HW1 IDS 572

Jitesh Patil, Pratik Sharma, Vibhav Mayekar

2/8/2022

```
#Question1
x <- c(1,2.3,2,3,4,8,12,43,-4,-1)
x
## [1]  1.0  2.3  2.0  3.0  4.0  8.0 12.0 43.0 -4.0 -1.0
#a) Creates a vector of integers and assigns the vector to x.

max(x)
## [1] 43
#b) Finds the max value from the vector x. The answer is 43.

y <- c(x, NA)
y
##  [1]  1.0  2.3  2.0  3.0  4.0  8.0 12.0 43.0 -4.0 -1.0   NA
#c) This command creates a new vector with the same values as x and adds a NA
at the end. This vector is assigned to y. The NA represents a missing value.

max(y, na.rm = T)
## [1] 43
#d) It calculates the max of y while ignoring the NA value. The answer is 43.

x2 <- c(-100,-43,0,3,1,-3)
min(x,x2)
## [1] -100
#e) Creates a new vector named x2. Find the min value from both x and x2. The
answer is -100.

sample(4:10)
## [1]  8 10  5  9  6  7  4
#f) Creates a sample from the sequence of numbers between 4 and 10.

sample(c(2,5,3), size=3, replace=FALSE)
## [1] 3 2 5
#g) Creates a sample of the vector specified. The size of the sample created
is 3 and the numbers are not replaced.

sample(c(2,5,3), size=3, replace= TRUE)
## [1] 2 2 5
#h) Creates a sample of the vector specified. The size of the sample created
is 3 and the numbers are replaced.

sample(2, 10, replace = TRUE)
##  [1] 1 2 2 2 2 1 2 1 1 2
#i) Creates a sample of size 10 of numbers 1 and 2. If replace is set to
```

```
False it gives an error since a sample of a size larger than the sequence of
numbers can't be created with replace = False.

sample(1:2, size=10, prob=c(1,3), replace=TRUE)
##  [1] 2 2 2 2 2 1 2 2 1 1
#j) Creates a sample of numbers 1 and 2 of size 10 with replace = True. The
probability of the numbers appearing in the sample is uneven with 2 being
thrice as more probable as 1.

round(3.14159, digits = 2)
## [1] 3.14
#k) Rounds down the number to two digits after the decimal. The answer is
3.14.

range(100:400)
## [1] 100 400
#l) Calculates the range of the sequence. The lowest number and highest
number of the sequence.

matrix(c(1,2.3,2,3,4,8,12,43,-4,-1,9,14), nr=3, nc=4)
##      [,1] [,2] [,3] [,4]
## [1,]  1.0    3   12   -1
## [2,]  2.3    4   43    9
## [3,]  2.0    8   -4   14
#m) Creates a matrix of 3 rows and 4 columns of the integers from the vector.

matrix(c(1,2.3,2,3,4,8,12,43,-4,-1,9,14), nr=3, nc=4, byrow = T)
##      [,1] [,2] [,3] [,4]
## [1,]    1  2.3    2    3
## [2,]    4  8.0   12   43
## [3,]   -4 -1.0    9   14
#n) Creates a matrix of 3 rows and 4 columns of the integers from the vector.
Since byrow = True, the matrix is arranged in the order of the vector.

x <- matrix(c(4,3,4,6,7,6),3,2)
rownames(x) <- c("row1","row2","row3")
colnames(x) <- c("col1", "col2")
#o) Creates a matrix of 3 rows and 2 columns and names the rows and columns
using the rownames and colnames functions.

x <- rbind(c(1:4),c(5,8))
y <- cbind(c(1:4),c(5,8))
#p) Creates 2 matrices with the two vectors using rbind and cbind functions.
These functions combine the vectors to form a matrix.

y<-1:9
w<-2:10
z<-3:5
rbind(y,w,z)
```

```
##   [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9]
## y    1    2    3    4    5    6    7    8    9
## w    2    3    4    5    6    7    8    9   10
## z    3    4    5    3    4    5    3    4    5
#q) Creates three vectors and then uses the rbind function to combine them
 and form a matrix.

m<-matrix(1:36,9,4)
m[2,3]
## [1] 20
m[,3]
## [1] 19 20 21 22 23 24 25 26 27
m[2,]
## [1]  2 11 20 29
cbind(m[,3])
##       [,1]
##  [1,]   19
##  [2,]   20
##  [3,]   21
##  [4,]   22
##  [5,]   23
##  [6,]   24
##  [7,]   25
##  [8,]   26
##  [9,]   27
m[,-3]
##       [,1] [,2] [,3]
##  [1,]    1   10   28
##  [2,]    2   11   29
##  [3,]    3   12   30
##  [4,]    4   13   31
##  [5,]    5   14   32
##  [6,]    6   15   33
##  [7,]    7   16   34
##  [8,]    8   17   35
##  [9,]    9   18   36
m[-(3:8),2:4]
##      [,1] [,2] [,3]
## [1,]   10   19   28
## [2,]   11   20   29
## [3,]   18   27   36
#r)Creates a matrix of numbers in the sequence 1 to 36. The matrix has 9
rows and 4 columns. The answer for m[2,3] is 20. M[,3] displays the third
column of the matrix. M[2,] displays the second row of the matrix.
Cbind(m[,3]) assigns the 3rd column of the m matrix to another matrix with a
single ‘column. M[,-3] displays the matrix after removing the 3rd column.
M[-(3:8),2:4] removes row 3 to 8 and column 1 from the matrix and displays
it.
```

```
x<-cbind(x1=3,x2=c(4:1,2:5))
dimnames(x)[[1]]<-letters[1:8]
apply(x,2,mean,trim=.2)
## x1 x2
##  3  3
col.sums<-apply(x,2,sum)
row.sums<-apply(x,1,sum)
apply(x,2,sort)
##      x1 x2
## [1,]  3  1
## [2,]  3  2
## [3,]  3  2
## [4,]  3  3
## [5,]  3  3
## [6,]  3  4
## [7,]  3  4
## [8,]  3  5
#s) Creates a matrix using cbind of two vectors. Then assigns names to the
column using dimnames starting with a through h. Uses the apply function to
find the mean of each column. Uses the apply function to find the sums of
each row and column and assigns them to row.sums and column.sums respectively
.The final apply is used to sort the dataframe's columns.
```

```
#Question2
```

```
library(ggplot2)
library(tidyverse)

## -- Attaching packages --------------------------------------- tidyverse 1.
3.1 --

## v tibble  3.1.6     v dplyr   1.0.7
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.1.0     v forcats 0.5.1
## v purrr   0.3.4

## -- Conflicts ------------------------------------------ tidyverse_conflict
s() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
##Imports

library(dplyr)
#Q2(a) x*y=z
x <- 15
y <- c (1 , 2 , 3 , 10 , 100)
z <- x*y
```

```r
total <- sum(z)
total
```

```
## [1] 1740
```

```r
#Q2(b) Generate sequence 0 to 10 and a sequence from 5 to -5

seq1 <- seq(from=1, to=10)
seq2 <- seq(from=5, to=-5)

seq1
```

```
##  [1]  1  2  3  4  5  6  7  8  9 10
```

```r
seq2
```

```
##  [1]  5  4  3  2  1  0 -1 -2 -3 -4 -5
```

```r
#Q2(c) Sequence from -3 to 3 by 0.1 steps
seq3 <- seq(from=-3 , to=3 ,by=0.1)
seq3
```

```
##  [1] -3.0 -2.9 -2.8 -2.7 -2.6 -2.5 -2.4 -2.3 -2.2 -2.1 -2.0 -1.9 -1.8 -1.7
-1.6
## [16] -1.5 -1.4 -1.3 -1.2 -1.1 -1.0 -0.9 -0.8 -0.7 -0.6 -0.5 -0.4 -0.3 -0.2
-0.1
## [31]  0.0  0.1  0.2  0.3  0.4  0.5  0.6  0.7  0.8  0.9  1.0  1.1  1.2  1.3
1.4
## [46]  1.5  1.6  1.7  1.8  1.9  2.0  2.1  2.2  2.3  2.4  2.5  2.6  2.7  2.8
2.9
## [61]  3.0
```

```r
#Q2(d) Sequence from -3 to 3 by 0.1 steps

t <- c("mon", "tue", "wed", "thu","fri", "sat" )
m <- c( 90, 80, 50, 20, 5, 20 )
study <- matrix(c(t,m),nrow = 6,ncol=2, byrow = F)
study
```

```
##      [,1]  [,2]
## [1,] "mon" "90"
## [2,] "tue" "80"
## [3,] "wed" "50"
## [4,] "thu" "20"
## [5,] "fri" "5"
## [6,] "sat" "20"
```

```r
#Q2(e) dataframe

age <- c(21, 35 , 829 , 2)
sex <- c("m", "f" , "m", "e")
```

```
height <- c(181 , 173 , 171 , 166)
weight <- c(69, 58 , 75 , 60)
df1 <- data.frame(age,sex,height,weight)
df1
```

```
##   age sex height weight
## 1  21   m    181     69
## 2  35   f    173     58
## 3 829   m    171     75
## 4   2   e    166     60
```

```
#min and max age
min_age <- min(age)
max_age <- max(age)
min_age
```

```
## [1] 2
```

```
max_age
```

```
## [1] 829
```

```
#age less than 20 and more than 80
```

```
df1 <- df1 %>% mutate( age= ifelse(df1$age<20 | df1$age >80 , "NA",df1$age))%
>% mutate(BMI = round(weight*100/height))
df1
```

```
##   age sex height weight BMI
## 1  21   m    181     69  38
## 2  35   f    173     58  34
## 3  NA   m    171     75  44
## 4  NA   e    166     60  36
```

#Question3

```
(x <- c(9, 8, 12, 6, 1, 10, 10, 10, 8, 516, 8, 6, 4, 19, 100))
```

```
##  [1]   9   8  12   6   1  10  10  10   8 516   8   6   4  19 100
```

```
# 3 (a) compute mean of x
mean(x)
```

```
## [1] 48.46667
```

```
## [1] Ques 3b SD of x
```

```
sd(x)
```

```
## [1] 131.5261
```

```
#Ques 3c Range of x
```

```
range(x)
```

```
## [1]   1 516
```

```
#Ques 3d five number summary of x
```

```
fivenum(x)
```

```
## [1]   1   7   9  11 516
```

```
#Ques 3e NA in x
```

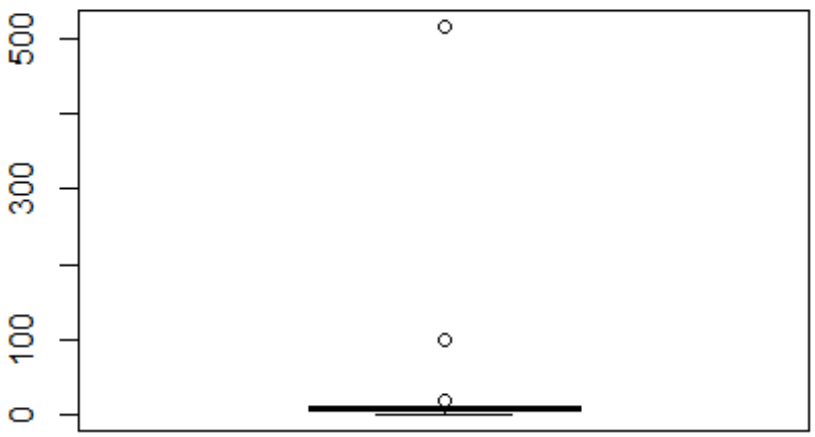
```
is.na(x)
```

```
##  [1] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
FALSE
```

```
#There is no NA in x


#[1] "Ques 3f outliers"
#Plotting the Boxplot for Vector x
(x <- c(9, 8, 12, 6, 1, 10, 10, 10, 8, 516, 8, 6, 4, 19, 100))
```

```
##  [1]   9   8  12   6   1  10  10  10   8 516   8   6   4  19 100
```

```
boxplot(x)


#display the outliers


boxplot(x)$out
```




```
## [1] 516  19 100
```

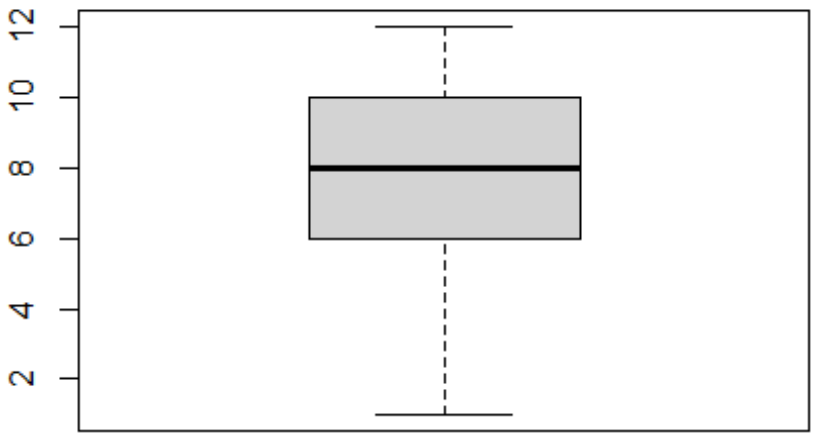
```
#There are 3 outliers in the data set 19,100 and 516
```

```
## [1] "There are 3 outliers in the data set 19,100 and 516"
```

```
#assign the outliers to another vector

outliers <-boxplot(x)$out
## display the vector containing outliers

outliers
```

```
## [1] 516  19 100
```

```
## Remove the outliers from the vector
x <- x[-which(x %in% outliers)]

## Plot again to verify
boxplot(x)$out
```




```
## numeric(0)
```

```
#Question4
```

```
#loading dataset
library(readr)

data1 <- read.csv("C:\\Users\\psharm50\\Desktop\\arbuthnot.csv")
data1
```

```
##     X year boys girls
## 1   1 1629 5218  4683
## 2   2 1630 4858  4457
## 3   3 1631 4422  4102
## 4   4 1632 4994  4590
## 5   5 1633 5158  4839
## 6   6 1634 5035  4820
## 7   7 1635 5106  4928
## 8   8 1636 4917  4605
## 9   9 1637 4703  4457
## 10 10 1638 5359  4952
## 11 11 1639 5366  4784
## 12 12 1640 5518  5332
## 13 13 1641 5470  5200
## 14 14 1642 5460  4910
```

```
## 15 15 1643 4793  4617
## 16 16 1644 4107  3997
## 17 17 1645 4047  3919
## 18 18 1646 3768  3395
## 19 19 1647 3796  3536
## 20 20 1648 3363  3181
## 21 21 1649 3079  2746
## 22 22 1650 2890  2722
## 23 23 1651 3231  2840
## 24 24 1652 3220  2908
## 25 25 1653 3196  2959
## 26 26 1654 3441  3179
## 27 27 1655 3655  3349
## 28 28 1656 3668  3382
## 29 29 1657 3396  3289
## 30 30 1658 3157  3013
## 31 31 1659 3209  2781
## 32 32 1660 3724  3247
## 33 33 1661 4748  4107
## 34 34 1662 5216  4803
## 35 35 1663 5411  4881
## 36 36 1664 6041  5681
## 37 37 1665 5114  4858
## 38 38 1666 4678  4319
## 39 39 1667 5616  5322
## 40 40 1668 6073  5560
## 41 41 1669 6506  5829
## 42 42 1670 6278  5719
## 43 43 1671 6449  6061
## 44 44 1672 6443  6120
## 45 45 1673 6073  5822
## 46 46 1674 6113  5738
## 47 47 1675 6058  5717
## 48 48 1676 6552  5847
## 49 49 1677 6423  6203
## 50 50 1678 6568  6033
## 51 51 1679 6247  6041
## 52 52 1680 6548  6299
## 53 53 1681 6822  6533
## 54 54 1682 6909  6744
## 55 55 1683 7577  7158
## 56 56 1684 7575  7127
## 57 57 1685 7484  7246
## 58 58 1686 7575  7119
## 59 59 1687 7737  7214
## 60 60 1688 7487  7101
## 61 61 1689 7604  7167
## 62 62 1690 7909  7302
## 63 63 1691 7662  7392
## 64 64 1692 7602  7316
```

```
## 65 65 1693 7676  7483
## 66 66 1694 6985  6647
## 67 67 1695 7263  6713
## 68 68 1696 7632  7229
## 69 69 1697 8062  7767
## 70 70 1698 8426  7626
## 71 71 1699 7911  7452
## 72 72 1700 7578  7061
## 73 73 1701 8102  7514
## 74 74 1702 8031  7656
## 75 75 1703 7765  7683
## 76 76 1704 6113  5738
## 77 77 1705 8366  7779
## 78 78 1706 7952  7417
## 79 79 1707 8379  7687
## 80 80 1708 8239  7623
## 81 81 1709 7840  7380
## 82 82 1710 7640  7288
```

```
#Question 4(a)
```

```
dim(data1)
```

```
## [1] 82  4
```

```
#a)The dimensions of the dataset are 82 rows and 4 columns. We use the dim()
function to find it.
```

```
#Question 4(b)
```

```
colnames(data1)
```

```
## [1] "X"     "year"  "boys"  "girls"
```

```
#b)The variables in the dataset are the serial number of the year starting
with 1, the year the children were born and if they were boy or girl.
We can find this by using the colnames function to find the names of these
Variables
```

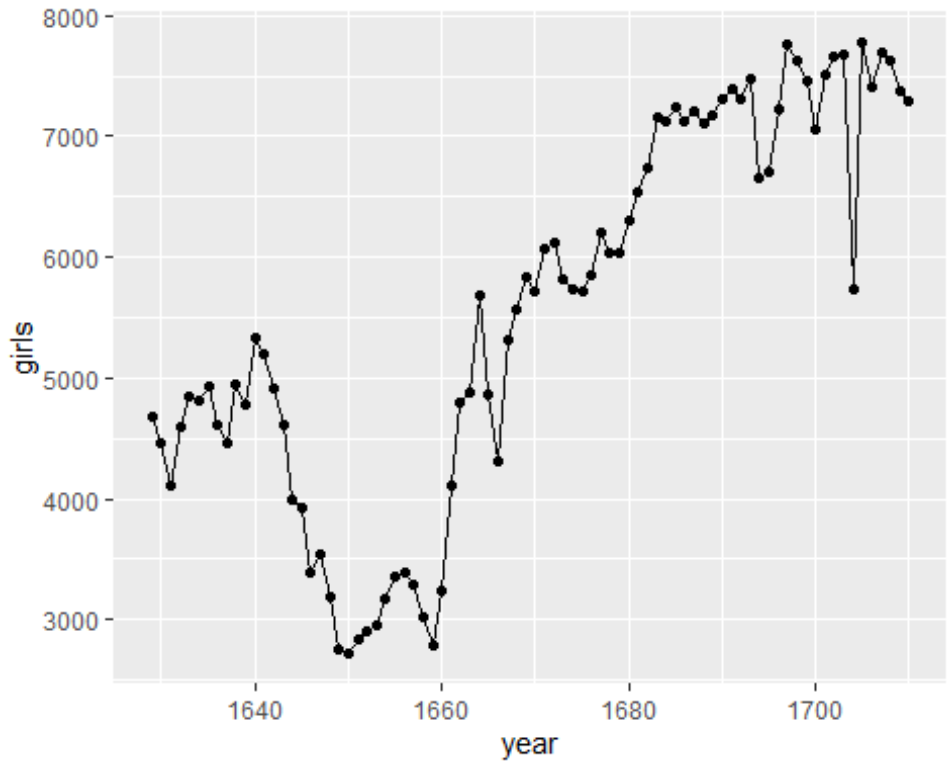
```
#Question 4(c)
```

```
sum(data1$girls)
```

```
## [1] 453841
```

```
#c) We use the sum(data1$girls) function to find the sum of all the girls
baptized
```

```
#Question 4(b=d)
#loading graph libraries
library(ggplot2)
library(stringr)
library(ggpubr)


#plotting graph
ggplot(data=data1, aes(x=year, y=girls, group=1)) +
  geom_line()+
  geom_point()
```

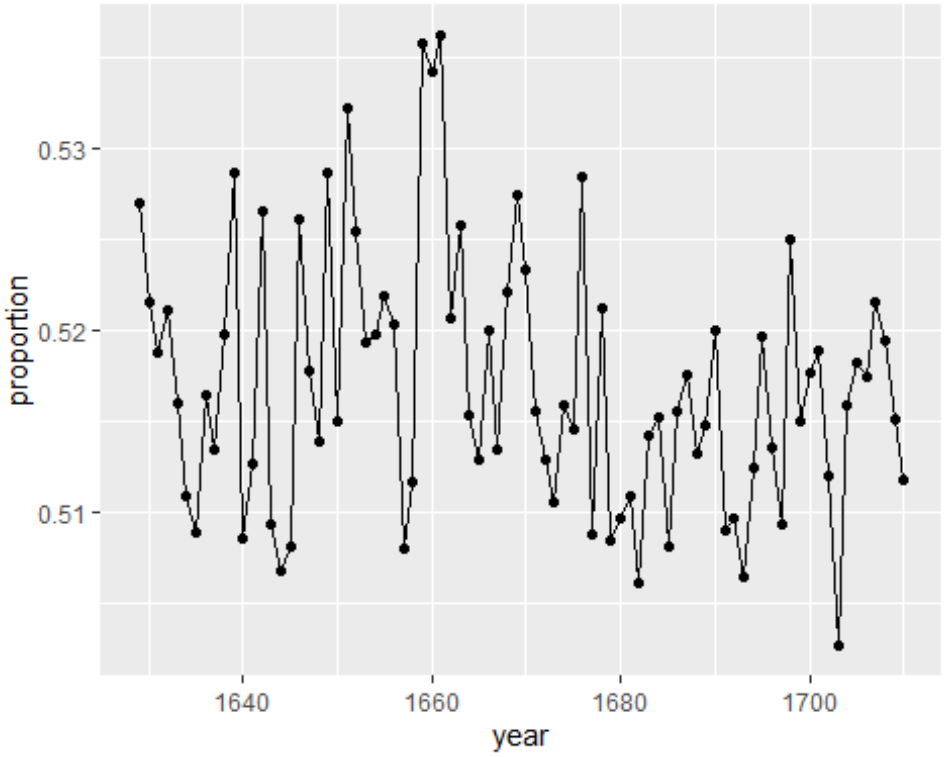



```
#d) There is a general rise in the number of girls baptized over the years
except between 1640-1660 where there was a steep decline. We find this out by
plotting a line graph of girls baptized over the years. There is also a big
dip in the year 1704


#Question 4(e)
totalbirths<- data.frame(data1$boys + data1$girls)
propb <- data.frame(data1$year, prop=data1$boys / totalbirths)
colnames(propb)<- c('year','proportion')
propb

##    year proportion
## 1  1629  0.5270175
```

```
## 2  1630  0.5215244
## 3  1631  0.5187705
## 4  1632  0.5210768
## 5  1633  0.5159548
## 6  1634  0.5109082
## 7  1635  0.5088698
## 8  1636  0.5163831
## 9  1637  0.5134279
## 10 1638  0.5197362
## 11 1639  0.5286700
## 12 1640  0.5085714
## 13 1641  0.5126523
## 14 1642  0.5265188
## 15 1643  0.5093518
## 16 1644  0.5067868
## 17 1645  0.5080341
## 18 1646  0.5260366
## 19 1647  0.5177305
## 20 1648  0.5139059
## 21 1649  0.5285837
## 22 1650  0.5149679
## 23 1651  0.5322023
## 24 1652  0.5254569
## 25 1653  0.5192526
## 26 1654  0.5197885
## 27 1655  0.5218447
## 28 1656  0.5202837
## 29 1657  0.5080030
## 30 1658  0.5116694
## 31 1659  0.5357262
## 32 1660  0.5342132
## 33 1661  0.5361942
## 34 1662  0.5206108
## 35 1663  0.5257482
## 36 1664  0.5153557
## 37 1665  0.5128359
## 38 1666  0.5199511
## 39 1667  0.5134394
## 40 1668  0.5220493
## 41 1669  0.5274422
## 42 1670  0.5232975
## 43 1671  0.5155076
## 44 1672  0.5128552
## 45 1673  0.5105507
## 46 1674  0.5158214
## 47 1675  0.5144798
## 48 1676  0.5284297
## 49 1677  0.5087122
## 50 1678  0.5212285
## 51 1679  0.5083822
```

```
## 52 1680  0.5096910
## 53 1681  0.5108199
## 54 1682  0.5060426
## 55 1683  0.5142178
## 56 1684  0.5152360
## 57 1685  0.5080788
## 58 1686  0.5155165
## 59 1687  0.5174905
## 60 1688  0.5132301
## 61 1689  0.5147925
## 62 1690  0.5199527
## 63 1691  0.5089677
## 64 1692  0.5095857
## 65 1693  0.5063659
## 66 1694  0.5123973
## 67 1695  0.5196766
## 68 1696  0.5135590
## 69 1697  0.5093183
## 70 1698  0.5249190
## 71 1699  0.5149385
## 72 1700  0.5176583
## 73 1701  0.5188268
## 74 1702  0.5119526
## 75 1703  0.5026541
## 76 1704  0.5158214
## 77 1705  0.5181790
## 78 1706  0.5174052
## 79 1707  0.5215362
## 80 1708  0.5194175
## 81 1709  0.5151117
## 82 1710  0.5117899
```

```
ggplot(data=propb, aes(x=year, y=proportion, group=1)) +
  geom_line()+
  geom_point()
```




```
#e)For the proportion of boys over the years, we see that the ratio is in the
0.5 to 0.53 range over the years with just a few exceptions

#Question 4(f)
maxdata <- data.frame(Total=rowSums(data1[,c(3,4)]))
row.names(maxdata) <- data1[,c("year")]
rownames(maxdata)[which.max(maxdata$Total)]

## [1] "1705"

#f)    We saw the greatest number of births in the year 1705. We find this
out by creating a dataframe where the column is the sum of births for girls
and boys for each corresponding year. The row names are the particular years,
and we fetch the rowname of the year with the max Total births
```

```
#Question5

##Imports
#if(!require("datasets"))install.packages("datasets")
library(dplyr)
library(datasets)
library(tidyverse)
data("attitude")
attitude

##    rating complaints privileges learning raises critical advance
## 1      43         51         30       39     61       92      45
```

```
## 2      63         64         51       54     63       73       47
## 3      71         70         68       69     76       86       48
## 4      61         63         45       47     54       84       35
## 5      81         78         56       66     71       83       47
## 6      43         55         49       44     54       49       34
## 7      58         67         42       56     66       68       35
## 8      71         75         50       55     70       66       41
## 9      72         82         72       67     71       83       31
## 10     67         61         45       47     62       80       41
## 11     64         53         53       58     58       67       34
## 12     67         60         47       39     59       74       41
## 13     69         62         57       42     55       63       25
## 14     68         83         83       45     59       77       35
## 15     77         77         54       72     79       77       46
## 16     81         90         50       72     60       54       36
## 17     74         85         64       69     79       79       63
## 18     65         60         65       75     55       80       60
## 19     65         70         46       57     75       85       46
## 20     50         58         68       54     64       78       52
## 21     50         40         33       34     43       64       33
## 22     64         61         52       62     66       80       41
## 23     53         66         52       50     63       80       37
## 24     40         37         42       58     50       57       49
## 25     63         54         42       48     66       75       33
## 26     66         77         66       63     88       76       72
## 27     78         75         58       74     80       78       49
## 28     48         57         44       45     51       83       38
## 29     85         85         71       71     77       74       55
## 30     82         82         39       59     64       78       39
```

```
#Q5(a)Summarize the main statistics of all the variables in the data set
```

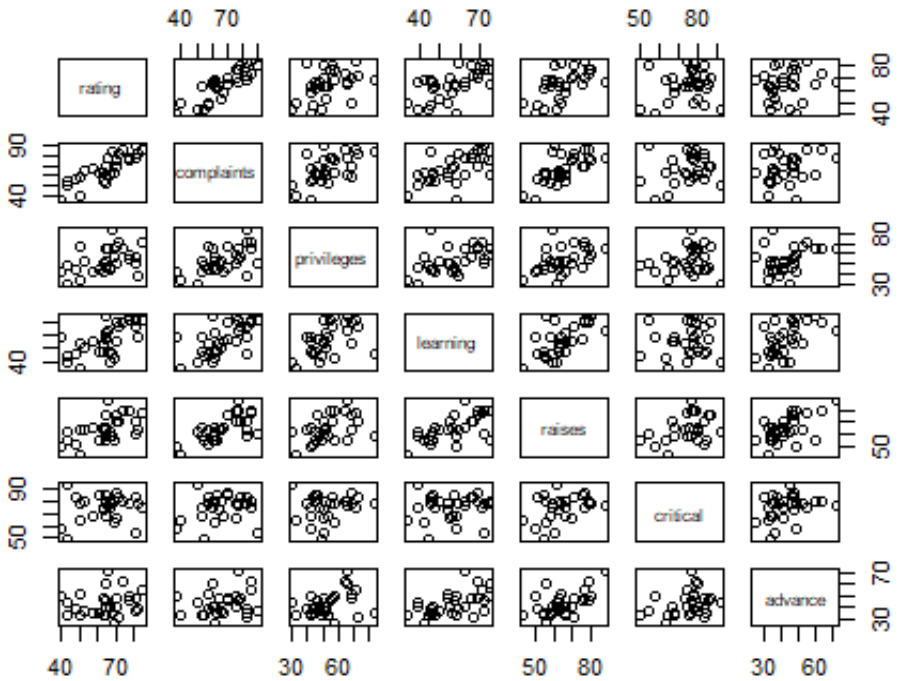
summary(attitude)

```
##      rating        complaints     privileges       learning         raises
##  Min.   :40.00   Min.   :37.0   Min.   :30.00   Min.   :34.00   Min.   :43
.00
##  1st Qu.:58.75   1st Qu.:58.5   1st Qu.:45.00   1st Qu.:47.00   1st Qu.:58
.25
##  Median :65.50   Median :65.0   Median :51.50   Median :56.50   Median :63
.50
##  Mean   :64.63   Mean   :66.6   Mean   :53.13   Mean   :56.37   Mean   :64
.63
##  3rd Qu.:71.75   3rd Qu.:77.0   3rd Qu.:62.50   3rd Qu.:66.75   3rd Qu.:71
.00
##  Max.   :85.00   Max.   :90.0   Max.   :83.00   Max.   :75.00   Max.   :88
.00
##     critical        advance
##  Min.   :49.00   Min.   :25.00
```

```
##  1st Qu.:69.25   1st Qu.:35.00
##  Median :77.50   Median :41.00
##  Mean   :74.77   Mean   :42.93
##  3rd Qu.:80.00   3rd Qu.:47.75
##  Max.   :92.00   Max.   :72.00
```

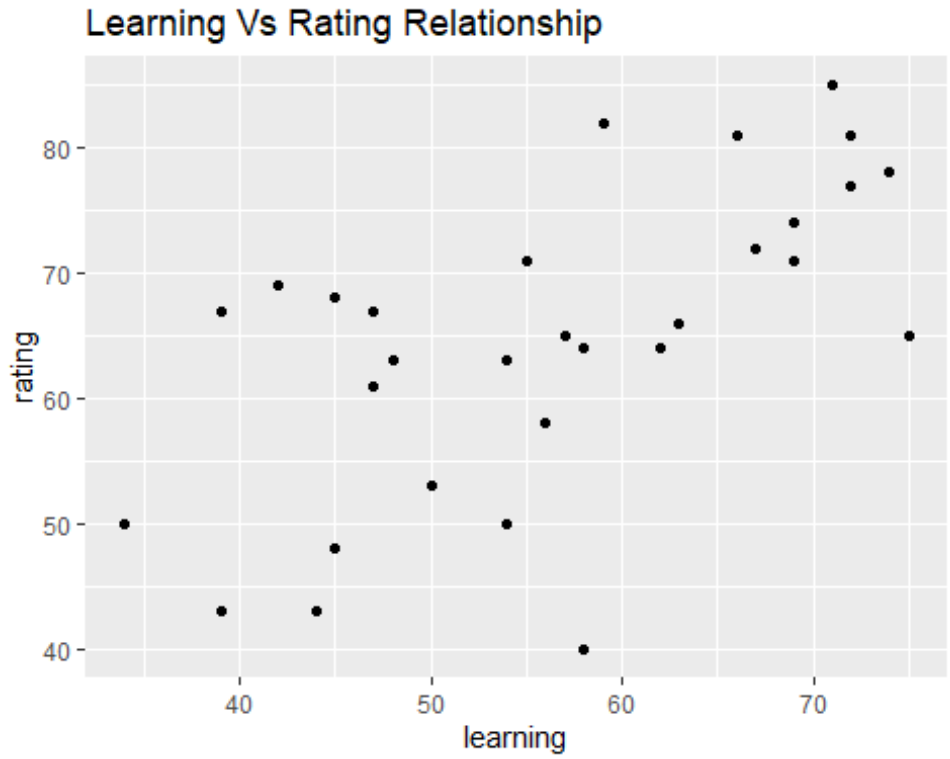
```
#summary(attitude) summarizes important statistical characteristics of each
variable such as mean , median , max. & min. values , 1st and 3rd quartile


#Q5(b)Summarize the main statistics of all the variables in the data set


View(attitude)

# => There are total 30 observation of 7 variables , Command used is view(x)
```

```
#Q5(c) scatter plot matrix of the variables in the attitude dataset
plot(attitude)
```

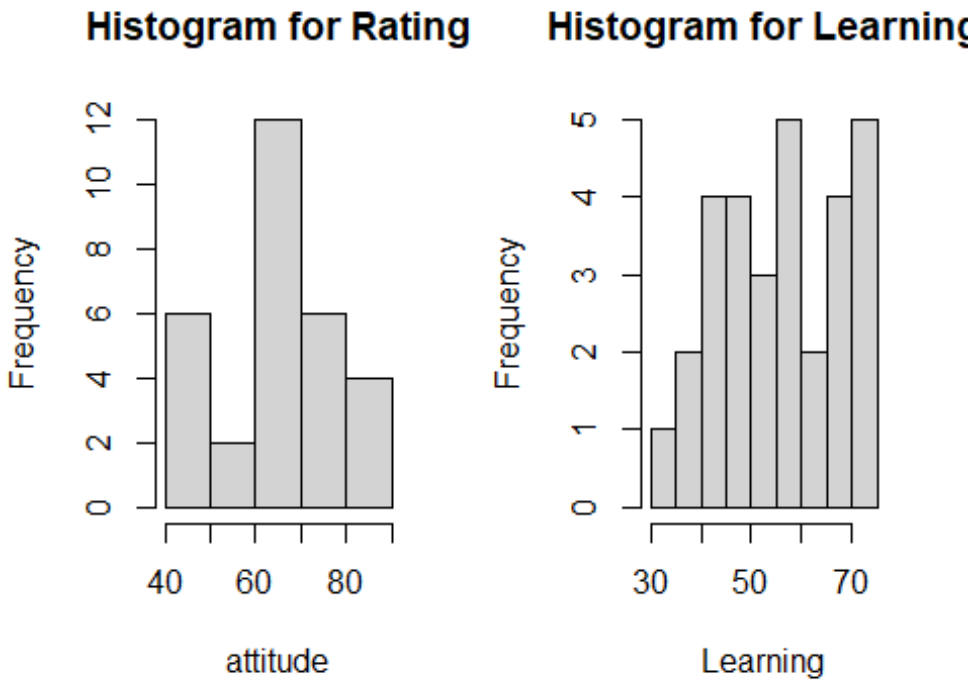



```
# => Correlation :- Linear Correlation between ratings and complaints.
```

```
#Q5(d) scatter plot of rating (on the y-axis) vs. learning (on the x-axis)

attitude %>% ggplot(data=attitude,mapping = aes(x=learning,y=rating))+ geom_p
oint()+labs(title= "Learning Vs Rating Relationship")
```




```
#Q5(e) Histogram for rating and learning variable
par(mfrow=c(1,2))
hist(attitude$rating,main = "Histogram for Rating",xlab = "attitude",ylab = "
Frequency")
hist(attitude$learning,main = "Histogram for Learning",xlab = "Learning",ylab
= "Frequency")
```




```
#Question6
```

```
data(mtcars)
head(mtcars, 6)
```

```
##                   mpg cyl disp  hp drat    wt  qsec vs am gear carb
## Mazda RX4         21.0   6  160 110 3.90 2.620 16.46  0  1    4    4
## Mazda RX4 Wag     21.0   6  160 110 3.90 2.875 17.02  0  1    4    4
## Datsun 710        22.8   4  108  93 3.85 2.320 18.61  1  1    4    1
## Hornet 4 Drive    21.4   6  258 110 3.08 3.215 19.44  1  0    3    1
## Hornet Sportabout 18.7   8  360 175 3.15 3.440 17.02  0  0    3    2
## Valiant           18.1   6  225 105 2.76 3.460 20.22  1  0    3    1
```

```
mtcars
```

```
##                     mpg cyl  disp  hp drat    wt  qsec vs am gear carb
## Mazda RX4          21.0   6 160.0 110 3.90 2.620 16.46  0  1    4    4
## Mazda RX4 Wag      21.0   6 160.0 110 3.90 2.875 17.02  0  1    4    4
## Datsun 710         22.8   4 108.0  93 3.85 2.320 18.61  1  1    4    1
## Hornet 4 Drive     21.4   6 258.0 110 3.08 3.215 19.44  1  0    3    1
## Hornet Sportabout  18.7   8 360.0 175 3.15 3.440 17.02  0  0    3    2
## Valiant            18.1   6 225.0 105 2.76 3.460 20.22  1  0    3    1
## Duster 360         14.3   8 360.0 245 3.21 3.570 15.84  0  0    3    4
## Merc 240D          24.4   4 146.7  62 3.69 3.190 20.00  1  0    4    2
## Merc 230           22.8   4 140.8  95 3.92 3.150 22.90  1  0    4    2
## Merc 280           19.2   6 167.6 123 3.92 3.440 18.30  1  0    4    4
## Merc 280C          17.8   6 167.6 123 3.92 3.440 18.90  1  0    4    4
```

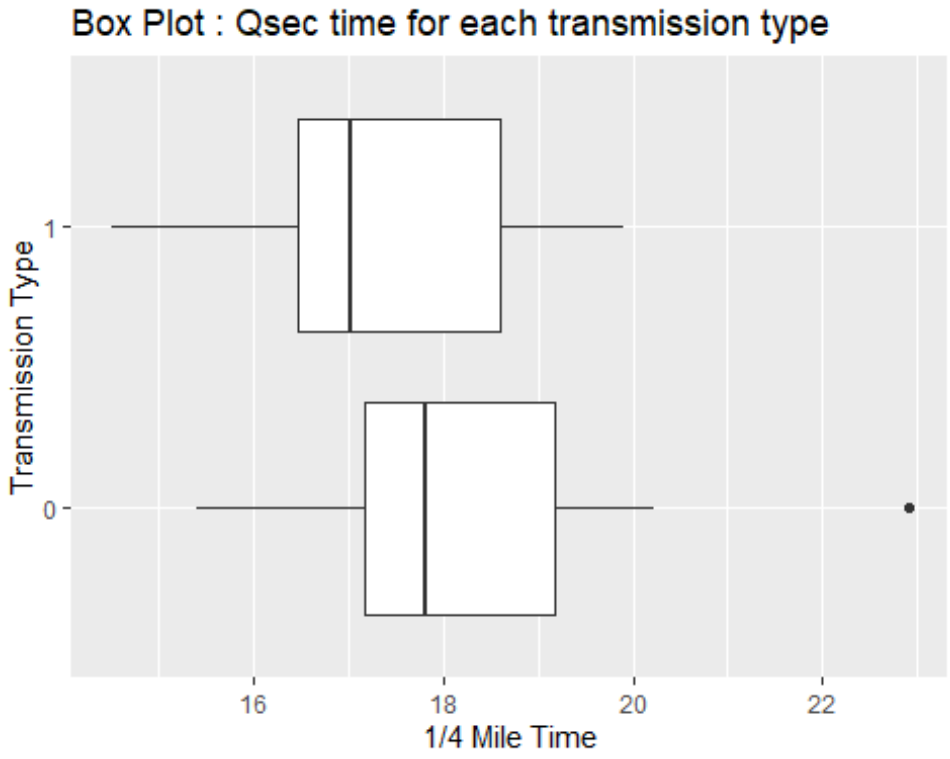
```
## Merc 450SE          16.4   8 275.8 180 3.07 4.070 17.40  0  0    3    3
## Merc 450SL          17.3   8 275.8 180 3.07 3.730 17.60  0  0    3    3
## Merc 450SLC         15.2   8 275.8 180 3.07 3.780 18.00  0  0    3    3
## Cadillac Fleetwood  10.4   8 472.0 205 2.93 5.250 17.98  0  0    3    4
## Lincoln Continental 10.4   8 460.0 215 3.00 5.424 17.82  0  0    3    4
## Chrysler Imperial   14.7   8 440.0 230 3.23 5.345 17.42  0  0    3    4
## Fiat 128            32.4   4  78.7  66 4.08 2.200 19.47  1  1    4    1
## Honda Civic         30.4   4  75.7  52 4.93 1.615 18.52  1  1    4    2
## Toyota Corolla      33.9   4  71.1  65 4.22 1.835 19.90  1  1    4    1
## Toyota Corona       21.5   4 120.1  97 3.70 2.465 20.01  1  0    3    1
## Dodge Challenger    15.5   8 318.0 150 2.76 3.520 16.87  0  0    3    2
## AMC Javelin         15.2   8 304.0 150 3.15 3.435 17.30  0  0    3    2
## Camaro Z28          13.3   8 350.0 245 3.73 3.840 15.41  0  0    3    4
## Pontiac Firebird    19.2   8 400.0 175 3.08 3.845 17.05  0  0    3    2
## Fiat X1-9           27.3   4  79.0  66 4.08 1.935 18.90  1  1    4    1
## Porsche 914-2       26.0   4 120.3  91 4.43 2.140 16.70  0  1    5    2
## Lotus Europa        30.4   4  95.1 113 3.77 1.513 16.90  1  1    5    2
## Ford Pantera L      15.8   8 351.0 264 4.22 3.170 14.50  0  1    5    4
## Ferrari Dino        19.7   6 145.0 175 3.62 2.770 15.50  0  1    5    6
## Maserati Bora       15.0   8 301.0 335 3.54 3.570 14.60  0  1    5    8
## Volvo 142E          21.4   4 121.0 109 4.11 2.780 18.60  1  1    4    2
```

```
# View Details of mtcars dataset


?mtcars
```

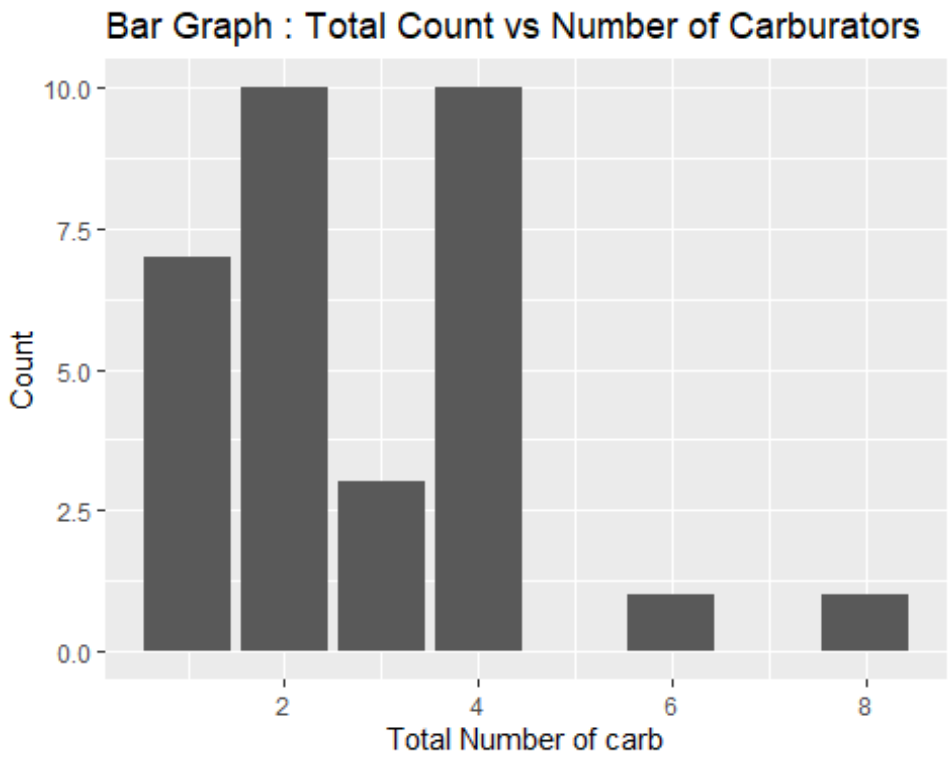
```
## starting httpd help server ... done
```

```
#6(a)

#The dataset mtcars comprises of fuel consumption and other 10 aspects of
automobile design and performance for 32 automobiles (1973-1974 models)

#There are 11 variables in the data set
#[, 1]   mpg Miles/(US) gallon
#[, 2]   cyl Number of cylinders
#[, 3]   disp    Displacement (cu.in.)
#[, 4]   hp  Gross horsepower
#[, 5]   drat    Rear axle ratio
# [, 6]   wt  Weight (1000 lbs)
# [, 7]   qsec    1/4 mile time
# [, 8]   vs  Engine (0 = V-shaped, 1 = straight)
# [, 9]   am  Transmission (0 = automatic, 1 = manual)
# [,10]   gear    Number of forward gears
# [,11]   carb    Number of carburetors "
```

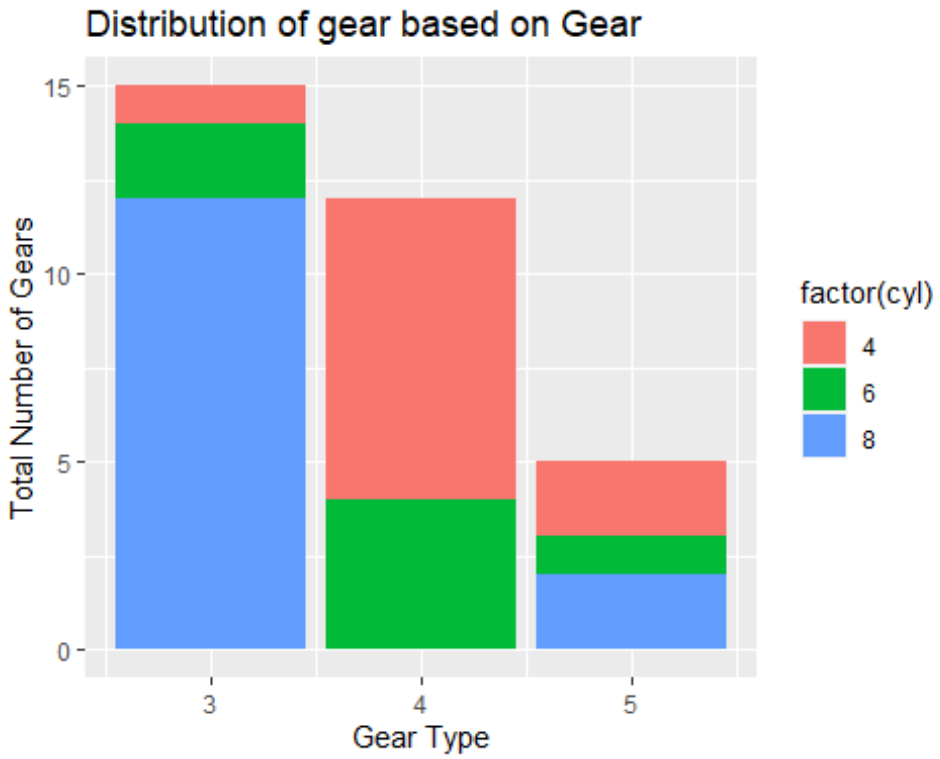
```
#Ques 6.b GG PLOT Q sec vs Transmission Type


ggplot(mtcars)+
  ggtitle("Box Plot : Qsec time for each transmission type ")+
  xlab("1/4 Mile Time") + ylab("Transmission Type")+
  geom_boxplot(mapping=aes(qsec,factor(am)))
```



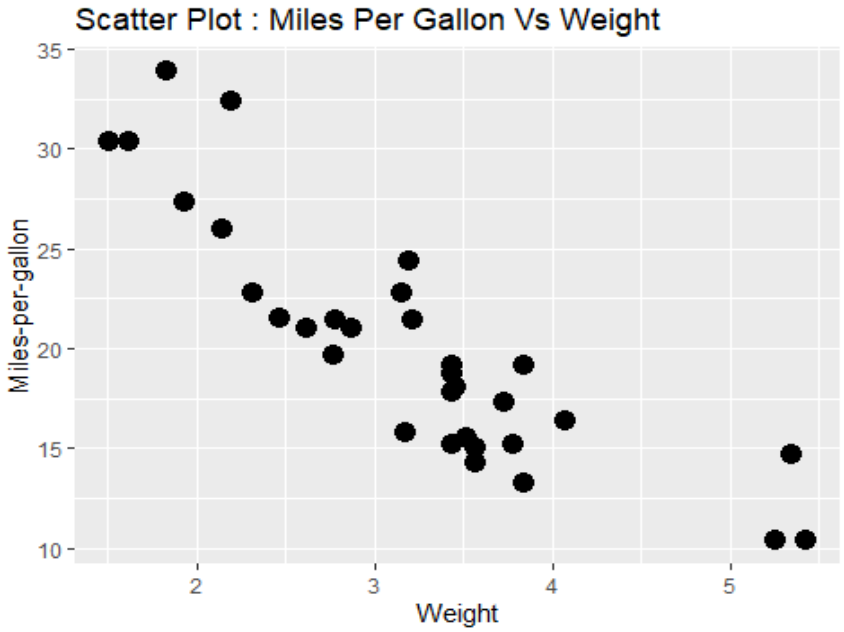

```
#6.c Bar Graph Total Count vs Type of Carb


ggplot(mtcars,aes(x=carb))+
         ggtitle("Bar Graph : Total Count vs Number of Carburators") +
  xlab("Total Number of carb") +
  ylab("Count") + geom_bar(stat="Count")
```



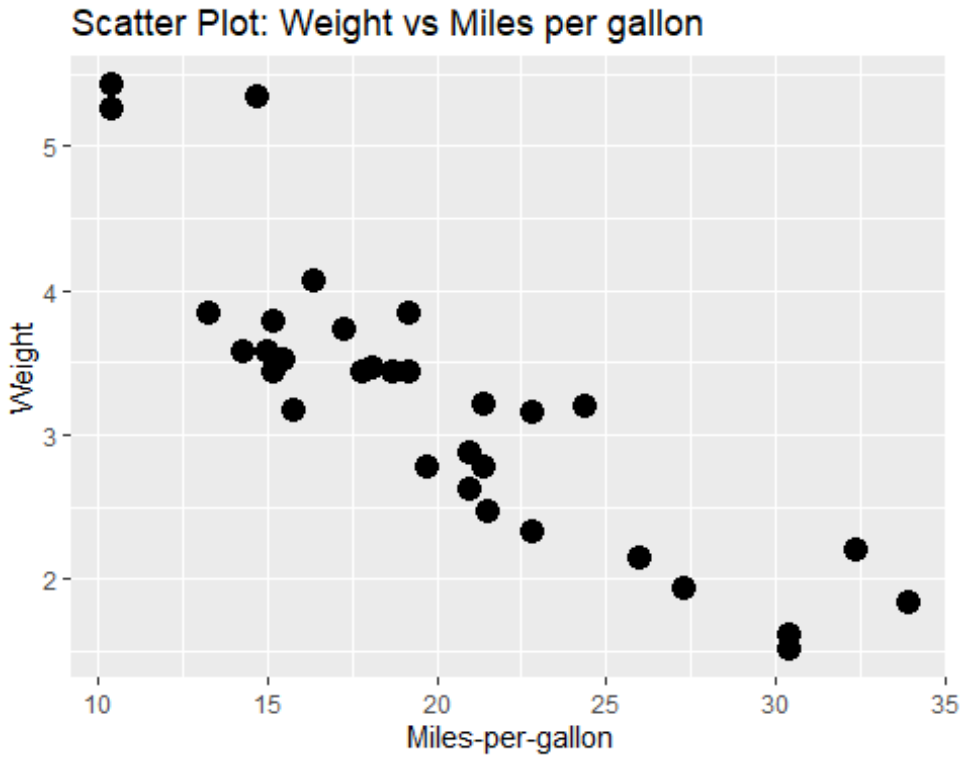

```
##6.d Stacked Bar Graph based on gear type and number of cylinders


ggplot(data=mtcars)+ggtitle("Distribution of gear based on Gear ")+
xlab("Gear Type") + ylab("Total Number of Gears")+
geom_bar(mapping=aes(x=(gear),fill=factor(cyl)))
```



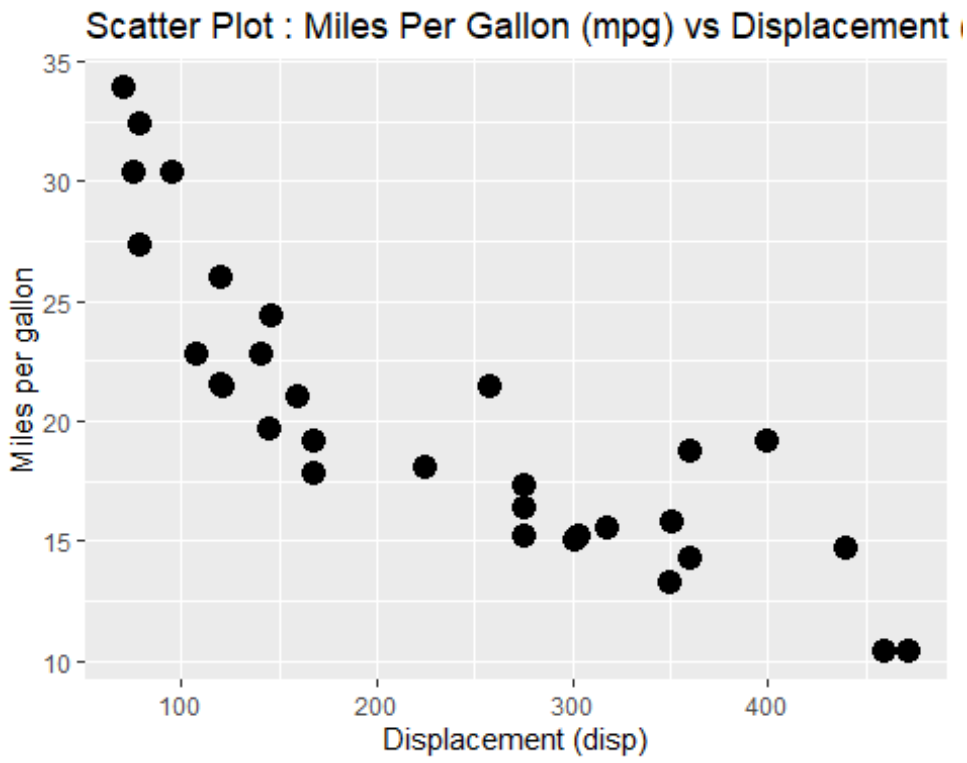

```
##6.e Scatter Plot Wt vs mpg


ggplot(data=mtcars,aes(y=mpg,x=wt))+
  ggtitle("Scatter Plot : Miles Per Gallon Vs Weight")+
xlab("Weight")+
ylab("Miles-per-gallon")+geom_point(size=4)
```



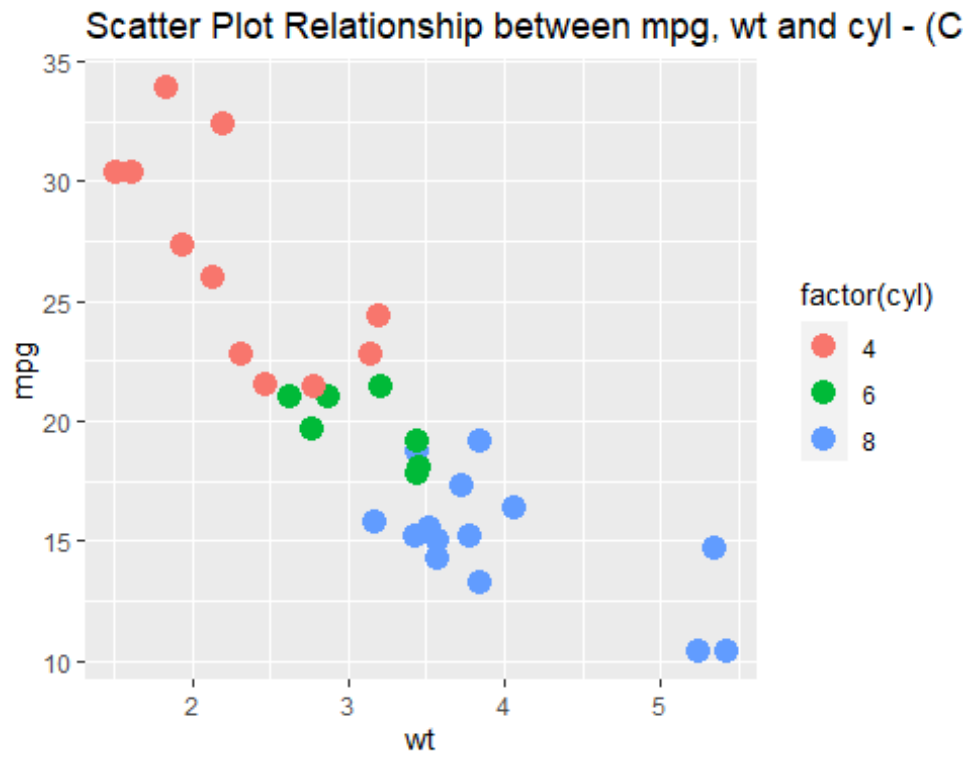
```
##Ques 6 Scatter plot wt vs mpg

ggplot(data=mtcars,aes(x=mpg,y=wt))+
  ggtitle("Scatter Plot: Weight vs Miles per gallon")+
  ylab("Weight")+
  xlab("Miles-per-gallon")+geom_point(size=4)
```



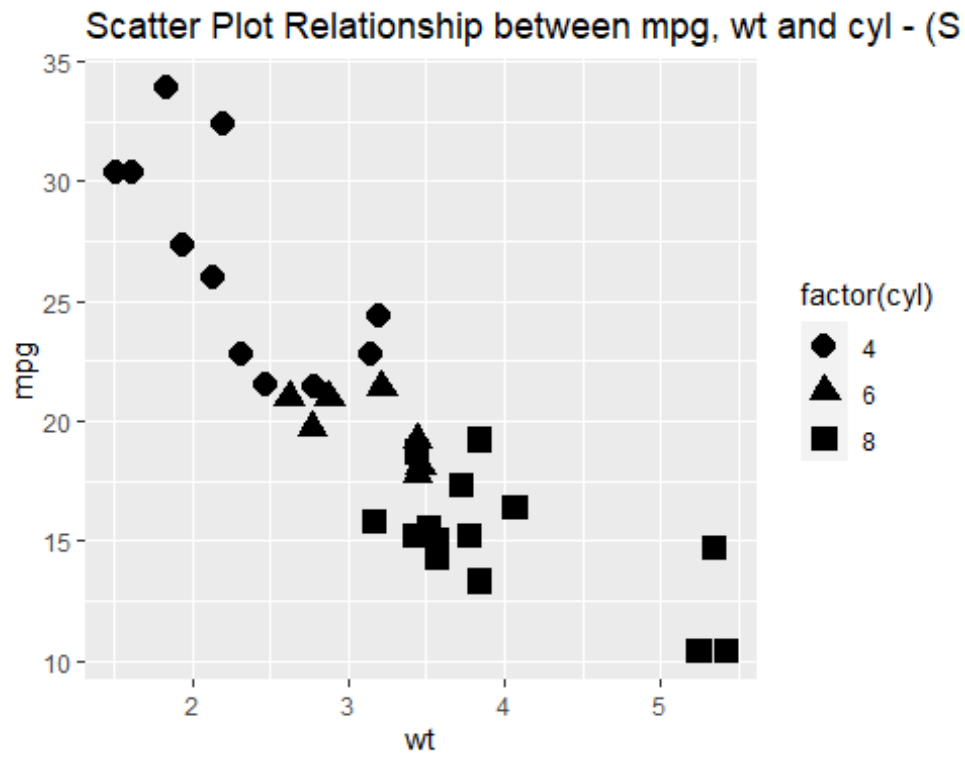

```
#Ques 6f Scatter plot disp vs mpg

ggplot(data=mtcars,aes(x=disp,y=mpg))+
  ggtitle("Scatter Plot : Miles Per Gallon (mpg) vs Displacement (disp)")+
  ylab("Miles per gallon")+
  xlab("Displacement (disp)")+geom_point(size=4)
```




```
#As value of disp Incresesd the value of mpg decreases


# Ques 6(g) Scatter Plot 1 relationship among mpg, wt and cyl

ggplot(mtcars, aes( x=wt ,y=mpg,color = factor(cyl))) +
  ggtitle("Scatter Plot Relationship between mpg, wt and cyl - (Color)")+
  geom_point(size= 4)
```




```
# Ques 6(h) Scatter Plot 2 relationship among mpg, wt and cyl


ggplot(mtcars, aes( x=wt ,y=mpg,shape = factor(cyl))) +
  ggtitle("Scatter Plot Relationship between mpg, wt and cyl - (Shape)") +
  geom_point(size= 4)
```



#Question 7

```
#loading dataset
library(readr)
data2 <- read.csv("C:\\Users\\psharm50\\Desktop\\gapminder.csv")


#Ques 7(a)
data_count_1 <- aggregate(data = data2,country ~ continent,function(country)
length(unique(country)))
data_count_1
```

```
##   continent country
## 1    Africa      52
## 2  Americas      25
## 3      Asia      33
## 4    Europe      30
## 5   Oceania       2
```

```
#We use the aggregate function to group the countries by continent and find
the count of unique countries in each continent
```

```r
#Question 7(b)


library(dplyr)
datanew <- filter(data2,continent=="Europe" & year == "1997")
filter(data2, gdpPercap == min(datanew$gdpPercap))
```

```
##   country continent year lifeExp     pop gdpPercap
## 1 Albania    Europe 1997   72.95 3428038  3193.055
```

```r
#we use the filter function in dplyr to filter the data for Europe in 1997
 and assign it to datanew. We find the min gdpPercap from this filtered data
and find it in the original dataframe. We find the answer is Albania.


#Question 7(c)

datanew1 <- filter(data2, year %in% c(1980:1989))
datanew2 <- aggregate(data=datanew1, lifeExp ~ continent, function(lifeExp) m
ean(lifeExp))
colnames(datanew2)[2]<- "Average Life Exp"
datanew2
```

```
##   continent Average Life Exp
## 1    Africa         52.46883
## 2  Americas         67.15978
## 3      Asia         63.73456
## 4    Europe         73.22428
## 5   Oceania         74.80500
```

```r
#We use the filter function to filter the data for 1980-1989. Using the
 aggregate function, We find the mean Life expectancy for each continent


#Question 7(d)


data_count_2 <- aggregate(data = data2,gdpPercap ~ country,function(gdpPercap
) sum(gdpPercap))

data_count_2 %>%
arrange(desc(gdpPercap)) %>%
slice(1:5)
```

```
##         country gdpPercap
## 1        Kuwait  783994.9
## 2   Switzerland  324892.0
## 3        Norway  320967.7
## 4 United States  315133.8
## 5        Canada  268929.0
```

```
#Using the aggregate function, we find the sum of gdpPercap for each country
over the years. We then arrange these in a descencding order and select the
 top 5 to get the 5 countries with max total gdppercap


#Question 7(f)
datanew3 <- filter(data2, lifeExp >= 80)
datanew3 <- datanew3[,c("country","lifeExp","year")]
datanew3
```

```
##             country lifeExp year
## 1         Australia  80.370 2002
## 2         Australia  81.235 2007
## 3            Canada  80.653 2007
## 4            France  80.657 2007
## 5  Hong Kong, China  80.000 1997
## 6  Hong Kong, China  81.495 2002
## 7  Hong Kong, China  82.208 2007
## 8           Iceland  80.500 2002
## 9           Iceland  81.757 2007
## 10           Israel  80.745 2007
## 11            Italy  80.240 2002
## 12            Italy  80.546 2007
## 13            Japan  80.690 1997
## 14            Japan  82.000 2002
## 15            Japan  82.603 2007
## 16      New Zealand  80.204 2007
## 17           Norway  80.196 2007
## 18            Spain  80.941 2007
## 19           Sweden  80.040 2002
## 20           Sweden  80.884 2007
## 21      Switzerland  80.620 2002
## 22      Switzerland  81.701 2007
```

```
#We filter the data to find the lifeexp over 80, then we select the relevant
columns from the data frame and display them
```

```r
#Question 8

#Imports
#install.packages("hflights")
library(hflights)
library(dplyr)

#Import data
flightdata <- hflights

#Q8(a) first 20 instances
head(flightdata, 20L)
```

```
##      Year Month DayofMonth DayOfWeek DepTime ArrTime UniqueCarrier FlightN
um
## 5424 2011     1          1         6    1400    1500            AA       4
28
## 5425 2011     1          2         7    1401    1501            AA       4
28
## 5426 2011     1          3         1    1352    1502            AA       4
28
## 5427 2011     1          4         2    1403    1513            AA       4
28
## 5428 2011     1          5         3    1405    1507            AA       4
28
## 5429 2011     1          6         4    1359    1503            AA       4
28
## 5430 2011     1          7         5    1359    1509            AA       4
28
## 5431 2011     1          8         6    1355    1454            AA       4
28
## 5432 2011     1          9         7    1443    1554            AA       4
28
## 5433 2011     1         10         1    1443    1553            AA       4
28
## 5434 2011     1         11         2    1429    1539            AA       4
28
## 5435 2011     1         12         3    1419    1515            AA       4
28
## 5436 2011     1         13         4    1358    1501            AA       4
28
## 5437 2011     1         14         5    1357    1504            AA       4
28
## 5438 2011     1         15         6    1359    1459            AA       4
28
## 5439 2011     1         16         7    1359    1509            AA       4
28
## 5440 2011     1         17         1    1530    1634            AA       4
28
## 5441 2011     1         18         2    1408    1508            AA       4
```

```
28
## 5442 2011     1         19         3    1356    1503            AA       4
28
## 5443 2011     1         20         4    1507    1622            AA       4
28
##      TailNum ActualElapsedTime AirTime ArrDelay DepDelay Origin Dest Dista
nce
## 5424  N576AA                60      40      -10        0    IAH  DFW
224
## 5425  N557AA                60      45       -9        1    IAH  DFW
224
## 5426  N541AA                70      48       -8       -8    IAH  DFW
224
## 5427  N403AA                70      39        3        3    IAH  DFW
224
## 5428  N492AA                62      44       -3        5    IAH  DFW
224
## 5429  N262AA                64      45       -7       -1    IAH  DFW
224
## 5430  N493AA                70      43       -1       -1    IAH  DFW
224
## 5431  N477AA                59      40      -16       -5    IAH  DFW
224
## 5432  N476AA                71      41       44       43    IAH  DFW
224
## 5433  N504AA                70      45       43       43    IAH  DFW
224
## 5434  N565AA                70      42       29       29    IAH  DFW
224
## 5435  N577AA                56      41        5       19    IAH  DFW
224
## 5436  N476AA                63      44       -9       -2    IAH  DFW
224
## 5437  N552AA                67      47       -6       -3    IAH  DFW
224
## 5438  N462AA                60      44      -11       -1    IAH  DFW
224
## 5439  N555AA                70      41       -1       -1    IAH  DFW
224
## 5440  N518AA                64      48       84       90    IAH  DFW
224
## 5441  N507AA                60      42       -2        8    IAH  DFW
224
## 5442  N523AA                67      46       -7       -4    IAH  DFW
224
## 5443  N425AA                75      42       72       67    IAH  DFW
224
##      TaxiIn TaxiOut Cancelled CancellationCode Diverted
## 5424      7      13         0                        0
## 5425      6       9         0                        0
```

```
## 5426      5      17         0                        0
## 5427      9      22         0                        0
## 5428      9       9         0                        0
## 5429      6      13         0                        0
## 5430     12      15         0                        0
## 5431      7      12         0                        0
## 5432      8      22         0                        0
## 5433      6      19         0                        0
## 5434      8      20         0                        0
## 5435      4      11         0                        0
## 5436      6      13         0                        0
## 5437      5      15         0                        0
## 5438      6      10         0                        0
## 5439     12      17         0                        0
## 5440      8       8         0                        0
## 5441      7      11         0                        0
## 5442     10      11         0                        0
## 5443      9      24         0                        0
```

```
##flightdata

#Q8(b) Fights on January 1st
firstjanflightdata <- filter(flightdata, Month == 1 & DayofMonth == 1)
head(firstjanflightdata,10)
```

```
##    Year Month DayofMonth DayOfWeek DepTime ArrTime UniqueCarrier FlightNum
## 1  2011     1          1         6    1400    1500            AA       428
## 2  2011     1          1         6     728     840            AA       460
## 3  2011     1          1         6    1631    1736            AA      1121
## 4  2011     1          1         6    1756    2112            AA      1294
## 5  2011     1          1         6    1012    1347            AA      1700
## 6  2011     1          1         6    1211    1325            AA      1820
## 7  2011     1          1         6     557     906            AA      1994
## 8  2011     1          1         6    1824    2106            AS       731
## 9  2011     1          1         6     654    1124            B6       620
## 10 2011     1          1         6    1639    2110            B6       622
##    TailNum ActualElapsedTime AirTime ArrDelay DepDelay Origin Dest Distanc
e
## 1   N576AA                60      40      -10        0    IAH  DFW      22
4
## 2   N520AA                72      41        5        8    IAH  DFW      22
4
## 3   N4WVAA                65      37       -9        1    IAH  DFW      22
4
## 4   N3DGAA               136     113       -3        1    IAH  MIA      96
4
## 5   N3DAAA               155     117        7       -8    IAH  MIA      96
4
## 6   N593AA                74      39       15        6    IAH  DFW      22
4
```

```
## 7   N3BBAA              129     113       -9       -3    IAH  MIA      96
4
## 8   N614AS              282     255       -4       -1    IAH  SEA     187
4
## 9   N324JB              210     181        5       -6    HOU  JFK     142
8
## 10  N324JB              211     188       61       54    HOU  JFK     142
8
##    TaxiIn TaxiOut Cancelled CancellationCode Diverted
## 1       7      13         0                         0
## 2       6      25         0                         0
## 3      16      12         0                         0
## 4       9      14         0                         0
## 5      12      26         0                         0
## 6       6      29         0                         0
## 7       5      11         0                         0
## 8       7      20         0                         0
## 9       6      23         0                         0
## 10     12      11         0                         0
```

```
##View(firstjanflightdata)

#Q8(C) Dataset related to American or United Airlines carriers


AAUAairlinesdata <- filter(flightdata , UniqueCarrier == "AA" |UniqueCarrier
== "UA" )


#head(AAUAairlinesdata,10)
#View(AAUAairlinesdata)

#Q8(d) Year, Month, DayofMonth and variables with word Taxi and Delay
data1 <- flightdata %>% select(c('Year', 'Month', 'DayofMonth'))
data2 <- flightdata %>% select(starts_with("Taxi") | ends_with("Delay"))
cmbdata <- data.frame(data1, data2)
head(cmbdata,10)
```

```
##      Year Month DayofMonth TaxiIn TaxiOut ArrDelay DepDelay
## 5424 2011     1          1      7      13      -10        0
## 5425 2011     1          2      6       9       -9        1
## 5426 2011     1          3      5      17       -8       -8
## 5427 2011     1          4      9      22        3        3
## 5428 2011     1          5      9       9       -3        5
## 5429 2011     1          6      6      13       -7       -1
## 5430 2011     1          7     12      15       -1       -1
## 5431 2011     1          8      7      12      -16       -5
## 5432 2011     1          9      8      22       44       43
## 5433 2011     1         10      6      19       43       43
```

```
##View(cmbdata)

#Q8(e) Subset data Departure Time, Arrival Time and Flight Number
newData <- flightdata %>% select(c('DepTime', 'ArrTime', 'FlightNum'))
head(newData,10)

##      DepTime ArrTime FlightNum
## 5424    1400    1500       428
## 5425    1401    1501       428
## 5426    1352    1502       428
## 5427    1403    1513       428
## 5428    1405    1507       428
## 5429    1359    1503       428
## 5430    1359    1509       428
## 5431    1355    1454       428
## 5432    1443    1554       428
## 5433    1443    1553       428

##View(newData)

#Q8(f) Dep. Delay more than 60 mins


flightdelay <- filter(flightdata , DepDelay > 60)
head(flightdelay,10)

##    Year Month DayofMonth DayOfWeek DepTime ArrTime UniqueCarrier FlightNum
## 1  2011     1         17         1    1530    1634            AA       428
## 2  2011     1         20         4    1507    1622            AA       428
## 3  2011     1         14         5    2119    2229            AA       533
## 4  2011     1          9         7    1835    1951            AA      1121
## 5  2011     1         11         2    1752    1855            AA      1121
## 6  2011     1         10         1    1934    2235            AA      1294
## 7  2011     1         26         3    1905    2211            AA      1294
## 8  2011     1         30         7    1856    2209            AA      1294
## 9  2011     1         11         2    1134    1454            AA      1700
## 10 2011     1          9         7    1938    2228            AS       731
##    TailNum ActualElapsedTime AirTime ArrDelay DepDelay Origin Dest Distanc
e
## 1   N518AA                64      48       84       90    IAH  DFW      22
4
## 2   N425AA                75      42       72       67    IAH  DFW      22
4
## 3   N549AA                70      45       69       74    IAH  DFW      22
4
## 4   N574AA                76      50      126      125    IAH  DFW      22
4
## 5   N586AA                63      41       70       82    IAH  DFW      22
4
## 6   N3BXAA               121     107       80       99    IAH  MIA      96
4
```

```
## 7   N3BXAA              126     111       56       70    IAH  MIA      96
4
## 8   N3CPAA              133     108       54       61    IAH  MIA      96
4
## 9   N3ALAA              140     115       74       74    IAH  MIA      96
4
## 10  N609AS              290     253       78       73    IAH  SEA     187
4
##    TaxiIn TaxiOut Cancelled CancellationCode Diverted
## 1       8       8         0                         0
## 2       9      24         0                         0
## 3       5      20         0                         0
## 4       9      17         0                         0
## 5       8      14         0                         0
## 6       3      11         0                         0
## 7       5      10         0                         0
## 8       7      18         0                         0
## 9      11      14         0                         0
## 10      5      32         0                         0
```

```
#Q8(g) Sorting departure Delay


carrierDeptDelay <- na.omit(flightdata)%>% select(c('UniqueCarrier', 'DepDela
y' ))
sortedIndices <- order(carrierDeptDelay$DepDelay)
carrierDeptDelay <- carrierDeptDelay[sortedIndices,]
head(carrierDeptDelay,10)
```

```
##         UniqueCarrier DepDelay
## 5996719            OO      -33
## 927973             MQ      -23
## 1694833            XE      -19
## 3814017            XE      -19
## 83407              CO      -18
## 5035285            EV      -18
## 457114             XE      -17
## 1043606            CO      -17
## 1442181            XE      -17
## 1965737            MQ      -17
```

#Question 9

Frequency Tables

		Yes	No
Income	high	0	3
	medium	4	2
	low	3	2

If Income = high ==> NO = 0/3
If Income = medium ==> Yes = 2/6
If Income = low ==> Yes = 2/5
Overall Misclassification rate = **4/14**

		buys computer	
		yes	no
student	yes	5	3
	no	2	4

If student = yes ==> yes = 3/8
If student = no ==> no = 2/6

Overall Misclassification rate = **5/14**

		buys computer	
		yes	no
credit rating	excellent	5	2
	fair	2	5

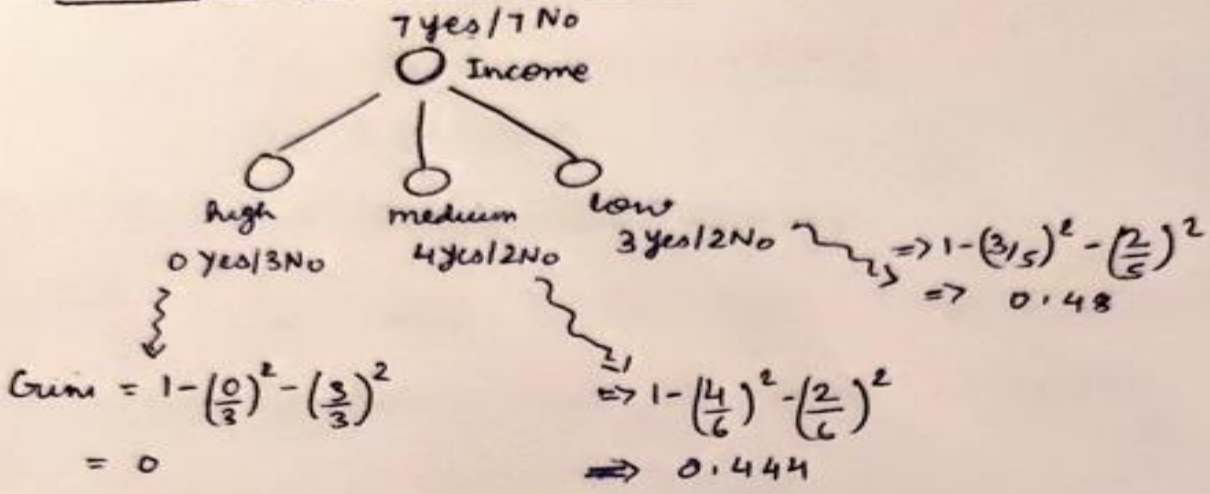
If credit rating = excellent ==> yes = 2/7
If credit rating = fair ==> no = 2/7

Overall Misclassification rate = 4/14

Lowest Misclassification is for Credit rating rule and Income

Record number	income	student	credit-rating	buys-computer
1	high	no	Fair	no
2	high	no	excellent	no
3	low	no	excellent	yes
4	medium	no	Fair	no
5	low	yes	Fair	no
6	low	yes	excellent	yes
7	low	no	excellent	yes
8	medium	yes	Fair	yes
9	low	yes	Fair	no
10	medium	yes	Fair	yes
11	medium	yes	excellent	yes
12	medium	no	excellent	no
13	high	yes	Fair	no
14	medium	yes	excellent	yes

Gini Index for income variable



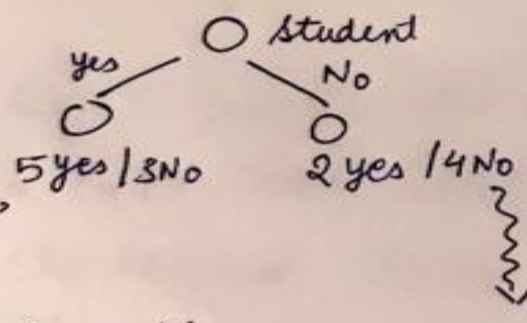
$$\text{Gini} = 1 - \left(\frac{0}{3}\right)^2 - \left(\frac{3}{3}\right)^2 = 0$$

$$\Rightarrow 1 - \left(\frac{4}{6}\right)^2 - \left(\frac{2}{6}\right)^2 \Rightarrow 0.444$$

$$\text{Gini(Income)} = \frac{3}{14}(0) + \frac{6}{14}(0.444) + \frac{5}{14}(0.48)$$

$$= 0.3619$$

Gini Index for Student Variable



$$= 1 - \left(\frac{5}{8}\right)^2 - \left(\frac{3}{8}\right)^2 = 0.468$$

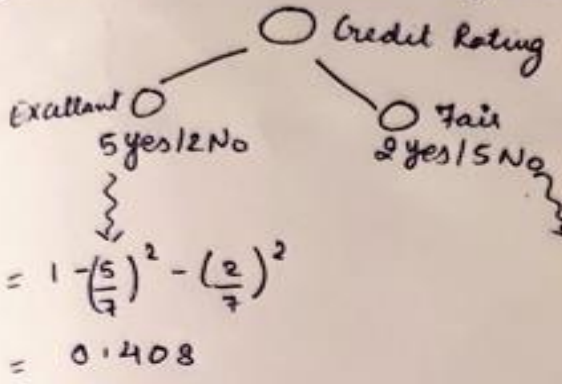
$$1 - \left(\frac{2}{6}\right)^2 - \left(\frac{4}{6}\right)^2 = 0.444$$

$$\text{Gini(Student)} = \frac{8}{14}(0.468) + \frac{6}{14}(0.444)$$

$$= 0.267 + 0.190$$

$$= 0.45$$

Gini Index for Credit-Rating Variable



$= 1 - \left(\frac{5}{7}\right)^2 - \left(\frac{2}{7}\right)^2$

$= 0.408$

$= 1 - \left(\frac{2}{7}\right)^2 - \left(\frac{4}{7}\right)^2$

$= 0.408$

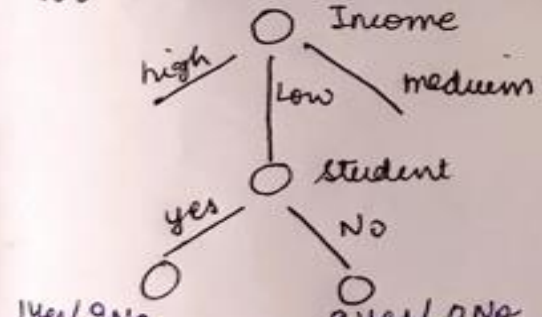
$\text{Gini (Credit Rating)} = \frac{7}{14}(0.408) + \frac{7}{14}(0.408)$

$= 0.408$

Gini (Income)	0.3619
Gini (Student)	0.45
Gini (Credit Rating)	0.408

Gini (Income) is the lowest. Hence root node is Income

For Income = Low & Student



$= 1 - \left(\frac{1}{3}\right)^2 - \left(\frac{2}{3}\right)^2$

$= 0.266$

$= 1 - \left(\frac{2}{2}\right)^2 - \left(\frac{0}{2}\right)^2$

$= 0$

Gini (Income = Low & Student)

$= \frac{3}{5} \times (0.266) + \left(\frac{2}{5}\right)(0) = 0.1596$

For Income = Low & Credit Rating

$= 1 - \left(\frac{3}{3}\right)^2 - \left(\frac{0}{3}\right)^2$

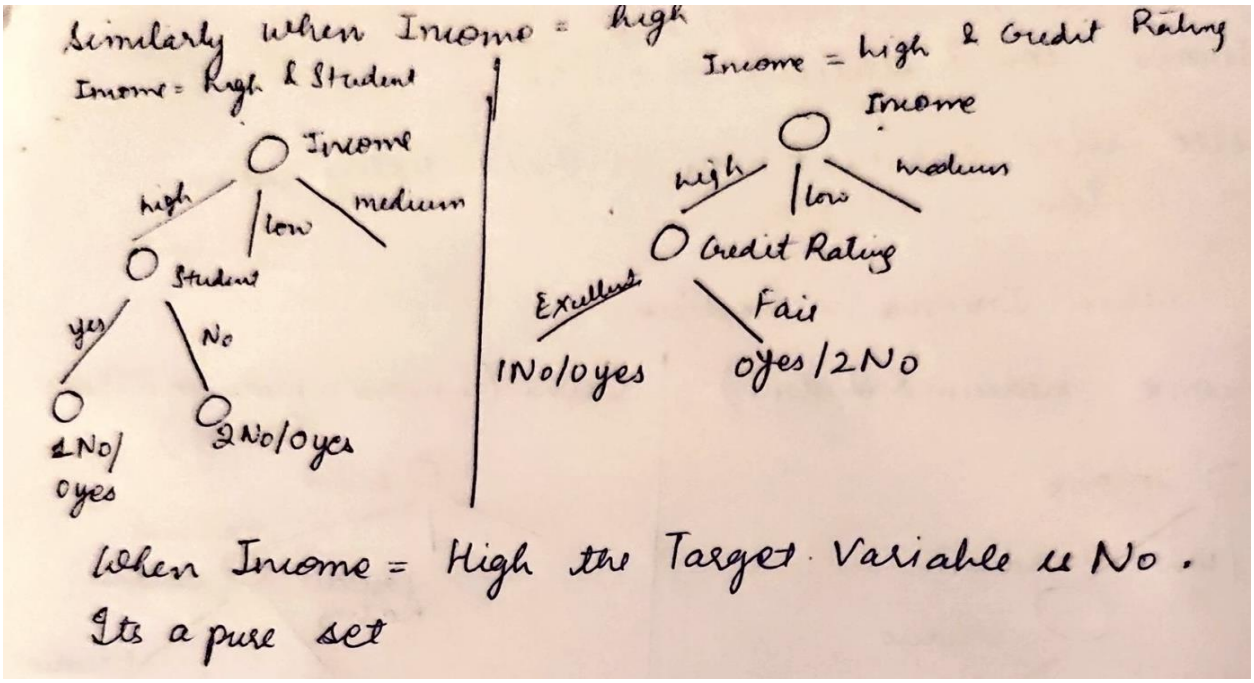
$= 0$

$= 1 - \left(\frac{0}{2}\right)^2 - \left(\frac{2}{2}\right)^2$

$= 0$

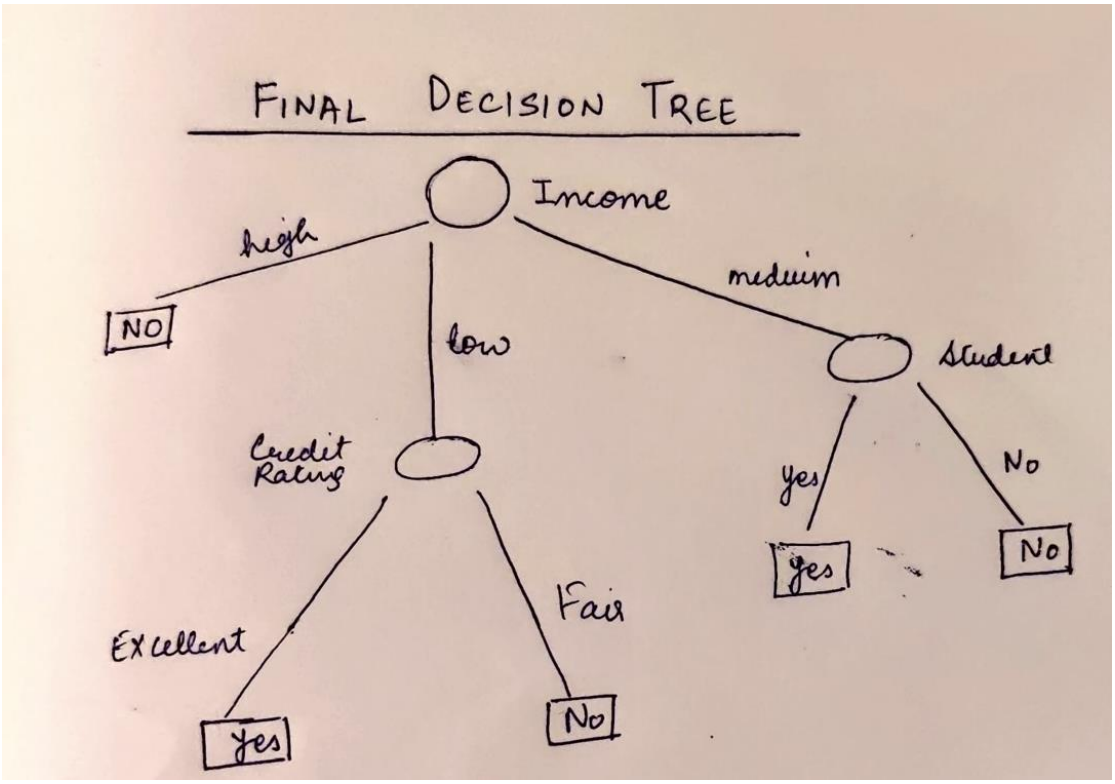
Gini (Income = Low & Credit Rating)

$= 0$



9.(b) As explained above, Using the Gini Index impurity measure the attribute selected as the root node for the decision tree will be Income.

9.(c) Below is the Full decision tree for the given data set.



9.(d) As observed in the decision tree:

1. For Income = High the target variable computer buy is always No. It's a pure set.
   For all the 3 cases given in the data, irrespective of Credit rating or Student (yes or no) the Target variable computer buy is always **No** when **Income = High**
2. For Income = Medium and Student = Yes, the target variable computer buy is always Yes.
   For all the 4 cases given in the data set, irrespective of credit rating the target variable buy computer is always **Yes** for Income = **Medium** and student = **Yes**.

9.(e) Since all the leaf/terminal nodes are pure, the accuracy of the decision tree is 100%