Hierarchical Cluster Analysis

Johnny Lo

June 2021

Table of Contents

Hierarchical Cluster Analysis 2
Divisive Analysis (DIANA) 5
Agglomerative Nesting (AGNES) 9
Combined Visualisation of Heatmaps and Dendrograms 12

Hierarchical Cluster Analysis

Hierarchical clustering is a technique for identifying groups or clusters in multivariate datasets. We do not need to pre-specify the number of clusters beforehand in order to peform the analysis, unlike say, *k*-means clustering (not part of the unit). A great feature of hierarchical clustering is that the sequence of groupings of observation or features is represented in a tree-based diagram, called **dendrogram**.

There are two main types of hierarchical clustering:

1. Agglomerative clustering - a bottom-up clustering approach and is also known as Agglomerative Nesting (AGNES).

2. Divisive clustering - a top-down clustering approach and is also known as Divisive Analysis (DIANA).

In this workshop, we will implement both approaches.

To start, you need to install and load the relevant packages for this workshop.

```
#De-comment to install the packages below
#install.packages(c("tidyverse","timeDate","dendextend","cluster","factoextra
","gplots",
#                   "mvbund","vegan","RColorBrewer"))
library(tidyverse)
library(cluster)  #For AGNES and DIANA
library(dendextend)  #For dendrogram and for calculating the cophenetic
correlation
library(factoextra)  #For visualisation of clusters
library(gplots)  #For the heatmaps
library(mvabund)  #For spider dataset
library(vegan)  #For vegdist(.) function
library(RColorBrewer)  #For more colour palettes
```

We begin with the public utility data of 22 U.S. power companies, which are given in the *Public Utility.csv* file. The relevant features are as follows:

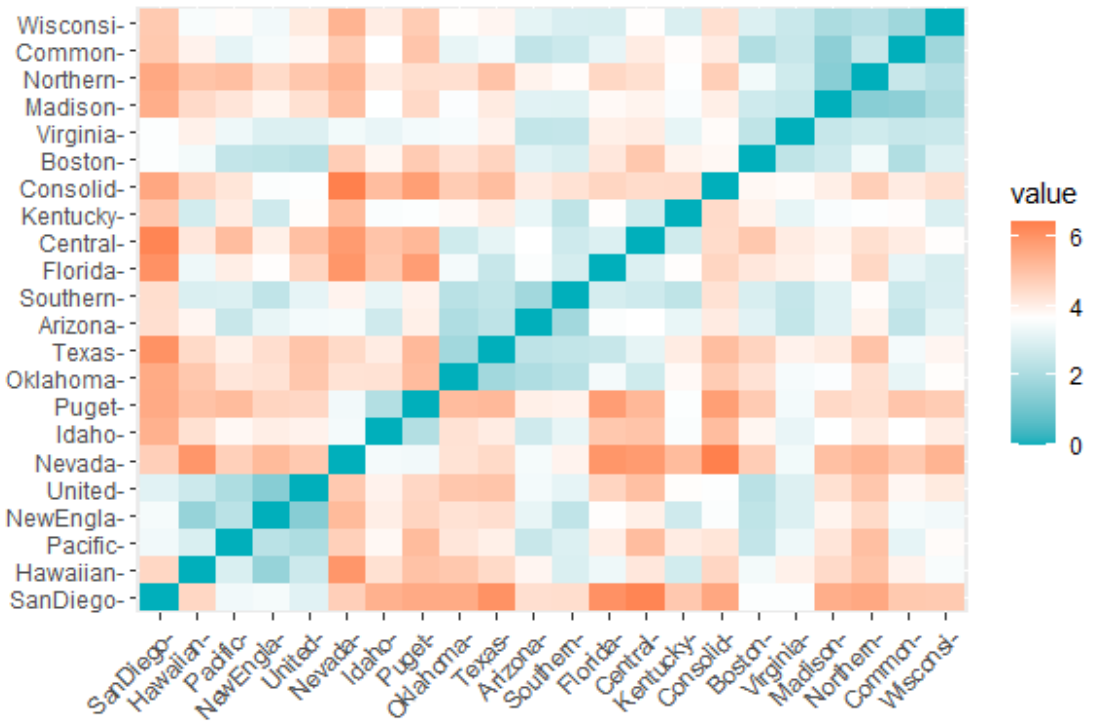
1. Fixed charge coverage ratio (income/debt)
2. Rate of return on capital
3. Cost per KW capacity in place
4. Annual load factor
5. Peak KWH demand growth
6. Sales (KWH use per year)
7. Percent nuclear
8. Total fuel cost (cents per KWH)

Given that the features are defined in different units, we should standardise the data before we create the distance matrix for hierarchical clustering.

```
dat <- read.csv("Public Utility.csv",header=TRUE,stringsAsFactors=TRUE);
rownames(dat) <- dat[,ncol(dat)]; #Name the rows by the company names
dat <- dat[,1:(ncol(dat)-1)]; #Remove the "Company" variable as it's not
required anymore.
View(dat)

dat <- na.omit(dat)  #Omit missing values although there isn't any here.
dat.z <- scale(dat);  #Standardise the variables
```

Next we will create the distance matrix using Euclidean distance, and visualise it with a heatmap.

```
#Create the distance matrix with Euclidean distance.
dist.pu <- dist(dat.z,method="euclidean");

#heat map for distance matrix.
fviz_dist(dist.pu,gradient = list(low = "#00AFBB", mid = "white", high =
"#FC4E07"))
```

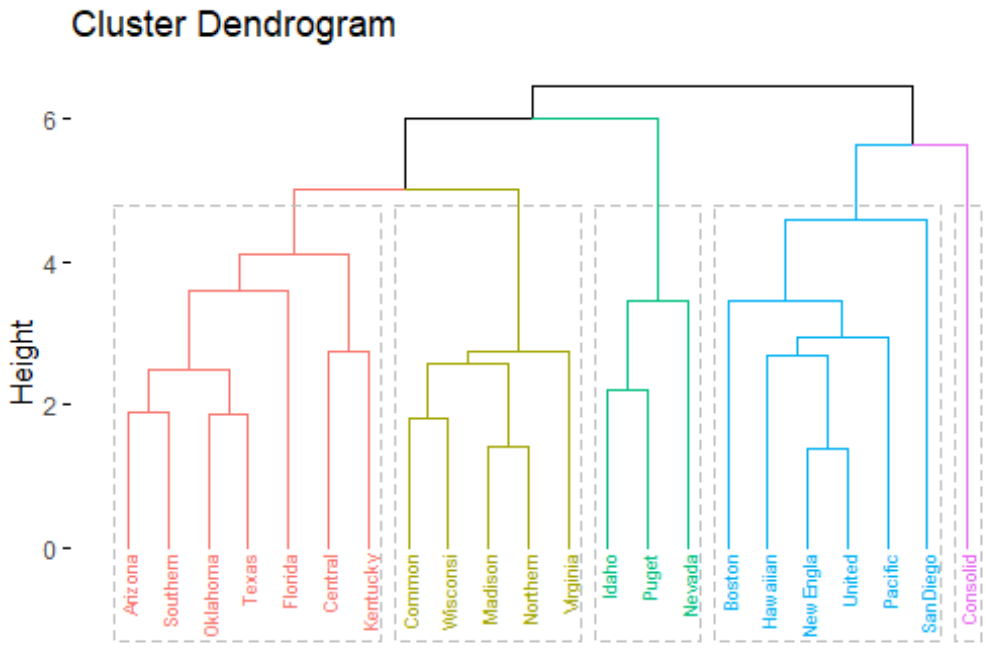


Exercise: How many major clusters can you see here? Are there companies that appear to operate differently to the rest?

Divisive Analysis (DIANA)

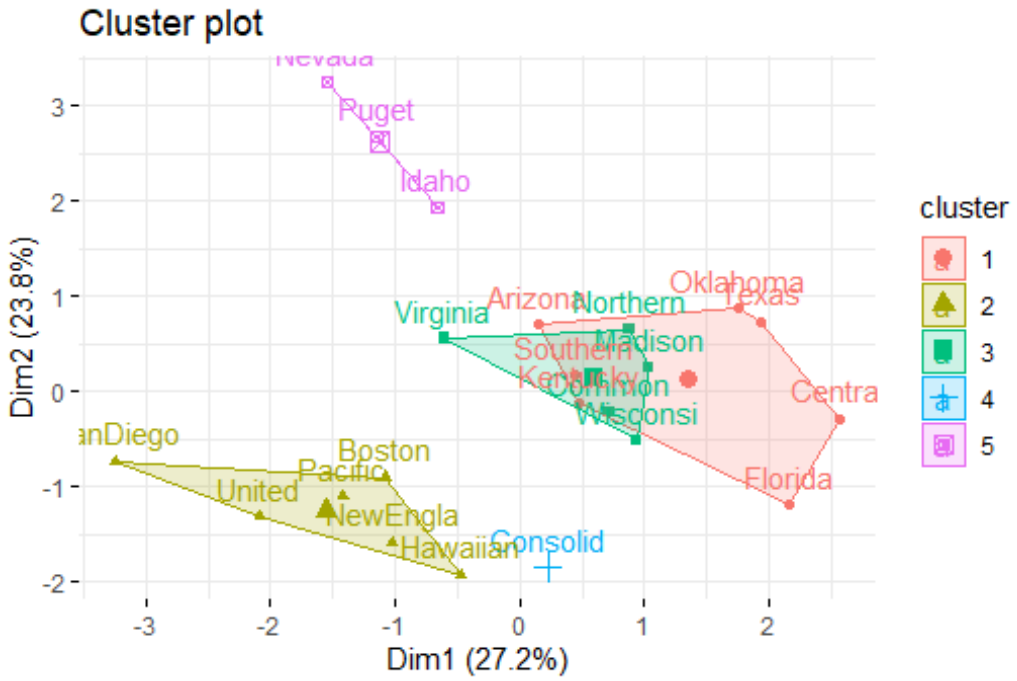
We will firstly perform DIANA on the public utility data.

```
#Divisive hierarchical clustering
dhc.pu <- diana(dat.z,
                #Standardise data? Not require here since data are already
standardised
                stand=FALSE);

#Plot the dendrogram
fviz_dend(dhc.pu,  #Clustering model
          cex=0.5,  #Font size of texts
          k=5,  #Number of clusters. To be defined by user
          color_labels_by_k=TRUE,  #Differentiate cluster by different
colours
          rect=TRUE)  #Draw a rectangle to encompass each cluster
```



We can also visualise the groups in a PCA plot using the *fviz_cluster(.)* function from the **factoextra** package. If Euclidean distance is not used to define the distance matrix, then do not use this function. In this instance, you can create, say a PCoA plot and overlay the information from hierarchical clustering over it.

```
#Assign group membership to objects
Group <- cutree(as.hclust(dhc.pu), k = 5);

#PCA plot where the observations are colour-coded by their respective
clusters
fviz_cluster(list(data=dat.z,cluster=Group),ggtheme=theme_minimal())
```

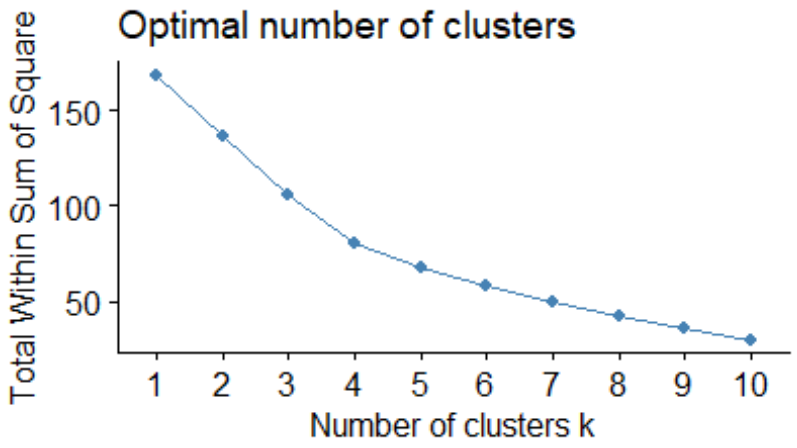


We can refer to the division coefficient to describe the strength of the clustering structure. The cophenetic correlation measures how well the dendrogram preserves the original distances or similarities. Both measures are defined within the 0-1 range. In the absolute sense for both quantities, the closer they are to one, the better. In relative terms, both measures are useful for comparing different clustering methods, i.e. DIANA vs AGNES or between AGNES models with different linkage methods.

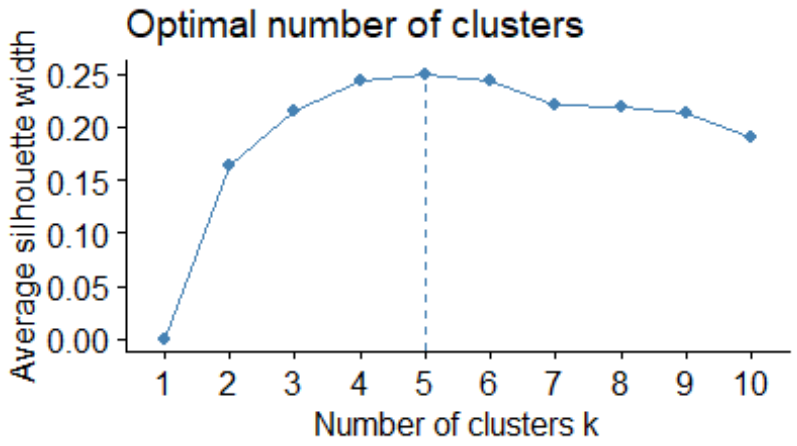
```
divcoef.pu <- dhc.pu$dc; divcoef.pu   #Divisive coefficient

## [1] 0.6081477

coph.dhc.pu <- cor_cophenetic(dhc.pu,dist.pu); coph.dhc.pu #Cophenetic
correlation

## [1] 0.6564673
```

With the previous cluster plot, 5 clusters were arbitrarily chosen for demonstration purpose. To formally decide on the optimal number of clusters, we can refer to the 1) Elbow Method, 2) Average Silhouette Method, and/or 3) Gap Statistic Method.

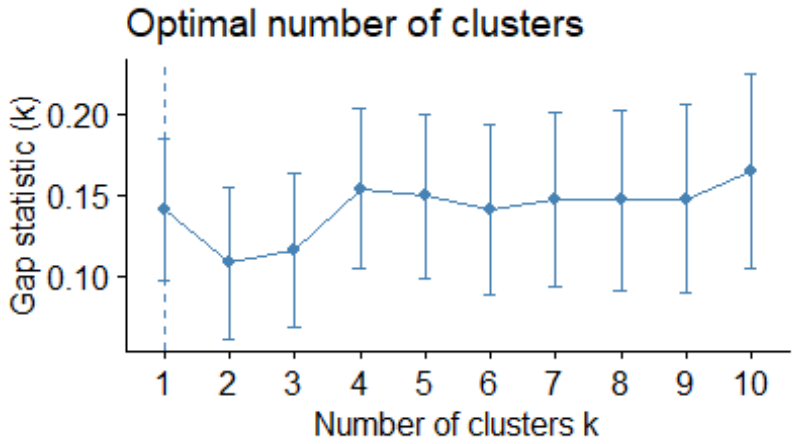
```
#Elbow Method
fviz_nbclust(dat.z, FUN=hcut, method="wss")
```




```r
#Average Silhouette Method
fviz_nbclust(dat.z, FUN=hcut, method="silhouette")
```




```r
#Gap Statistic Method
gap_stat <- clusGap(dat.z, FUN=hcut, nstart=25, K.max=10, B=500)
fviz_gap_stat(gap_stat)
```

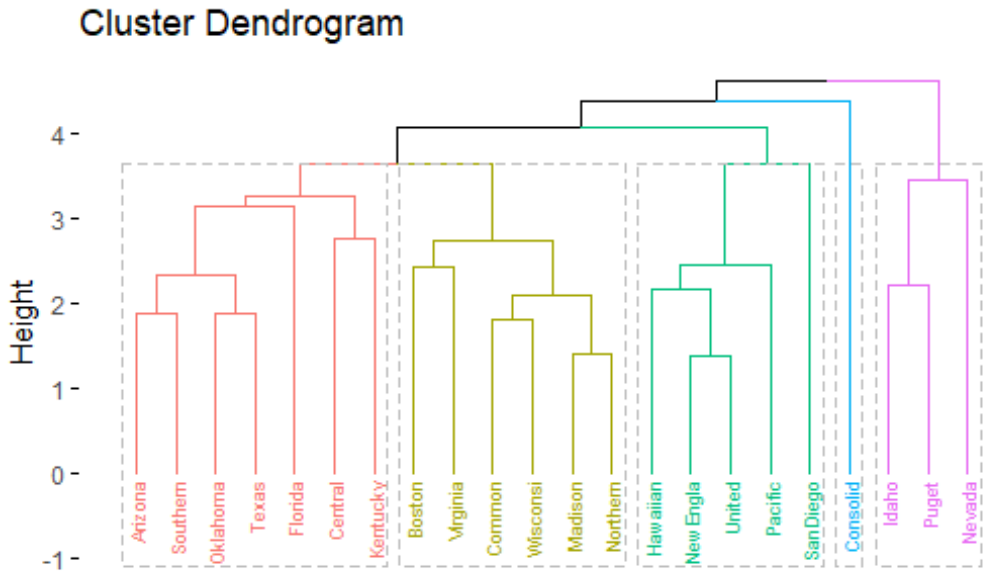


Exercise: How many clusters do you believe is appropriate here?

Agglomerative Nesting (AGNES)

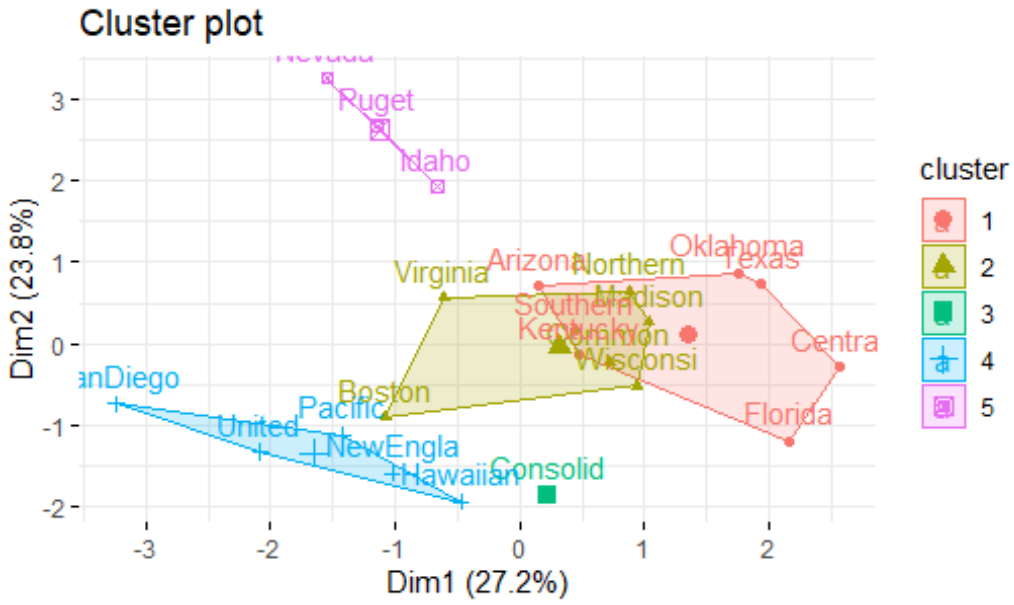
Next, we will perform AGNES using the Average Linkage Method.

```
#Agglomerative hierarchical clustering using Average Linkage
ahc.pu.ave <- agnes(dat.z,method="average");

#Plot the dendrogram
fviz_dend(ahc.pu.ave, cex=0.5, k=5,
          color_labels_by_k=TRUE, rect=TRUE)
```



Next, we can view the results in a scatter plot and whose axes are defined by PC1 and PC2 in from PCA process.

```
Group <- cutree(as.hclust(ahc.pu.ave), k = 5); #Assign group membership to
objects.
fviz_cluster(list(data=dat.z,cluster=Group),ggtheme=theme_minimal())
```



The agglomerative coefficient and cophenetic correlation coefficient are determined as follows.

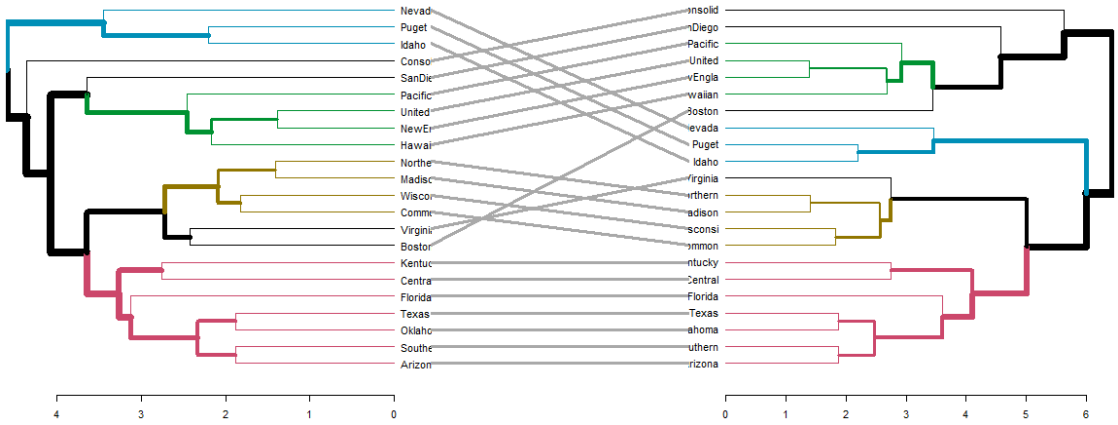
```
aggcoef.ave <- ahc.pu.ave$ac; aggcoef.ave  #Agglomerative coefficient

## [1] 0.5000999

coph.cor.ave <- cor_cophenetic(ahc.pu.ave,dist.pu); coph.cor.ave  #Cophenetic
corrrelation

## [1] 0.7044192
```

Exercise: Based on the agglomerative and divisive coefficients, and the cophenetic correlation coefficients for AGNES and DIANA, which approach do you think is better?

We can also compare the dendrograms from DIANA and AGNES and see how well they align with each other (1) quantitatively using the entanglement measure, and (2) visually using tanglegram plot. Entanglement is a measure between 0 (no entanglement) and 1 (full entanglement). The lower the entanglement coefficient, the better the alignment.

```
#Convert clustering information to dendrograms
dend1 <- as.dendrogram(ahc.pu.ave);
dend2 <- as.dendrogram(dhc.pu);

entanglement(dend1,dend2)  #Entanglement measure

## [1] 0.138784

tanglegram(dend1,dend2,
           highlight_distinct_edges = FALSE, # Turn-off dashed lines
```

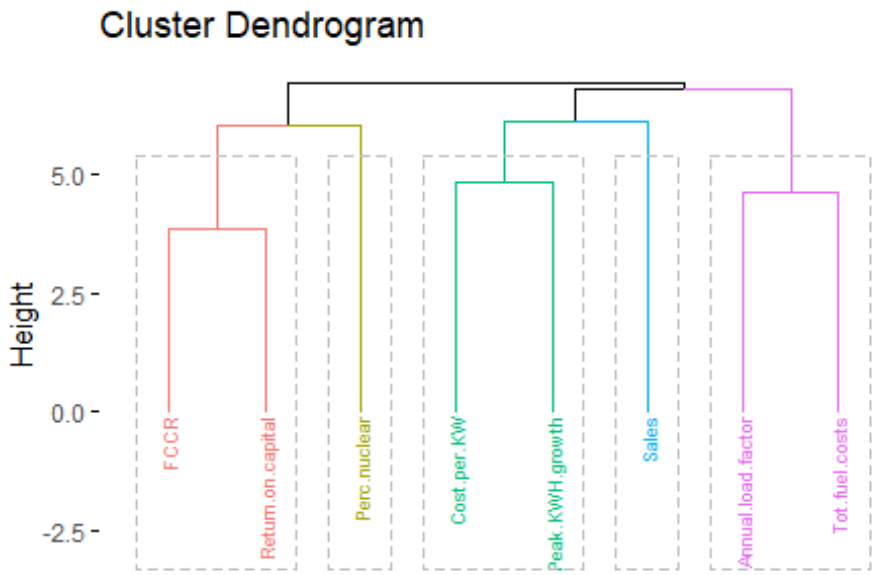
```
            common_subtrees_color_lines = FALSE, # Turn-off line colors
            common_subtrees_color_branches = TRUE) # Colour common branches
```



Exercise: Perform AGNES with **single** and **complete** linkage methods, and compare them to the average linkage method.

By transposing the data matrix and then applying either AGNES or DIANA, we switch our focus to the grouping of the features instead.

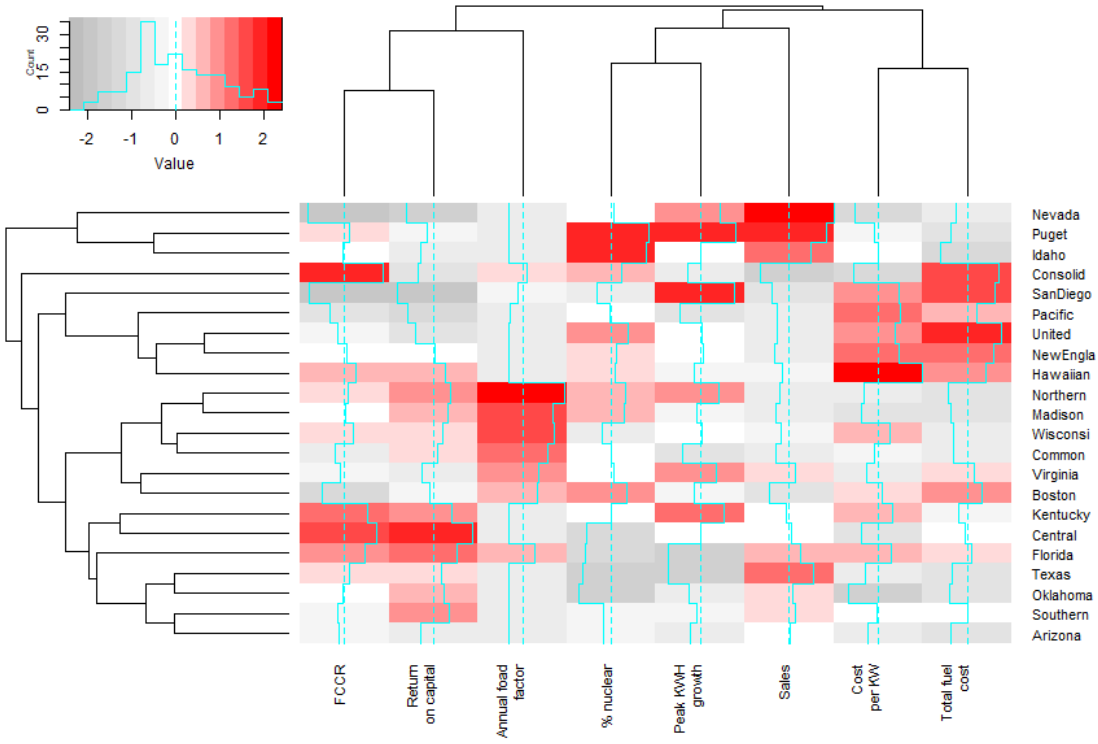
```
#Agglomerative hierarchical clustering using Average Linkage
dat.z.trp <- t(dat.z);   #Transposing the standardised data
dist.pu.trp <- dist(dat.z.trp,method="euclidean");
ahc.pu.ave2 <- agnes(dist.pu.trp,method="average");
aggcoef.ave2 <- ahc.pu.ave2$ac; aggcoef.ave2   #Agglomerative coefficient

## [1] 0.2982958

coph.cor.ave2 <- cor_cophenetic(ahc.pu.ave2,dist.pu.trp); coph.cor.ave2

## [1] 0.866669

#Plot the dendrogram
fviz_dend(ahc.pu.ave2, cex=0.5, k=5,
          color_labels_by_k=TRUE, rect=TRUE, labels_track_height=2.8)
```



Combined Visualisation of Heatmaps and Dendrograms

Now that we have performed AGNES on both the samples and the features, we can now combined the two results, together with a distance matrix heatmap, to better understand the relationship between the U.S. utility companies and their operations as it related to the measured features.

```
colfunc <- colorRampPalette(c("gray", "white", "red"))  #Custom colour
palette
heatmap.2(dat.z,  #Input data
          Rowv=as.dendrogram(ahc.pu.ave),
          Colv=as.dendrogram(ahc.pu.ave2),
          cexRow=0.8,  #Font size of row labels
          cexCol=0.8,  #Font size of column labels
          #Relabel the column text
          labCol=c("FCCR","Return\non capital","% nuclear","Cost\nper KW",
                   "Peak KWH\ngrowth","Sales","Annual foad\nfactor","Total
fuel\ncost"),
          key.title=NA,
          col=colfunc(15))
```

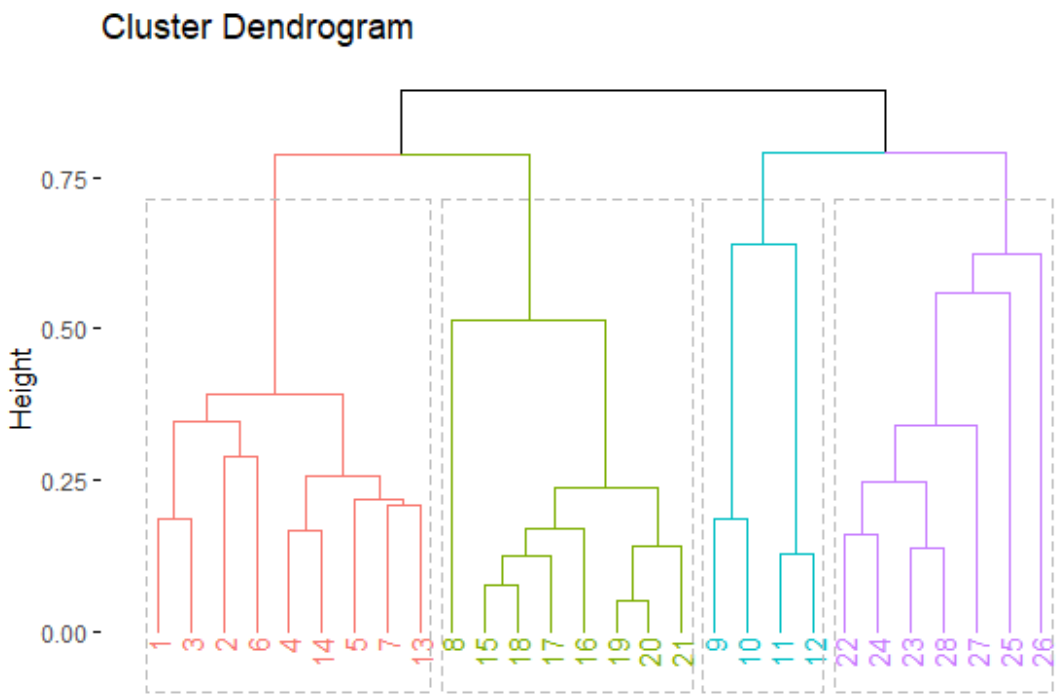


Exercise: The companies, Nevada Power Co., Puget Sound Power and Light Co. and Idaho Co. form one of the major clusters accordingly to both AGNES and DIANA. Based on the above figure, in what ways are these three companies are similar to one another, and at the same time, different to the other utility companies? Also, Consolidated Edison Co. (**Consolid**) was deemed to be different to all other U.S. companies according to results from AGNES and DIANA. Using the above heatmap, comment as to how this company is different to the rest.

Here is another example with the spider dataset. We begin with the clusering of sites.

```
data(spider)  #Load the spider data
dat.abund <- spider$abund  #Extract the abundance data

#Clustering of sites
dist.site <- vegdist(dat.abund,method="bray");
ahc.site <- agnes(dist.site,method="average");
aggcoef.site <- ahc.site$ac; aggcoef.site

## [1] 0.7656291

coph.cor.site <- cor_cophenetic(ahc.site,dist.site); coph.cor.site

## [1] 0.9298845
```

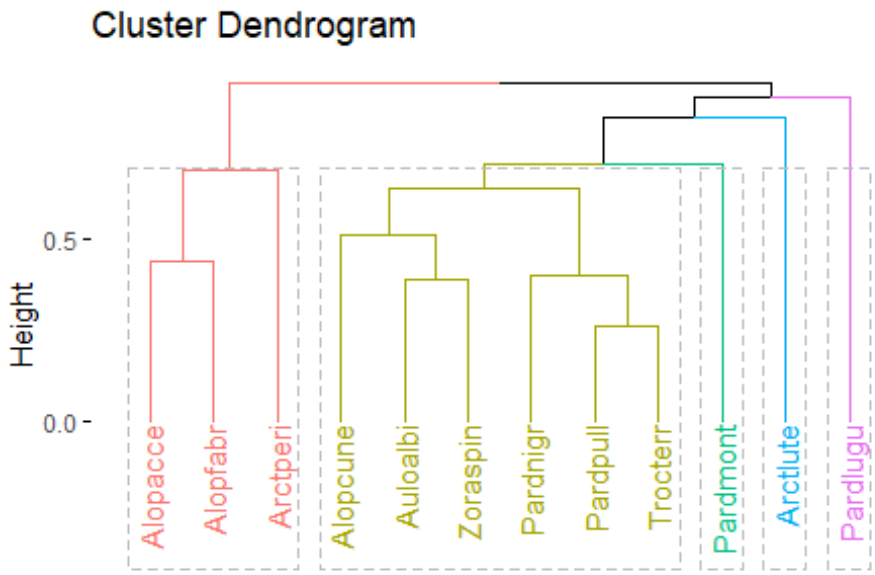
```
#Plot the dendrogram
fviz_dend(ahc.site, cex=0.8, k=4,
          color_labels_by_k=TRUE, rect=TRUE, labels_track_height=-0.4)
```



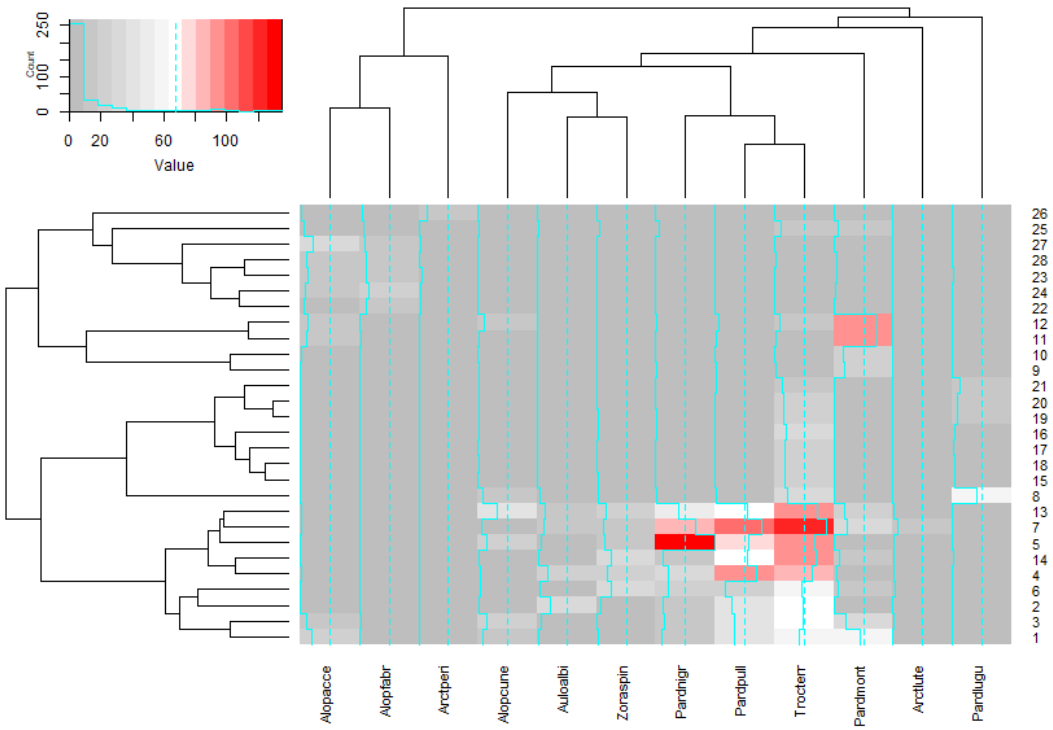
Examine the results below to the PCoA plot from the previous workshop sesison.

Next, we will cluster the species.

```
#Clustering of species
dist.species <- vegdist(t(dat.abund),method="bray");
ahc.species <- agnes(dist.species,method="average");
aggcoef.species <- ahc.species$ac; aggcoef.species

## [1] 0.4414149

coph.cor.species <- cor_cophenetic(ahc.species,dist.species);
coph.cor.species

## [1] 0.9098113

#Plot the dendrogram
fviz_dend(ahc.species, cex=0.8, k=5,
          color_labels_by_k=TRUE, rect=TRUE, labels_track_height=-0.1)
```




```r
heatmap.2(as.matrix(dat.abund),  #Input data. Must be a numeric matrix.
          Rowv=as.dendrogram(ahc.site),
          Colv=as.dendrogram(ahc.species),
          cexRow=0.8,  #Font size of row labels
          cexCol=0.8,  #Font size of column labels
          key.title=NA,col=colfunc(15))
```



Besides the obvious with **Pardnigr**, **Pardpull**, **Trocterr** and **Pardmont** and the corresponding sites, it is difficult to make any sensible observation with the other species/sites. This is largely due to the fact the few large counts (as seen by the red cells in the above plot) that are >80 are drowning out the smaller counts (around 10 or less), and the latter make up the majority of the cases (see legend key). A way to overcome this is by transforming the whole dataset using, say the fourth-root transformation. A transformation reducing the distances between the low and high counts, and as a result, the lower values are given more emphasis. In biology, rare species are often times just as important as the common species.

```
dat.abund.4root <- (dat.abund)^(1/4); #Fourth root transformation

#AGNES by sites on the transformed data
dist.site.4root <- vegdist(dat.abund.4root,method="bray");
ahc.site.4root <- agnes(dist.site.4root,method="average");

#AGNES by species on the transformed data
dist.species.4root <- vegdist(t(dat.abund.4root),method="bray");
ahc.species.4root <- agnes(dist.species.4root,method="average");

heatmap.2(as.matrix(dat.abund.4root),  #Input data. Must be a numeric matrix.
          Rowv=as.dendrogram(ahc.site.4root),
          Colv=as.dendrogram(ahc.species.4root),
          cexRow=0.8,  #Font size of row labels
          cexCol=0.8,  #Font size of column labels
          key.title=NA,col=colfunc(15))
```