Data Mining – Individual Assignment

Engg\_College\_data.csv

Subashree Bhaskaran

2019-06-12

# Introduction:

We have a set of 26 Engineering colleges rating for 5 features / variables.

This attempt is to cluster the college data using Hierarchical clustering & k-means and compare the output. Also, for the given data set, understand the optimal value for clustering. As we have 26 observations, let us consider 2 - 4 clusters, beyond which clusters will become too granular for analysis as it may not add better value.

**Pre-requisite & EDA:**

#Pre-requisite packages  
#install.packages("factoextra", "cluster", "tidyverse")  
library(factoextra)

## Warning: package 'factoextra' was built under R version 3.4.1

## Loading required package: ggplot2

## Welcome! Related Books: `Practical Guide To Cluster Analysis in R` at https://goo.gl/13EFCZ

library(cluster)  
library(tidyverse)

## Loading tidyverse: tibble  
## Loading tidyverse: tidyr  
## Loading tidyverse: readr  
## Loading tidyverse: purrr  
## Loading tidyverse: dplyr

## Warning: package 'tidyr' was built under R version 3.4.1

## Warning: package 'purrr' was built under R version 3.4.1

## Warning: package 'dplyr' was built under R version 3.4.1

## Conflicts with tidy packages ----------------------------------------------

## filter(): dplyr, stats  
## lag(): dplyr, stats

#Read data & EDA  
setwd("/Users/anand/Documents/BABI/DataMining/EnggCollege\_Indiv\_assignment")  
data\_engg <- read.csv("Engg\_College\_Data.csv")  
  
str(data\_engg)

## 'data.frame': 26 obs. of 7 variables:  
## $ SR\_NO : int 1 2 3 4 5 6 7 8 9 10 ...  
## $ Engg\_College : Factor w/ 26 levels "A","B","C","D",..: 1 2 3 4 5 6 7 8 9 10 ...  
## $ Teaching : int 5 4 4 5 2 3 1 4 4 3 ...  
## $ Fees : int 2 2 5 4 5 4 3 4 4 4 ...  
## $ Placements : int 5 5 5 5 2 3 1 5 4 3 ...  
## $ Internship : int 5 5 4 4 2 3 1 5 4 4 ...  
## $ Infrastructure: int 3 3 5 4 5 4 2 5 4 5 ...

data\_engg$SR\_NO <- as.factor(data\_engg$SR\_NO)  
summary(data\_engg)

## SR\_NO Engg\_College Teaching Fees   
## 1 : 1 A : 1 Min. :1.000 Min. :1.000   
## 2 : 1 B : 1 1st Qu.:2.000 1st Qu.:2.250   
## 3 : 1 C : 1 Median :3.000 Median :4.000   
## 4 : 1 D : 1 Mean :2.808 Mean :3.577   
## 5 : 1 E : 1 3rd Qu.:3.000 3rd Qu.:5.000   
## 6 : 1 F : 1 Max. :5.000 Max. :5.000   
## (Other):20 (Other):20   
## Placements Internship Infrastructure   
## Min. :1.000 Min. :1.000 Min. :1.000   
## 1st Qu.:1.250 1st Qu.:2.000 1st Qu.:3.000   
## Median :3.000 Median :2.000 Median :3.000   
## Mean :2.885 Mean :2.769 Mean :3.385   
## 3rd Qu.:4.000 3rd Qu.:4.000 3rd Qu.:4.750   
## Max. :5.000 Max. :5.000 Max. :5.000   
##

**Inference**: No missing data found / No scaling required

**Hierarchical Clustering:**

#Calculate the Euclidean distance  
data\_engg\_euc <- dist(data\_engg[3:7], method = "euclidean")  
#Visualize the distance matrix  
fviz\_dist(data\_engg\_euc, gradient = list(low = "#008000", mid = "white", high = "#FC4E07"))

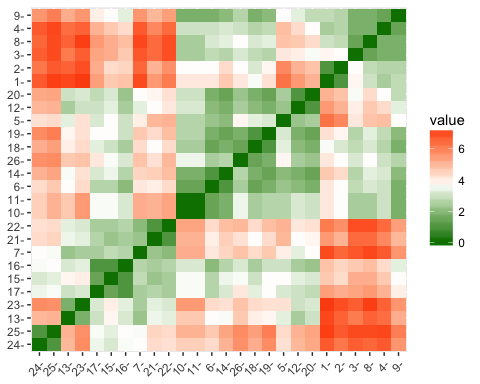
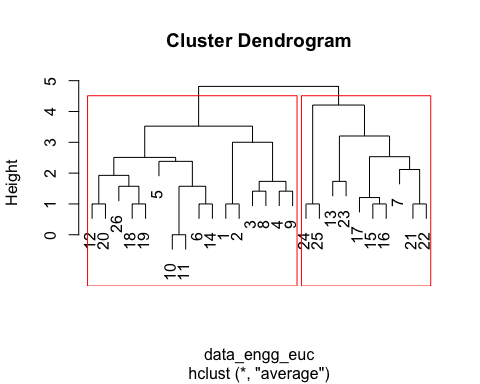


Fig 1: Distance matrix visualization for 26 observations

This gives a basic idea which observations are closer (green) with each other & vice versa.

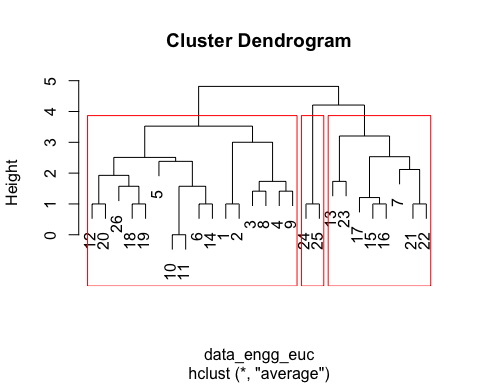
#Do Clustering using Hierarchical with Average Linkage  
clust\_avg1 <- hclust(data\_engg\_euc, method = "average")

#Visualize clusters with Dendogram for k=2  
plot(clust\_avg1, labels = data\_engg$SR\_NO)  
rect.hclust(clust\_avg1, k=2 , border = "red")



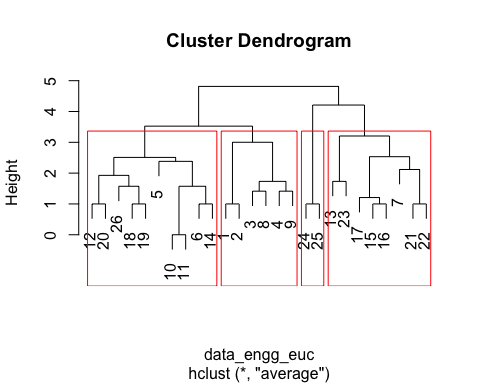
**Fig 2: Hierarchical Cluster Dendogram for Average linkage showing 2 Clusters**

#Visualize clusters with Dendogram for k=3  
plot(clust\_avg1, labels = data\_engg$SR\_NO)  
rect.hclust(clust\_avg1, k=3 , border = "red")



**Fig 3: Hierarchical Cluster Dendogram for Average linkage showing 3 Clusters**

#Visualize clusters with Dendogram for k=4  
plot(clust\_avg1, labels = data\_engg$SR\_NO)  
rect.hclust(clust\_avg1, k=4 , border = "red")

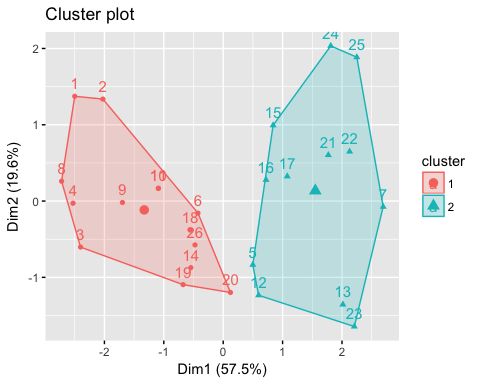


**Fig 4: Hierarchical Cluster Dendogram for Average linkage showing 4 Clusters**

**K-Means Clustering:**

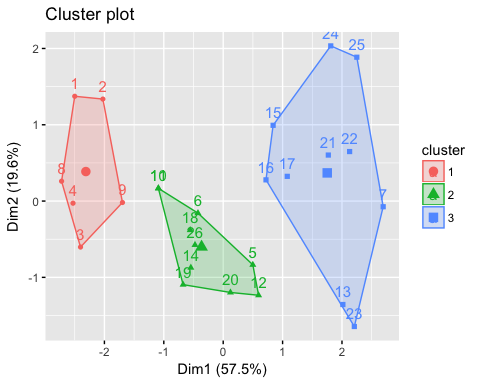
#Do Clustering with k-means  
set.seed(1234)  
#No change in data - hence, the same data\_engg variable is used

#Perform k-means for k=2 & Visualize  
clust\_kmeans2 <- kmeans(data\_engg[3:7], centers = 2, nstart = 25)  
fviz\_cluster(clust\_kmeans2, data\_engg[3:7])



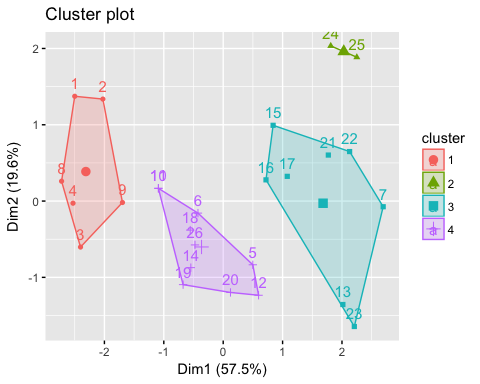
**Fig 5: K-means Cluster Analysis showing 2 Clusters**

#Perform k-means for k=3 & Visualize  
clust\_kmeans3 <- kmeans(data\_engg[3:7], centers = 3, nstart = 25)  
fviz\_cluster(clust\_kmeans3, data\_engg[3:7])



**Fig 6: K-means Cluster Analysis showing 3 Clusters**

#Perform k-means for k=4 & Visualize  
clust\_kmeans4 <- kmeans(data\_engg[3:7], centers = 4, nstart = 25)  
fviz\_cluster(clust\_kmeans4, data\_engg[3:7])



**Fig 7: K-means Cluster Analysis showing 4 Clusters**

Now that, cluster analysis is performed using Hierarchical clustering & K-means clustering techniques, let us find out the optimal values of k for the given data set, before proceeding to cluster performance analysis.

**Optimal k-value:**

An optimal value of ‘k’ can be known by computing the Average Silhouette values for different ‘k’ values. The ‘k’ value with higher average Silhouette value will be the optimum.

# To know optimal k value: Compute average silhouette for k clusters  
  
#Silhouette   
avg\_sil <- function(k) {  
 km.res <- kmeans(data\_engg[3:7], centers = k, nstart = 25)  
 ss <- silhouette(km.res$cluster, dist(data\_engg[3:7]))  
 mean(ss[, 3])  
}

# Compute and plot wss for k = 2 to k = 16  
k.values <- 2: 16  
  
# extract avg silhouette for 2-16 clusters  
avg\_sil\_values <- map\_dbl(k.values, avg\_sil)  
  
plot(k.values, avg\_sil\_values,  
 type = "b", pch = 19, frame = FALSE,   
 xlab = "Number of clusters K",  
 ylab = "Average Silhouettes")



**Fig 8: Average Silhouette for range of ‘k’ values**

From average Silhouette plot above, though k= 8 or 13 has highest value in Y-axis, we will ignore k values beyond 4, so as to not make very granular clusters.

k=5 has same silhouette value as that of 4. Hence, having 5 clusters is same as having 4.

Optimal k will be 4 followed closely by 2. Let us compare results of both clustering techniques.

**Hierarchical vs K-means Comparison:**

For values of k=2,3,4, let us compare the results in tabular format as below:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **K=2** | |  | **K=3** | |  | **K=4** | |
| **Average** | **K-means** |  | **Average** | **K-means** |  | **Average** | **K-means** |
| 20 | 20 |  | 12 | 12 |  | 12 | 12 |
| 12 |  |  | 20 | 20 |  | 20 | 20 |
| 26 | 26 |  | 26 | 26 |  | 26 | 26 |
| 18 | 18 |  | 18 | 18 |  | 18 | 18 |
| 19 | 19 |  | 19 | 19 |  | 19 | 19 |
| 5 | 9 |  | 5 | 5 |  | 5 | 5 |
| 10 | 10 |  | 10 | 10 |  | 10 | 10 |
| 11 | 11 |  | 11 | 11 |  | 11 | 11 |
| 6 | 6 |  | 6 | 6 |  | 6 | 6 |
| 14 | 14 |  | 14 | 14 |  | 14 | 14 |
| 1 | 1 |  | 1 |  |  | 1 | 1 |
| 2 | 2 |  | 2 |  |  | 2 | 2 |
| 3 | 3 |  | 3 |  |  | 3 | 3 |
| 4 | 4 |  | 4 |  |  | 4 | 4 |
| 8 | 8 |  | 8 |  |  | 8 | 8 |
| 9 |  |  | 9 |  |  | 9 | 9 |
| 24 | 5 |  | 24 | 1 |  | 24 | 24 |
| 25 | 7 |  | 25 | 2 |  | 25 | 25 |
| 13 | 13 |  |  | 3 |  | 13 | 13 |
| 23 | 12 |  |  | 4 |  | 23 | 23 |
| 17 | 15 |  |  | 8 |  | 17 | 17 |
| 15 | 16 |  |  | 9 |  | 15 | 15 |
| 16 | 17 |  | 13 | 13 |  | 16 | 16 |
| 7 | 21 |  | 23 | 23 |  | 7 | 7 |
| 21 | 22 |  | 17 | 17 |  | 21 | 21 |
| 22 | 23 |  | 15 | 15 |  | 22 | 22 |
|  | 24 |  | 16 | 16 |  |  |  |
|  | 25 |  | 7 | 7 |  |  |  |
|  |  |  | 21 | 21 |  |  |  |
|  |  |  | 22 | 22 |  |  |  |
|  |  |  |  | 24 |  |  |  |
|  |  |  |  | 25 |  |  |  |

**Table 1: Comparison of Hierarchical clustering (Average Linkage) and K-means**

**Inference:** For k=4, the Hierarchical clustering with Average linkage exactly matches the k-means clustering result. Hence, there is no change in the linkage to be used.   
However, I have used all other different linkage to further substantiate my answer.

clust\_wardD <- hclust(data\_engg\_euc, method = "ward.D")

clust\_wardD2 <- hclust(data\_engg\_euc, method = "ward.D2")

clust\_single <- hclust(data\_engg\_euc, method = "single")

clust\_complete <- hclust(data\_engg\_euc, method = "complete")

clust\_mcquitty <- hclust(data\_engg\_euc, method = "mcquitty")

clust\_median <- hclust(data\_engg\_euc, method = "median")

clust\_centroid <- hclust(data\_engg\_euc, method = "centroid")

From the output, it is clear that the mcquitty, ward.D2 linkages perform the same as average and exactly match the k-means output. Please note, this could be the inference for this data set for the 4 clusters and may not apply all the time.

For easy reference, I have the result output put in the below tabular form:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **K=4** | | | | | | | |
| **Average** | **K-means** | **mcquitty** | **ward.D2** | **ward.D** | **single** | **complete** | **median / centroid** |
| 12 | 12 | 12 | 12 | 12 | 12 | 12 | 12 |
| 20 | 20 | 20 | 20 | 20 | 20 | 20 | 20 |
| 26 | 26 | 26 | 26 | 26 | 26 | 6 | 26 |
| 18 | 18 | 18 | 18 | 18 | 18 | 14 | 18 |
| 19 | 19 | 19 | 19 | 19 | 19 | 10 | 19 |
| 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 |
| 10 | 10 | 10 | 10 | 6 | 10 | 11 | 10 |
| 11 | 11 | 11 | 11 | 14 | 11 | 26 | 11 |
| 6 | 6 | 6 | 6 | 10 | 6 | 18 | 6 |
| 14 | 14 | 14 | 14 | 11 | 14 | 19 | 14 |
| 1 | 1 | 1 | 1 | 1 | 21 | 1 | 8 |
| 2 | 2 | 2 | 2 | 2 | 22 | 2 | 9 |
| 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 |
| 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 |
| 8 | 8 | 8 | 8 | 8 | 8 | 8 | 13 |
| 9 | 9 | 9 | 9 | 9 | 9 | 9 | 23 |
| 24 | 24 | 24 | 24 | 24 | 17 | 24 | 17 |
| 25 | 25 | 25 | 25 | 25 | 15 | 25 | 15 |
| 13 | 13 | 13 | 13 | 13 | 16 | 13 | 16 |
| 23 | 23 | 23 | 23 | 23 | 7 | 23 | 7 |
| 17 | 17 | 17 | 17 | 17 | 13 | 17 | 21 |
| 15 | 15 | 15 | 15 | 15 | 23 | 15 | 22 |
| 16 | 16 | 16 | 16 | 16 | 24 | 16 | 24 |
| 7 | 7 | 7 | 7 | 7 | 25 | 7 | 25 |
| 21 | 21 | 21 | 21 | 21 | 1 | 21 | 1 |
| 22 | 22 | 22 | 22 | 22 | 2 | 22 | 2 |

**Table 2: Comparison of Hierarchical clustering (all Linkages)**

**and K-means for k=4**

# Thank you!