Analyzing and processing Big Data using R - Data Science Project

Team Members

- Ahmed Ayman Aabed - 2018170019
- Amany Salah Zahran - 2018170094
- Moumen Hamada Nagah - 2018170420

Team ID - 97

Review of Big Data Analytic Methods

Step 1: Retrieve and Clean Up Data using R

1. Analyze the zeta table (zeta.csv), which has data on households in different zip codes. Look at the column descriptions and record the column names.

    - Columns names:
        - zcta, sex, meanage, meaneducation, meanemployment, meanhouseholdincome

```
> # Read the zeta file
> zeta <- read.csv("./Data/zeta.csv")
>
> # Print the columns  names
> colnames(zeta)
[1] "zcta"                "sex"                 "meanage"
[4] "meaneducation"       "meanemployment"      "meanhouseholdincome"
```

2. How many rows of data are there in the zeta table?

    - 64076 Rows

```
> # print number of rows
> print(nrow(zeta))
[1] 64076
```

3. Are there any duplicate rows of data in the zeta table? If so, how can you tell?
    - No, there are no duplicate rows.
    - The length of the unique rows and the original rows are equal.

```
> # check if there any dups
> nrow(unique(zeta))== nrow(zeta)
[1] TRUE
```

4. If there are duplicates, make a new table called zeta_nodupes that has no duplicates. Now are there any duplicate rows of data? How can you tell?
    - There are no duplicate rows.
5. Save the table in a file named “zeta_nodupes.csv”
    - There are no duplicate rows.

Step 2: Data Analysis in R

1. Load the text file of income data (zipIncome.txt) into R.

```
> # Read zipincome.csv file
> zipincome = read.csv("./Data/zipIncome.csv")
> colnames(zipincome)
[1] "Zip1"                "MeanHouseholdIncome"
> head(zipincome)
  Zip1 MeanHouseholdIncome
1    0            19109.25
2    0            21591.53
3    0            20005.68
4    0            12013.34
5    0            19188.84
6    0            16687.42
```

2. Change the column names of your data frame so that zcta becomes zipCode and meanhouseholdincome becomes income.

```
> # Change zcta column name to zipCode and meanhouseholdincome to income, in zipincome dataframe
> colnames(zipincome) <- c("zipCode", "income")
> colnames(zipincome)
[1] "zipCode" "income"
```

3. Analyze the summary of your data. What are the mean and median average incomes?

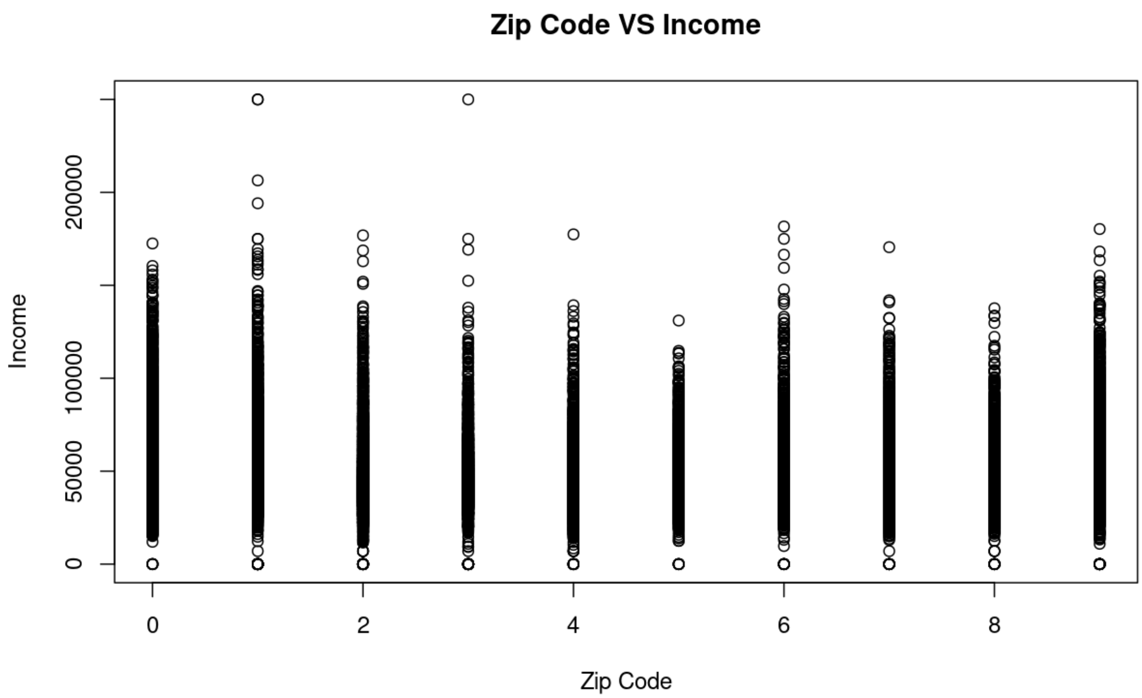
    - Average incomes mean : 48245
    - Average incomes median : 44163

```
> # Print the summary of the zeta table to analyze
> summary(zipincome)
    zipCode          income
 Min.   :0.000   Min.   :     0
 1st Qu.:2.000   1st Qu.: 37644
 Median :4.000   Median : 44163
 Mean   :4.473   Mean   : 48245
 3rd Qu.:7.000   3rd Qu.: 54373
 Max.   :9.000   Max.   :250000
```

4. Plot a scatter plot of the data. Although this graph is not too informative, do you see any outlier values? If so, what are they?

    - It appears that there are some outliers values above income $200,000 and near $0

```
> # Plot the zipcode against the income in a scatter plot
> plot(x = zeta$zipCode,
+      y = zeta$income,
+      xlab = "Zip Code",
+      ylab = "Income",
+      main = "Zip Code VS Income")
```



5. In order to omit outliers, create a subset of the data so that:
   $7,000 < income < $200,000

```
> # Ommit outliers, by limiting the income to 7000 > income > 200,000
> zipincome_omitted <- zipincome[(zipincome$income > 7000 & zipincome$income < 200000),]
> # Number of rows before ommitting outliers
> nrow(zipincome)
[1] 32038
> # Number of rows after ommitting outliers
> nrow(zipincome_omitted)
[1] 31871
```

6. What’s your new mean?

   - New Mean of Zeta$income : 48465
   - New Median of Zeta$income : 44234

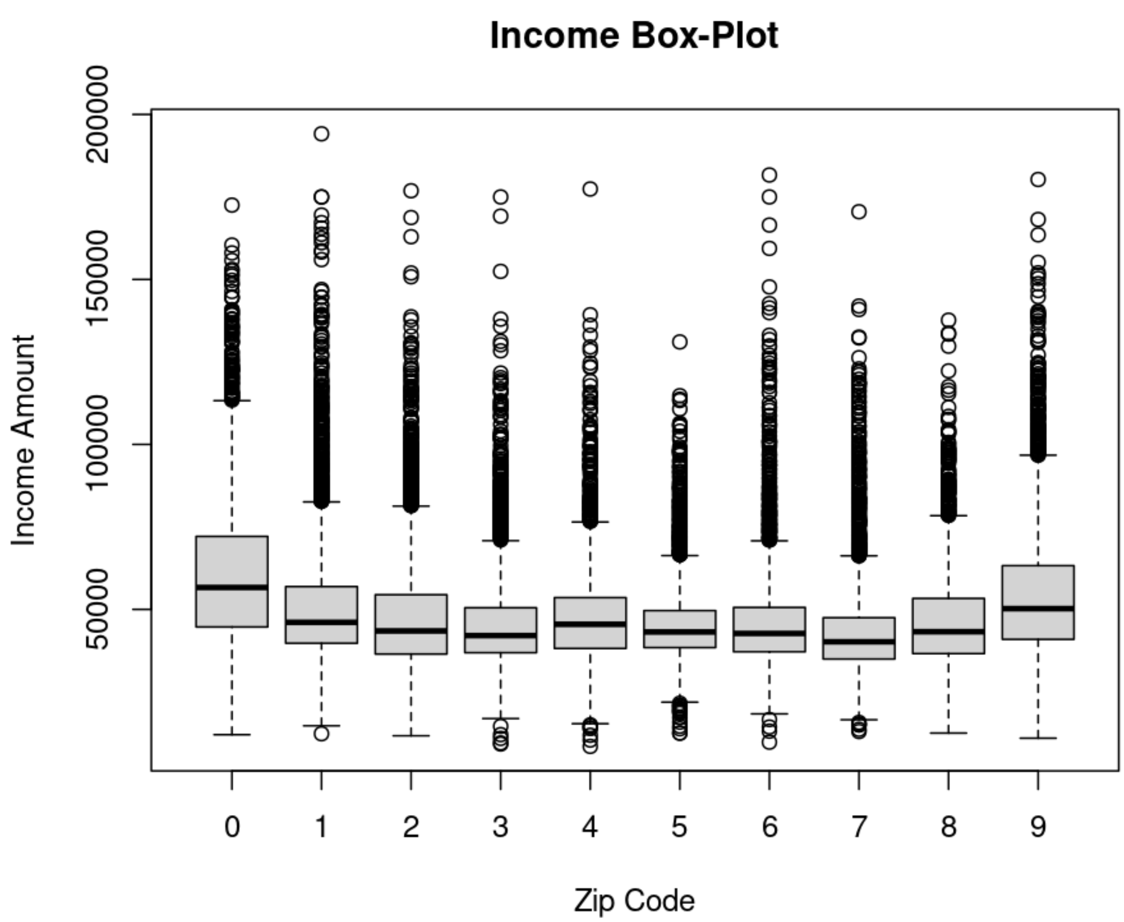
```
> # Print the summary of the zeta table after ommitting outliers
> summary(zipincome_omitted)
    zipCode          income
 Min.   :0.000   Min.   :  8465
 1st Qu.:2.000   1st Qu.: 37755
 Median :4.000   Median : 44234
 Mean   :4.474   Mean   : 48465
```

```
 3rd Qu.:7.000   3rd Qu.: 54444
 Max.   :9.000   Max.   :194135
```

Step 3: Visualize your data

1. Create a simple box plot of your data. Be sure to add a title and label

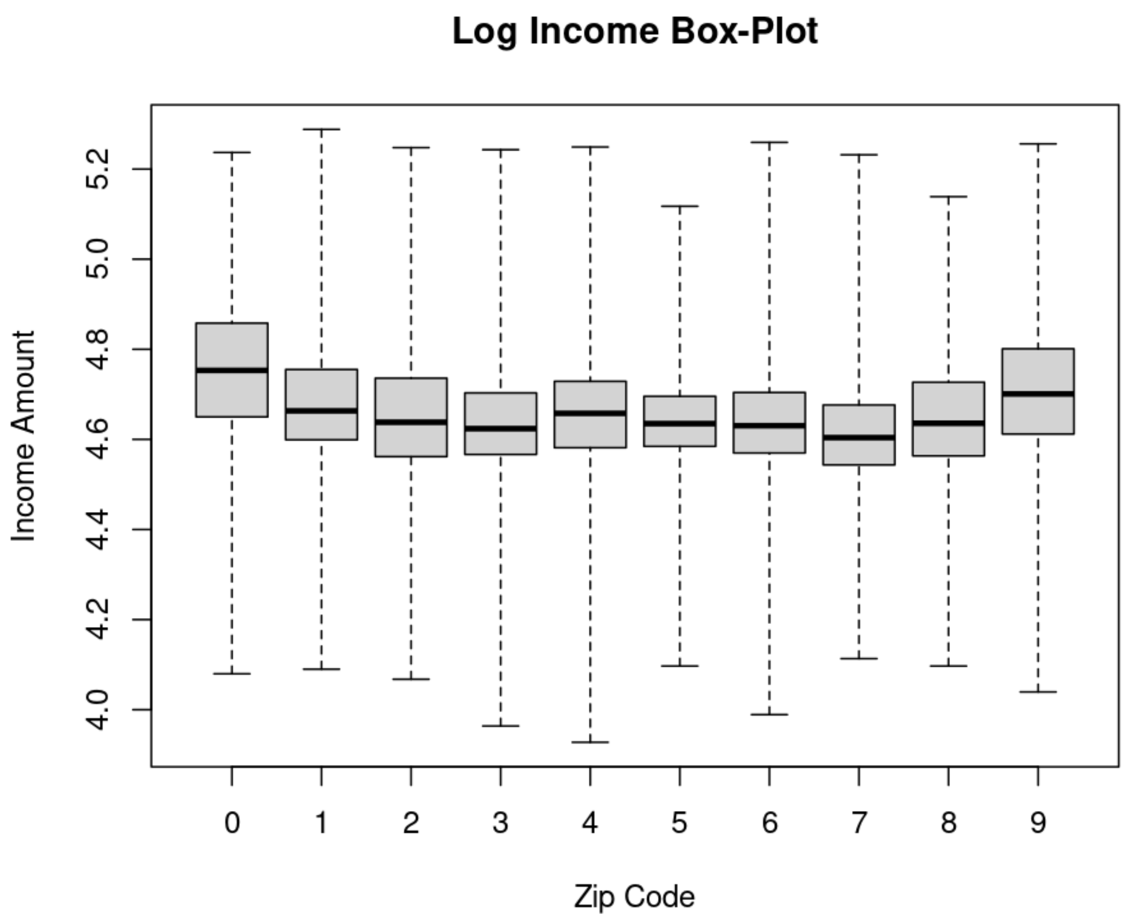
```
> # Box Plot for income
> boxplot(zipincome_omitted$income~as.factor(zipincome_omitted$zipCode),
+         data=zipincome_omitted,
+         xlab = "Zip Code",
+         ylab = "Income Amount",
+         main = "Income Box-Plot")
```



2. In the box plot you created, notice that all of the income data is pushed towards the bottom of the graph because most average incomes tend to be low. Create a new box plot where

the y-axis uses a log scale. Be sure to add a title and label the axes.

```
> logincome = log10(zipincome_omitted$income)
>
> # Box Plot for income log10
> boxplot(logincome~as.factor(zipincome_omitted$zipCode),
+         data=zipincome_omitted,
+         range=0,
+         xlab = "Zip Code",
+         ylab = "Income Amount",
+         main = "Log Income Box-Plot")
```



3. What can you conclude from this data analysis/visualization?.

   There are some outliers in the data, and the data after applying a log scale appears to be normally distributed. also the maximum income as below 200,000 and the minimum below 10,000.

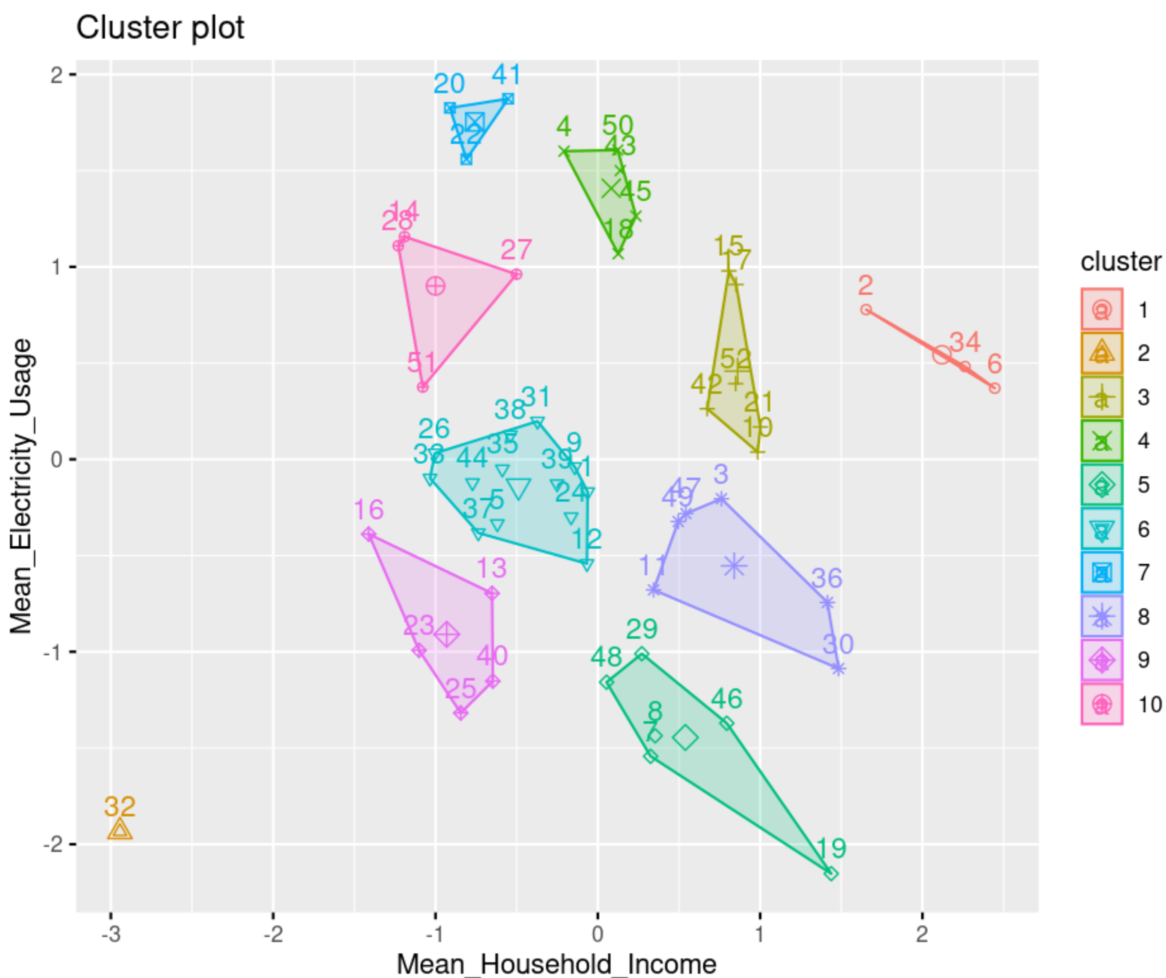
Advanced Analytics/Methods (K-means)

1. Access the census data saved as ‘income_elect_state.csv’ provided to you. Create a table with three columns: state, mean household income, and mean electricity usage.

```
> # Read income_elec_state.csv
> income_elec <- read.csv("./Data/income_elec_state.csv")
>
> # Change col names to State, Mean Household Income, Mean Electricity Usage
> colnames(income_elec) <- c("State", " Mean Household Income", "Mean Electricity Usage")
>
> head(income_elec)
  State  Mean Household Income Mean Electricity Usage
1    OH                  52516                    939
2    MD                  68760                   1099
3    IL                  60317                    933
4    NC                  51135                   1238
5    NE                  47244                    911
6    CT                  76265                   1030
```

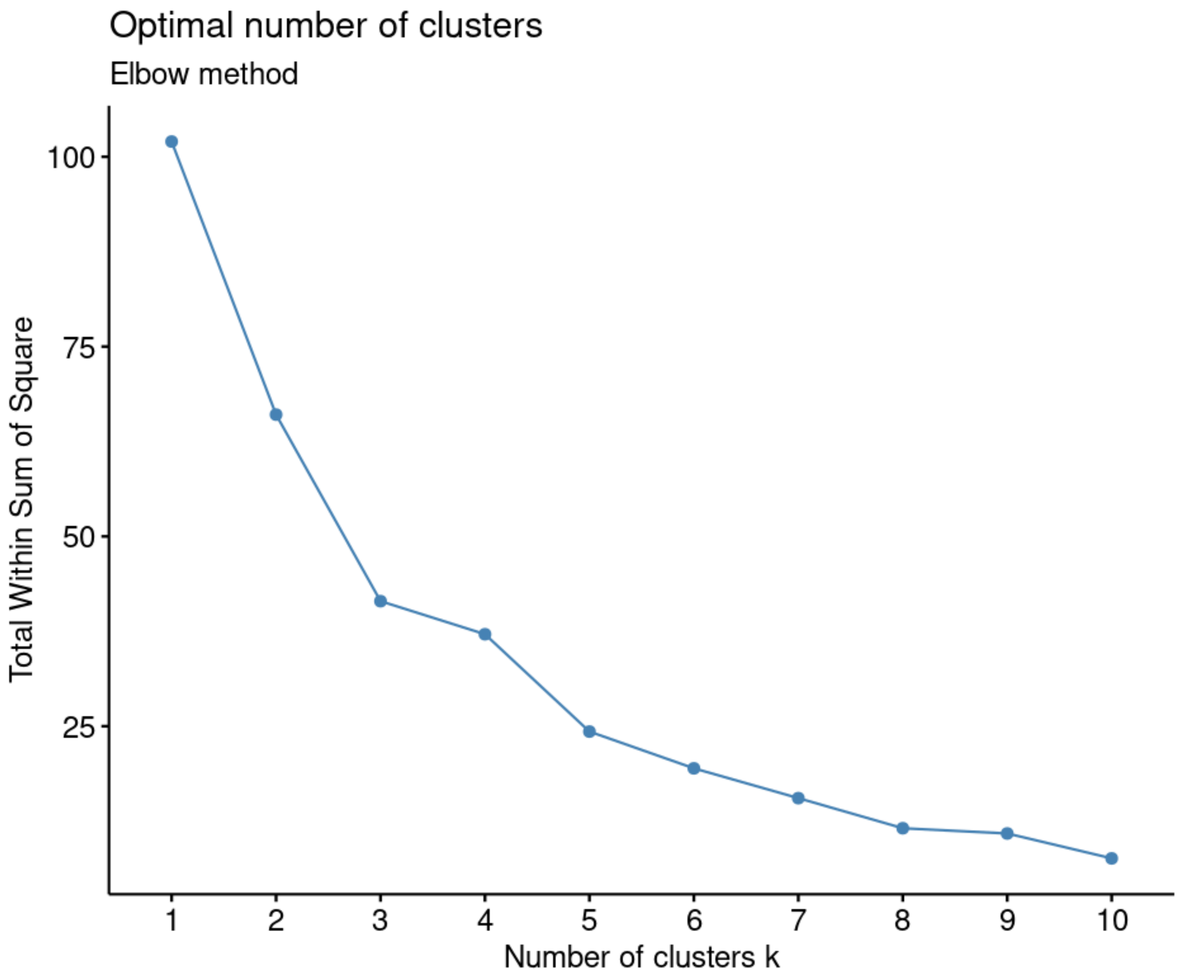
2. Cluster the data using k-means function and plot all 52 data points, along with the centroids. Mark all data points and centroids belonging to a given cluster with their own color. Here, let k=10

```
> # Read income_elec_state data and rename cols
> income_elec <- read.csv("./Data/income_elec_state.csv")
> colnames(income_elec) <- c("State", "Mean_Household_Income", "Mean_Electricity_Usage")
> head(income_elec)
  State Mean_Household_Income Mean_Electricity_Usage
1    OH                 52516                    939
2    MD                 68760                   1099
3    IL                 60317                    933
4    NC                 51135                   1238
5    NE                 47244                    911
6    CT                 76265                   1030
>
> # Take the Mean_Household_Income, and Mean_Electricity_Usage cols
> x <- income_elec[, 2:3]
> x_sc <- scale(x)
>
> # Calculate K-means with k = 10 and visualize the clusters alongside their centroids
> k_out <- kmeans(x_sc, 10, 25)
> fviz_cluster(object = list(data=x_sc,cluster=k_out$cluster))
```



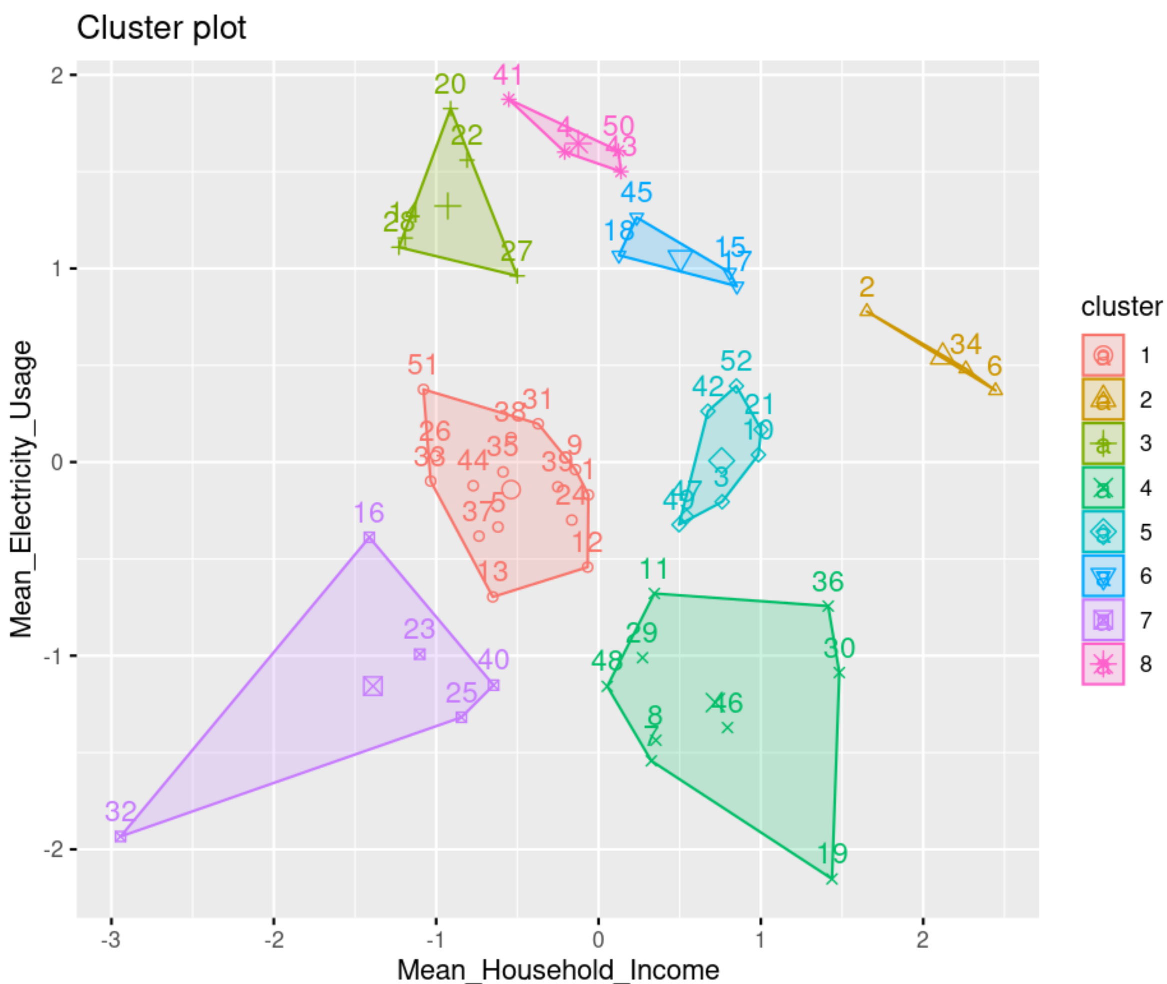
3. Determine a reasonable value of k using the “elbow” of the plot of the within-cluster sum of squares.

```
> # Visualize the Elbow to find a more suitable K value
> fviz_nbclust(x_sc, kmeans, method="wss") + labs(subtitle="Elbow method")
```



K = 8

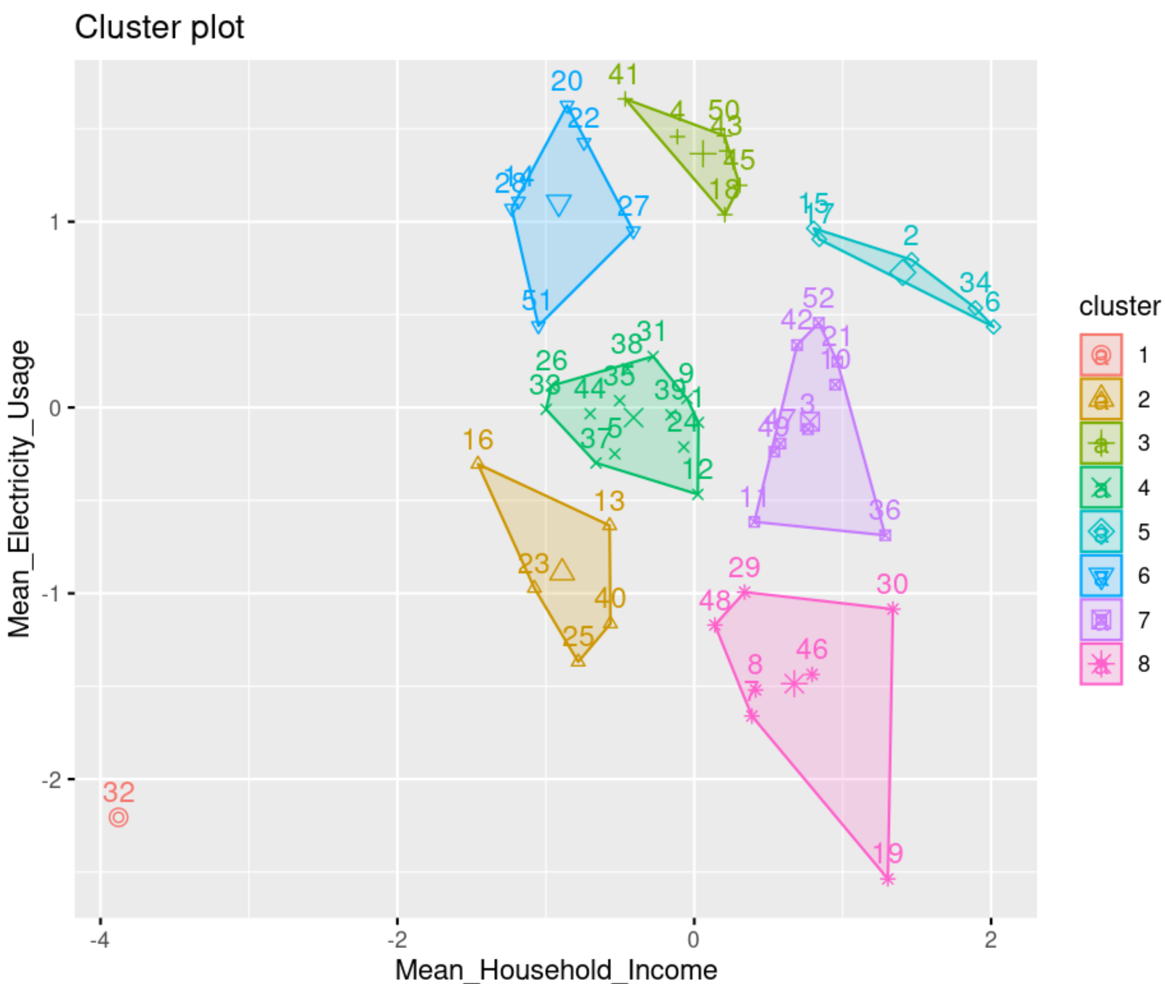
```
> # Calculate K-means with k = 5 and visualize the clusters alongside their centroids
> k_out_2 <- kmeans(x_sc, 8, 25)
> fviz_cluster(object = list(data=x_sc,cluster=k_out_2$cluster))
```



4. Convert the mean household income and mean electricity usage to a log10 scale and cluster this transformed dataset. How has the clustering changed? Why?

   The samples tend to be more normally distributed and classified into reasonable clusters, because the log10 operation stretched the data.

```
> # Convert the data to log10
> x_lg <- log10(x)
> x_lg_sc <- scale(x_lg)
>
> # Calculate K-means with k = 5 and visualize the clusters alongside their centroids
> k_out_3 <- kmeans(x_lg_sc, 8, 25)
> fviz_cluster(object = list(data=x_lg_sc,cluster=k_out_3$cluster))
```



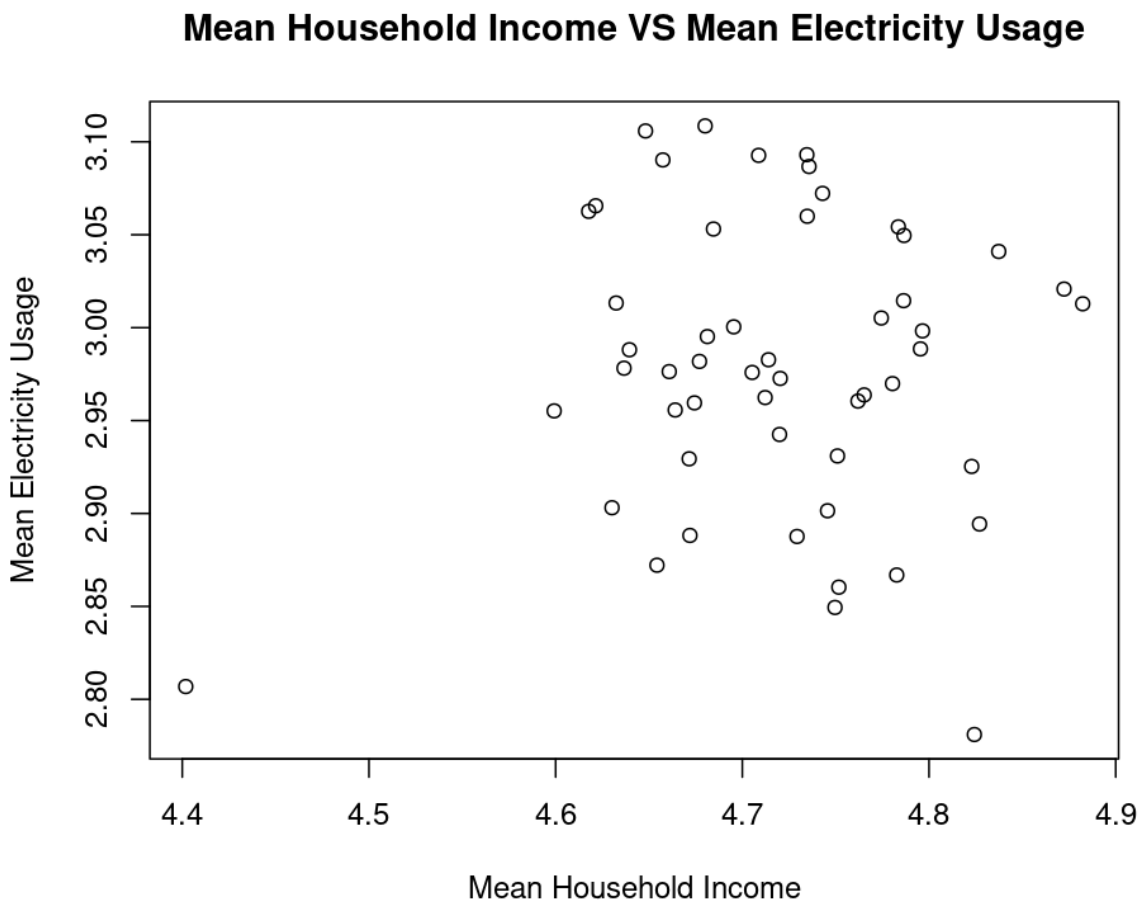
5. Reevaluate your choice of k. Would you now choose k differently? Why or why not?

   Yes, because there are outliers that distracts the clustering like point 32 and 19 in the above plot.

6. Have you observed an outlier in the data? Remove the outlier and, once again, reevaluate your choice of k.

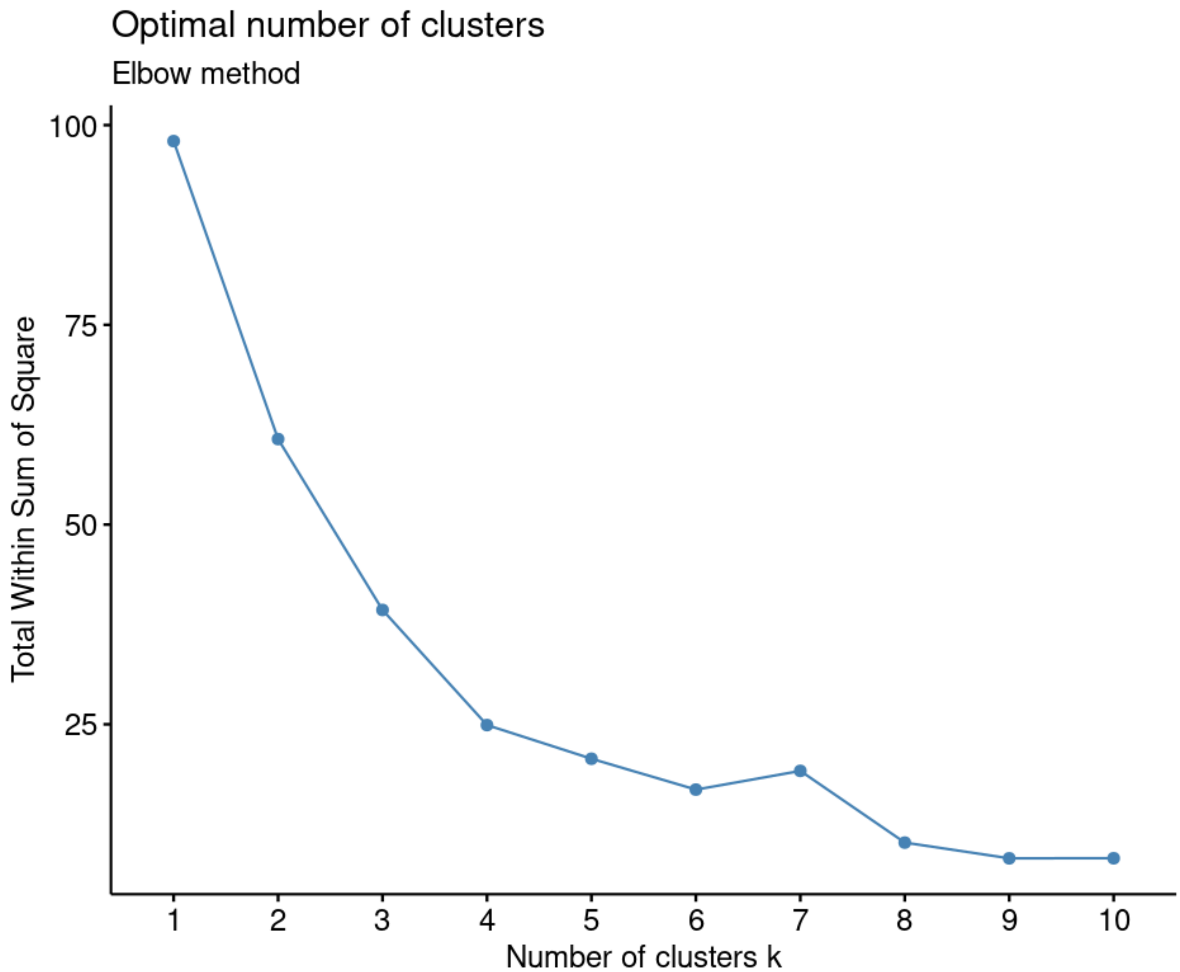
   - Find the outliers by plotting the data

```
> # Plot the data  to find the outliers
> plot(x = x_lg$Mean_Household_Income,
+      y = x_lg$Mean_Electricity_Usage,
+      xlab = "Mean Household Income",
+      ylab = "Mean Electricity Usage",
+      main = "Mean Household Income VS Mean Electricity Usage")
> # found outliers at Mean_Electricity_Usage > 2.83
```



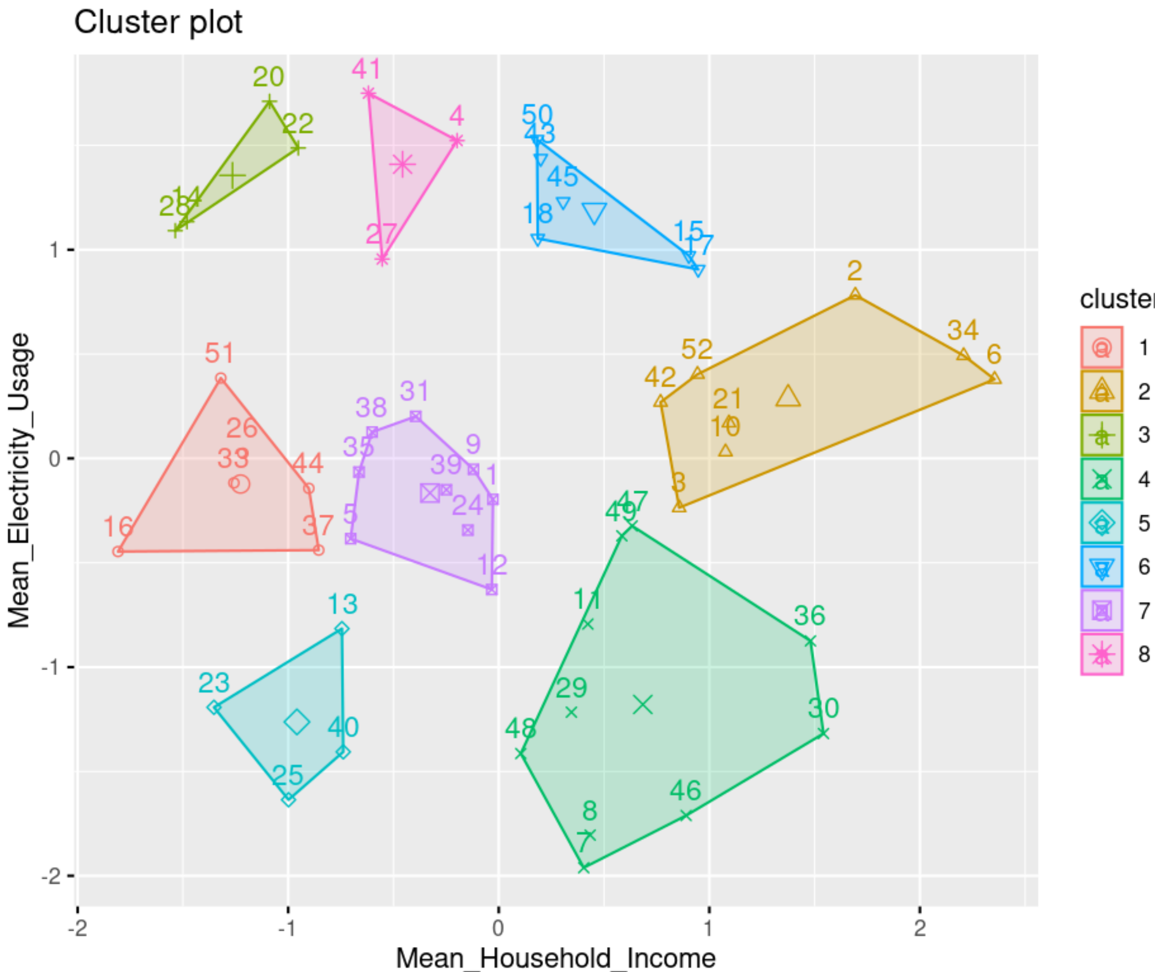
- Remove the outlier at Mean_Electricity_Usage > 2.83 and plot the Elbow to find the suitable K value

```
> x_lg_om <- x_lg[(x_lg$Mean_Electricity_Usage > 2.83) ,]
> x_lg_om_sc <- scale(x_lg_om)
> head(x_lg_om_sc)
  Mean_Household_Income Mean_Electricity_Usage
1           -0.02767118             -0.1963223
2            1.69332675              0.7821846
3            0.85673590             -0.2361881
4           -0.19784413              1.5228506
5           -0.70323814             -0.3845888
6            2.35484315              0.3789303
>
> # Visualize the Elbow to find a more suitable K value
> fviz_nbclust(x_lg_om_sc, kmeans, method="wss") + labs(subtitle="Elbow method")
> # found k = 8 to be more suitable
```



- Cluster the data using K = 8 and visualize the clusters

```
> # Calculate K-means with k = 8 and visualize the clusters alongside their centroids
> k_out_4 <- kmeans(x_lg_om_sc, 8, 25)
> fviz_cluster(object = list(data=x_lg_om_sc,cluster=k_out_4$cluster))
```


Cluster plot
Mean_Electricity_Usage
Mean_Household_Income
cluster
1
2
3
4
5
6
7
8
-2
-1
0
1
2