DATA FRAMES, CO VARIANCE

```
emp.date<-data.frame
age1=c("5-6","7-8","9-10")
a=c(12,34,45)
b=c(56,67,78)
c=c(89,90,12)
photo1=data.frame(age1,a,b,c)
photo1
s1=cov(a,b)
s1
photo1=data.frame(a,b,c)
photo1
s2=cov(photo1)
s2
```

HISTOGRAM GRAPH

```
c1=c(1,1,5,5,5,5,5,8,8,10,10,10,10,12,14,14,14,15,15,15,15,18,18,18,18,18,0,20,20,20,20,20,21,21,2
1,21,25,25,25,25,25,28,28,30)
hist(c1)
c2=c(1,1,5,5,5,5,5,8,8,10,10,10,10,12,14)
c3=c(14,14,15,15,15,15,18,18,18,18,18,0,20,20,20)
c4=c(20,20,21,21,21,21,25,25,25,25,25,28,28,30)
s1=mean(c2)
s1
s2=mean(c3)
s2
s3=mean(c4)
s3
```

BOX PLOT

```
c1=c(76,35,47,64,95,66,89,36,84,76,35,47,64,95,66,89,36,84)

c2=c(51,56,84,60,59,70,63,66,50,51,56,84,60,59,70,63,66,50)

s1=mean(c1)

s1

s2=mean(c2)

s2

s3=median(c1)

s3

s4=median(c2)

s4

s5=range(c1)

s6=range(c2)

s6

boxplot(c1~c2,xlab="x values",ylab="y values",main="sample")

head(ToothGrowth)

boxplot(c1~c2,xlab="class c1",ylab="class b",main="class 9 maths performance")
```

mean, median and standard deviation

```
age=c(23,23,27,27,39,41,47,49,50,52,54,54,56,57,58,58,60,61)

fact=c(9.5,26.5,7.8,17.8,31.4,25.9,27.4,27.2,31.2,34.6,42.5,28.8,33.4,30.2,34.1,32.9,41.2,35.7)

s1=mean(age)

s1

s2=mean(fact)

s2

s3=median(age)

s3

s4=median(fact)

s4

s5=sd(age)

s5

s6=sd(fact)
```

s6

boxplot(age~fact,xlab="fact values",ylab="age values",main="sample")

MAX MIN NORMALISE

age=c(23,23,27,27,39,41,47,49,50,52,54,54,56,57,58,58,60,61)

fact=c(9.5,26.5,7.8,17.8,31.4,25.9,27.4,27.2,31.2,34.6,42.5,28.8,33.4,30.2,34.1,32.9,41.2,35.7)

min_age<-min(age)

max_age<-max(age)

norm_age_minmax<-(39-min_age)/(max_age-min_age)

norm_age_minmax

mean_age=mean(age)

mean_age

sd_age=sd(age)

sd_age

norm_age_zscore<-(39-mean_age)/sd_age

norm_age_zscore

MINMAX, ZSCORE, DECIMAL

c1=c(200,300,400,600,1000)

max(c1,nm.rm=TRUE)

min(c1,nm.rm=TRUE)

z_score_norm<-function(x){(x-mean(x))/sd(x)}

norm_c1<-z_score_norm(c1)

cat("Normalised c1:",norm_c1,"\n")

max_abs_value<-max(abs(c1))

scale_factor<-10^(ceiling(log10(max_abs_value))+1)

scaled_c1<-c1/scale_factor

cat("original c1:",c1,"\n")

cat("scaled numbers:",scaled_c1,"\n")

box scatter, plot

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

class_b <- c(51,56,84,60,59,70,63,66,50) boxplot(class_a, class_b, main = "Boxplot of Exam Scores", names = c("class A", "class B"), ylab = "Score")

plot(class_a, class_b, main = "Scatter plot of Exam Scores", xlab = "class A scores", ylab = "class B Scores")

BOX, SCATTER, QQ PLOT

age=c(23,23,27,27,39,41,47,49,50,52,54,54,56,57,58,58,60,61)

fact=c(9.5,26.5,7.8,17.8,31.4,25.9,27.4,27.2,31.2,34.6,42.5,28.8,33.4,30.2,34.1,32.9,41.2,35.7)

boxplot(age,fact,names=c("AGE","FACT"),col="red",main="AGE and FACT data")

plot(age, fact, main="AGE and FACT data", xlab="AGE", ylab="FACT", col="green")

qqnorm(age)

qqline(age,col="red")

qqnorm(fact)

qqline(fact,col="red")

ARFF FOR GIVEN DATA

@relation supermarket

@attribute hotdogs{yes,no}

@attribute buns{yes,no}

@attribute ketchup{yes,no}

@attribute coke{yes,no}

@attribute chips{yes,no}

@data

yes,yes,yes,no,no

yes,yes,no,no,no

yes,no,no,yes,yes

no,no,no,yes,yes

no,no,yes,no,,yes

yes,no,no,yes,yes

CREATE A ARFF FOR GIVEN DATA

@relation breakfast

@attribute bread{yes,no}

@attribute peanuts{yes,no}

@attribute milk{yes,no}

@attribute fruit{yes,no}

@attribute jam{yes,no}

@attribute soda{yes,no}

@attribute chips{yes,no}

@attribute steak{yes,no}

@attribute yogurt{yes,no}

@attribute cheese{yes,no}

@data

yes,yes,yes,yes,yes,no,no,no,no,no

yes,yes,no,yes,yes,yes,yes,yes,yes,no

yes,no,no,no,yes,yes,yes,yes,no,no

no,yes,yes,yes,yes,yes,no,no,no,no

yes,no,yes,no,yes,yes,yes,no,no,no

no,no,yes,yes,no,yes,yes,no,no,no

no,yes,no,yes,no,no,no,no,yes,yes

ARFF

@relation playtennis

@attribute outlook{sunny,overcast,rain}

@attribute temperature{hot,mild,cold}

@attribute humidity{high,normal}

@attribute wind{strong,weak}

@data

sunny,hot,high,weak,no

sunny,hot,high,strong,no

overcast,hot,high,weak,yes

rain,mild,high,weak,yes

rain,cold,normal,weak,yes

rain,cold,normal,strong,no

overcast,cold,normal,strong,yes

sunny,mild,high,weak,no

sunny,cold,normal,weak,yes

rain,mild,normal,weak,yes

sunny,mild,normal,strong,yes

overcast,mild,high,strong,yes

overcast,hot,normal,weak,yes

rain,mild,high,strong,no

CLUSTER

@relation employee

@attribute employeid numeric

@attribute gender{male,female}

@attribute age numeric

@attribute salary numeric

@attribute credit numeric

@data

1111,male,28,150000,39

2222,male,25,150000,27

3333,female,26,160000,42

4444,female,25,160000,40

5555,female,30,170000,64

6666,male,29,200000,72

Incorrect:-

@relation employee

@attribute employeid numeric

@attribute gender{male,female}

@attribute age numeric

@attribute salary numeric

@attribute credit numeric

@data

1111,female,28,150000,39

2222,male,25,150000,67

3333,female,26,160000,42

4444,female,25,160000,40

5555,male,30,170000,64

6666,male,29,200000,72

DECISION TREE

@relation dataset

@attribute height numeric

@attribute weight numeric

@attribute gender{male,female}

@data

180,60,male

120,81,male

125,55,female

Incorrect:-

@relation dataset

@attribute height numeric

@attribute weight numeric

@attribute gender{male,female}

@data

180,60,female

120,81,male

125,55,male

FP GROWTH

@relation t_id

@attribute sony{yes,no}

@attribute bpl{yes,no}

@attribute lg{yes,no}

@attribute samsung{yes,no}

@attribute onida{yes,no}

@data

yes,yes,yes,no,no

no,yes,no,yes,no

no,yes,no,no,yes

yes,yes,no,yes,no

yes,no,no,no,yes

no,yes,no,no,yes

yes,no,no,no,yes

yes,yes,yes,no,yes

yes,yes,no,no,yes

MIN MAX SCORE NORMALISATION

F_min <- 50000

F_max <- 100000

v <- 80000

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

min_max_norm <- function(x){(x-F_min)/(F_max-F_min)}

data_min_max_norm <- min_max_norm(data)

z_score_norm <- function(x){(x-mean(data))/sd(data)}

data_z_score_norm <- z_score_norm(data)

cat("Min-Max normalised data:",data_min_max_norm,"\n")

DECISION TREE USING WEKA

@relation dataset

@attribute height numeric

@attribute weight numeric

@attribute gender{male,female}

@data

180,60,male

120,81,male

125,55,female

Incorrect:-

@relation dataset

@attribute height numeric

@attribute weight numeric

@attribute gender{male,female}

```
@data
180,60,female
120,81,male
125,55,male
```

SD AND VARIANCE

```
avgspeed=c(78,81,82,74,83,82,77)
ttime=c(39,37,36,42,35,36,40)
sd(avgspeed)
sd(ttime)
var(avgspeed)
var(ttime)
```

SCATTER

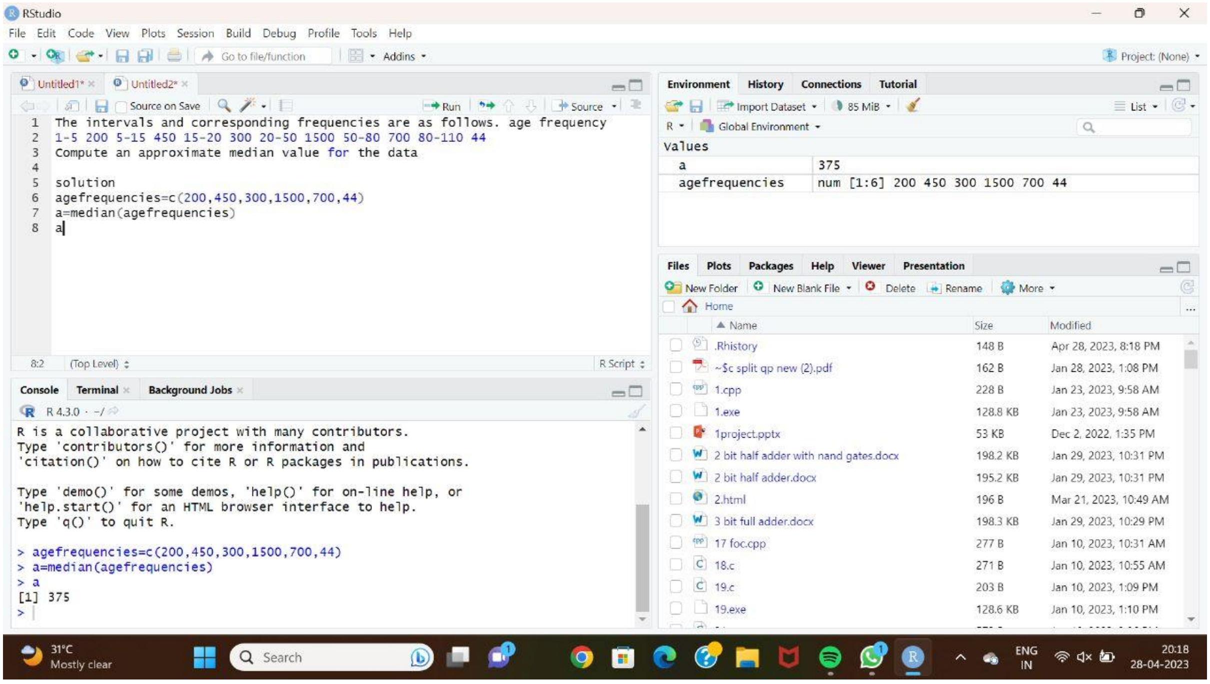
```
v1=read.csv("C:/Users/shail/Downloads/cancer.csv")
v2=scatter.smooth(v1$tumour.size)
v3=boxplot(v1)
v4=hist(v1$age)
```

APPLES AND STRAWBERRY

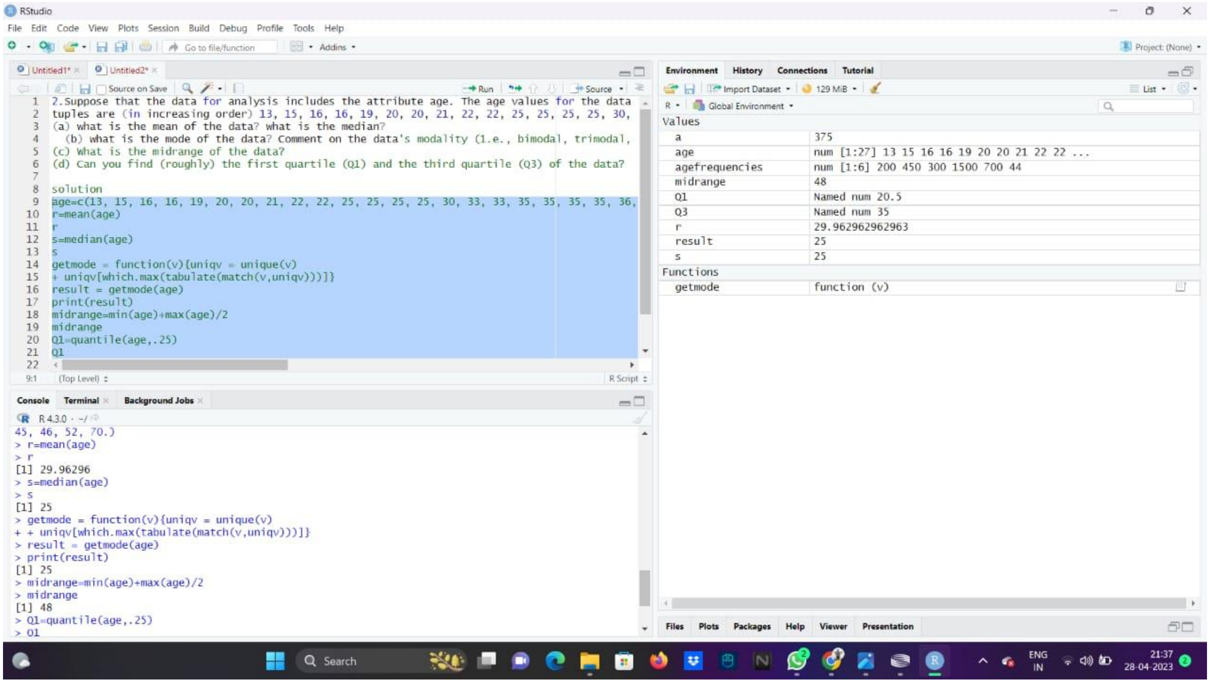
```
true_apples <- 9
true_strawberries <- 10
correct_apples <- 6
correct_strawberries <- 8
misclassified_apples <- 3
misclassified_strawberries <- 2
total_identified <- correct_apples + correct_strawberries + misclassified_apples +
misclassified_strawberries
accuracy_apples <- correct_apples / true_apples
accuracy_strawberries <- correct_strawberries / true_strawberries
precision_apples <- correct_apples / (correct_apples + misclassified_strawberries)
precision_strawberries <- correct_strawberries / (correct_strawberries + misclassified_apples)
recall_apples <- correct_apples / true_apples
recall_strawberries <- correct_strawberries / true_strawberries
```

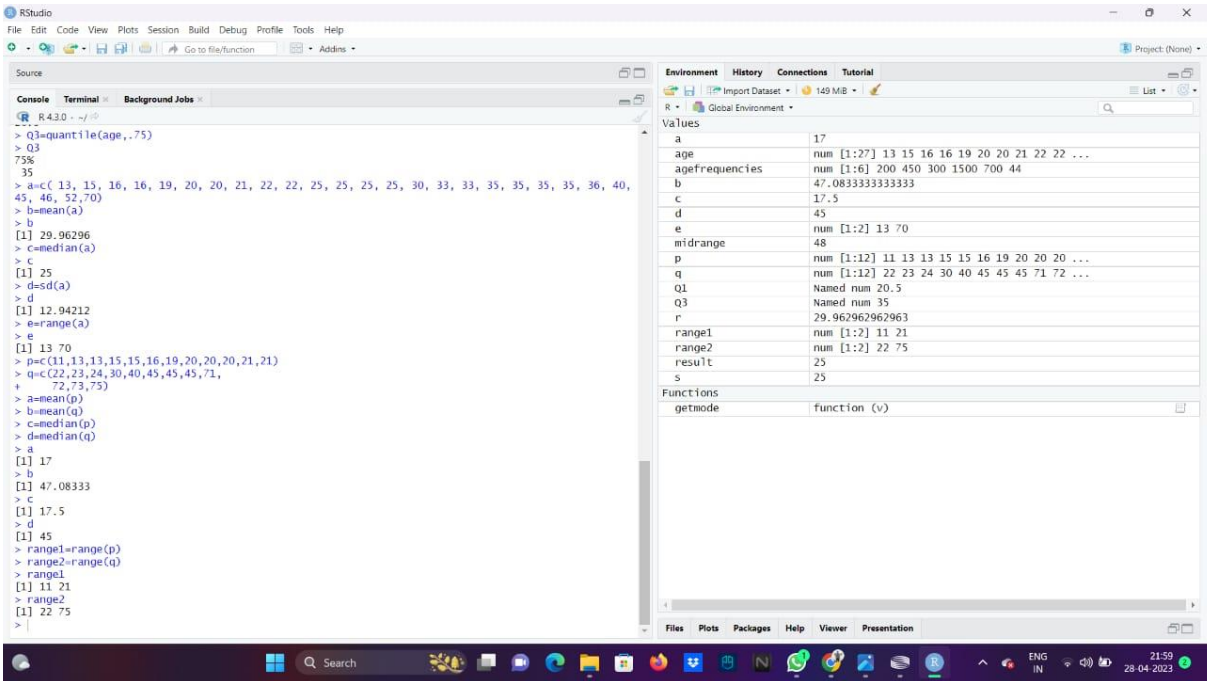
```
cat("total identified fruits:", total_identified, "\n")
cat("accuracy for apples:", round(accuracy_apples, 2), "\n")
cat("accuracy for strawberries:", round(accuracy_strawberries, 2), "\n")
cat("precision for apples:", round(precision_apples, 2), "\n")
cat("precision for strawberries:", round(precision_strawberries, 2), "\n")
cat("recall for apples:", round(recall_apples, 2), "\n")
cat("recall for strawberries:", round(recall_strawberries, 2), "\n")
```

INTERVALS

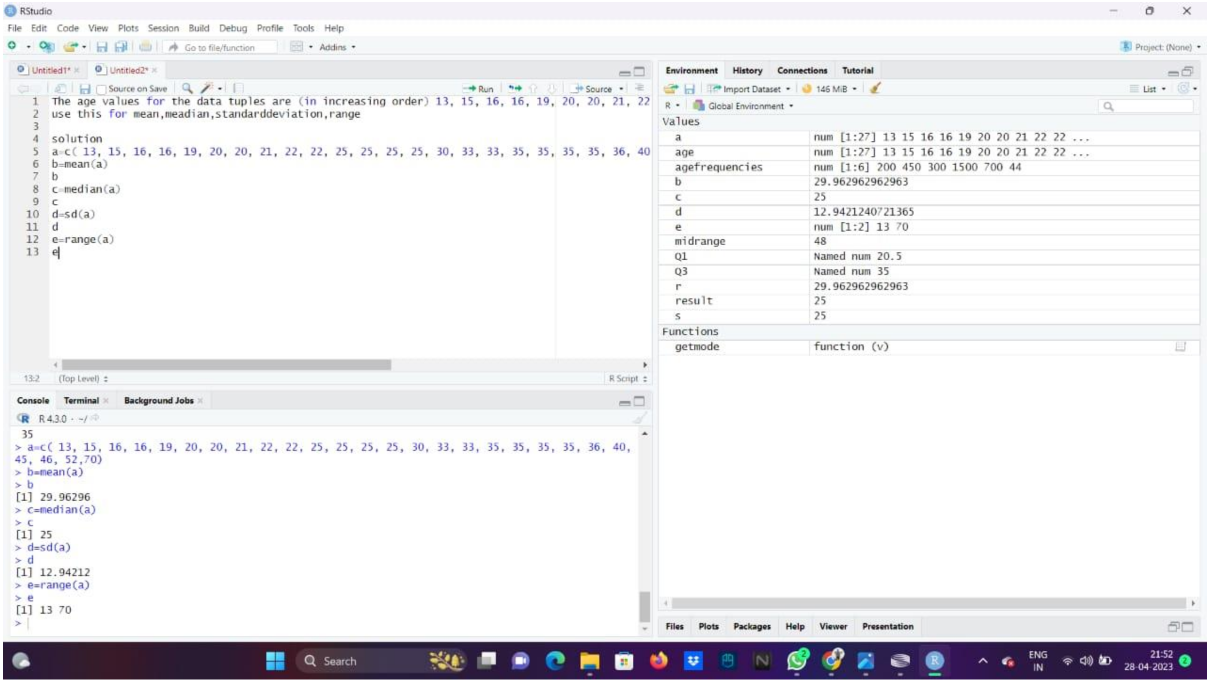


AGE





MEAN, MEDIAN AND RANGE



MEAN, MEDIA,MODE, MID RANGE

2.Suppose that the data for analysis includes the attribute age. The age values for the data tuples are (in increasing order) 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. tuples are (in increasing order) 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.

(a) what is the mean of the data? what is the median?

(b) what is the mode of the data? Comment on the data's modality (1.e., bimodal, trimodal, etc.).

(c) What is the midrange of the data?

(d) Can you find (roughly) the first quartile (Q1) and the third quartile (Q3) of the data?

solution

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.)

r=mean(age)

s=median(age)

getmode = function(v){uniqv = unique(v)+ uniqv[which.max(tabulate(match(v,uniqv)))]}

result = getmode(age)

print(result)

midrange=min(age)+max(age)/2

midrange

Q1=quantile(age,.25)

Q1

Q3=quantile(age,.75)

Q3

MEDIAN

1.The intervals and corresponding frequencies are as follows. age frequency

1-5 200 5-15 450 15-20 300 20-50 1500 50-80 700 80-110 44

Compute an approximate median value for the data

solution

agefrequencies=c(200,450,300,1500,700,44)

a=median(agefrequencies)

SMOOTHING BIN

Data:11,13,13,15,15,16,19,20,20,20,21,21,22,23,24,30,40,45,45,45,71,72,73,75

a) Smoothing by bin mean

b) Smoothing by bin median

c) Smoothing by bin boundaries

p=c(11,13,13,15,15,16,19,20,20,20,21,21)

q=c(22,23,24,30,40,45,45,45,71, 72,73,75)

a=mean(p)

b=mean(q)

c=median(p)

d=median(q)

a

b

c

d

range1=range(p)

range2=range(q)

range1

range2

STANDARD DEVIATION

The age values for the data tuples are (in increasing order) 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 use this for mean, median, standard deviation, range solution

```
a=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)

b=mean(a)

c=median(a)

d=sd(a)

e=range(a)
```