

# Exploratory Data Analysis Assignments

Washim Ahmed

12/14/2016

## Assignment 1

Write the R Program for the following steps.

(a) Load mtcars() dataframe

```
my_cars <- mtcars
head(my_cars)
```

```
##           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
```

(b) Add a column name qualCat, a new column which takes the value of A if mpg < 15 and B if mpg >=15

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##   filter, lag

## The following objects are masked from 'package:base':
##
##   intersect, setdiff, setequal, union
```

```
my_cars <- my_cars %>%
  mutate(qualCat = ifelse(mpg >= 15, "B", "A"))
head(my_cars,50)
```

```
##      mpg cyl  disp  hp drat   wt  qsec vs am gear carb qualCat
## 1  21.0   6  160.0 110 3.90 2.620 16.46  0  1   4    4         B
## 2  21.0   6  160.0 110 3.90 2.875 17.02  0  1   4    4         B
## 3  22.8   4  108.0  93 3.85 2.320 18.61  1  1   4    1         B
## 4  21.4   6  258.0 110 3.08 3.215 19.44  1  0   3    1         B
## 5  18.7   8  360.0 175 3.15 3.440 17.02  0  0   3    2         B
## 6  18.1   6  225.0 105 2.76 3.460 20.22  1  0   3    1         B
## 7  14.3   8  360.0 245 3.21 3.570 15.84  0  0   3    4         A
```

```
## 8 24.4 4 146.7 62 3.69 3.190 20.00 1 0 4 2 B
## 9 22.8 4 140.8 95 3.92 3.150 22.90 1 0 4 2 B
## 10 19.2 6 167.6 123 3.92 3.440 18.30 1 0 4 4 B
## 11 17.8 6 167.6 123 3.92 3.440 18.90 1 0 4 4 B
## 12 16.4 8 275.8 180 3.07 4.070 17.40 0 0 3 3 B
## 13 17.3 8 275.8 180 3.07 3.730 17.60 0 0 3 3 B
## 14 15.2 8 275.8 180 3.07 3.780 18.00 0 0 3 3 B
## 15 10.4 8 472.0 205 2.93 5.250 17.98 0 0 3 4 A
## 16 10.4 8 460.0 215 3.00 5.424 17.82 0 0 3 4 A
## 17 14.7 8 440.0 230 3.23 5.345 17.42 0 0 3 4 A
## 18 32.4 4 78.7 66 4.08 2.200 19.47 1 1 4 1 B
## 19 30.4 4 75.7 52 4.93 1.615 18.52 1 1 4 2 B
## 20 33.9 4 71.1 65 4.22 1.835 19.90 1 1 4 1 B
## 21 21.5 4 120.1 97 3.70 2.465 20.01 1 0 3 1 B
## 22 15.5 8 318.0 150 2.76 3.520 16.87 0 0 3 2 B
## 23 15.2 8 304.0 150 3.15 3.435 17.30 0 0 3 2 B
## 24 13.3 8 350.0 245 3.73 3.840 15.41 0 0 3 4 A
## 25 19.2 8 400.0 175 3.08 3.845 17.05 0 0 3 2 B
## 26 27.3 4 79.0 66 4.08 1.935 18.90 1 1 4 1 B
## 27 26.0 4 120.3 91 4.43 2.140 16.70 0 1 5 2 B
## 28 30.4 4 95.1 113 3.77 1.513 16.90 1 1 5 2 B
## 29 15.8 8 351.0 264 4.22 3.170 14.50 0 1 5 4 B
## 30 19.7 6 145.0 175 3.62 2.770 15.50 0 1 5 6 B
## 31 15.0 8 301.0 335 3.54 3.570 14.60 0 1 5 8 B
## 32 21.4 4 121.0 109 4.11 2.780 18.60 1 1 4 2 B
```

(c) Save this data frame locally

```
write.csv(my_cars, file = "~/manipal_practice/mycars.csv") # Given my Unix system path
```

(d) Load this file on to RStudio and know the different types of variables

```
mycars <- read.csv("~/manipal_practice/mycars.csv")
str(mycars) # Display structure of variables with observations
```

```
## 'data.frame': 32 obs. of 13 variables:
## $ X : int 1 2 3 4 5 6 7 8 9 10 ...
## $ mpg : num 21 21 22.8 21.4 18.7 18.1 14.3 24.4 22.8 19.2 ...
## $ cyl : int 6 6 4 6 8 6 8 4 4 6 ...
## $ disp : num 160 160 108 258 360 ...
## $ hp : int 110 110 93 110 175 105 245 62 95 123 ...
## $ drat : num 3.9 3.9 3.85 3.08 3.15 2.76 3.21 3.69 3.92 3.92 ...
## $ wt : num 2.62 2.88 2.32 3.21 3.44 ...
## $ qsec : num 16.5 17 18.6 19.4 17 ...
## $ vs : int 0 0 1 1 0 1 0 1 1 1 ...
## $ am : int 1 1 1 0 0 0 0 0 0 0 ...
## $ gear : int 4 4 4 3 3 3 3 4 4 4 ...
## $ carb : int 4 4 1 1 2 1 4 2 2 4 ...
## $ qualCat: Factor w/ 2 levels "A","B": 2 2 2 2 2 2 1 2 2 2 ...
```

```
summary(mycars) # Display summary
```

```
##           X           mpg           cyl           disp
## Min.      : 1.00    Min.   :10.40    Min.   :4.000    Min.   : 71.1
## 1st Qu.:  8.75    1st Qu.:15.43    1st Qu.:4.000    1st Qu.:120.8
## Median :16.50    Median :19.20    Median :6.000    Median :196.3
## Mean   :16.50    Mean   :20.09    Mean   :6.188    Mean   :230.7
## 3rd Qu.:24.25    3rd Qu.:22.80    3rd Qu.:8.000    3rd Qu.:326.0
## Max.    :32.00    Max.    :33.90    Max.    :8.000    Max.    :472.0
##           hp           drat           wt           qsec
## Min.      : 52.0    Min.   :2.760    Min.   :1.513    Min.   :14.50
## 1st Qu.:  96.5    1st Qu.:3.080    1st Qu.:2.581    1st Qu.:16.89
## Median :123.0    Median :3.695    Median :3.325    Median :17.71
## Mean   :146.7    Mean   :3.597    Mean   :3.217    Mean   :17.85
## 3rd Qu.:180.0    3rd Qu.:3.920    3rd Qu.:3.610    3rd Qu.:18.90
## Max.    :335.0    Max.    :4.930    Max.    :5.424    Max.    :22.90
##           vs           am           gear           carb           qualCat
## Min.      :0.0000    Min.   :0.0000    Min.   :3.000    Min.   :1.000    A: 5
## 1st Qu.:0.0000    1st Qu.:0.0000    1st Qu.:3.000    1st Qu.:2.000    B:27
## Median :0.0000    Median :0.0000    Median :4.000    Median :2.000
## Mean   :0.4375    Mean   :0.4062    Mean   :3.688    Mean   :2.812
## 3rd Qu.:1.0000    3rd Qu.:1.0000    3rd Qu.:4.000    3rd Qu.:4.000
## Max.    :1.0000    Max.    :1.0000    Max.    :5.000    Max.    :8.000
```

```
colnames(mycars) # Display Columns
```

```
## [1] "X"      "mpg"     "cyl"     "disp"     "hp"      "drat"     "wt"
## [8] "qsec"    "vs"      "am"      "gear"     "carb"    "qualCat"
```

```
dim(mycars) # Display total observations and total variables
```

```
## [1] 32 13
```

## Assignment 2

Write an R program for clustering the drinks.csv file using k-means clustering.

```
library(ggplot2)
library(gridExtra)
```

```
##
## Attaching package: 'gridExtra'
```

```
## The following object is masked from 'package:dplyr':
```

```
##
## combine
```

```
drinks <- read.csv(file = "~/manipal_practice/drinks.csv")
drinks <- na.omit(drinks)
cluster1 <- kmeans(drinks[,c(2,5)],4)
p1 <- ggplot(data = drinks, aes(x = beer_servings,
                                y = total_litres_of_pure_alcohol,
```

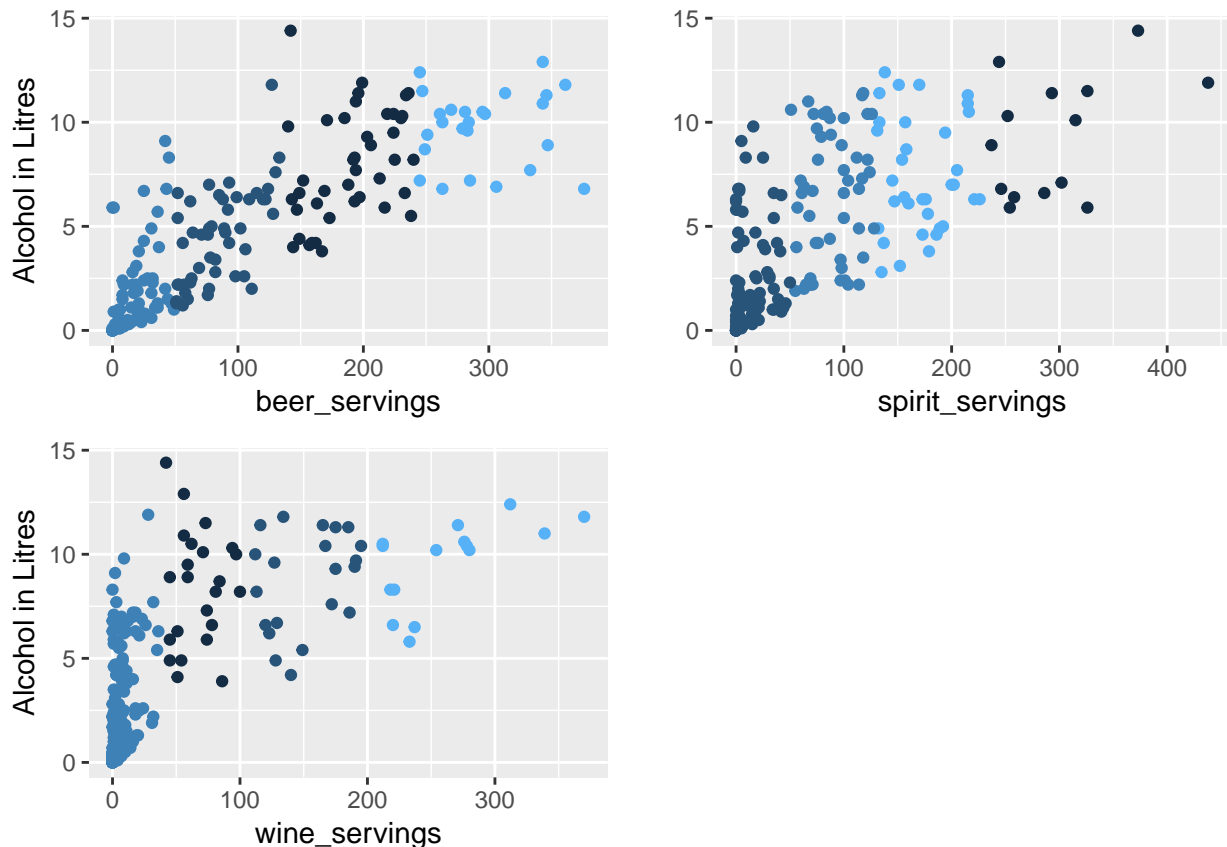
```

                                color = cluster1$cluster))
p1 <- p1 + geom_point() + ylab("Alcohol in Litres") + theme(legend.position="none")

cluster2 <- kmeans(drinks[,c(3,5)],4)
p2 <- ggplot(data = drinks, aes(x = spirit_servings,
                                y = total_litres_of_pure_alcohol,
                                color = cluster2$cluster))
p2 <- p2 + geom_point() + ylab("") + theme(legend.position="none")

cluster3 <- kmeans(drinks[,c(4,5)],4)
p3 <- ggplot(data = drinks, aes(x = wine_servings,
                                y = total_litres_of_pure_alcohol,
                                color = cluster3$cluster))
p3 <- p3 + geom_point() + ylab("Alcohol in Litres") + theme(legend.position="none")
grid.arrange(p1, p2, p3, ncol=2)

```



Also use the elbow method to infer the optimal value of K

```

maxk = 10
drinks <- data.matrix(drinks)

wssdist <- 1 #It may be any value which will be replaced with actual value later
wssdist[1:maxk] <- sapply(1:maxk, function(x,y){
  k <- kmeans(x,y)
  return(k$tot.withinss)
}, x=drinks)

```

```

totssdist <- 1 #It may be any value which will be replaced with actual value later
totssdist[1:maxk] <- sapply(1:maxk, function(x,y){
  k <- kmeans(x, y)
  return(k$totss)
}, x = drinks)

betweendist <- 1 #It may be any value which will be replaced with actual value later
betweendist[1:maxk] <- sapply(1:maxk, function(x,y){
  k <- kmeans(x, y)
  return(k$between)
}, x = drinks)

distdf <- data.frame(cluster = 1:maxk, wssdist = wssdist, totss = totssdist, between = betweendist)

g <- ggplot(data = distdf, aes(x = cluster, y = wssdist))
g <- g + geom_line(aes(color = "tot.withinss")) + geom_point()
g <- g + geom_line(aes(y = totss, color = "totss")) + geom_point(aes(y = totss))
g <- g + geom_line(aes(y = between, color = "between")) + geom_point(aes(y = between))
g <- g + ylab("Distortion") + xlab("Clusters")
g <- g + ggtitle("Elbow method Graph for Sum of Square(Within, Total, Between)")
g <- g + scale_x_continuous(breaks = 1:10)
g

```

