Code

ANLY_699_Assignment5

Subhash Pemmaraju

July 12, 2020

Variables for Cluster Analysis

Hide

```
clust_data <- merged_data[, c(10:27)]
clust_data1 <- clust_data[complete.cases(clust_data),]
#dim(fa_data1)
str(clust_data1)
```

```
## 'data.frame':    3120 obs. of  18 variables:
##  $ perc_fair_poor_health: int  21 18 30 19 22 31 28 23 24 21 ...
##  $ avg_phy_unh_days     : num  4.7 4.2 5.4 4.6 4.9 5.4 5.4 4.9 4.9 4.8 ...
##  $ avg_mental_unh_days  : num  4.7 4.3 5.2 4.6 4.9 4.9 5.3 4.8 4.9 4.7 ...
##  $ perc_smokers         : int  18 17 22 19 19 23 22 21 19 17 ...
##  $ perc_obese           : int  33 31 42 38 34 37 43 39 40 35 ...
##  $ food_env_ind         : num  7.2 8 5.6 7.8 8.4 4.3 6.6 6.9 6.4 8.3 ...
##  $ perc_phy_inact       : int  35 27 24 34 30 25 40 32 30 31 ...
##  $ perc_excess_drink    : num  15 18 12.8 15.6 14.2 ...
##  $ perc_uninsured       : int  9 11 12 10 13 11 11 12 12 11 ...
##  $ perc_college         : int  62 67 35 44 53 35 42 59 48 52 ...
##  $ perc_unemp           : num  3.6 3.6 5.2 4 3.5 4.7 4.8 4.7 3.9 3.6 ...
##  $ perc_child_pov       : int  19 14 44 28 18 68 36 27 31 25 ...
##  $ perc_diabetes        : int  11 11 18 15 17 24 19 18 20 15 ...
##  $ median_income        : int  59338 57588 34382 46064 50412 29267 37365 45400 39917 42132
...
##  $ perc_65up            : num  15.6 20.4 19.4 16.5 18.2 16.4 20.3 17.7 19.5 23 ...
##  $ perc_black           : num  19.3 8.8 48 21.1 1.5 69.5 44.6 20.9 39.6 4.2 ...
##  $ perc_female          : num  51.4 51.5 47.2 46.8 50.7 45.5 53.4 51.9 52.1 50.5 ...
##  $ perc_18less          : num  23.7 21.6 20.9 20.5 23.2 21.1 22.2 21.6 20.8 19.2 ...
```

Hide

```
#names(fa_data1)
```

The 18 variables above are used to identify clusters. The categories of the variables are broadly in 3 categories:

- Demographic Data (Age, Race, Gender)
- Socio-economic Data (Education, income, unemployment, insurane coverage, poverty etc.)
- Health Data (Obesity, Smoking, physical activity, mental health, drinking etc.)

Optimal number of clusters and K-Means Clustering

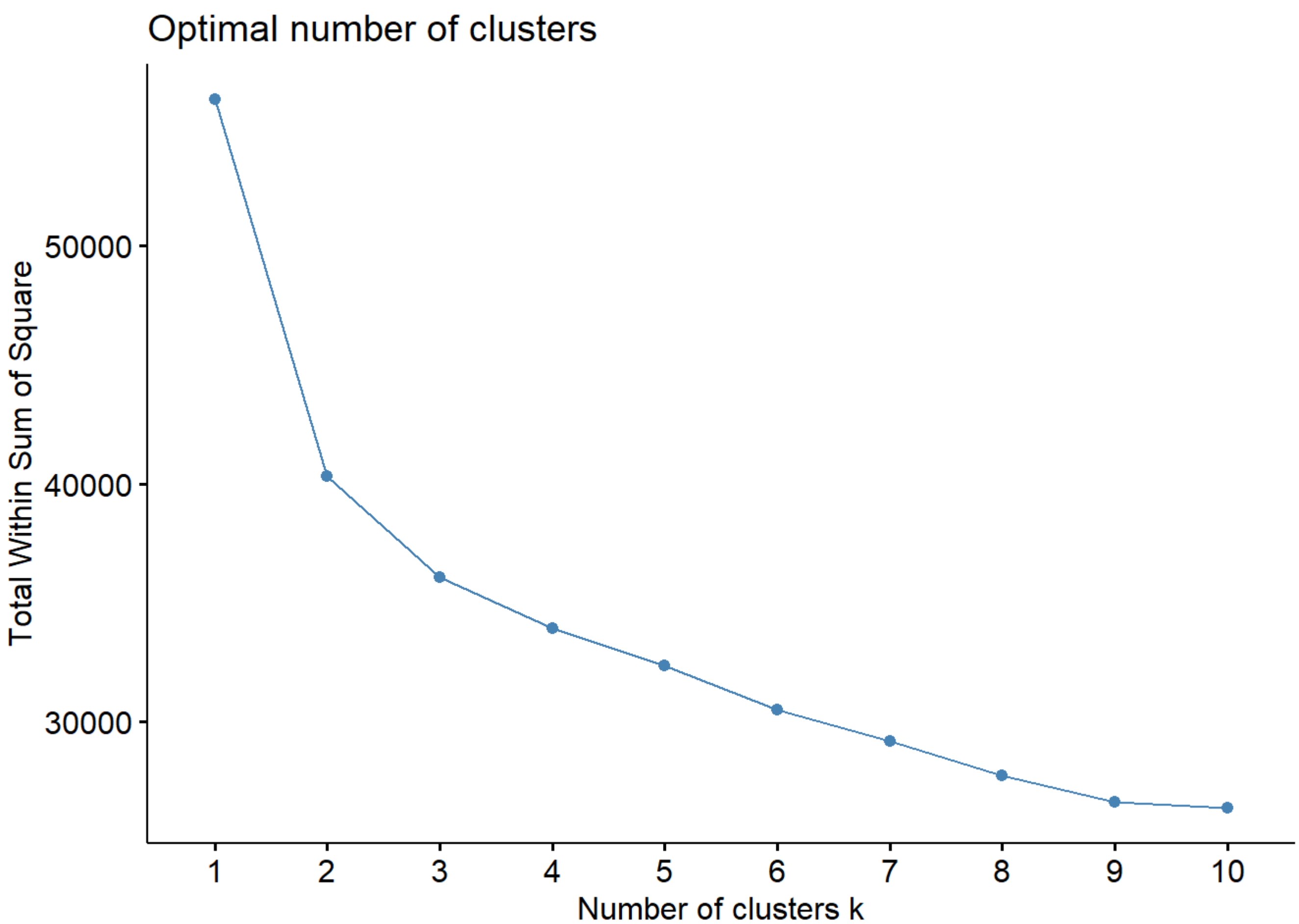
Hide

```
df<-scale(clust_data1)

set.seed(123)

# function to compute total within-cluster sum of square

fviz_nbclust(df, kmeans, method="wss")
```



Based on the elbow plot, we can see that the bend in the elbow occurs at 3 clusters. Therefore, optimal number of clusters = 3. We also plot a range of cluster plots.

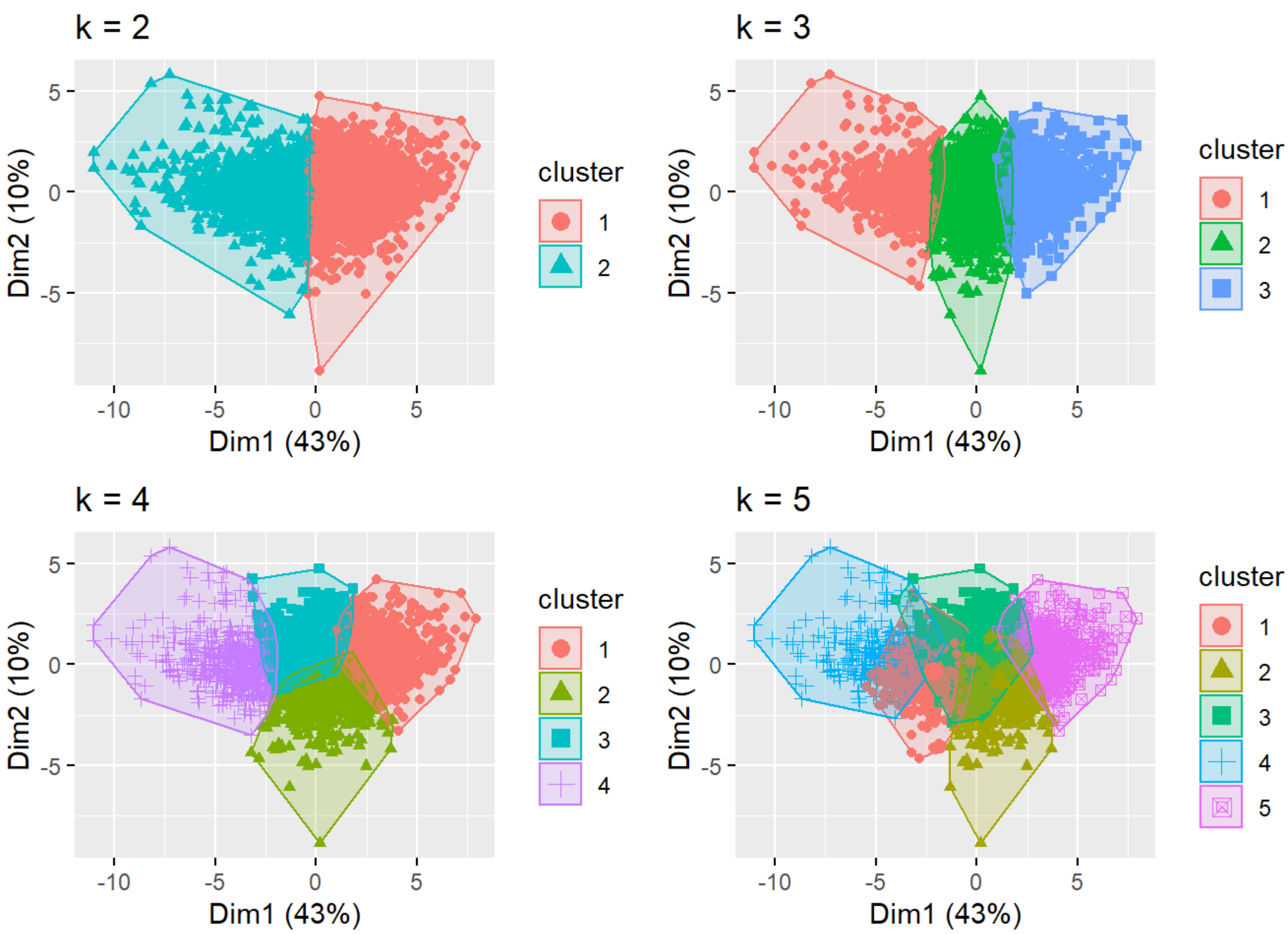
K means cluster plots

Hide

```
k2 <- kmeans(df, centers = 2, nstart = 25)
k3 <- kmeans(df, centers = 3, nstart = 25)
k4 <- kmeans(df, centers = 4, nstart = 25)
k5 <- kmeans(df, centers = 5, nstart = 25)

# plots to compare
p1 <- fviz_cluster(k2, geom = "point", data = df) + ggtitle("k = 2")
p2 <- fviz_cluster(k3, geom = "point",  data = df) + ggtitle("k = 3")
p3 <- fviz_cluster(k4, geom = "point",  data = df) + ggtitle("k = 4")
p4 <- fviz_cluster(k5, geom = "point",  data = df) + ggtitle("k = 5")

library(gridExtra)
grid.arrange(p1, p2, p3, p4, nrow = 2)
```



Hide

```
# Compute k-means clustering with k = 4
set.seed(123)
final <- kmeans(df, 3, nstart = 25)
print(final$centers)
```

```
##   perc_fair_poor_health avg_phy_unh_days avg_mental_unh_days perc_smokers
## 1           -0.92989169      -0.95904340          -0.9291810  -0.77697241
## 2            1.31418986       1.25994498           1.1231680   1.17792691
## 3            0.03208322       0.08324124           0.1319055  -0.01515278
##    perc_obese food_env_ind perc_phy_inact perc_excess_drink perc_uninsured
## 1 -0.52949167   0.62335994    -0.72647152        0.81573070     -0.5608488
## 2  0.74804259  -0.87444631     0.95114208       -1.02001347      0.3427638
## 3  0.01841211  -0.02493893     0.06476852       -0.09794453      0.2557387
##   perc_college    perc_unemp perc_child_pov perc_diabetes median_income
## 1    0.8746322 -0.5423748512    -0.90382624   -0.65413900     0.8861153
## 2   -0.8990331  0.8007697181     1.25782305    0.91006045    -0.9576412
## 3   -0.2072887  0.0007178443     0.04144577    0.03014415    -0.1854162
##     perc_65up perc_black perc_female  perc_18less
## 1 -0.10153935 -0.3622087 -0.01467851  0.004262794
## 2 -0.09262455  0.8785237  0.14545539  0.122109812
## 3  0.12757905 -0.1801499 -0.06502418 -0.067476685
```

Hide

```
print(final$size)
```

```
## [1] 1053  712 1355
```

With 3 clusters, we can see the centers for each cluster as shown above. Cluster1 has 712 elements, Cluster2 has 1053 elements and Cluster3 has 1355 elements.

Hierarchical Clustering

We use Agnes to conduct agglomeration clustering and identify which method has the highest coefficient. As can be seen below, that is the Ward method.

Hide

```
# methods to assess
m <- c( "average", "single", "complete", "ward")
names(m) <- c( "average", "single", "complete", "ward")

# function to compute coefficient
ac <- function(x) {
  agnes(df, method = x)$ac
}

map_dbl(m, ac)
```

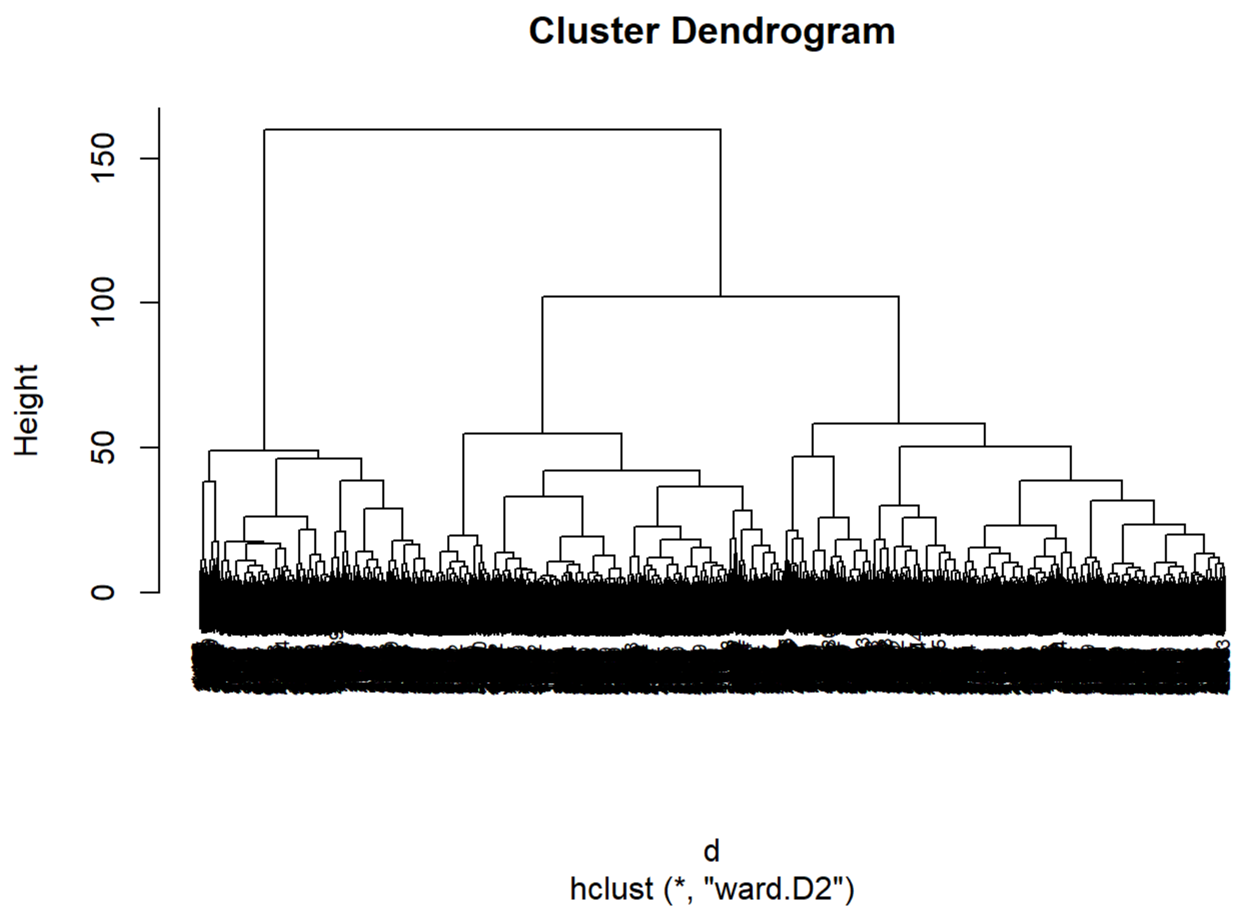
```
##   average    single  complete      ward
## 0.8478885 0.7031994 0.9049859 0.9882121
```

Hide

```
# Dissimilarity matrix
d <- dist(df, method = "euclidean")

# Ward's method
hc3 <- hclust(d, method = "ward.D2" )

plot(hc3, cex = 0.6)
```



From the dendrogram, we can see that the number of optimal clusters is again 3, similar to K-means clustering.

Hierarchical clustering analysis

Hide

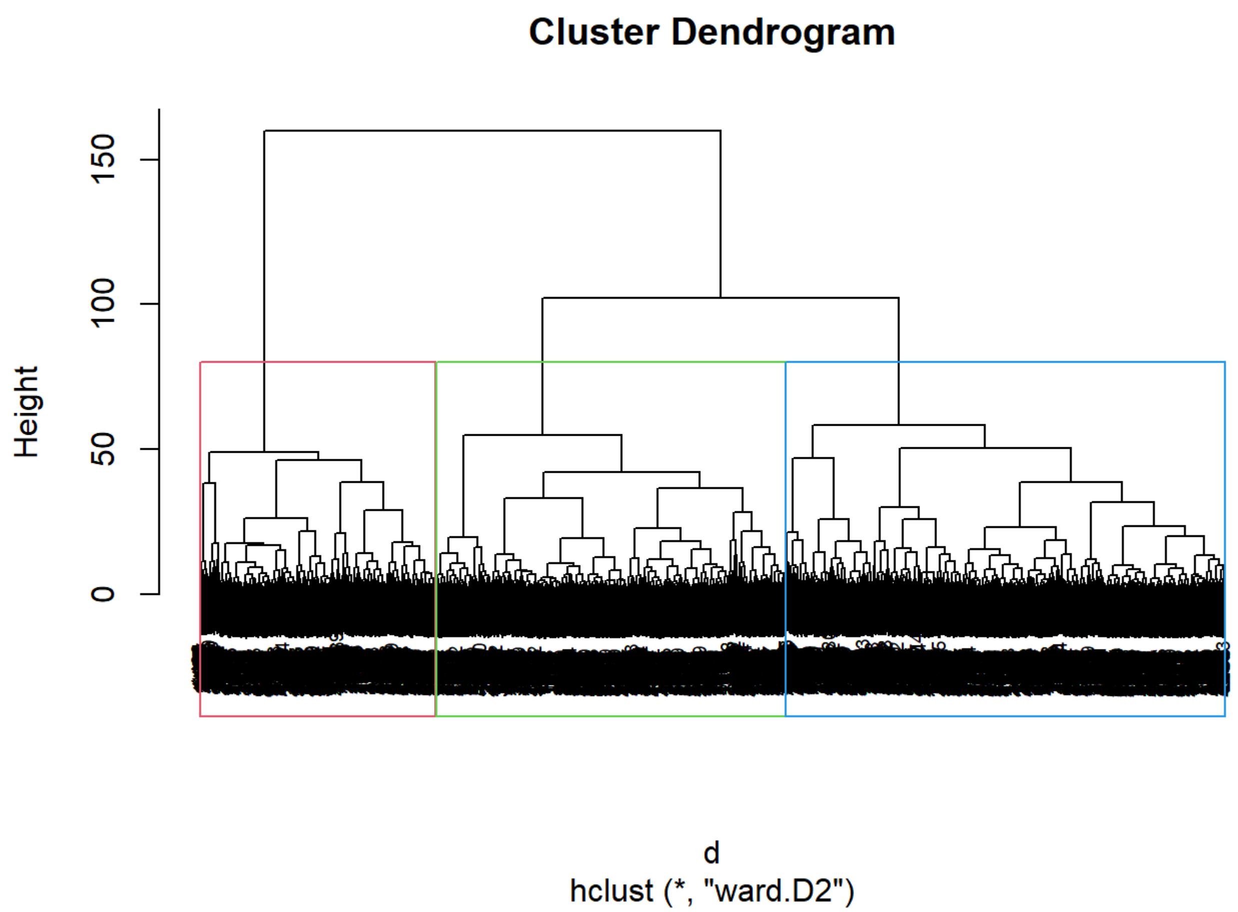
```
# Cut tree into 4 groups
sub_grp <- cutree(hc3, k = 3)

# Number of members in each cluster
table(sub_grp)
```

```
## sub_grp
##    1    2    3
## 1338  718 1064
```

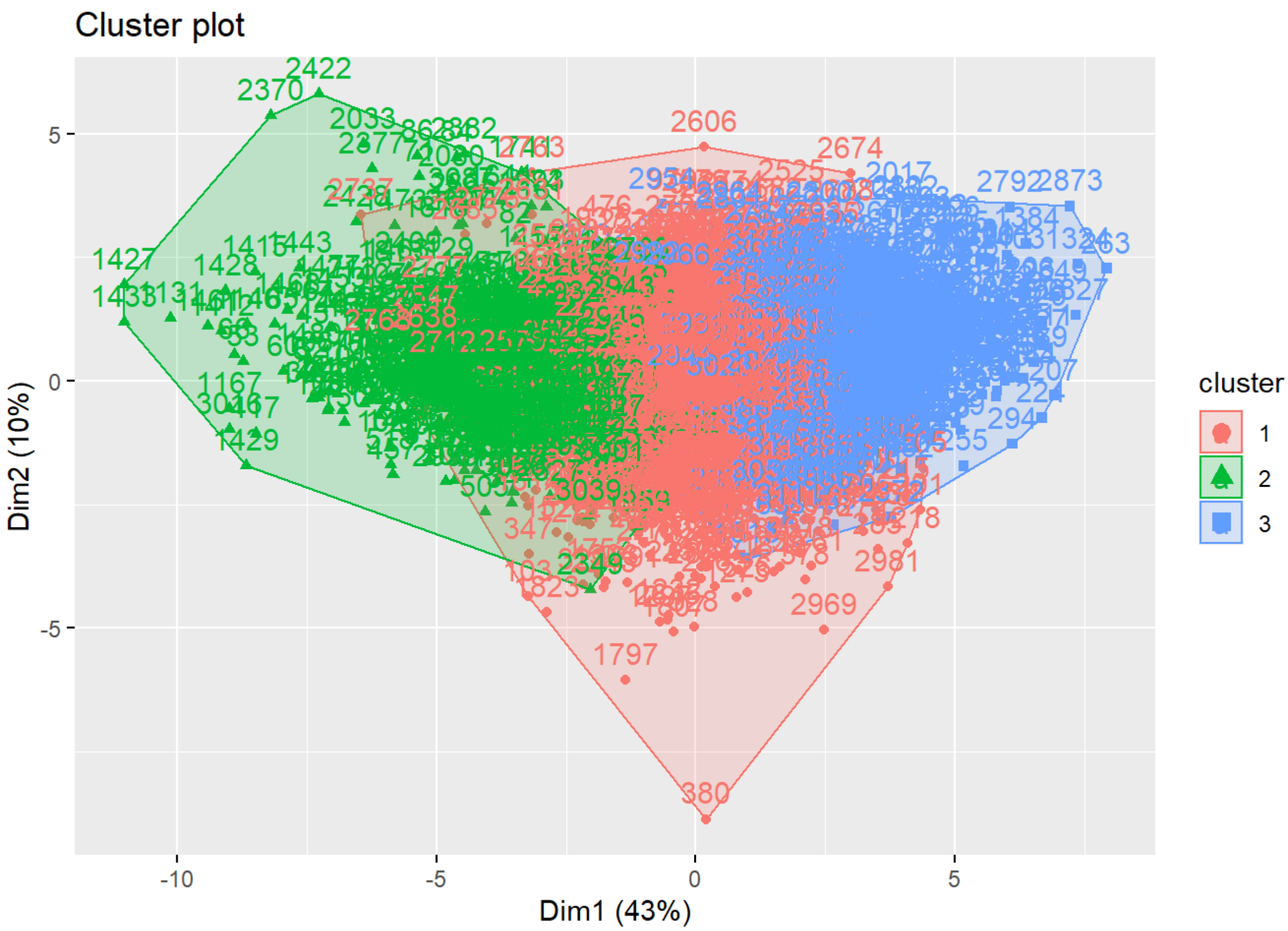
Hide

```
plot(hc3, cex = 0.6)
rect.hclust(hc3, k = 3, border = 2:5)
```



Hide

```
fviz_cluster(list(data = df, cluster = sub_grp))
```



Number of members in each cluster and the boundaries of each clusters are shown in the plot above.