A.Bakiu

Cluster analysis pt.2

Hierarchical analysis

Hierarchical clustering

- Unlike kmeans we do not need to know the number of clusters since the beginning

Two types:

- ➢ Divisive (from «top» - «down»)
- ➢ Agglomerative (from «bottom» - «up») done with hclust() function in R

Agglomerative hierarchical clustering

✶ Step 1

Compute the distance matrix

✶ Step 2

Make every observation its own cluster, for a total of n cluster

✶ Step 3

Combine the two most similar cluster into one.

✶ Step 4

Update the distance matrix Compute pairwise distance between each pair of cluster.

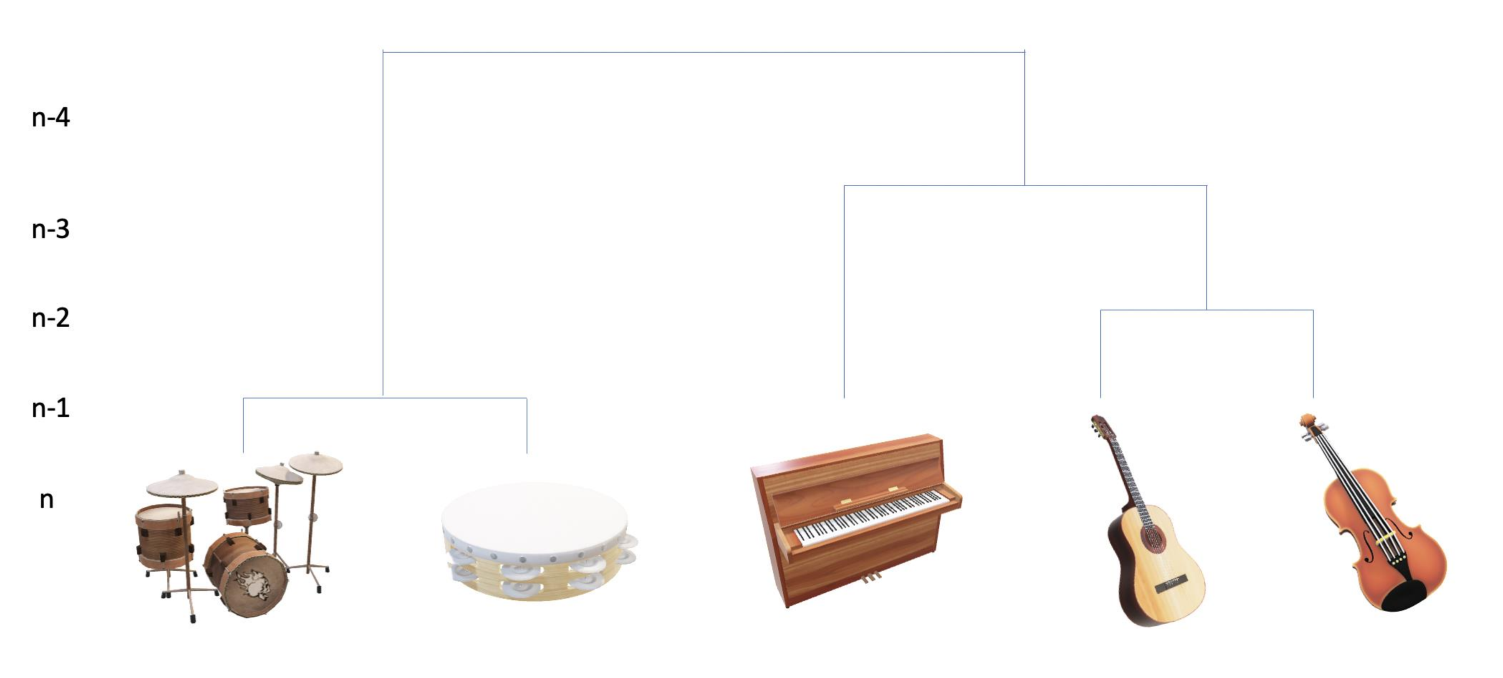
✶ Step 5

Iterate until there is only 1 cluster.

Example of the algorithm

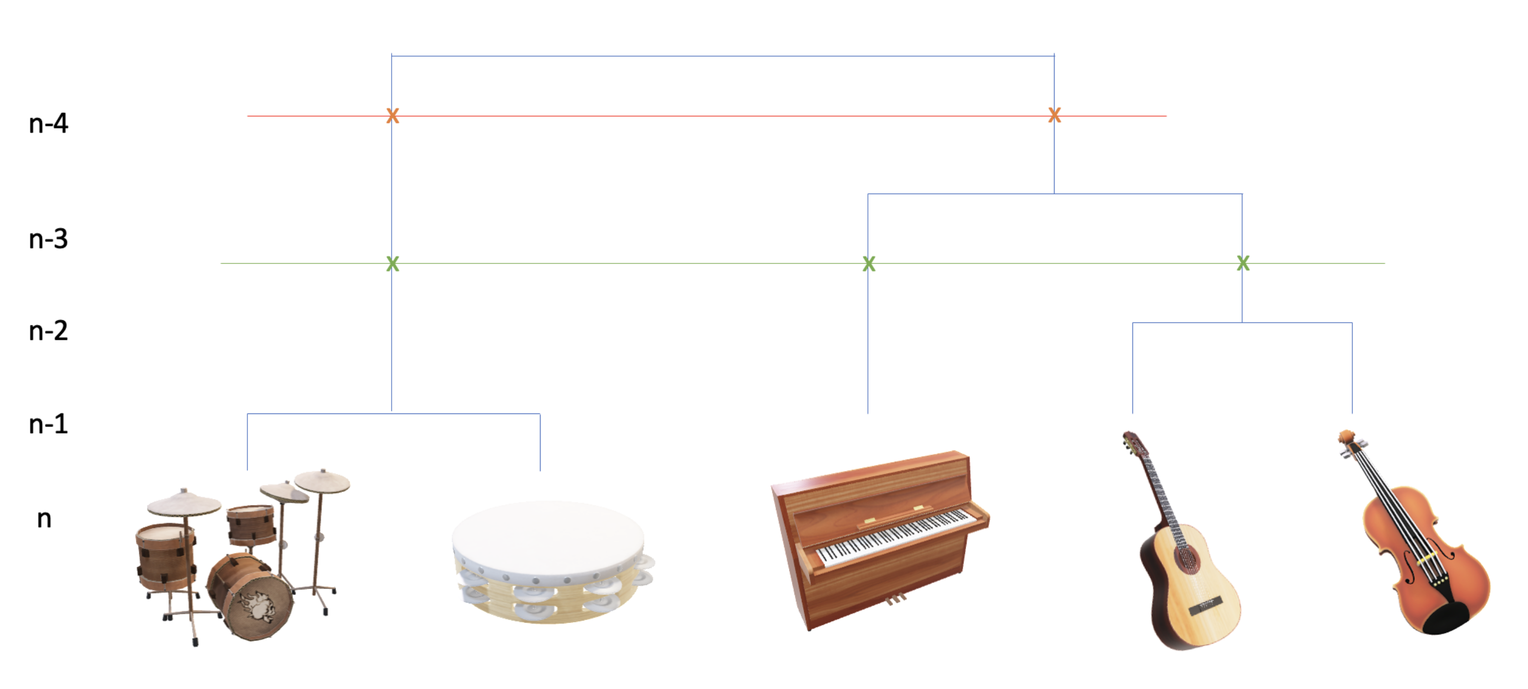
- Let’s do an example where we draw the dendrogram (output of hierarchical clustering) might look like for 5 different music instruments.

* The more similar are the clusters, the shorter is the line that combines them.
** We have combined piano with the violin and the guitar, since it is more similar to them then the drums,
because piano produces sounds thanks to the vibrations of the cords inside it.



* If we want to create 2 clusters we simply draw a horizontal line on the location where our line will create 2 broken links. (shown in red).

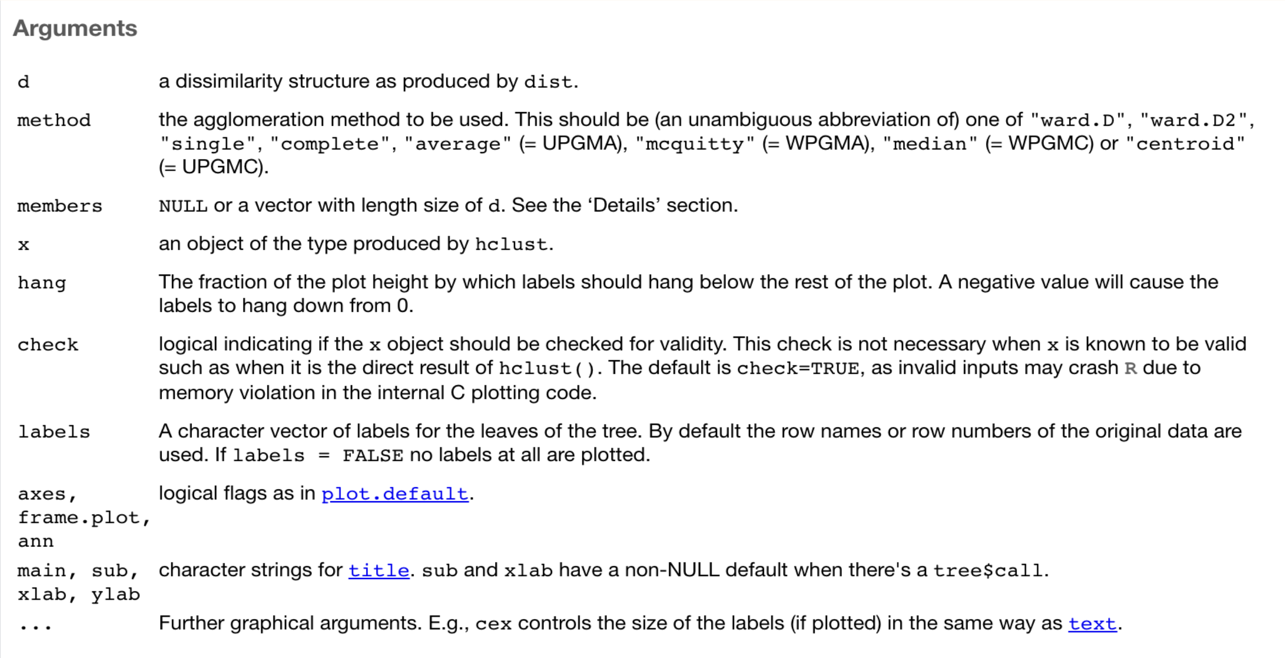
** The same if we want 3 clusters or more. (shown in green).



02 Hierarchical analysis in R

How to perform hierarchical cluster analysis in R

Arguments of hclust() in R



Arguments

`d` a dissimilarity structure as produced by `dist`.

`method` the agglomeration method to be used. This should be (an unambiguous abbreviation of) one of `"ward.D"`, `"ward.D2"`, `"single"`, `"complete"`, `"average"` (= UPGMA), `"mcquitty"` (= WPGMA), `"median"` (= WPGMC) or `"centroid"` (= UPGMC).

`members` `NULL` or a vector with length size of `d`. See the 'Details' section.

`x` an object of the type produced by `hclust`.

`hang` The fraction of the plot height by which labels should hang below the rest of the plot. A negative value will cause the labels to hang down from 0.

`check` logical indicating if the `x` object should be checked for validity. This check is not necessary when `x` is known to be valid such as when it is the direct result of `hclust()`. The default is `check=TRUE`, as invalid inputs may crash R due to memory violation in the internal C plotting code.

`labels` A character vector of labels for the leaves of the tree. By default the row names or row numbers of the original data are used. If `labels = FALSE` no labels at all are plotted.

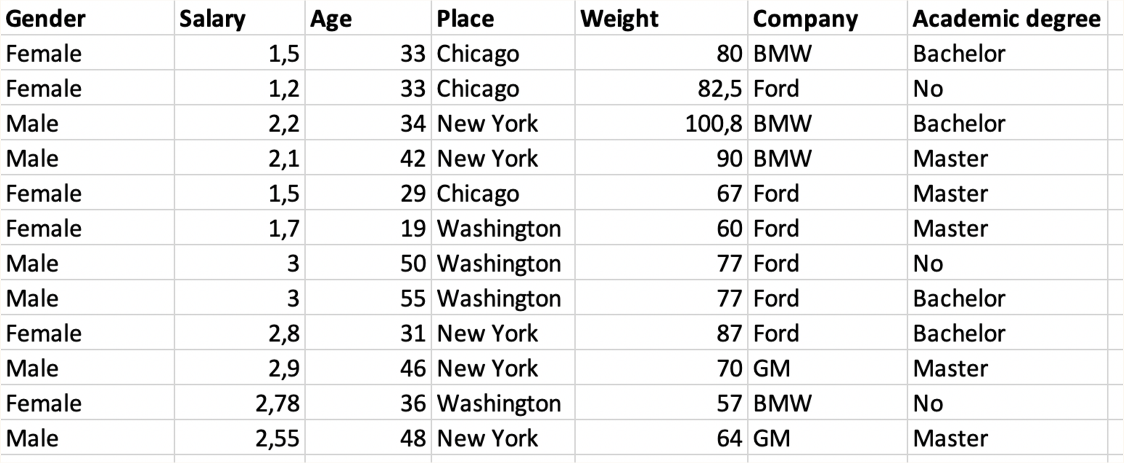
`axes, frame.plot, ann` logical flags as in `plot.default`.

`main, sub, xlab, ylab` character strings for `title`. `sub` and `xlab` have a non-NULL default when there's a `tree$call`.

`...` Further graphical arguments. E.g., `cex` controls the size of the labels (if plotted) in the same way as `text`.

Agglomerative hierarchical clustering on R

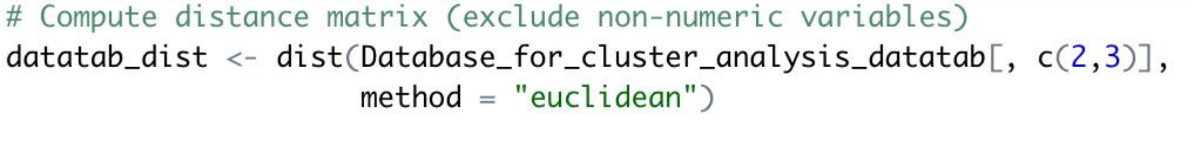
- We're going to perform agglomerative hierarchical clustering on the dataset below, into base R:



Gender	Salary	Age	Place	Weight	Company	Academic degree
Female	1,5	33	Chicago	80	BMW	Bachelor
Female	1,2	33	Chicago	82,5	Ford	No
Male	2,2	34	New York	100,8	BMW	Bachelor
Male	2,1	42	New York	90	BMW	Master
Female	1,5	29	Chicago	67	Ford	Master
Female	1,7	19	Washington	60	Ford	Master
Male	3	50	Washington	77	Ford	No
Male	3	55	Washington	77	Ford	Bachelor
Female	2,8	31	New York	87	Ford	Bachelor
Male	2,9	46	New York	70	GM	Master
Female	2,78	36	Washington	57	BMW	No
Male	2,55	48	New York	64	GM	Master

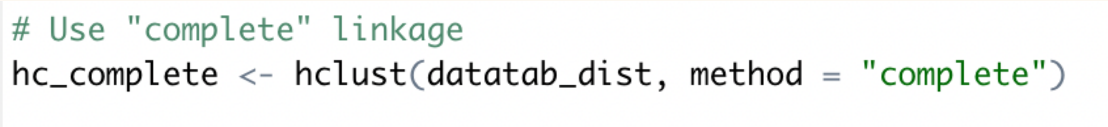
Agglomerative hierarchical clustering on R

- Firstly we need to compute the distance matrix using the Euclidian distance metric.



```r
# Compute distance matrix (exclude non-numeric variables)
datatab_dist <- dist(Database_for_cluster_analysis_datatab[, c(2,3)],
                     method = "euclidean")
```

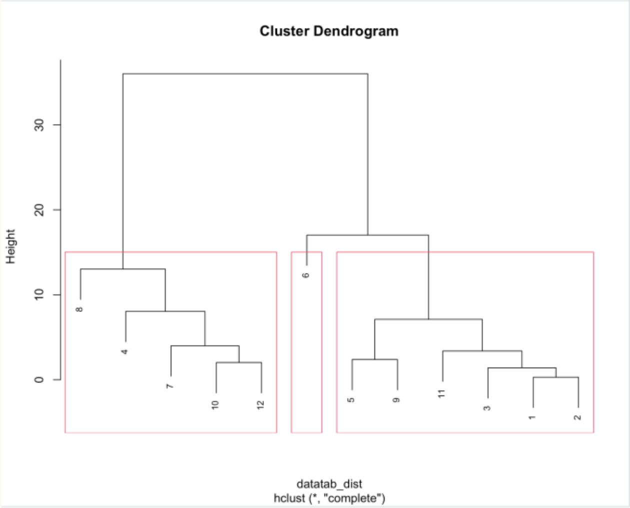
- Let's put also the type of linkage we will use.

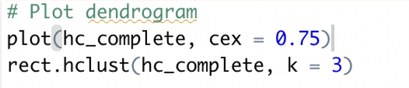


```r
# Use "complete" linkage
hc_complete <- hclust(datatab_dist, method = "complete")
```

Agglomerative hierarchical clustering on R

- Now we have to plot the dendrogram for hc_complete objcet that we just created. We can use the plot function directly on that object.
- We are also using rect.hclust to create 3 rectangles displaying what the cluster labels would be if we picked 3 clusters.

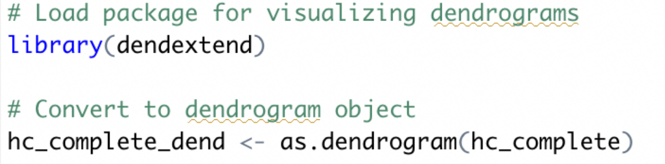




```r
# Plot dendrogram
plot(hc_complete, cex = 0.75)
rect.hclust(hc_complete, k = 3)
```

Agglomerative hierarchical clustering on R

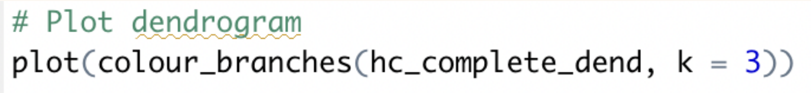
- Next we have to load dendextend package in order to to make it look better with the use of an external package.



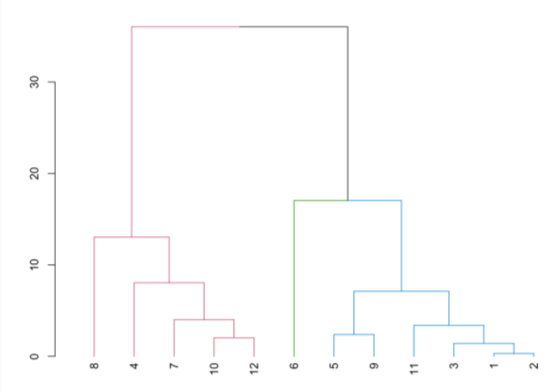
```
# Load package for visualizing dendrograms
library(dendextend)

# Convert to dendrogram object
hc_complete_dend <- as.dendrogram(hc_complete)
```

- Now we have to use the colour_branches

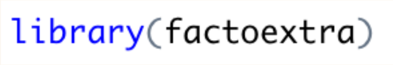


```
# Plot dendrogram
plot(colour_branches(hc_complete_dend, k = 3))
```



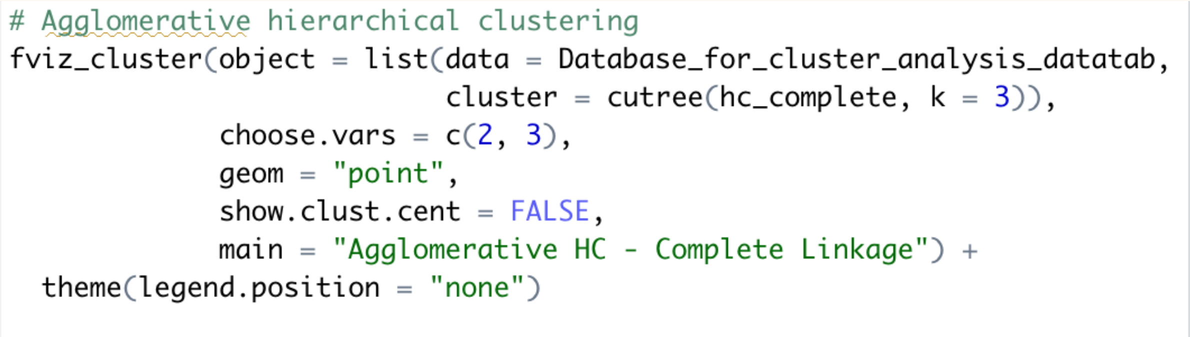
Agglomerative hierarchical clustering on R

- Now we have to load factoextra package, which will help us visualize the clusters for the 2 different algorithms.



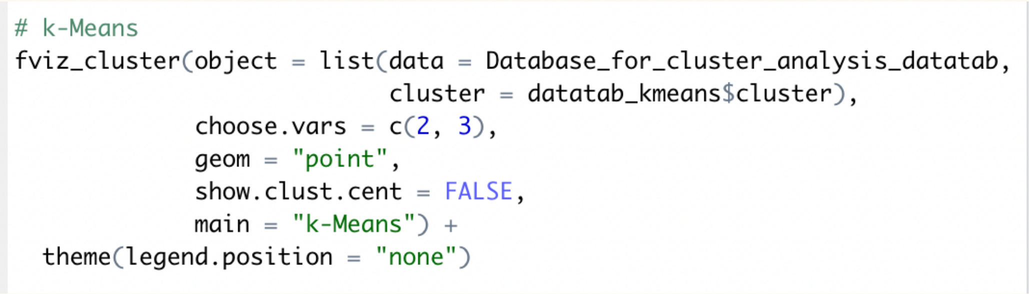
```
library(factoextra)
```

- Then we will use fviz_cluster from the factoextra to plot the cluster lables.

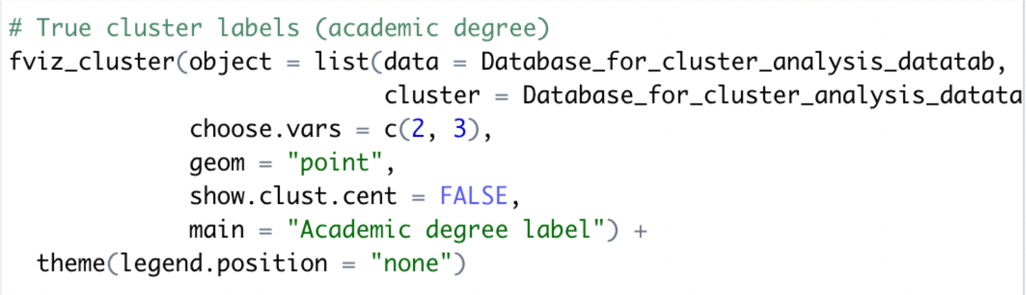


```
# Agglomerative hierarchical clustering
fviz_cluster(object = list(data = Database_for_cluster_analysis_datatab,
                           cluster = cutree(hc_complete, k = 3)),
             choose.vars = c(2, 3),
             geom = "point",
             show.clust.cent = FALSE,
             main = "Agglomerative HC - Complete Linkage") +
  theme(legend.position = "none")
```

Agglomerative hierarchical clustering on R



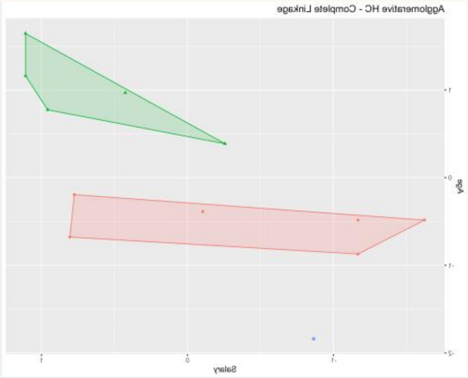
```
# k-Means
fviz_cluster(object = list(data = Database_for_cluster_analysis_datatab,
                           cluster = datatab_kmeans$cluster),
             choose.vars = c(2, 3),
             geom = "point",
             show.clust.cent = FALSE,
             main = "k-Means") +
  theme(legend.position = "none")
```

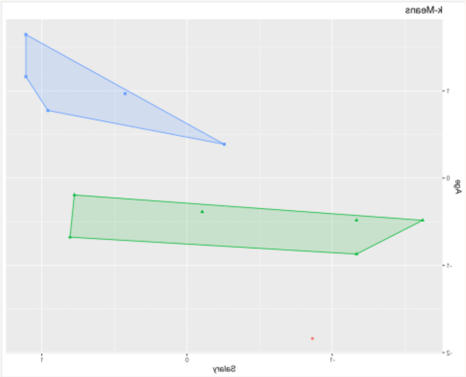


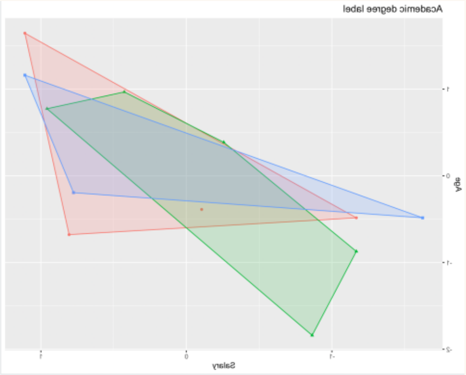
```
# True cluster labels (academic degree)
fviz_cluster(object = list(data = Database_for_cluster_analysis_datatab,
                           cluster = Database_for_cluster_analysis_datata
             choose.vars = c(2, 3),
             geom = "point",
             show.clust.cent = FALSE,
             main = "Academic degree label") +
  theme(legend.position = "none")
```

We have to run all these syntax and then we can compare the results between k-means and hierarchical clustering

Agglomerative hierarchical clustering on R







We may notice that the clusters in Agglomerative hc complete linkage and k-means are the same.

Comparison k-means and agglomerative clustering

✦ k-means

- Much faster and better choice for very large datasets.
- Sensitive to initializations.
- Restricted to Euclidean technique

✦ Hierarchical

- We do not need to know k ahead of time.
- More freedom how to define similarity.

The end.