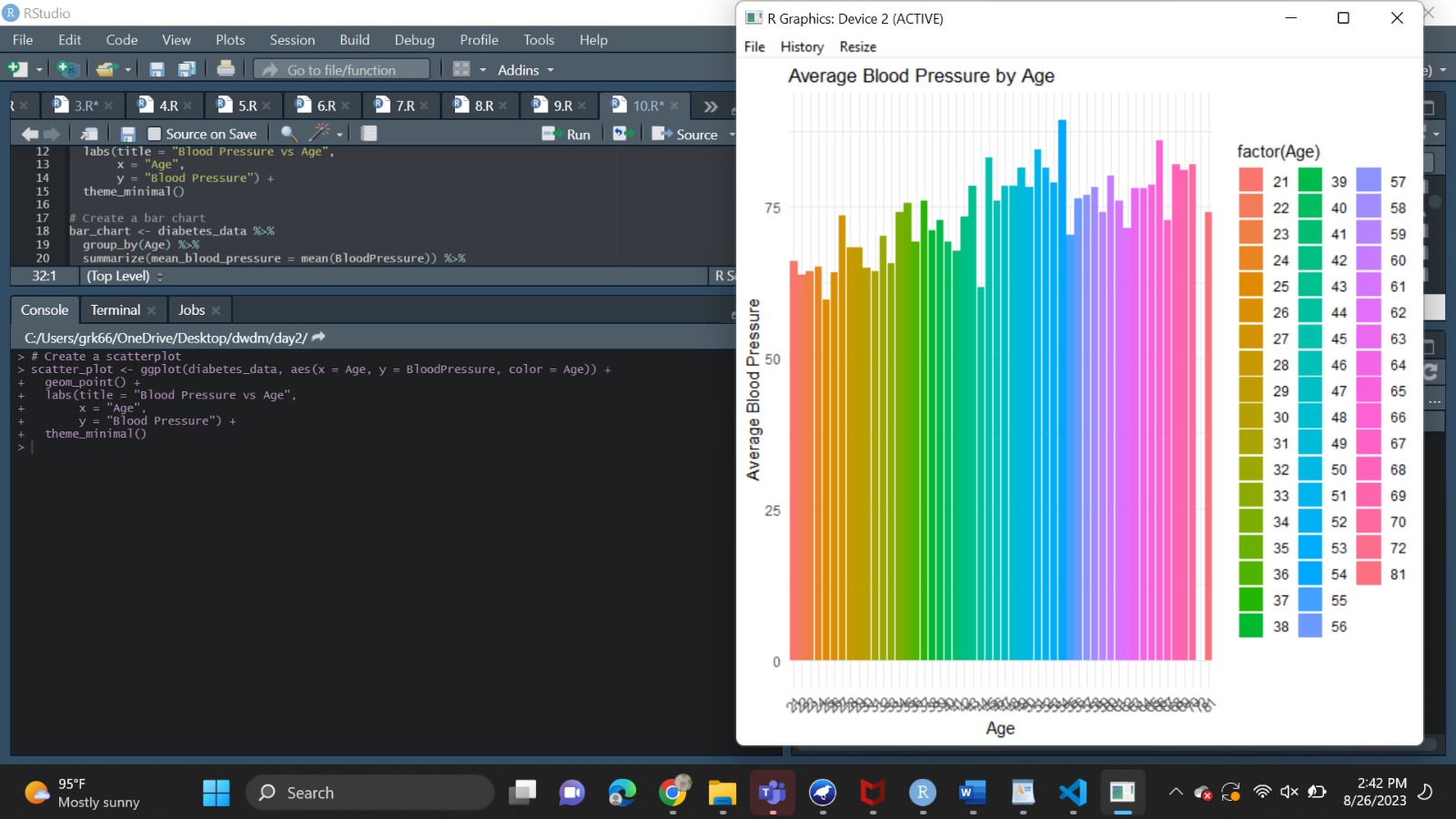
View(dia)

plot(dia$Age, dia$BloodPressure, xlab = "Age", ylab = "Blood Pressure", main = "Blood Pressure vs. Age", col = "blue",pch = 16)

age\_group\_labels <- cut(dia$Age, breaks = c(0, 35, 55, Inf), labels = c("Young", "Middle-aged", "Elderly"))

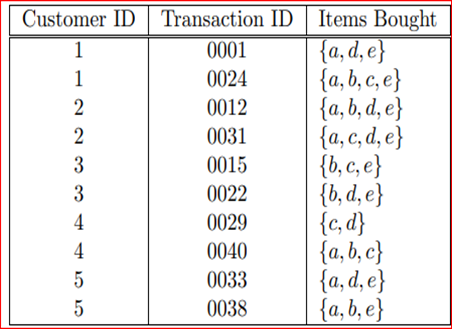
age\_group\_avg\_bp <- tapply(dia$BloodPressure, age\_group\_labels, mean)

barplot(age\_group\_avg\_bp, main = "Average Blood Pressure by Age Group",xlab = "Age Group",ylab = "Average Blood Pressure", col = "steelblue", ylim = c(0, max(age\_group\_avg\_bp) \* 1.2))



1.Consider the data set and perform the Apriori Algorithm and FP algorithm support:3 and

confidence=50%



Input:

@relation dataset

@attribute a{true,false}

@attribute b{true,false}

@attribute c{true,false}

@attribute d{true,false}

@attribute e{true,false}

@data

true false false true true

true true true false true

true true false true true

true false true true true

false true true false true

false true false true true

false false true true false

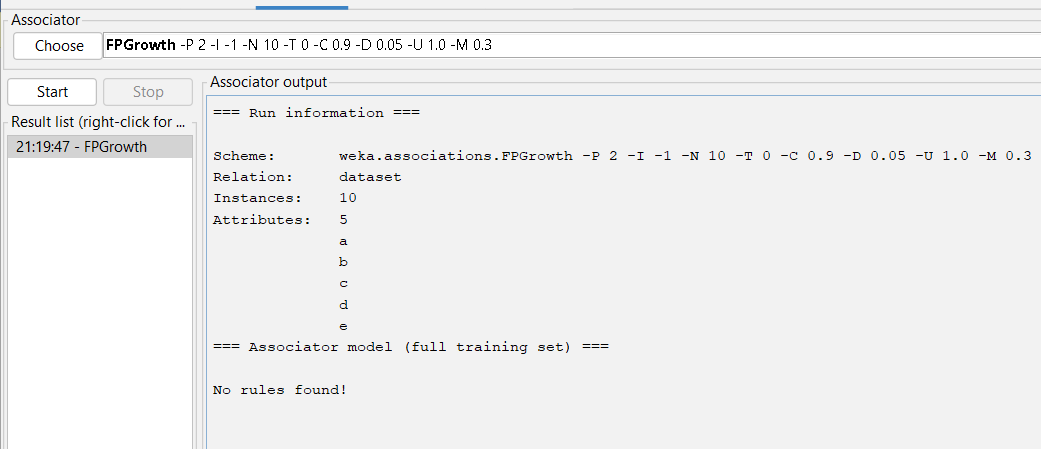
true true true false false

true false false true true

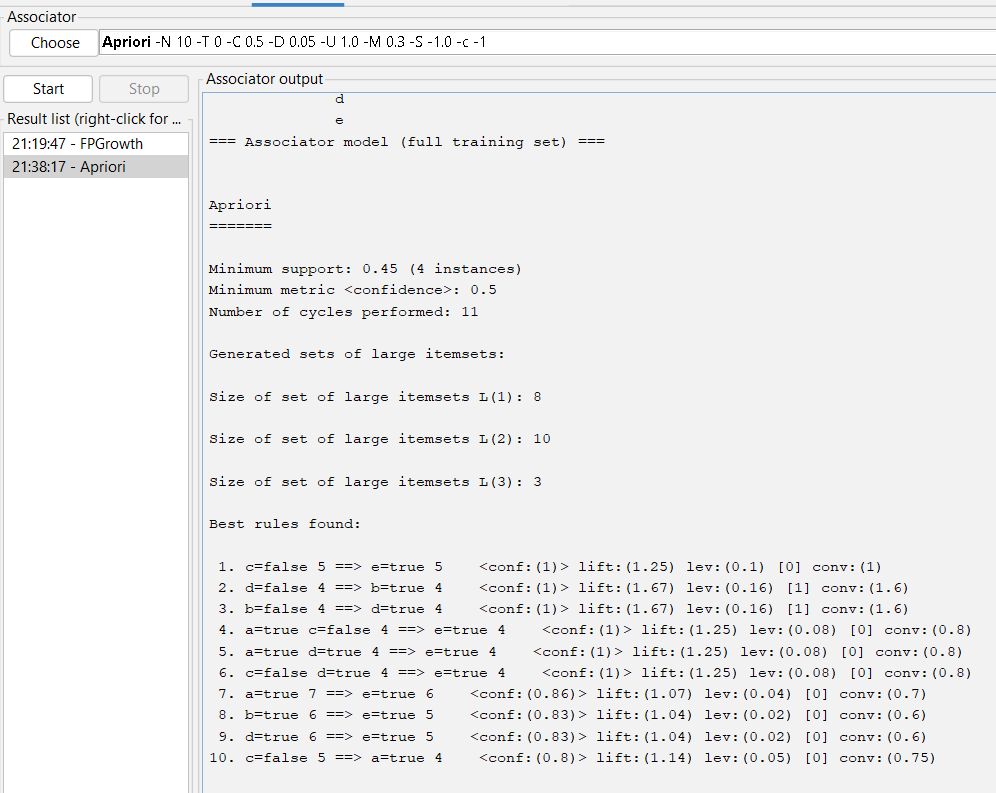
true true false false true

output:

FPGROWTH:



APRIORI ALGORITHM:



2.Consider the data set and perform the Apriori Algorithm and FP algorithm support:3 and confidence=50%

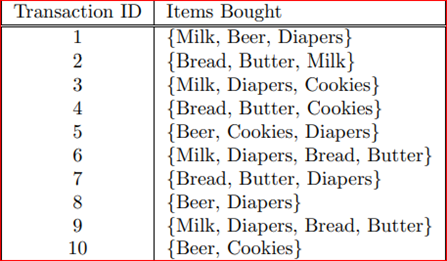
Consider the market basket transactions shown in the above table.

(a) What is the maximum number of association rules that can be extracted

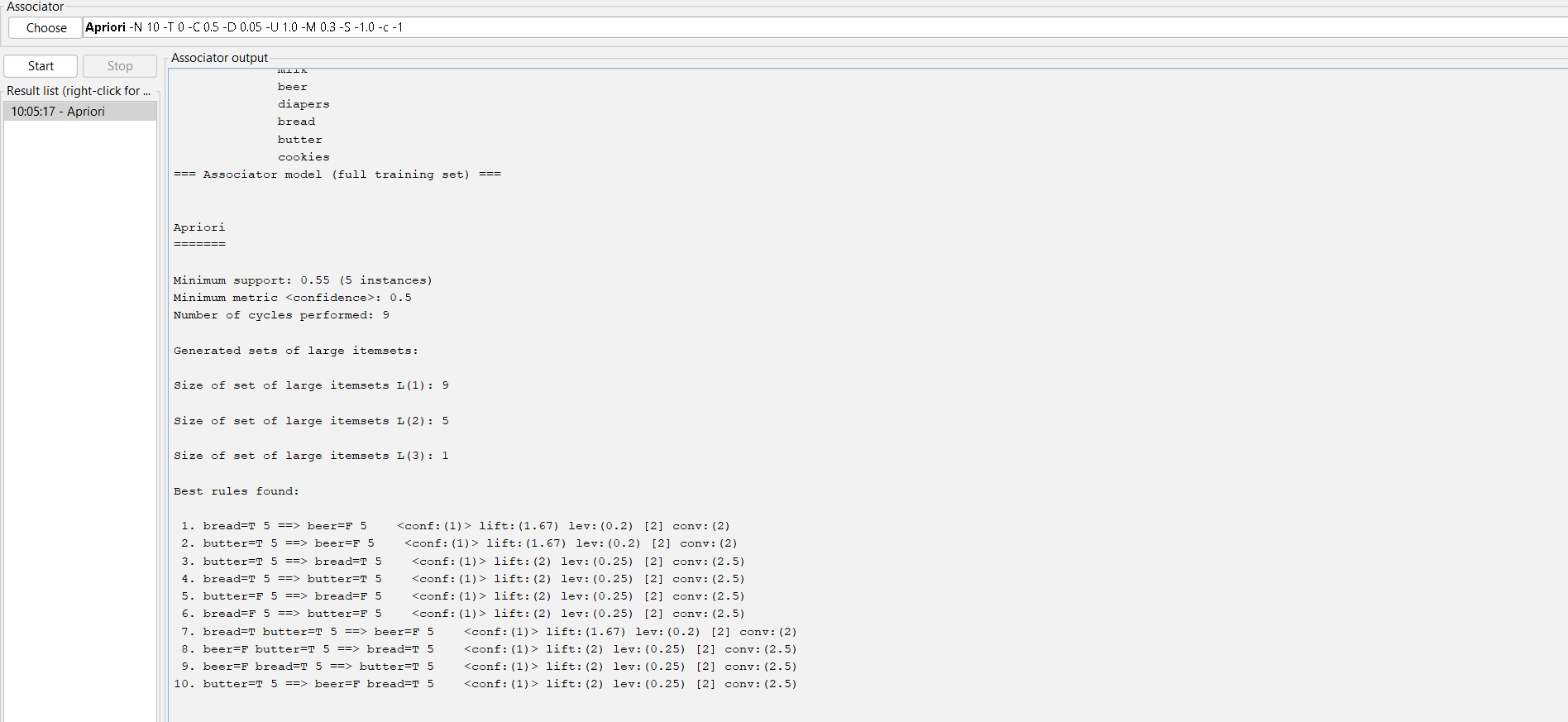
from this data (including rules that have zero support)?

(b) What is the maximum size of frequent itemsets that can be extracted

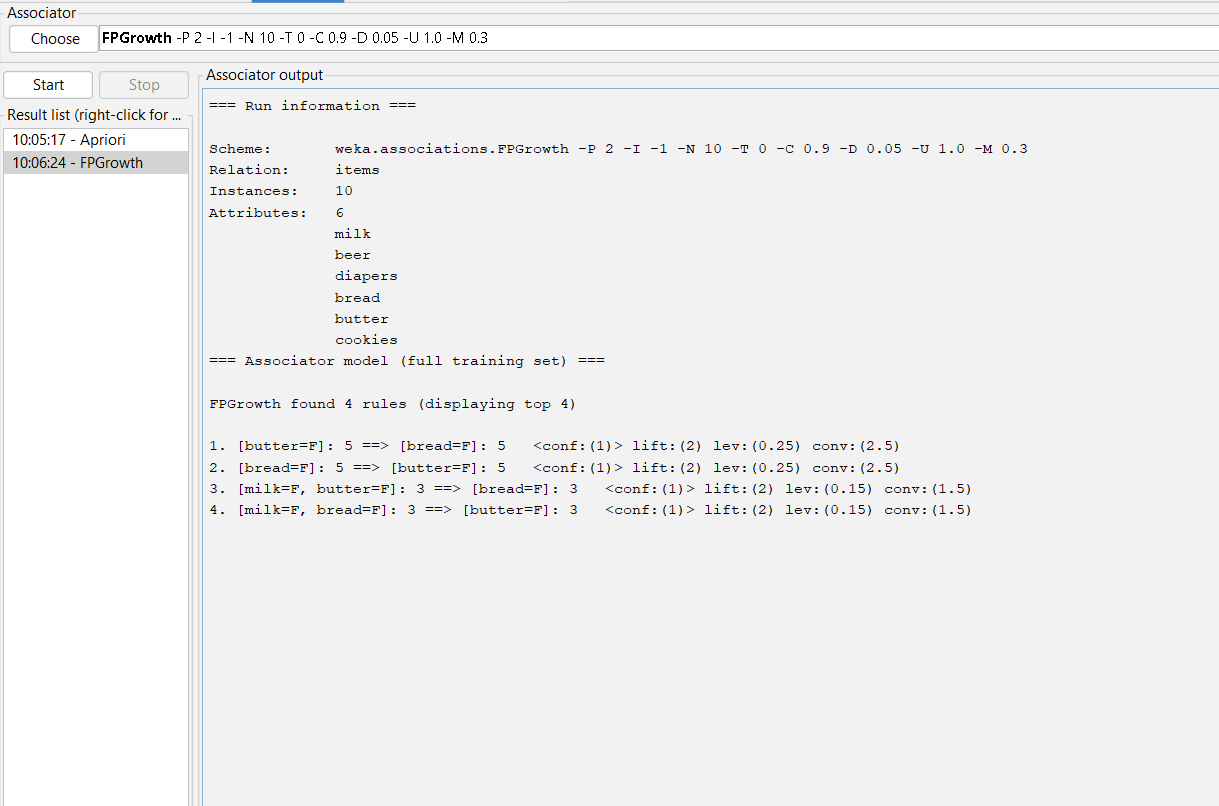
(assuming minsup > 0)?



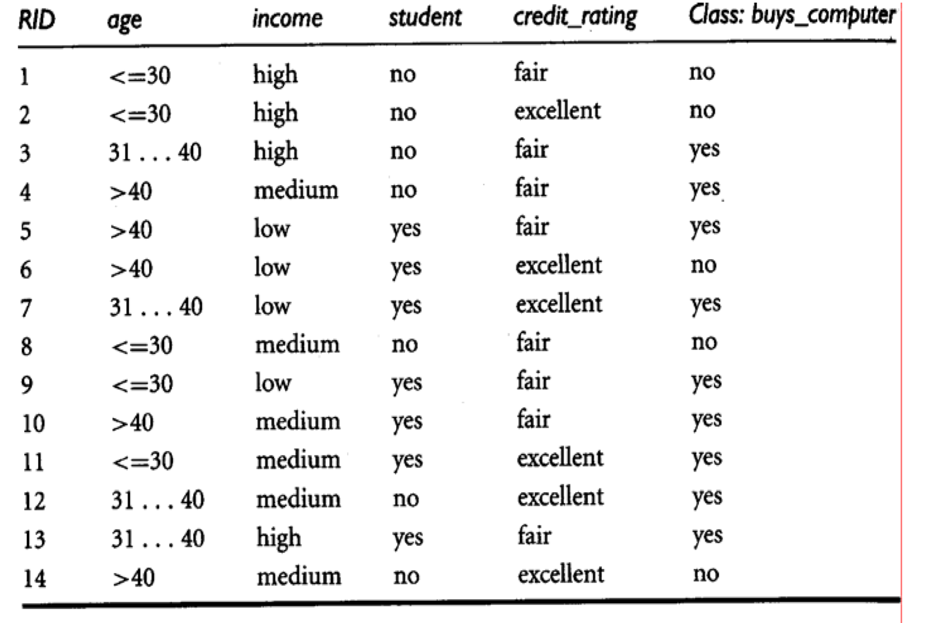
Apriori algorithm:



Fp growth algorithm:



3.Bayes classification and descion tree (using training and test data)



Input:

@relation decision\_tree

@attribute age{young,middle,old}

@attribute income{low,medium,high}

@attribute student{yes,no}

@attribute Creit\_rating{fair,excellent}

@attribute class{yes,no}

@data

young high no fair no

young high no excellent no

middle high no fair yes

old medium no fair yes

old low yes fair yes

old low yes excellent no

middle low yes excellent yes

young medium no fair no

young low yes fair yes

old medium yes fair yes

young medium yes excellent yes

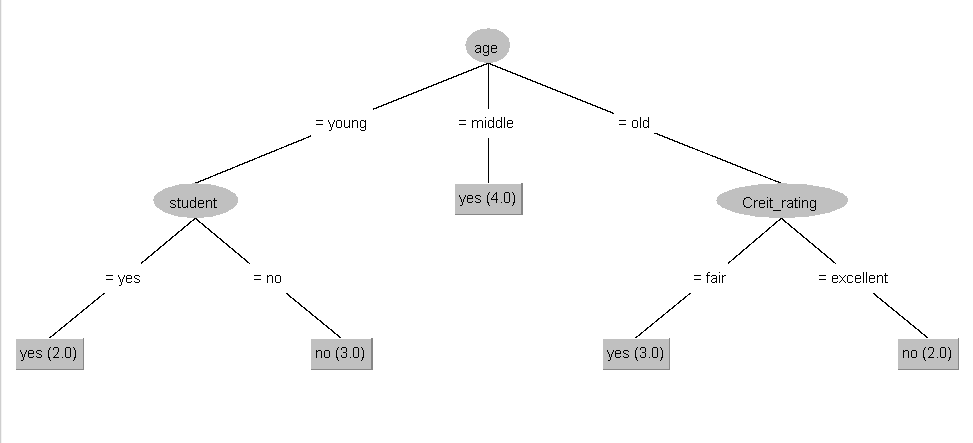
middle medium no excellent yes

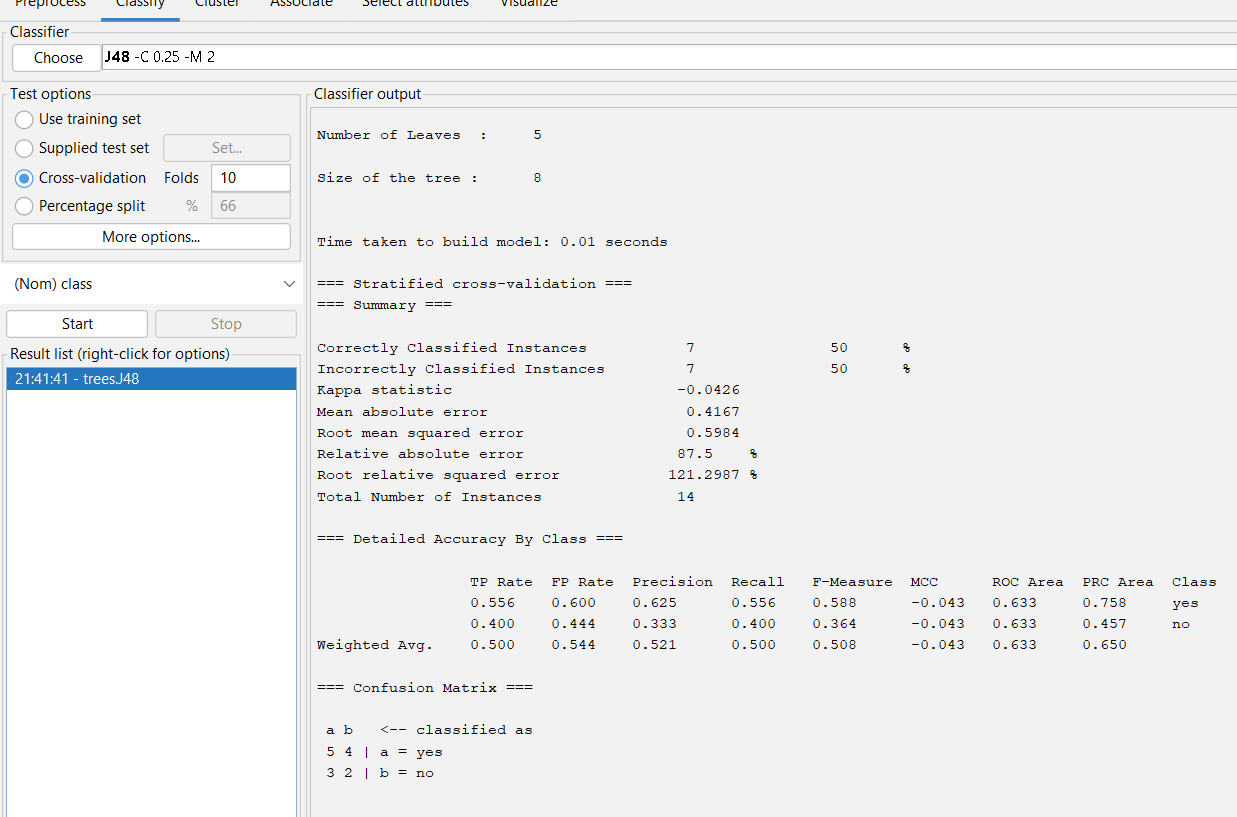
middle high yes fair yes

old medium no  excellent no

output:

tree:





4.Analysis the dataset “diabetes. csv” how the diabetes trend is for different age people, using linear regression and multiple regression.

Input:

data<-read.csv("C:/Users/Hari Naidu/Desktop/POM/download papers/diabetes.csv")

data

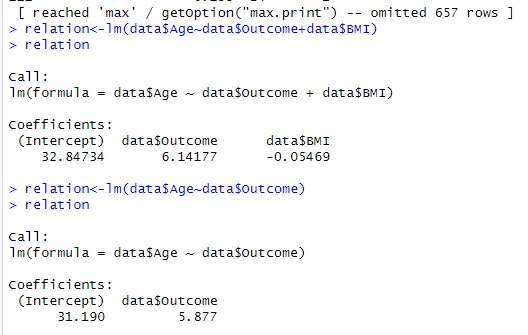
relation<-lm(data$Age~data$Outcome)

relation

relation<-lm(data$Age~data$Outcome+data$BMI)

relation

output:



5.Implement using WEKA for the given Suppose a database has five transactions. Let min sup= 50%(2) and min con f = 80%.

**Transactions Items**

T1 (M, O, N, K, E, Y)

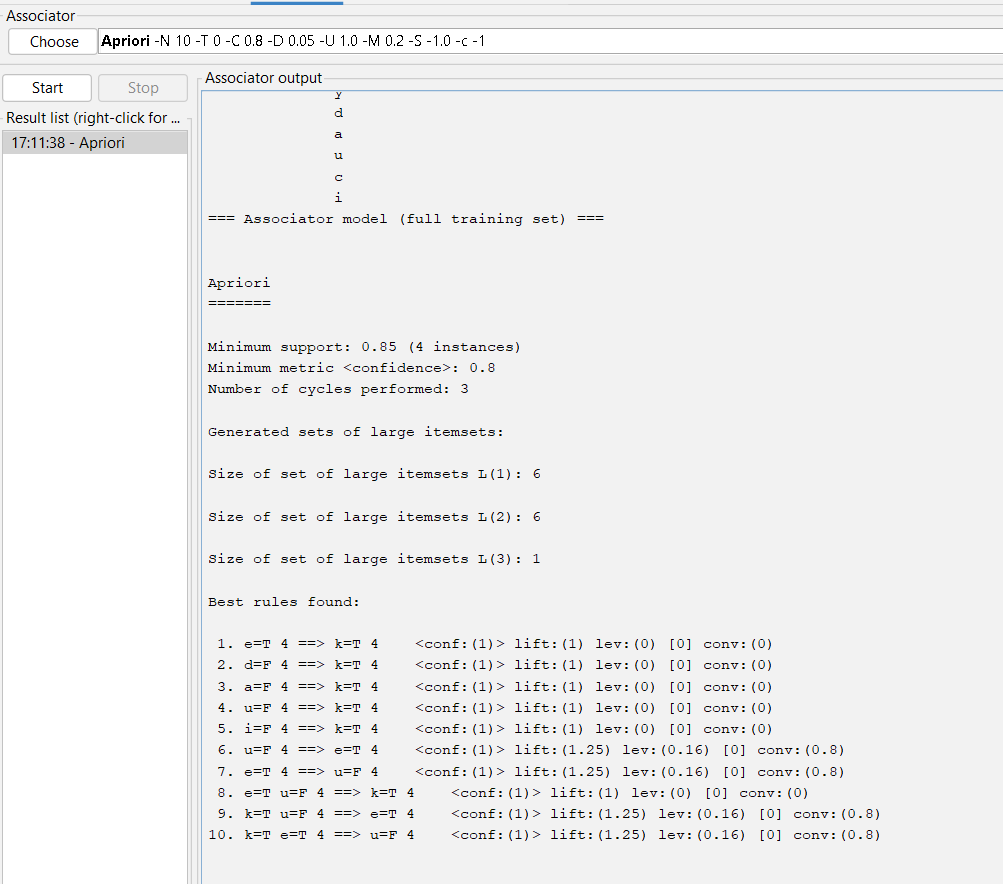
T2 (D, O, N, K, E, Y)

T3 (M, A, K, E)

T4 (M, U, C, K, Y)

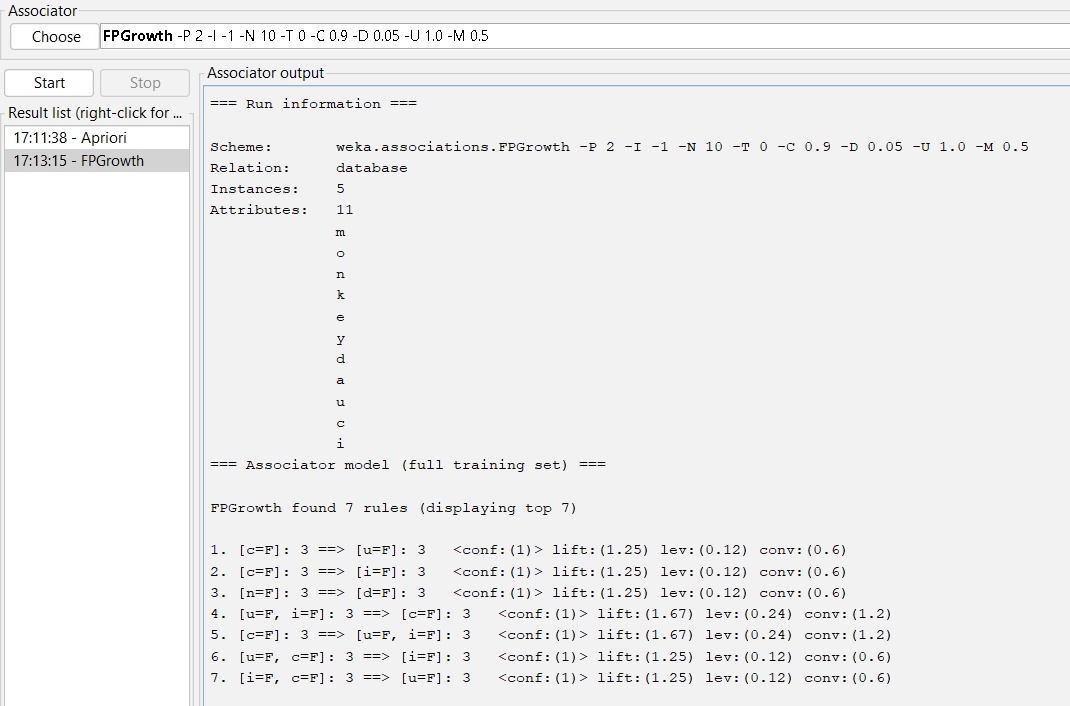
T5 (C,O, O, K, I ,E)

* Find all frequent item sets using Apriori algorithm
* Also draw FP-Growth Tree

Input:

Apriori algorithm:

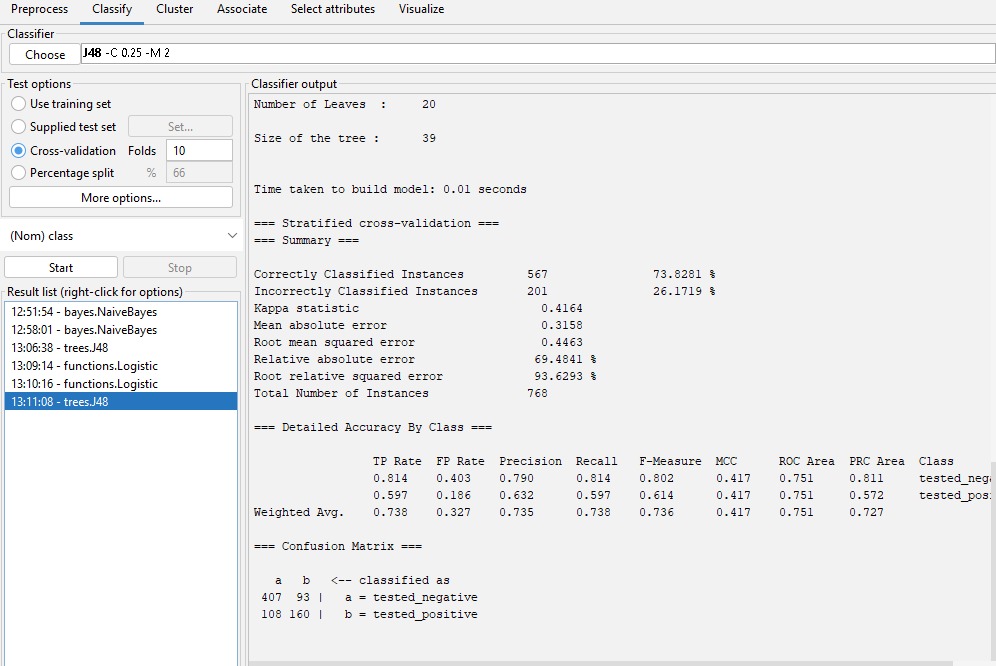
Fpgrowth algorithm:



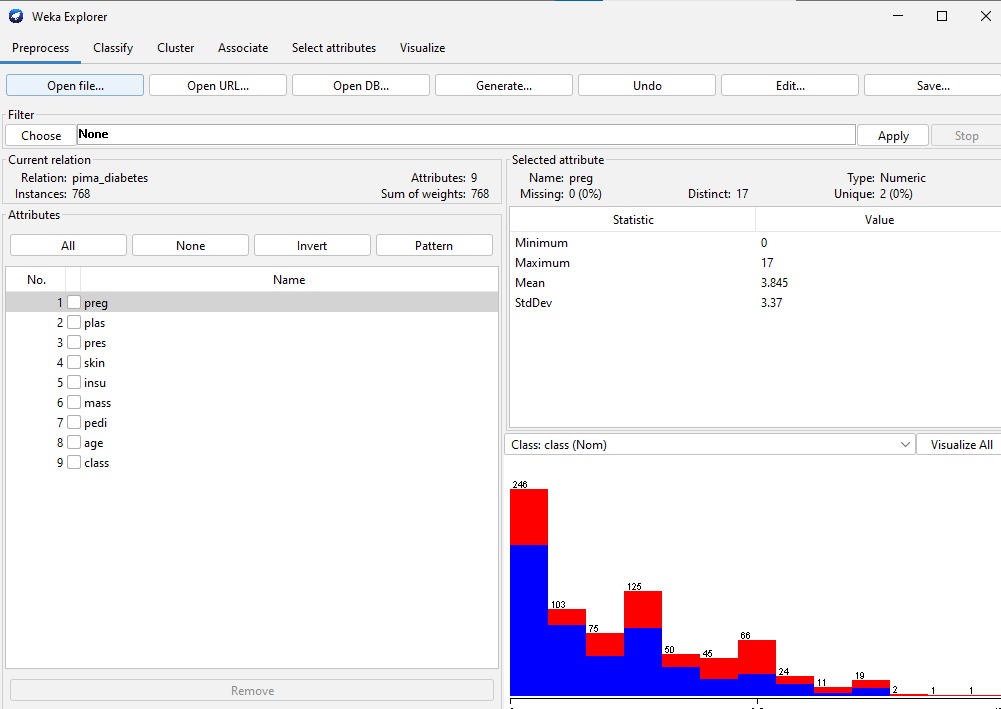
6. Prediction of Categorical Data using Decision Tree Algorithm through WEKA using any datasets. a) Tree b) Preprocess c) Logistic

Output:

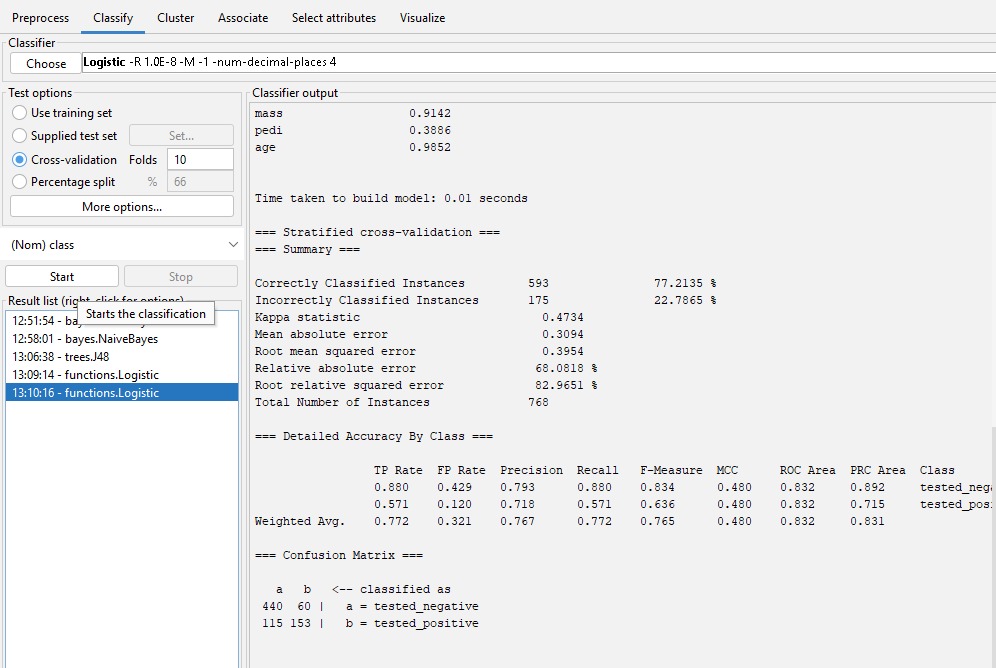
Tree:

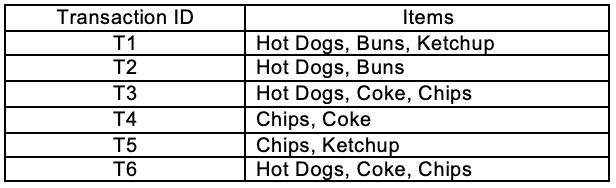


Preprocessor:



Logistic:



7.Create the dataset using ARFF file format:

a.Find the **frequent itemsets** and generate **association rules** on this. Assume that minimum support threshold (s = 33.33%) and minimum confident threshold (c = 60%).

b.List the various rule generated by apriori and FP tree algorthim ,mention wheather accepted or rejcted.

Input:

@relation hotdogs

@attribute hotdogs{t,f}

@attribute buns{t,f}

@attribute ketchup{t,f}

@attribute coke{t,f}

@attribute chips{t,f}

@data

t t t f f

t t f f f

t f f t t

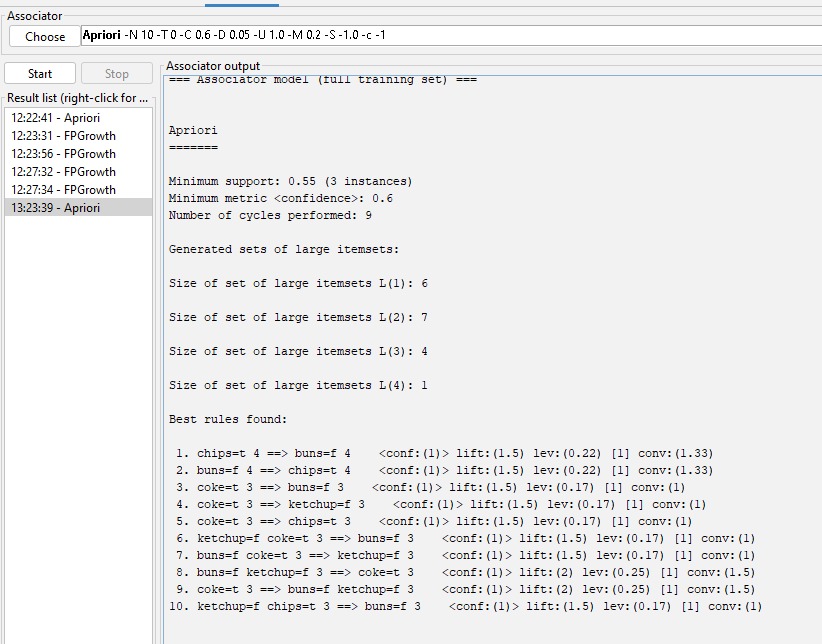
f f f t t

f f t f t

t f f t t

output:

apriori algorithm:



Fp growth:

