

**HO CHI MINH CITY UNIVERSITY TECHNOLOGY AND EDUCATION**

🞠◊🞠◊🞠

**GVHD**

**TOPIC:HIERARCHICAL CLUSTERING**

**Project 1**

**GROUP: 02**

**SEMESTER : 1 – YEAR: 2018-2019**

**TP. HỒ CHÍ MINH –**

**Participating members list**

Group: 02

*Nguyen Tran Nguyet Minh*

*ID: 17110052*

*Tran Thuan Tuong Vy*

*ID: 17110093*

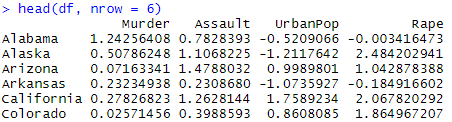
***Comment***

In this project, we use R language to describe the agglomerative clustering algorithms.

|  |
| --- |
| # Load the data  data("USArrests") |

We uses the dataset called “USArrests”. This is a public dataset available in R. Now, we’ll show the first 6 rows

|  |
| --- |
| head(df, nrow = 6) |

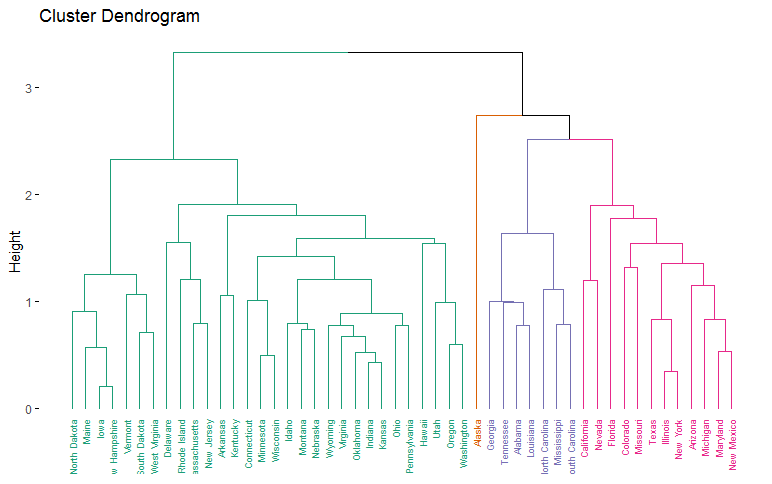


We use ‘Euclidean distance’ to calculates the distance between the data matrix rows; however, we can use the argument method to show other metrics. We try creating the hierarchical tree with average method

|  |
| --- |
| # Load Library  library("cluster")  res.agnes <- agnes(x = USArrests, # data matrix  stand = TRUE, # Standardize the data  metric = "euclidean", # metric for distance matrix  method = "average" # Linkage method  ) |

To display the hierarchical tree generated by the function hclust(), we can use the base function plot(res.hc). But to make Dendrograms observation easier, we use the function fviz\_dend() in factoextra R package.

|  |
| --- |
| fviz\_dend(res.agnes, cex = 0.5, k = 4, k\_colors = c("#1B9E77", "#D95F02", "#7570B3", "#E7298A")) |



|  |
| --- |
| # Compute cophentic distance  res.coph <- cophenetic(res.agnes)  # Correlation between cophenetic distance and the original distance  cor(res.dist, res.coph) |



To show the different between linkage methods, we try to use other method

|  |
| --- |
| res.agnes1 <- agnes(x = USArrests,  stand = TRUE,  metric = "euclidean",  method = "complete"  )  cor(res.dist, cophenetic(res.agnes1)) |



The correlation coefficient shows that using a different linkage method creates a tree that represents another original distance.

|  |
| --- |
| # Cut tree into 4 groups  grp <- cutree(res.agnes, k = 4)  head(grp, n = 4)  # Number of members in each cluster  table(grp) |





To present the result in a scatter plot, we also use function fviz\_cluster().

|  |
| --- |
| Mylist <- list(data = scale(USArrests), cluster = grp)  fviz\_cluster(mylist,  palette = c("#1B9E77", "#D95F02", "#7570B3", "#E7298A"),  ggtheme = theme\_minimal()) |

