ASEEM MALHOTRA

101557400

STA 30005

MULTIVARIATE ANALYSIS

ASSIGNMENT 1

PART 1

**Question (a)**

In *utilities.csv* we are presented with public utility usage pattern for 22 different entities. The data is distributed in 8 different variables. The purpose of this report is to cluster the entities into groups which will helps us in making business decisions. In this report, we will consider how many clusters are ideal for classifying entities.

We start examining the variables and Immediately we notice that there is a huge difference in the scale of measurement of each variable (e.g. mean sales X6 is 8914 whereas mean coverage ratio and fuel costs are 1.1 each). To compare and cluster this data we normalized the dataset I.e. .

Now that all variables are scaled and ready to be compared we calculated the Euclidean distance between each entity. The Euclidean distance will be used to divide the dataset by k-means clustering into 2, 3, 4 and 5 clusters. The division of clusters for each k-value is given below in *figure 1* representing a scatter plot showing ROC on y-axis and Sales on x-axis.

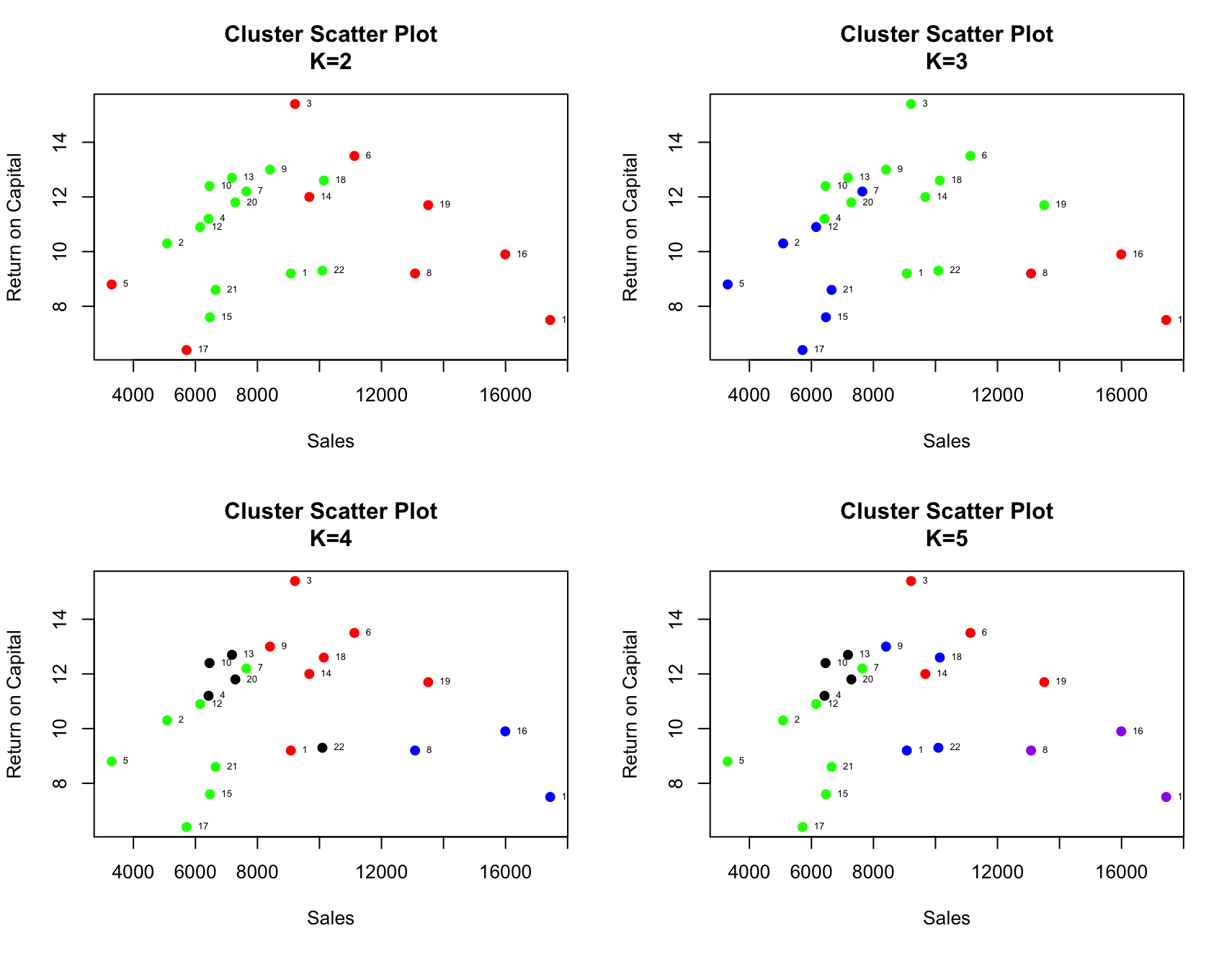


Figure : Showing scatter plot for k = 2, 3, 4 and 5.

By viewing the above plots the plot with 3 clusters seems to divide the chart in 3 equal portions and *table 1* below shows a significant increase in variation between sum of squares and ratio of between and total sum of squares from 2 to 3 clusters. This increase slows down from 3 to 4 clusters. Hence 3 clusters seem to be ideal.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *Table 1: shows group sizes and Sum of squares for each k-value.* | | | | |  |
| **Numbers of Clusters (k)** | **Group Sizes** | **iterations** | **Total within**  **Sum of Squares** | **Total Between**  **Sum of Squares** | **Between SS/**  **Total SS** |
| 2 | 9 & 13 | 1 | 569.84 | 123.5 | 17.8% |
| 3 | 3, 12 & 7 | 2 | 409.98 | 283.36 | 40.9% |
| 4 | 7, 7, 3 & 5 | 3 | 325.54 | 367.80 | 53% |
| 5 | 4, 7, 4, 4 & 3 | 4 | 273.61 | 419.73 | 60.5% |

**Question (b)**

Choosing the number of clusters is the most difficult part of any cluster analysis. An ideal number of clusters can group the variables based on characteristics that can be very informative about any data.

There are many methods used to find the optimal levels of clusters. One such method is elbow method. It measures the drop-in total within SoS. In *figure 2* we can see a sharp drop from clusters 2 to 3 and a smaller drop from cluster 3 to 4. The proportion of between SoS and total SoS also sees a large increase from cluster 2 to 3 and small increase from 3 to 4. Ideally a higher proportion (closer to 1) is preferable.

Another method of determining ideal number of clusters is Bayesian information criterion (BIC). It looks at several different models and their optimal levels based on number of clusters. The BIC chart in *figure 2* shows 4 to be the ideal number of clusters.

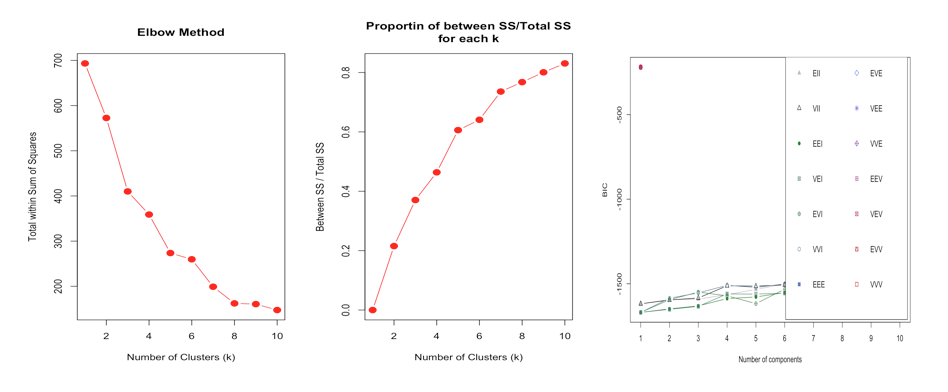


Figure : Elbow method (left), proportion of sum of errors (Centre), BIC (right)

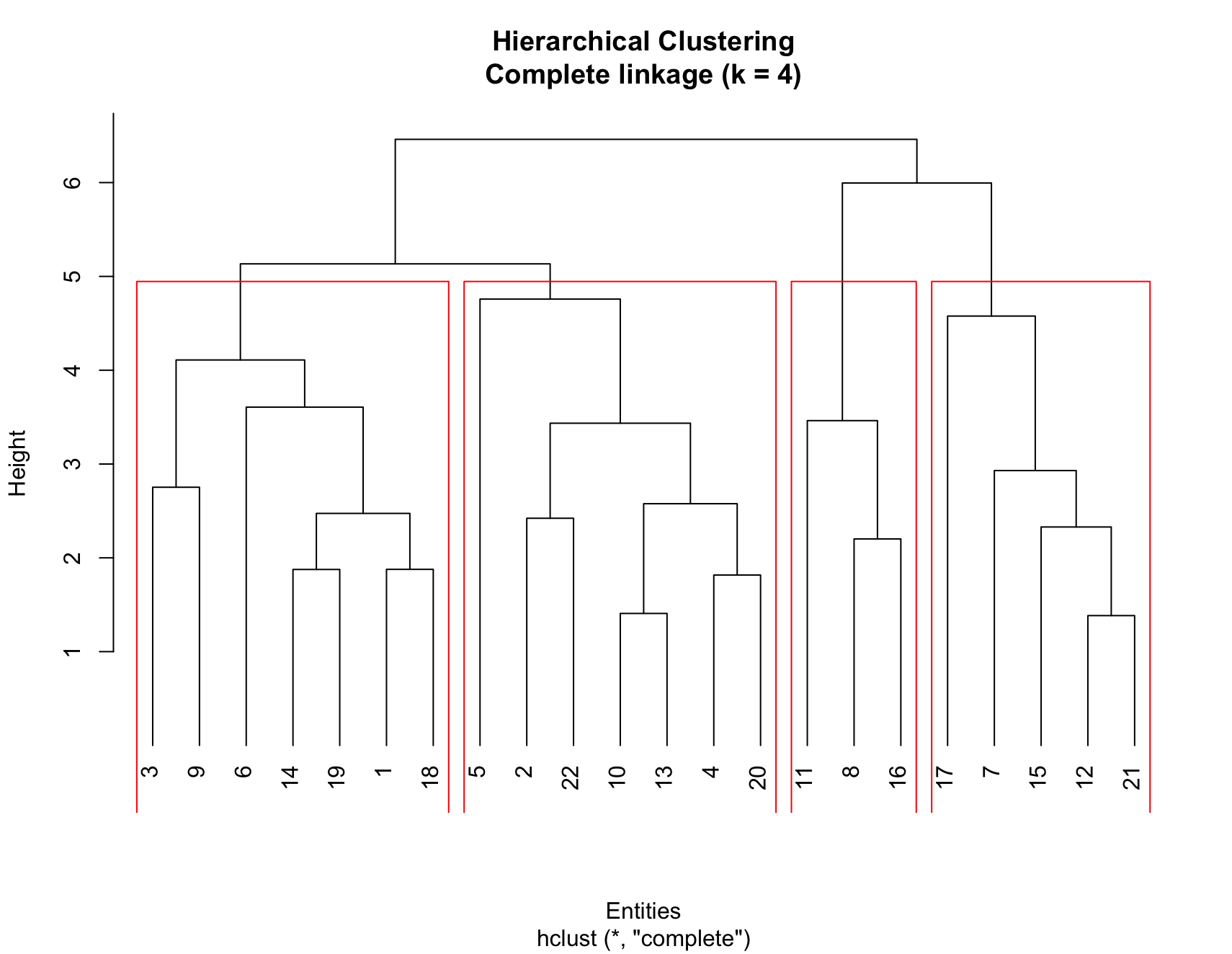
There are many internal and stability measures to determine ideal clusters. Out of 7 such measures three (Silhouette, ADM, FOM) choose 4 clusters as ideal and only one (APN) choose 3 clusters.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Table 2: Internal and External measures for optimum number of Clusters* | | | | |
| **Name** | **Measure Type** | **Score** | **Method** | **Optimal Clusters** |
| Connectivity | Internal | 7.30 | k-means | 2 |
| Dunn | Internal | 0.23 | k-means | 5 |
| Silhouette | Internal | 0.62 | k-means | 4 |
| APN | Stability | 0.04 | k-means | 3 |
| AD | Stability | 921.59 | k-means | 5 |
| ADM | Stability | 219.38 | k-means | 4 |
| FOM | Stability | 349.46 | k-means | 4 |

In part (a) we decided on 3 clusters on the primary basis of just by looking at the scatterplot. Now we have the option of choosing between 3 and 4 clusters.

**Question (c)**

Hierarchical clustering is another widely used clustering method. It starts with *n-*number of clusters where each element is in its own cluster and joins similar clusters one by one till they all merge in two groups. In *figure 3* all entities are clustered together using complete linkage method based on Euclidean distance. Complete linkage concatenates items together which are furthest apart in clusters.



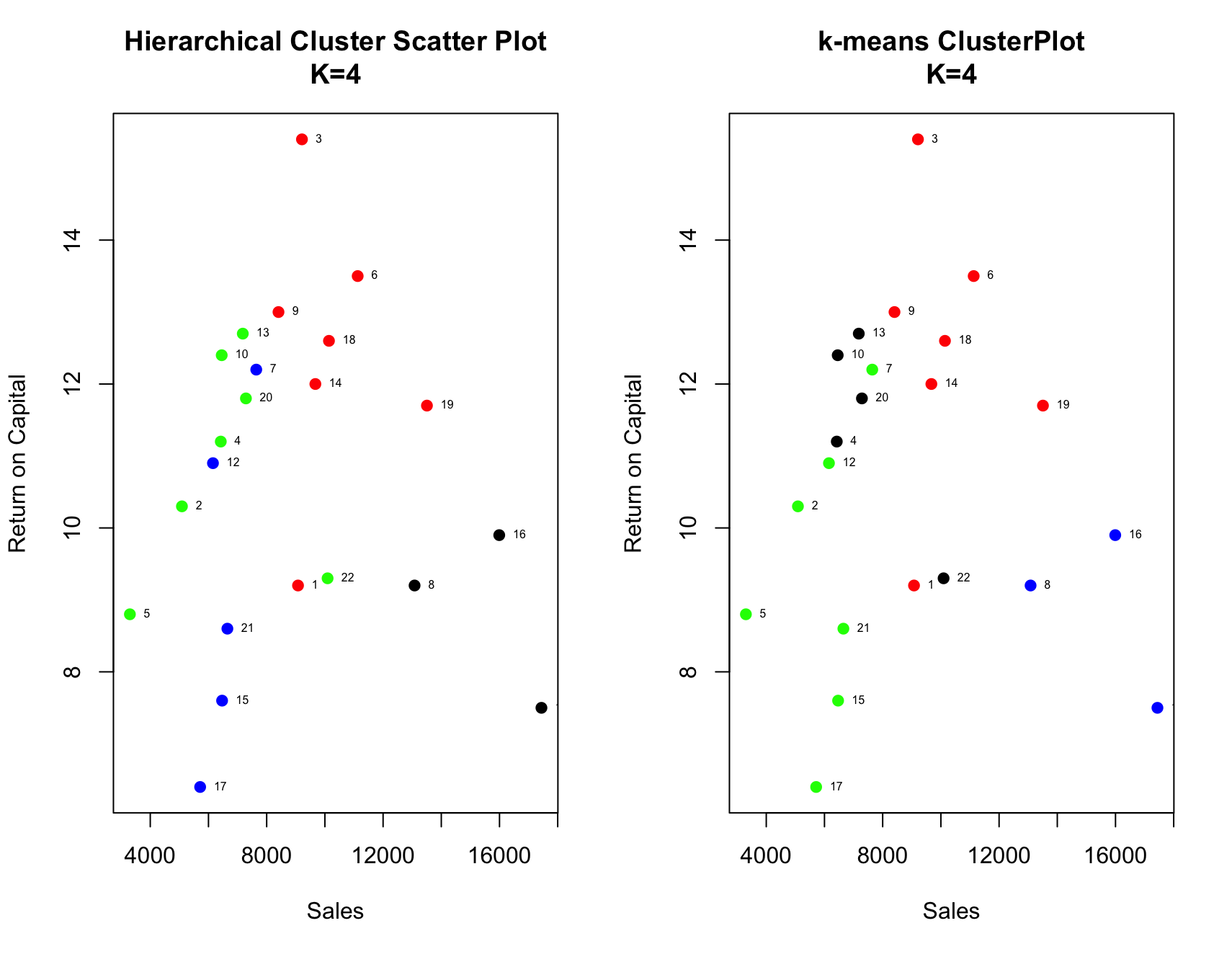
Figure 3: Hierarchical Clustering (complete linkage)

Figure 4: Hierarchical Clustering VS k-means clustering

When we compare the Hierarchical clustering with k-means clustering we can notice that the size of the groups is same (7, 7, 3 and 5) but there’s a notable difference in the distribution inside clusters (*figure 4),* especially entities in green clusters. We use complete linkage to cluster our data which considers the furthest distance in each group. So each method will cluster different entities together and it’s up to the individual to decide which one to use.

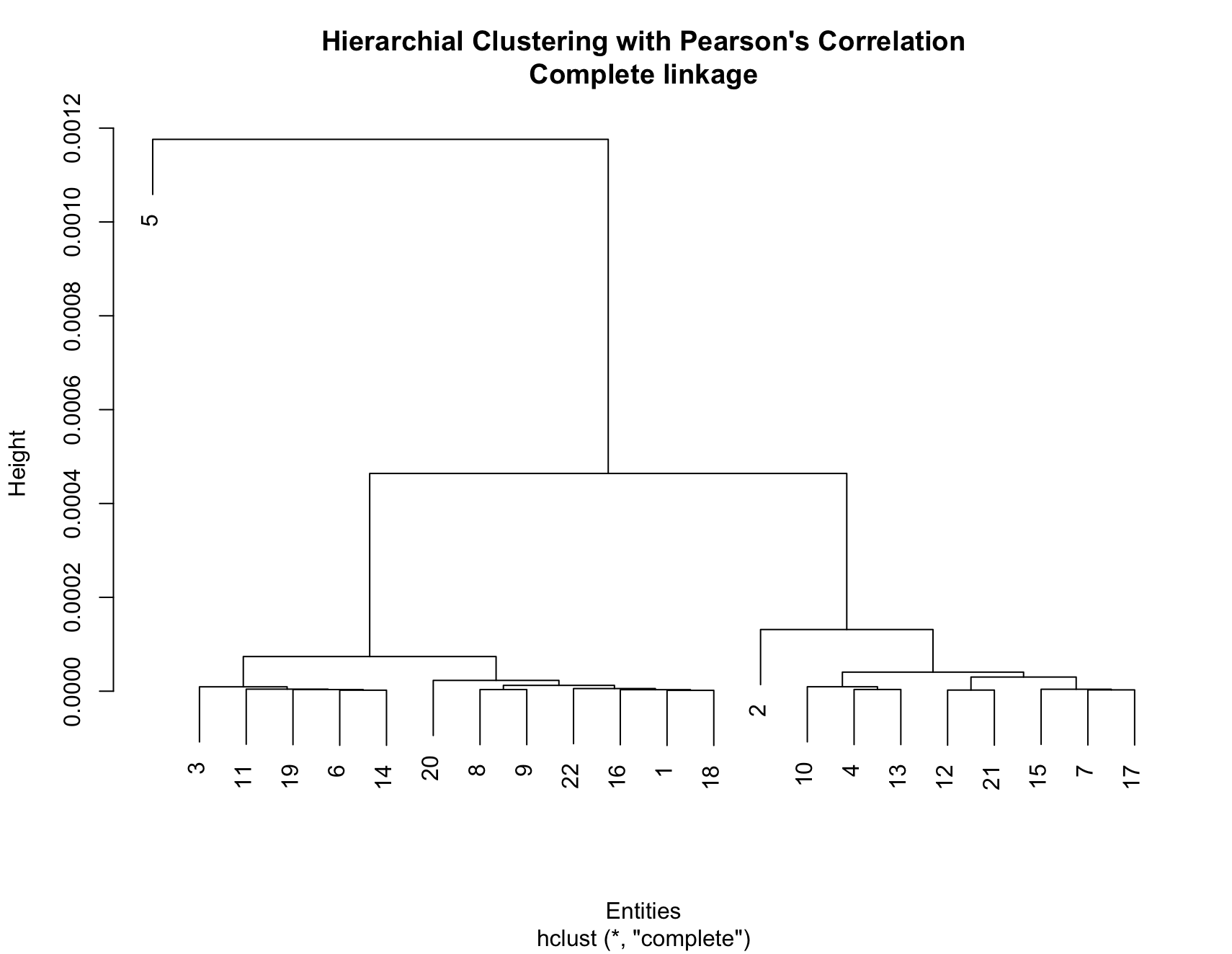
**Question (d)**

Figure : Hierarchical Clustering (complete linkage) with Pearson's Correlation

In *figure 5* we used Pearson’s correlation distance instead of Euclidean distance to group our clusters. Its clusters are quite different from what we got in the previous section. It got 2 clusters of 1 entity each. Entity 5 (Consolid) and entity 2 (Boston) are clustered alone along with two other groups of 12 and 8 entities.

Clusters which are extremely large or small are not that helpful in business decision making, but is useful in identifying outliers.

**Question (e)**

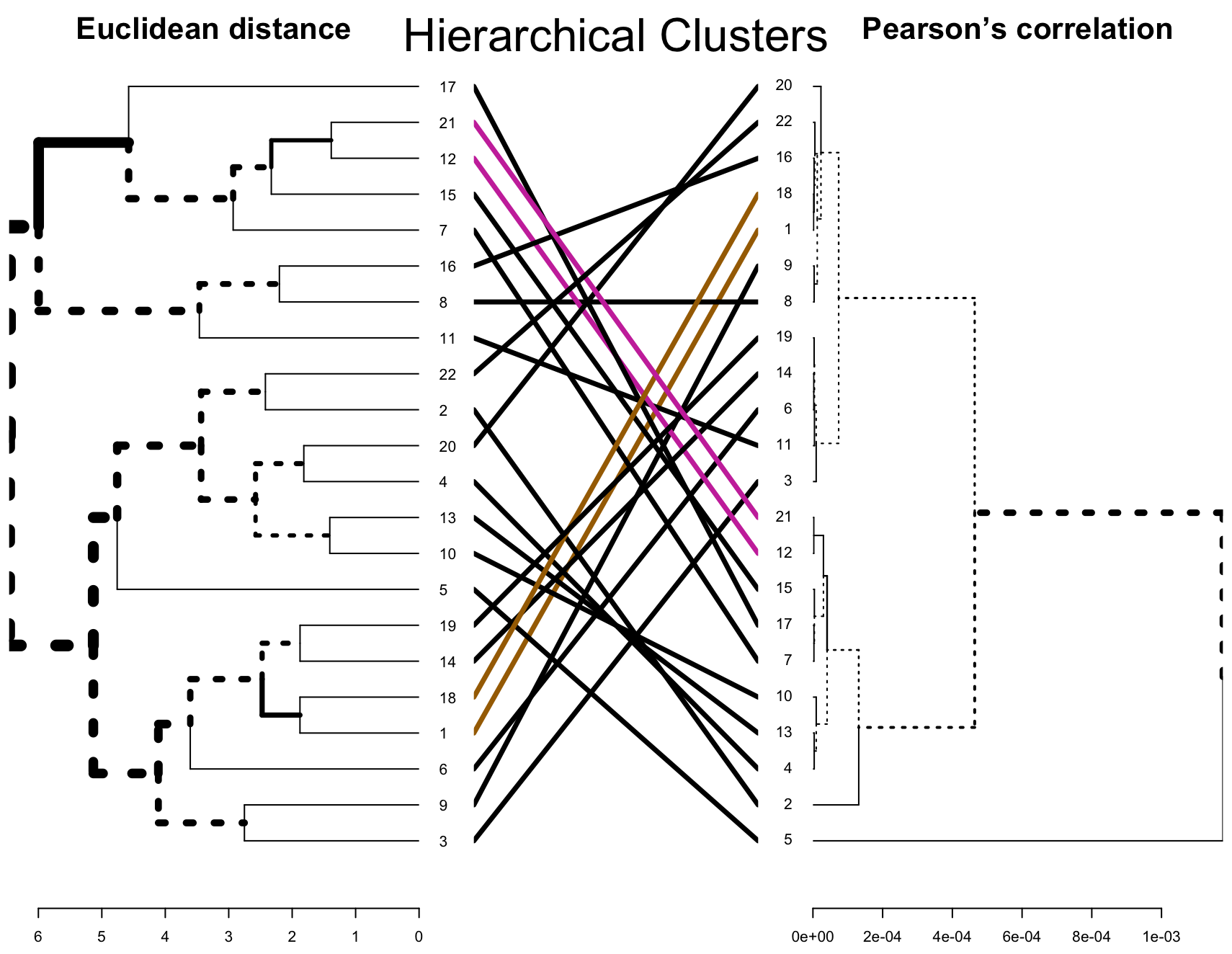
When we compare all three methods we can see the they all produce different results. Some entities that are clustered together in k-means clustering are don’t cluster together under hierarchical method unless we move up to 2 or 1 clusters. Even the two Hierarchical methods produce significant difference as shown in the dendrogram comparison in *figure 6.*

Figure 6: Difference in results in the two Hierarchical clustering methods

The advantage of Hierarchical method is that we get the result in form of a dendrogram and we don’t need to reproduce results based on how many clusters we decide to produce. That’s a major advantage it has over k-means clustering where we need to pre-specify the number of clusters. Even though there are many internal and external measures to determine the optimal numbers of clusters it does require some trial and error.

Form a presentation point of view Hierarchical method seems most appropriate. As statistician are often required to present their data to non-statistician a dendrogram seems the ideal way to present your findings.

APPENDIX Part 1

**Question (a)**

*# read file*

utilities <- read.csv(file.choose())

attach(utilities)

*# mean of variables*

mean(X6)

mean(X8)

mean(X1)

*# Scaling and Euclidean distance*

utilities.scale <- scale(utilities[, 2:9])

utilities.dist <- dist(utilities.scale, method = “uclidean”)

*# clustering with k = 2, 3, 4, 5*

kmeans\_2 <- kmeans(utilities.dist, 2, 25)

kmeans\_3 <- kmeans(utilities.dist, 3, 25)

kmeans\_4 <- kmeans(utilities.dist, 4, 25)

kmeans\_5 <- kmeans(utilities.dist, 5, 25)

*# color specification for consistency*

colo <- c(“red”,”green”,”blue”,”black”,”purple”)

par(mfrow = c(2, 2))

*# plot k = 2*

plot(X2~X6, col = colo[kmeans\_2$cluster], xlab = “Sales”, ylab = “Return on Capital”,

main = “Cluster Scatter Plot\nK=2”, pch = 19) +

with(text(X2~X6, pos = 4, cex = .5))

*# plot k = 3*

plot(X2~X6, col = colo[kmeans\_3$cluster], xlab = “Sales”, ylab = “Return on Capital”,

main = “Cluster Scatter Plot\nK=3”, pch = 19) +

with(text(X2~X6, pos = 4, cex = .5))

*# plot k = 4*

plot(X2~X6, col = colo[kmeans\_4$cluster], xlab = “Sales”, ylab = “Return on Capital”,

main = “Cluster Scatter Plot\nK=4”, pch = 19) +

with(text(X2~X6, pos = 4, cex = .5))

*# plot k = 5*

plot(X2~X6, col = colo[kmeans\_5$cluster], xlab = “Sales”, ylab = “Return on Capital”,

main = “Cluster Scatter Plot\nK=5”, pch = 19) +

with(text(X2~X6, pos = 4, cex = .5))

*# clusters information*

*#k =2*

kmeans\_2

kmeans\_2$iter

kmeans\_2$betweenss

kmeans\_2$tot.withinss

*#k = 3*

kmeans\_3

kmeans\_3$iter

kmeans\_3$betweenss

kmeans\_3$tot.withinss

*#k = 4*

kmeans\_4

kmeans\_4$iter

kmeans\_4$betweenss

kmeans\_4$tot.withinss

*#k = 5*

kmeans\_5

kmeans\_5$iter

kmeans\_5$betweenss

kmeans\_5$tot.withinss

**Question (b)**

*# ELBOW METHOD for SoS*

total\_within <- NULL

for(i in 1:10){

total\_within <- c(total\_within, kmeans(utilities.dist, i, 25)$tot.withinss)

}

plot(total\_within, type = “b”, xlab = “Number of Clusters”, ylab = “Total within Sum of Squares”,

main = “Elbow Method”, pch = 19, cex = 1.5, col = “red”)

*# between SS / Total SS proportion*

*# create a list*

k\_utilities <- list()

# loop and store % in list

for (i in 1:10){

k\_utilities[[i]] <- kmeans(utilities.dist, i)

}

k\_utilities

*# store propotion chart of between SS / total SS for k 1 to 10 in the list*

bet.SS\_tot.SS <- list()

for (i in 1:10){

bet.SS\_tot.SS[[i]] <- k\_utilities[[i]]$betweenss/k\_utilities[[i]]$totss

}

plot(1:10, bet.SS\_tot.SS, type = “b”, ylab = “Between SS / Total SS”, xlab = “Number of Clusters (k)”,

main = “Proportin of between SS/Total SS\nfor each k”, pch = 19, cex = 1.5, col = “red”)

*#plot(gap\_check) Bayesian information criteria*

d\_clust <- Mclust(utilities.dist, G=1:10,

modelNames = mclust.options(“emModelNames”))

d\_clust$BIC

plot(d\_clust)

*# Both internal and stability measures*

int\_stab <- clValid(utilities[, 2:9], nClust = 2:5, clMethods = “kmeans”,

validation = c(“internal”, “stability”))

int\_stab

optimalScores(int\_stab)

**Question (c)**

*# clustering, plotting & border*

complete\_hclust <-hclust(utilities.dist, method = “complete”)

plot(complete\_hclust, hang = -1, xlab = “Entities”, main = “Hierarchial Clustering\nComplete linkage (k = 4)”)

rect.hclust(complete\_hclust, k = 4, border = “red”)

*# Both internal and stability measures*

int\_stab\_2 <- clValid(utilities[, 2:9], nClust = 2:5, clMethods = c(“kmeans”, “hierarchial”),

validation = c(“internal”, “stability”))

int\_stab\_2

optimalScores(int\_stab\_2)

*# k-means VS hierarchical*

*# plotting both plots and comparing them*

par(mfrow = c(1, 2))

plot(X2~X6, col = colo[kmeans\_4$cluster], xlab = “Sales”, ylab = “Return on Capital”,

main = “k-means ClusterPlot\nK=4”, pch = 19) +

with(text(X2~X6, pos = 4, cex = .5))

*# cluster map for plotting color on scatter plot*

cluster\_map <- cutree(complete\_hclust, 4)

cluster\_map

plot(utilities[,c(7,3)], col = colo[cluster\_map], xlab = “Sales”, ylab = “Return on Capital”,

main = “Hierarchical Cluster Scatter Plot\nK=4”, pch = 19) +

with(text(X2~X6, pos = 4, cex = .5))

**Question (d)**

*# calculating correlation matrix of after transpose*

cor\_ut <- cor(t(utilities[,2:9]))

#ecludean distance

dist\_cor\_ut <- as.dist(1-abs(cor\_ut))

*# clustering, plotting & border*

cor\_complete\_hclust <-hclust(dist\_cor\_ut, method = “complete”)

plot(cor\_complete\_hclust, hang = -1, xlab = “Entities”, main = “Hierarchial Clustering with Pearson’s Correlation\nComplete linkage (k = 4)”)

rect.hclust(cor\_complete\_hclust, k = 4, border = “red”)

*# cluster map for plotting color on scatter plot*

cluster\_map\_2 <- cutree(cor\_complete\_hclust, 4)

cluster\_map\_2

*# ploting Hierarchical Cluster With Pearson’s Correlation VS without*

par(mfrow = c(1, 2))

plot(utilities[,c(7,3)], col = colo[cluster\_map\_2], xlab = “Sales”, ylab = “Return on Capital”,

main = “Hierarchical Cluster Plot\nWith Pearson’s Correlation (K=4)”, pch = 19) +

with(text(X2~X6, pos = 4, cex = .5))

plot(utilities[,c(7,3)], col = colo[cluster\_map], xlab = “Sales”, ylab = “Return on Capital”,

main = “Hierarchical Cluster Scatter Plot\nK=4”, pch = 19) +

with(text(X2~X6, pos = 4, cex = .5))

**Question (e)**

# a tanglegram comparing dendrogram by Hierarchical complete method with dendrogram by pearnson correlation

den <- dendlist(as.dendrogram(complete\_hclust), as.dendrogram(cor\_complete\_hclust))

tanglegram(den, sort = TRUE, common\_subtrees\_color\_lines = T, highlight\_distinct\_edges = T, highlight\_branches\_lwd = T,

main\_left = “Euclidean distance”, main\_right = “Pearson’s correlation”, main = “Hierarchical Clusters”)

PART 2

**Summary of the paper**

One of the biggest challenge statisticians face is expressing multivariate data in a way that is understandable to everyone. In 1973 Herman Chertoff came up with a unique way of representing and clustering data. He assigned each variable a special feature of the face and started drawing faces that represents each item and feature of the faces depends on the variables they represent. He used the nummulited fossil data and the mineral data and used it to make faces and asked his colleague to group the faces based on visible similarities.

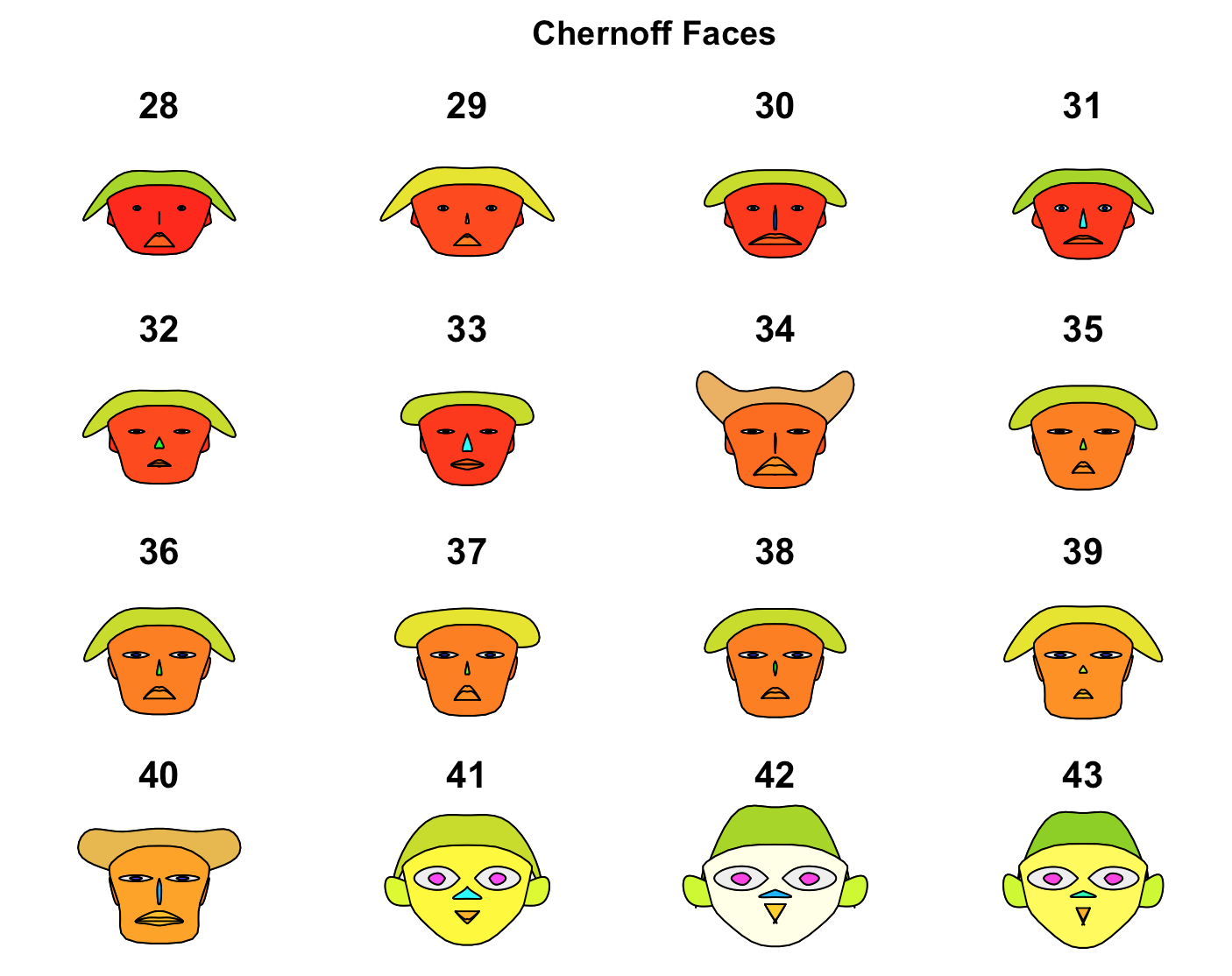


Figure 7: Chernoff faces

There were 88 nummlited specimen from the Eocene yellow limestone formation Jamaica. It had 8 variables but only 6 were used as 2 were considered age dependent, and 87 specimens were used as one was rejected. These were used to form faces like the one in *figure 7* on the right (the faces in *figure 7* were drawn using r)*.* The original drawings werehand drawn, black & white and looks very simple. These drawings were shown to people and they were asked to group them together on similar characteristics.

A similar process was performed on mineral data from 4500-foot core drilled from Colorado mountainside. 53 faces were drown based on 12 variables. The faces showed obvious signs of similarity and differences which correlated with the data.

**Can Chernoff faces be used to judge the quality of clusters?**

This method though be fun and gives a unique way of clustering is not a widely used as it’s a cumbersome and time consuming way to group data. But a good use of Chernoff faces could be to check how robust the clustering process is. When a data is clustered into different groups then Chernoff faces within these groups should display similar features.

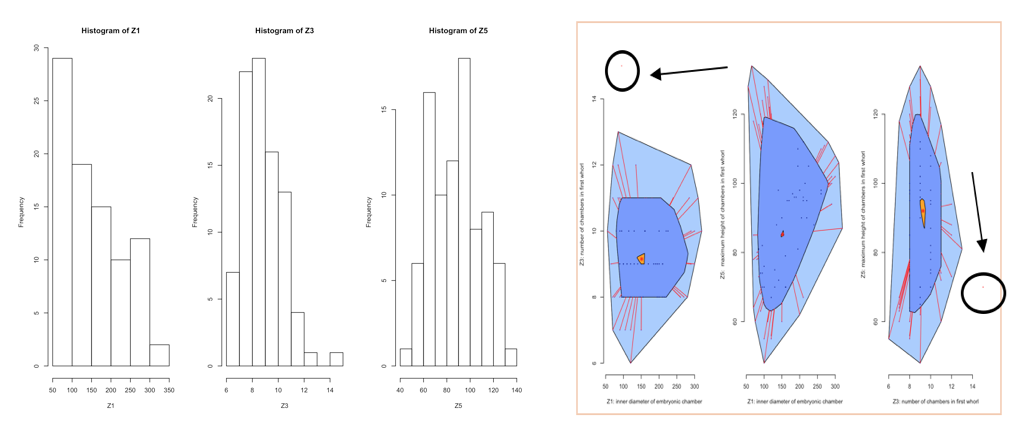
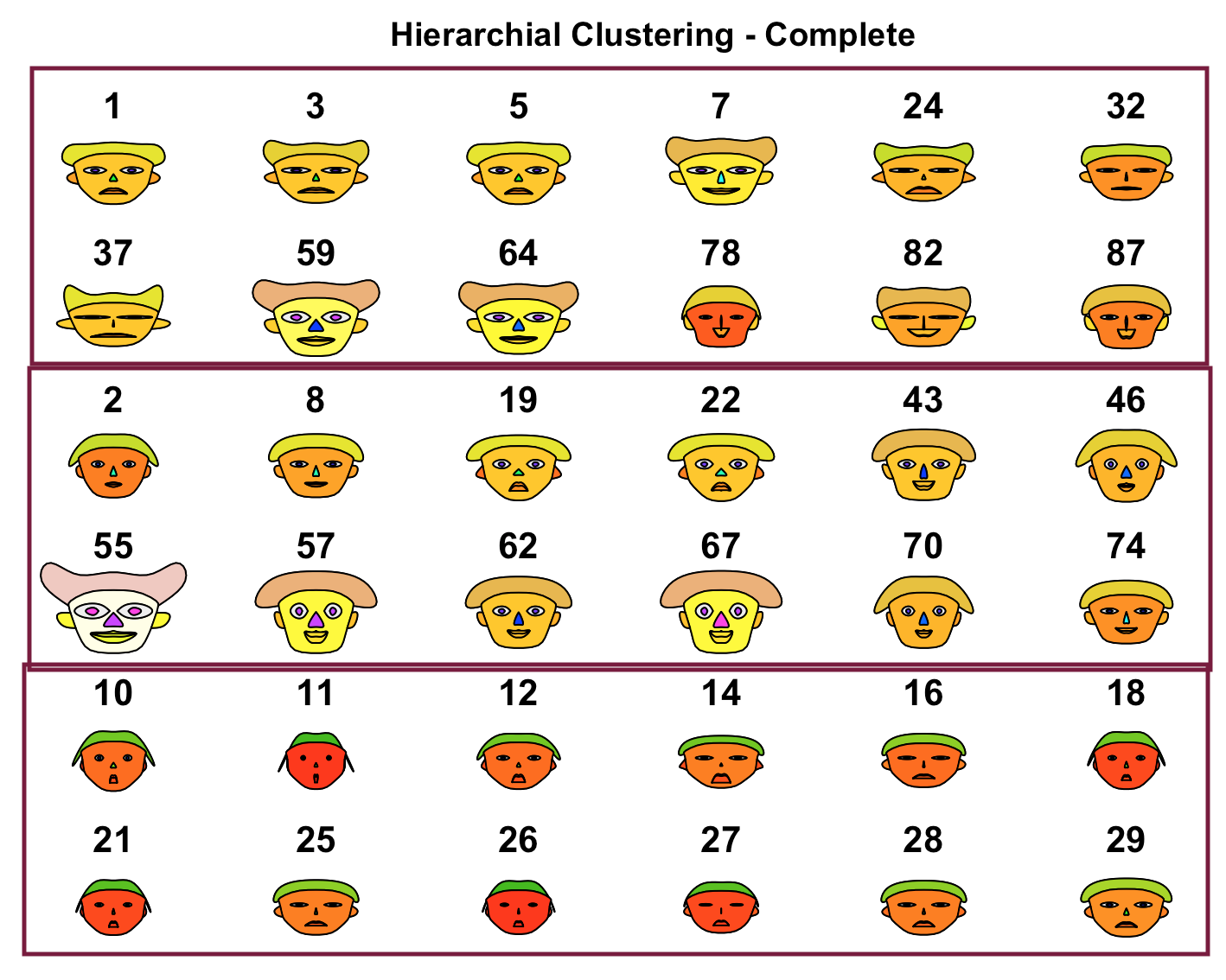


Figure 8: Histogram of Z1, Z2 & Z3 and Bagplot for outliers

Three variables of the nummulited fossils were considered. Z1: inner diameter of embryonic chambers, Z3: number of chambers in the first whorl and Z5: maximum height of chambers in first whorl. By choosing only three variables we are avoiding over fitting and each variable will have more impact in clustering and in characteristics of the faces. Z1 and Z2 are only slightly right skewed which. Bagplot was used (*figure 8*) to detect outliers and one (ID 35, inside circle) was deemed as one and removed from further analysis.

Figure 8: Bagplot for Outliers

These variables were clustered using different hierarchical methods (complete, simple and Ward) into 3 clusters and 12 faces from each cluster were compared. A high-quality cluster will have similar features within clusters and faces in one clusters should be easily distinguishable from other clusters.

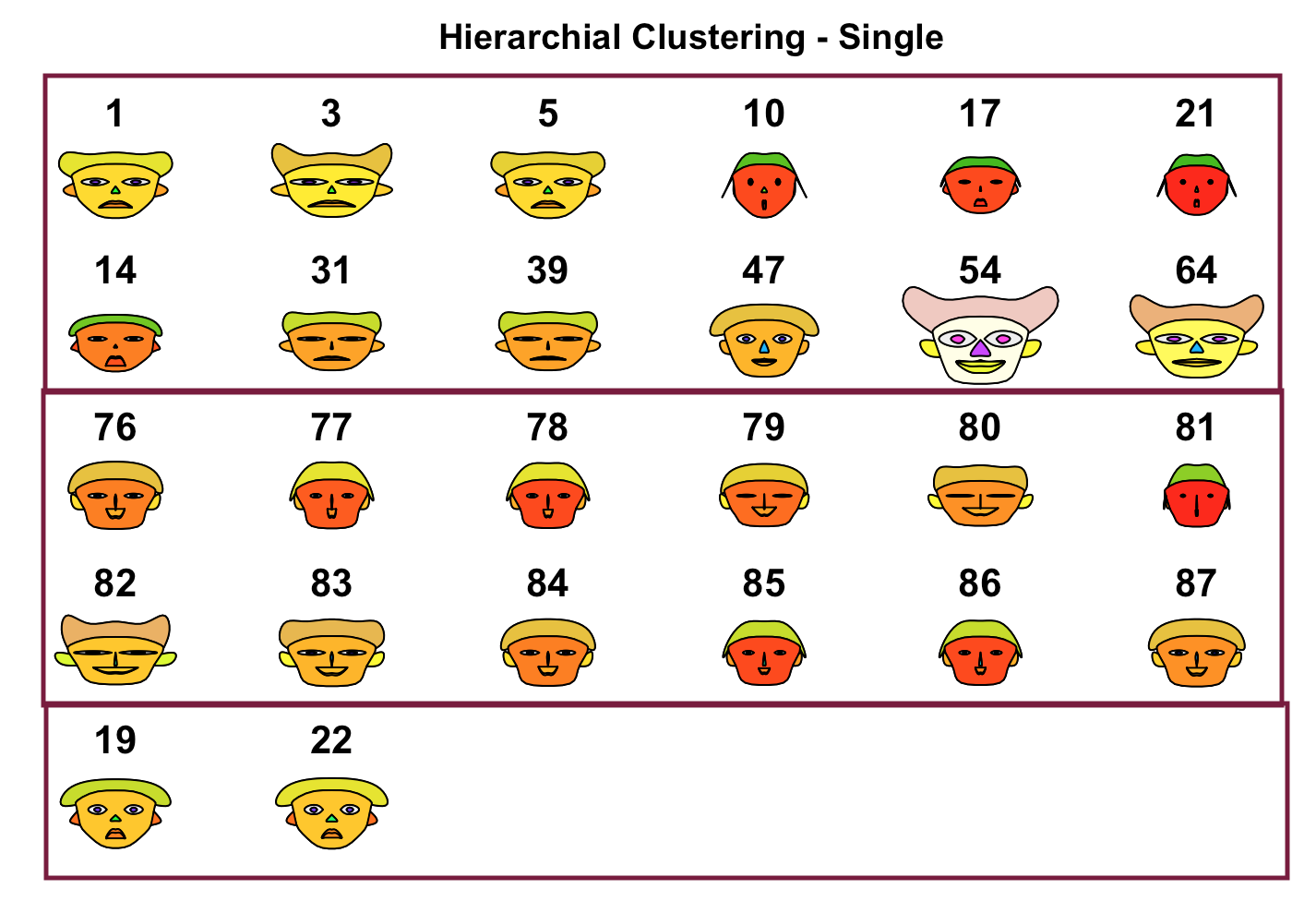
*Figure 9* shows fossils being divides into 3 clusters. There seems to be moderate similarity among clusters. Cluster 1 seems to be most diverse, group 2 seems more similar and maybe another cluster will make the group more homogeneous. Faces in cluster 3 seems to be similar in characteristics.

Figure 9: Complete Linkage

*Figure 10* are faces of hierarchical clustering by single method. Single method clusters items together which are closest to each other which leads to majority of items in one clusters. About 80% of fossils end up in group 1 and that’s why its faces are so dissimilar and group 3 end up with only 2 items. It’s one of the biggest limitation of single linkage and that’s why it’s rarely used.

Figure 10: Simple linkage

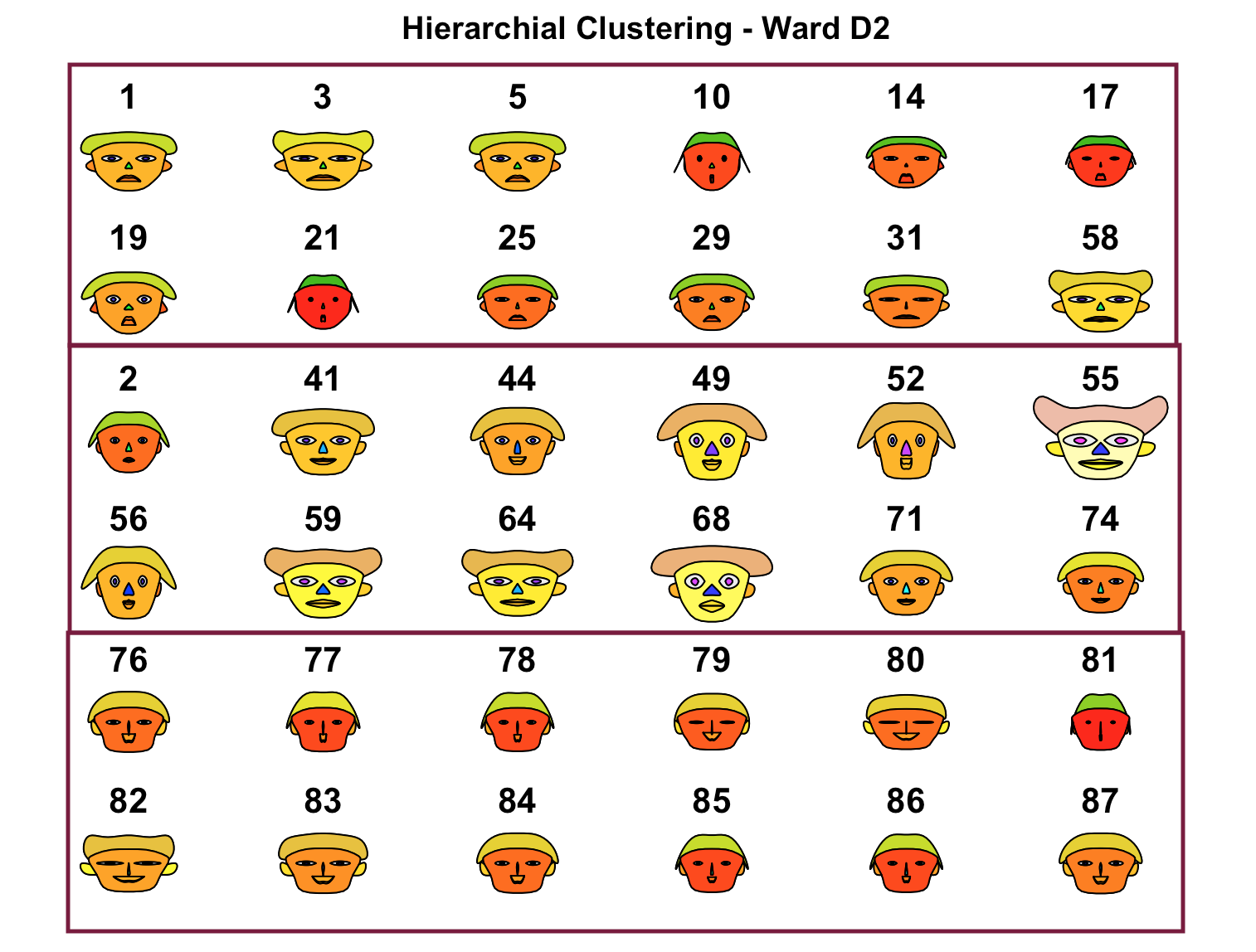


Figure 11: Ward clustering

Cluster 3 (*figure 11)* uses Ward method of clustering and seems the most robust of the three. Group 2 and 3 seems quite homogenous with some variation in group 1.

Chernoff faces might not be the most effective method for cluster analysis and are rarely used but they can be used as a measure of the quality clusters.

Appendix Part 2

# reading file

chernoff <- read.csv(file.choose())

attach(chernoff)

# drawing Chernoff faces

faces(chernoff[ c(28:43), c(2, 4, 6)], main = "Chernoff Faces")

# plotting Bagplot and looking for outliers

par(mfrow = c(1, 3))

a <- bagplot(cbind(Z1, Z3), xlab = "Z1: inner diameter of embryonic chamber",

ylab = "Z3: number of chambers in first whorl")

b <- bagplot(cbind(Z1, Z5), xlab = "Z1: inner diameter of embryonic chamber",

ylab = "Z5: maximum height of chambers in first whorl")

c <- bagplot(cbind(Z3, Z5), xlab = "Z3: number of chambers in first whorl",

ylab = "Z5: maximum height of chambers in first whorl")

a$pxy.outlier

b$pxy.outlier

c$pxy.outlier

# removing outlier, scaling and calculating Euclidean distance

chernoff\_2 <- chernoff[-34, c(2, 4, 6)]

chernoff\_2

chernoff\_2.scale <- scale(chernoff\_2)

chernoff\_2\_dist <- dist(chernoff\_2.scale, method = "euclidean")

# hierarchical clustering using complete linkage method

chernoff\_hc <- hclust(chernoff\_2\_dist, method = "complete")

chernoff\_hc\_cuttree <- cutree(chernoff\_hc, 3)

chernoff\_hc\_cuttree

# faces – 12 of each group

faces(chernoff\_2[c(1, 3, 5, 7, 24, 32, 36, 58, 63, 77, 81, 86,

2, 8, 19, 22, 42, 45, 54, 56, 61, 66, 69, 73,

10, 11, 12, 14, 16, 18, 21, 25, 26, 27, 28, 29), ], main = "Hierarchial Clustering - Complete")

# hierarchical clustering using single linkage method

chernoff\_hs <- hclust(chernoff\_2\_dist, method = "single")

chernoff\_hs\_cuttree <- cutree(chernoff\_hs, 3)

chernoff\_hs\_cuttree

# faces – 12 of each group

faces(chernoff\_2[c(1, 3, 5, 10, 17, 21, 14, 31, 38, 46, 53, 63,

75:86,

19, 22), ], main = "Hierarchial Clustering - Single")

# hierarchical clustering using ward 2 method

chernoff\_hw <- hclust(chernoff\_2\_dist, method = "ward.D2")

chernoff\_hw\_cuttree <- cutree(chernoff\_hw, 3)

chernoff\_hw\_cuttree

# faces – 12 of each group

faces(chernoff\_2[c(1, 3, 5, 10, 14, 17, 19, 21, 25, 29, 31, 57,

2, 40, 43, 48, 51, 54, 55, 58, 63, 67, 70, 73,

75:86), ], main = "Hierarchial Clustering - Ward D2")