Class 5 - Cluster Analysis

K-Means

```r
stations = read.csv("G:/My Drive/Master-Data-Science/Semester_1/Business_Analytics/Data/Ch5_bike_statio
```

```r
two = kmeans(stations, 2)
two
```

```
## K-means clustering with 2 clusters of sizes 118, 126
##
## Cluster means:
##   latitude longitude
## 1 38.88838 -76.97846
## 2 38.93855 -77.03975
##
## Clustering vector:
##   [1] 2 1 2 1 2 2 1 1 1 1 2 1 1 1 2 2 2 2 2 1 1 2 2 1 2 1 2 2 2 1 1 1 1 2 2 2 2
##  [38] 1 2 2 1 2 2 2 1 2 1 2 1 2 1 2 1 1 1 1 1 1 2 1 2 2 2 2 1 1 1 2 1 2 1 2 2 2
##  [75] 1 2 1 2 1 2 1 2 2 2 1 2 1 2 1 1 2 1 2 1 1 2 2 1 2 2 1 1 2 2 1 2 2 2 2 2 2
## [112] 2 1 2 2 1 2 2 1 1 1 1 2 2 2 1 1 1 1 1 1 2 2 2 2 1 1 1 2 1 2 2 2 2 2 2 2 1
## [149] 2 1 2 2 1 2 2 1 2 1 1 1 1 1 1 1 2 1 2 2 2 1 2 2 2 1 2 1 1 2 2 2 1 1 1 2 1
## [186] 1 1 2 1 2 1 1 2 2 1 1 1 2 1 2 2 2 2 2 2 1 1 2 2 2 2 1 1 2 2 1 2 1 1 2 1 2
## [223] 1 1 2 1 1 2 1 1 2 1 2 2 1 1 1 2 1 2 1 1 1 1
##
## Within cluster sum of squares by cluster:
## [1] 0.1754263 0.1575802
##  (between_SS / total_SS =  53.4 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
## [6] "betweenss"    "size"         "iter"         "ifault"
```

```r
three = kmeans(stations, 3)
three
```

```
## K-means clustering with 3 clusters of sizes 94, 93, 57
##
## Cluster means:
##   latitude longitude
## 1 38.93765 -77.01089
## 2 38.87904 -76.97566
## 3 38.93327 -77.06502
```

```
##
## Clustering vector:
##   [1] 1 1 3 2 3 3 2 2 2 2 3 2 2 2 1 3 3 1 3 2 2 1 1 2 3 1 3 1 1 2 2 2 1 3 1 3 1
##  [38] 2 1 1 2 1 3 1 2 1 2 3 2 3 2 1 2 2 2 2 2 2 1 2 3 3 3 1 2 2 2 1 1 1 1 1 1 1
##  [75] 2 1 1 1 2 3 2 3 3 3 1 3 2 1 2 2 1 2 3 2 2 1 3 2 1 3 1 2 3 3 1 1 3 1 1 1 1
## [112] 1 2 1 3 2 1 1 2 1 2 1 3 3 3 2 2 2 2 2 1 1 3 3 1 2 2 2 3 1 1 1 1 3 1 3 3 1
## [149] 1 2 3 1 2 1 1 2 3 2 1 1 2 2 2 2 1 2 3 3 1 2 1 1 1 2 3 1 2 1 1 3 2 2 2 3 2
## [186] 2 2 3 2 1 2 2 3 1 2 2 2 1 1 1 3 1 3 3 1 2 1 3 1 1 1 2 2 1 1 2 3 1 2 1 1 3
## [223] 1 2 3 2 2 1 2 2 1 1 1 3 2 2 2 3 2 3 2 1 2 1
##
## Within cluster sum of squares by cluster:
## [1] 0.07588127 0.12261951 0.04715762
##  (between_SS / total_SS =  65.7 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
## [6] "betweenss"    "size"         "iter"         "ifault"
```

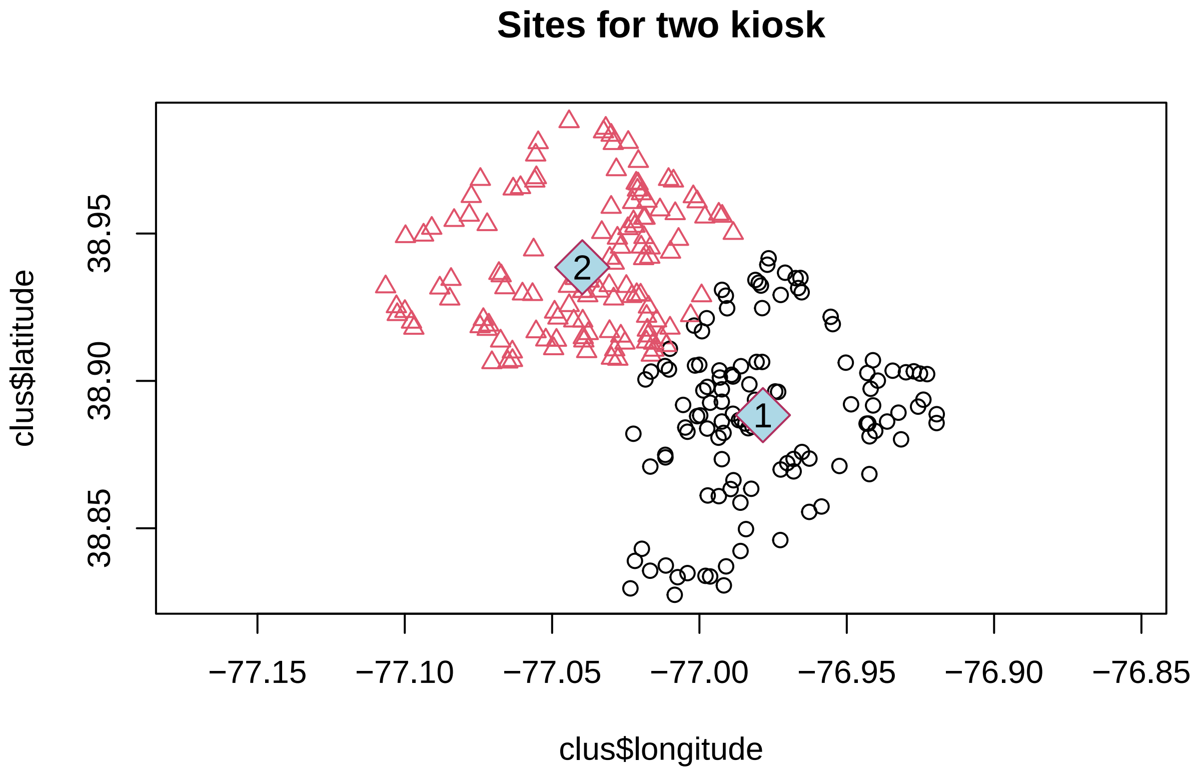
```
four = kmeans(stations, 4)
four
```

```
## K-means clustering with 4 clusters of sizes 32, 87, 70, 55
##
## Cluster means:
##   latitude longitude
## 1 38.90008 -76.94203
## 2 38.94000 -77.01424
## 3 38.87463 -76.99228
## 4 38.93235 -77.06589
##
## Clustering vector:
##   [1] 2 3 4 1 4 4 1 3 1 1 4 1 1 1 2 4 4 2 4 1 3 2 2 1 4 2 4 2 2 3 1 3 2 4 2 4 2
##  [38] 1 2 2 3 2 4 2 1 2 3 4 3 4 3 2 3 3 3 1 1 3 2 3 4 4 4 2 3 3 3 2 2 2 1 2 2 2
##  [75] 3 2 2 2 3 4 1 4 4 4 2 4 3 2 3 3 2 1 4 3 3 2 4 3 2 4 2 3 4 2 2 2 4 2 2 2 2
## [112] 2 3 2 4 3 2 2 3 1 1 2 4 4 4 3 3 3 3 3 2 2 4 4 2 3 1 3 4 2 2 2 2 4 2 4 4 2
## [149] 2 3 4 2 3 2 2 3 4 1 2 2 1 3 3 3 2 1 4 4 2 3 2 2 2 1 4 3 3 2 2 4 3 3 3 4 3
## [186] 3 3 4 3 2 3 3 4 2 3 1 3 2 2 2 4 2 4 4 2 3 3 4 2 2 2 3 3 2 2 1 4 1 3 2 3 4
## [223] 1 3 2 3 3 2 3 1 2 2 2 4 1 3 1 4 3 4 3 2 3 1
##
## Within cluster sum of squares by cluster:
## [1] 0.01710919 0.06435667 0.05568070 0.04289033
##  (between_SS / total_SS =  74.8 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
## [6] "betweenss"    "size"         "iter"         "ifault"
```

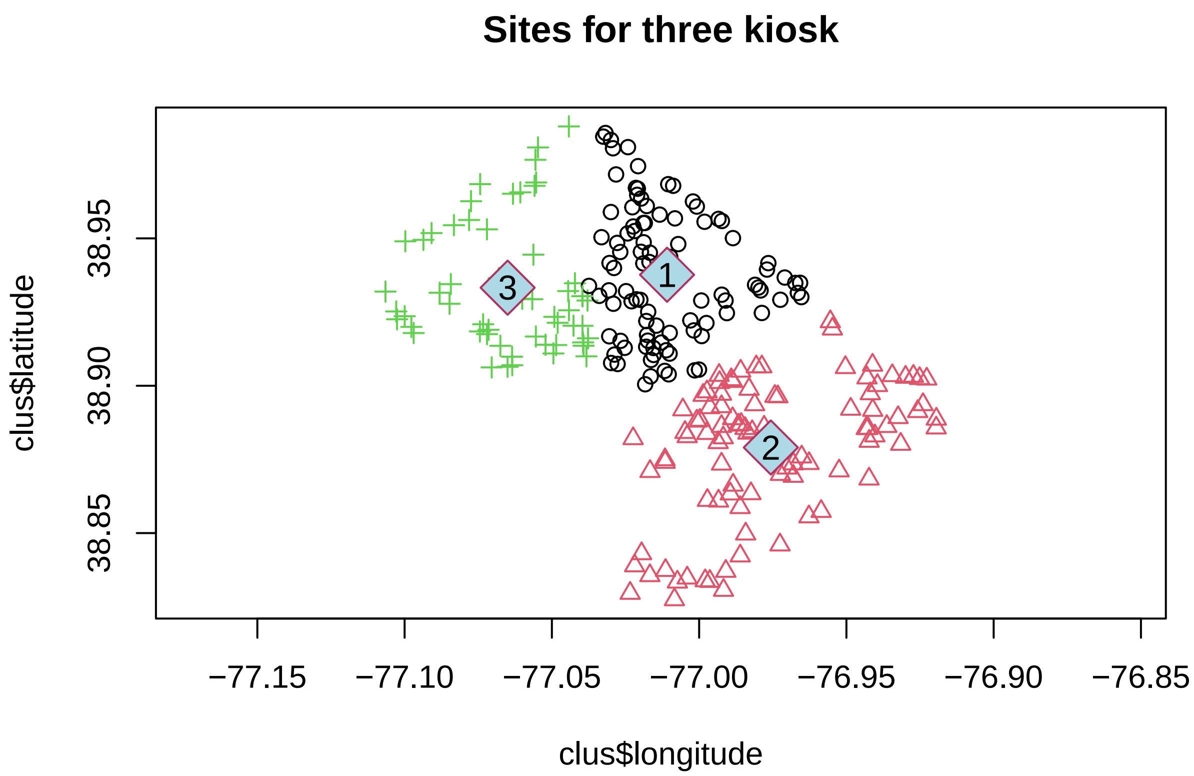
```
clus = cbind(stations, clu2 = two$cluster, clu3 = three$cluster)
head(clus)
```

```
##   latitude longitude clu2 clu3
## 1 38.95659 -76.99344    2    1
## 2 38.90522 -77.00150    1    1
## 3 38.98086 -77.05472    2    3
## 4 38.90293 -76.92991    1    2
## 5 38.94950 -77.09362    2    3
## 6 38.92780 -77.08474    2    3
```

```
plot(clus$longitude, clus$latitude, col = two$cluster, asp = 1, pch = two$cluster, main = 'Sites for tw
points(two$centers[,2], two$centers[,1], pch = 23, col = 'maroon', bg = 'lightblue', cex=3)
text(two$centers[,2], two$centers[,1], cex = 1.1, col = 'black', attributes(two$centers)$dimnames[[1]])
```



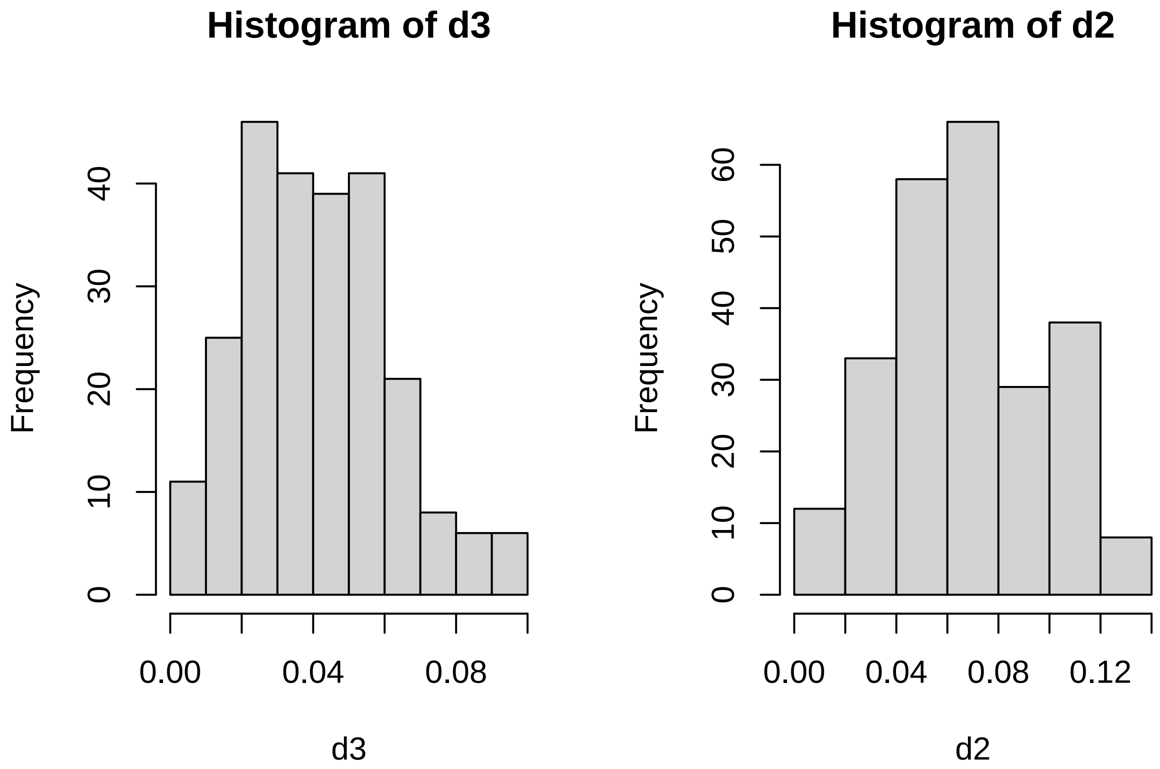
```
plot(clus$longitude, clus$latitude, col = three$cluster, asp = 1, pch = three$cluster, main = 'Sites fo
points(three$centers[,2], three$centers[,1], pch = 23, col = 'maroon', bg = 'lightblue', cex=3)
text(three$centers[,2], three$centers[,1], cex = 1.1, col = 'black', attributes(three$centers)$dimnames
```



```
d3 = rep(0,244)
for (i in 1:244) {
  if (three$cluster[i] == 1) {
    d3[i] = sqrt((clus$latitude[i] - three$centers[1,1])^2 + (clus$longitude[i] - three$centers[1,2])^2
  }
  if (three$cluster[i] == 2) {
    d3[i] = sqrt((clus$latitude[i] - three$centers[2,1])^2 + (clus$longitude[i] - three$centers[2,2])^2
  }
  else {
    d3[i] = sqrt((clus$latitude[i] - three$centers[3,1])^2 + (clus$longitude[i] - three$centers[3,2])^2
  }
}

d2 = rep(0,244)
for (i in 1:244) {
  if (three$cluster[i] == 1) {
    d2[i] = sqrt((clus$latitude[i] - two$centers[1,1])^2 + (clus$longitude[i] - two$centers[1,2])^2)
  }
  else {
    d2[i] = sqrt((clus$latitude[i] - two$centers[2,1])^2 + (clus$longitude[i] - two$centers[2,2])^2)
  }
}

par(mfrow = c(1,2))
hist(d3)
hist(d2)
```



```
test = data.frame(
  measure = c('mean distance', 'maximum distance'),
  '2-cluster' = c(mean(d2), max(d2)),
  '3-cluster' = c(mean(d3), max(d3))
)
test
```

```
##            measure X2.cluster X3.cluster
## 1    mean distance 0.06739649 0.04149592
## 2 maximum distance 0.13133490 0.09976565
```

Scaling Data

```
market = read.csv("G:/My Drive/Master-Data-Science/Semester_1/Business_Analytics/Data/Ch5_age_income_da
head(market)
```

```
##     bin age   income
## 1 60-69  64 87083.24
## 2 30-39  33 76807.82
## 3 20-29  24 12043.60
## 4 30-39  33 61972.00
## 5 70-79  78 60120.32
## 6 60-69  62 40058.42
```

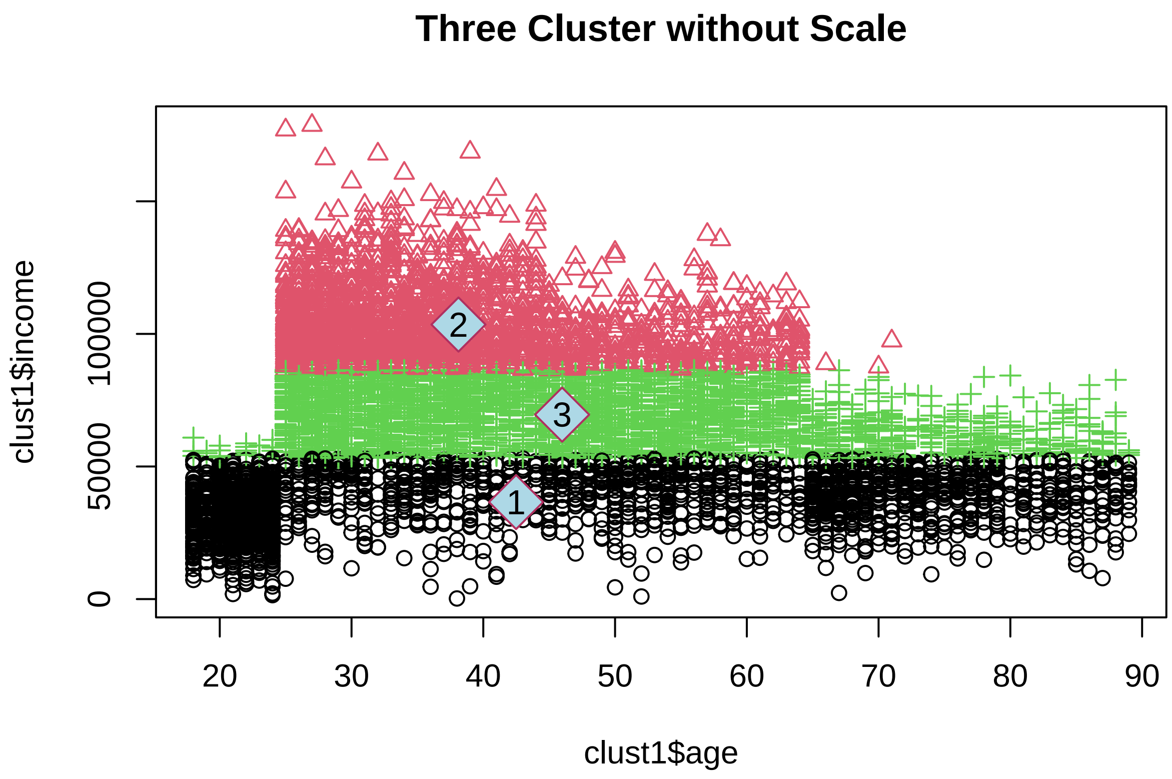
```
summary(market)
```

```
##      bin                  age            income
## Length:8105        Min.   :18.00   Min.   :   233.6
## Class :character   1st Qu.:28.00   1st Qu.: 43792.7
## Mode  :character   Median :39.00   Median : 65060.0
##                    Mean   :42.85   Mean   : 66223.6
##                    3rd Qu.:55.00   3rd Qu.: 85944.7
##                    Max.   :89.00   Max.   :178676.4
```

```
test1 = kmeans(market[,c(2,3)], 3)

clust1 = cbind(market, clu2 = test1$cluster)

plot(clust1$age, clust1$income, col = test1$cluster, pch = test1$cluster, main= 'Three Cluster without S
points(test1$centers[,1], test1$centers[,2], pch = 23, col = 'maroon', bg = 'lightblue', cex = 3)
text(test1$centers[,1], test1$centers[,2], cex = 1.1, col = 'black', attributes(test1$centers)$dimnames
```



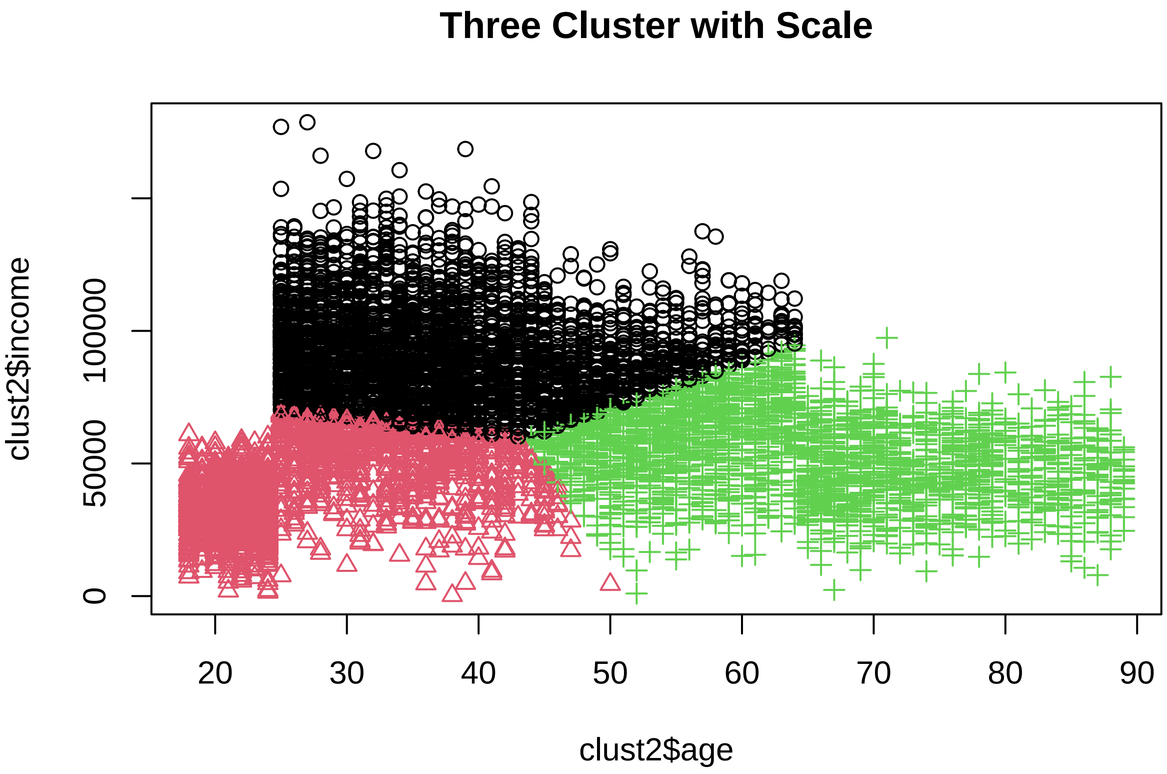
```
market$age_scale = as.numeric(scale(market$age))
market$income_scale = as.numeric(scale(market$income))
head(market,10)
```

```
##      bin age   income  age_scale income_scale
## 1  60-69  64 87083.24  1.2071346   0.75070999
## 2  30-39   33 76807.82 -0.5619790   0.38091240
## 3  20-29   24 12043.60 -1.0755926  -1.94986082
## 4  30-39   33 61972.00 -0.5619790  -0.15300793
## 5  70-79   78 60120.32  2.0060891  -0.21964755
## 6  60-69   62 40058.42  1.0929982  -0.94164698
## 7    80-   88 38850.72  2.5767709  -0.98511016
## 8  50-59   54 65239.05  0.6364528  -0.03543151
## 9  50-59   54 51362.92  0.6364528  -0.53481378
## 10 30-39   31 36418.25 -0.6761154  -1.07265161
```

```
test2 = kmeans(market[,c(4,5)], 3)

clust2 = cbind(market, clu2 = test2$cluster)

plot(clust2$age, clust2$income, col = test2$cluster, pch = test2$cluster, main= 'Three Cluster with Scal
points(test2$centers[,1], test2$centers[,2], pch = 23, col = 'maroon', bg = 'lightblue', cex = 3)
text(test2$centers[,1], test2$centers[,2], cex = 1.1, col = 'black', attributes(test2$centers)$dimnames
```



```
test2$centers
```

```
##    age_scale income_scale
## 1 -0.2813449    0.9080015
## 2 -0.9374135   -0.9586354
## 3  1.2165996   -0.4590239
```

Unscale data

```r
#unscale each of the scaled values in the scaled_data vector
#scaled_data * attr(scaled_data, 'scaled:scale') + attr(scaled_data, 'scaled:center')
```

Hierarchical Techniques

```r
set.seed(456)
hc_mod = hclust(dist(market[,4:5]), method = 'ward.D2')
hc_mod
```

```
##
## Call:
## hclust(d = dist(market[, 4:5]), method = "ward.D2")
##
## Cluster method   : ward.D2
## Distance         : euclidean
## Number of objects: 8105
```

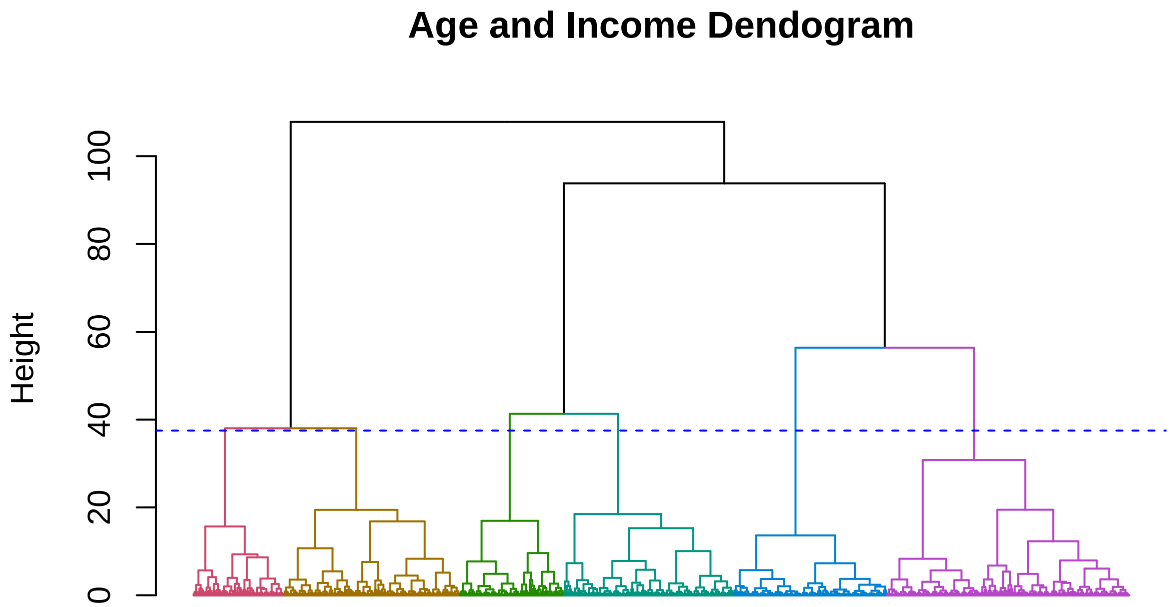
```r
library(dendextend)
```

```
##
## ---------------------
## Welcome to dendextend version 1.19.0
## Type citation(’dendextend’) for how to cite the package.
##
## Type browseVignettes(package = ’dendextend’) for the package vignette.
## The github page is: https://github.com/talgalili/dendextend/
##
## Suggestions and bug-reports can be submitted at: https://github.com/talgalili/dendextend/issues
## You may ask questions at stackoverflow, use the r and dendextend tags:
##   https://stackoverflow.com/questions/tagged/dendextend
##
##  To suppress this message use:  suppressPackageStartupMessages(library(dendextend))
## ---------------------
```

```
##
## Attaching package: ’dendextend’
```

```
## The following object is masked from ’package:stats’:
##
##     cutree
```

```r
dend = as.dendrogram(hc_mod)
dend_six_color = color_branches(dend, k = 6)
plot(dend_six_color, leaflab = "none", horiz = F,
     main = 'Age and Income Dendogram', ylab = 'Height')
abline(h = 37.5, lty= 'dashed', col = 'blue')
```



```
str(cut(dend, h =37.5)$upper)
```

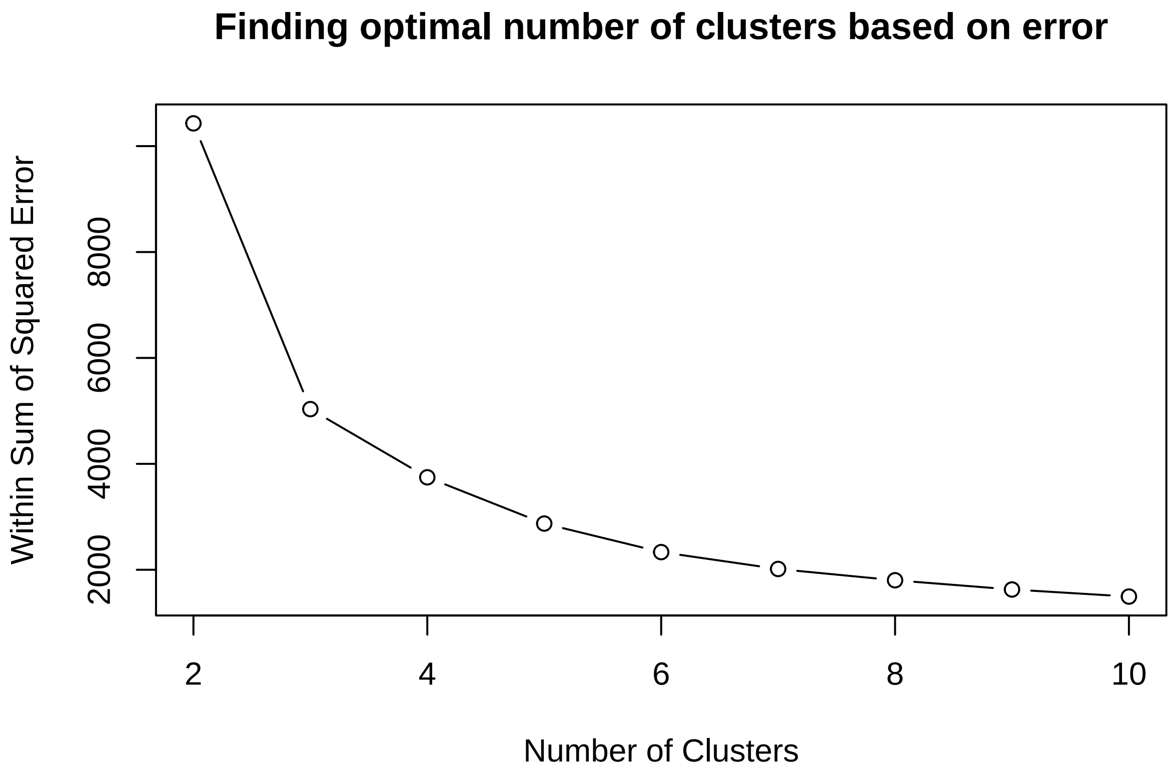
```
## --[dendrogram w/ 2 branches and 6 members at h = 108]
##   |--[dendrogram w/ 2 branches and 2 members at h = 38]
##   |  |--leaf "Branch 1" (h= 15.7 midpoint = 274, x.member = 782 )
##   |  `--leaf "Branch 2" (h= 19.5 midpoint = 628, x.member = 1526 )
##   `--[dendrogram w/ 2 branches and 4 members at h = 93.8]
##      |--[dendrogram w/ 2 branches and 2 members at h = 41.3]
##      |  |--leaf "Branch 3" (h= 17 midpoint = 431, x.member = 905 )
##      |  `--leaf "Branch 4" (h= 18.5 midpoint = 463, x.member = 1473 )
##      `--[dendrogram w/ 2 branches and 2 members at h = 56.4]
##         |--leaf "Branch 5" (h= 13.6 midpoint = 530, x.member = 1323 )
##         `--leaf "Branch 6" (h= 30.8 midpoint = 753, x.member = 2096 )
```

Evaluating models

```
optimize = data.frame(
  clusters = c(2:10),
  wss = rep(0,9)
)

for (i in seq(2, 10, by=1)) {
  x = kmeans(market[,4:5], i)
  optimize[i-1,2] = as.numeric(x$tot.withinss)
}

plot(optimize$wss ~ optimize$clusters, type = 'b',
     main = 'Finding optimal number of clusters based on error',
     xlab = 'Number of Clusters',
     ylab = 'Within Sum of Squared Error')
```



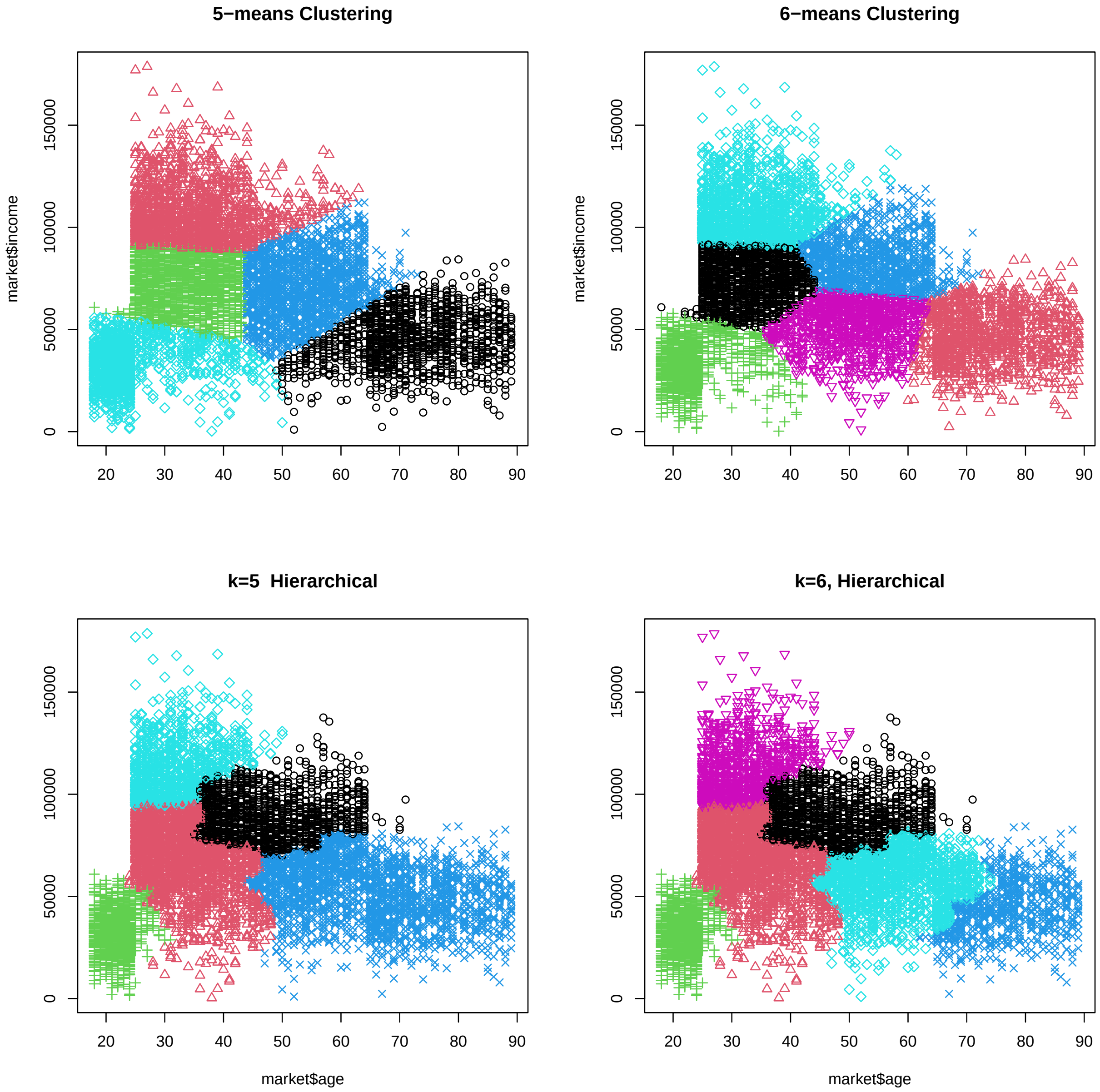
```
# Cluster 5
five = kmeans(market[,4:5], 5)

market$clus5 = five$cluster
dend_five = cutree(dend, k = 5)
market$dend5 = dend_five

# Cluster 6
six = kmeans(market[,4:5], 6)

market$clus6 = six$cluster
dend_six = cutree(dend, k = 6)
market$dend6 = dend_six
```

```
par(mfrow = c(2,2))
plot(market$age, market$income, col=five$cluster,
     pch = five$cluster, xlab='', main='5-means Clustering')
plot(market$age, market$income, col=six$cluster,
     pch = six$cluster, xlab='', main = '6-means Clustering')
plot(market$age, market$income, col=market$dend5,
     pch = market$dend5, ylab='', main='k=5  Hierarchical')
plot(market$age, market$income, col=market$dend6,
     pch=market$dend6, ylab='', main='k=6, Hierarchical')
```

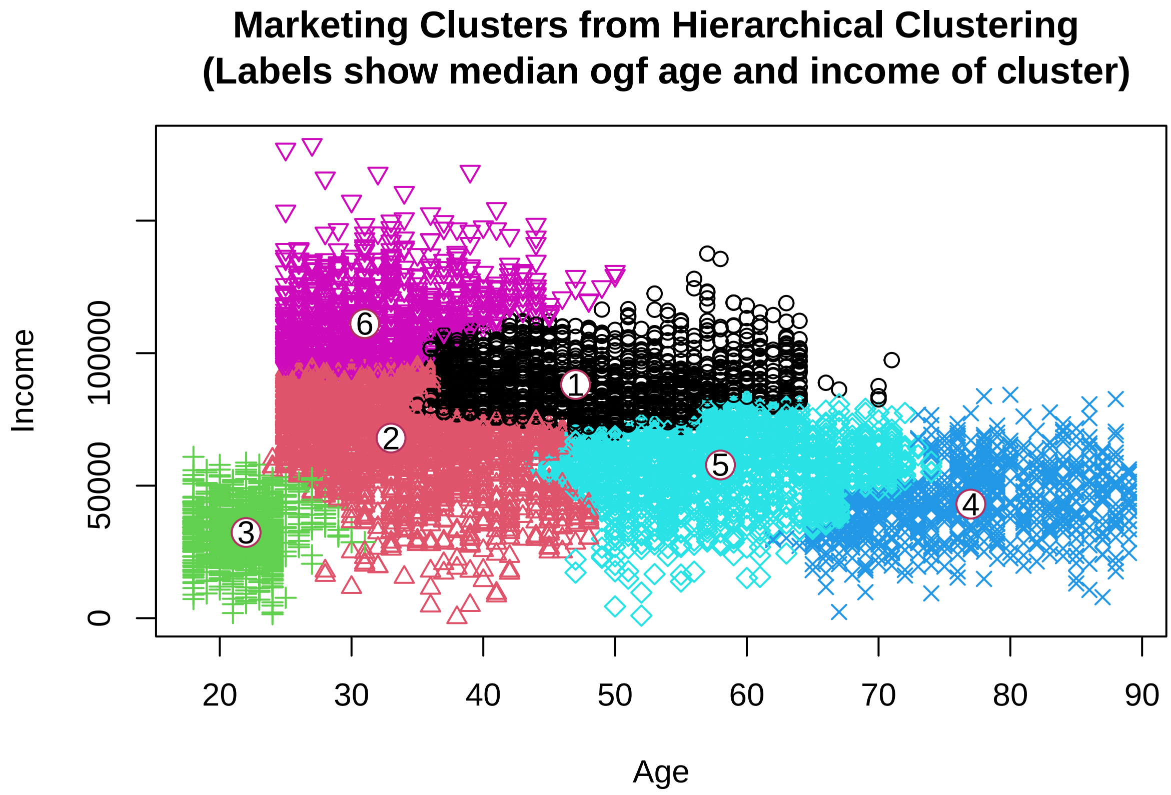


Showing median with labels

```
labels = as.data.frame(market %>%
                         group_by(dend6) %>%
                         summarise(
                           avg_age = median(age),
                           avg_inc = median(income)
                           )
                       )
labels
```

```
##   dend6 avg_age   avg_inc
## 1     1      47  88170.32
## 2     2      33  67957.66
## 3     3      22  32329.49
## 4     4      77  43044.21
## 5     5      58  57806.34
## 6     6      31 111124.93
```

```
plot(market$age, market$income, col = market$dend6,
     pch = market$dend6, xlab='Age', ylab='Income',
     main ='Marketing Clusters from Hierarchical Clustering \n (Labels show median ogf age and income o
points(labels[,2], labels[,3], pch = 21, col='maroon', bg='white', cex = 2)
text(labels[,2], labels[,3], col ='black', cex = 1, labels[,1])
```



```r
market %>% group_by(dend6) %>% summarise(Clustersize = n())
```

```
## # A tibble: 6 x 2
##   dend6 Clustersize
##   <int>       <int>
## 1     1        1473
## 2     2        2096
## 3     3        1323
## 4     4         782
## 5     5        1526
## 6     6         905
```

```r
market %>% group_by(dend6) %>%
  summarise(
    min_age = min(age),
    med_age = median(age),
    max_age = max(age),
    min_inc = min(income),
    med_inc = median(income),
    max_inc = max(income)
  )
```

```
## # A tibble: 6 x 7
##   dend6 min_age med_age max_age min_inc med_inc max_inc
##   <int>   <int>   <dbl>   <int>   <dbl>   <dbl>   <dbl>
## 1     1      35      47      71  69492.  88170. 137557.
## 2     2      24      33      48    234.  67958.  94709.
## 3     3      18      22      31   1485.  32329.  60887.
## 4     4      62      77      89   2319.  43044.  84301.
## 5     5      44      58      74    973.  57806.  81988.
## 6     6      25      31      50  93827. 111125. 178676.
```

```r
custom_labels = c(
  "Professionals",
  "Juniors",
  "Fresh Grads",
  "Pensioners",
  "Old Gov Servants",
  "High Achievers"
)

market %>% group_by(dend6) %>%
  summarise(
    Age_Range = paste(min(age),'-',max(age)),
    Age_Median = median(age),
    Income_range = paste(round(min(income),2),'-',round(max(income),2)),
    Income_Median = median(income)
  ) %>%
  mutate(Label = custom_labels)
```

```
## # A tibble: 6 x 6
##   dend6 Age_Range Age_Median Income_range         Income_Median Label
##   <int> <chr>          <dbl> <chr>                        <dbl> <chr>
## 1     1 35 - 71           47 69491.78 - 137557.18        88170. Professionals
## 2     2 24 - 48           33 233.63 - 94708.92           67958. Juniors
## 3     3 18 - 31           22 1484.85 - 60887.37          32329. Fresh Grads
## 4     4 62 - 89           77 2319.27 - 84300.56          43044. Pensioners
## 5     5 44 - 74           58 973.41 - 81988.14           57806. Old Gov Servants
## 6     6 25 - 50           31 93826.66 - 178676.37       111125. High Achievers
```