**5 - Cluster Analysis**

Cluster analysis involves finding group or clusters of similar observations on the basis of . For some cluster methods the must be numeric, for others that simply take a distance matrix as input the variables could be a variety of types (numeric/continuous, ordinal, or nominal). Once we have clusters of similar observations we generally try to determine what makes the clusters distinct and what features observations in the same cluster have in common.

**5.1 – Hierarchical Clustering**  
Starting with a dissimilarity/distance matrix containing “good” measures of pairwise distances between observations, cluster analysis seeks to form clusters/groups of observations that are similar. Ideally the clusters will have small intra-cluster distances between observations within the same cluster and have large inter-cluster distances between observations in different clusters. There are several different algorithms for forming clusters from an initial distance matrix. We will examine the main algorithms for forming clusters in the examples that follow.

We begin by examining Hierarchical Cluster Analysis using agglomerative methods. In agglomerative clustering each observation starts its own cluster and then observations that are most similar are fused together successively until all observations in the same cluster. The key part of this process is concept of measuring distances between clusters, which is called “linkage”. The example below will illustrate how hierarchical clustering works and explore two simple linkage options.

Example 5.1: Simple Examples (“by hand”)

Distance matrix for 5 pairs of points.

**Method: Single Linkage**

Step 1: Cluster observations #3 & #5

Step 2: Cluster observations #(35) with #1

Step 3: Cluster observations #2 with #4

Step 4: Last cluster – cluster observations #(135) with #(24)

* Cluster #(12345)

In R…

mymatrix <- matrix(c(0,9,3,6,11,9,0,7,5,10,3,7,0,9,2,6,5,9,0,8,11,10,2,8,0),nrow=5,ncol=5)

mydist <- as.dist(mymatrix)

mycluster <- hclust(mydist,method='sing')

plot(mycluster)

**Dendrogram for Single Linkage**

Chart, box and whisker chart

Description automatically generated

**Method: Complete Linkage**

Distance matrix for 5 pairs of points.

Step 1: Cluster observations #3 & #5

Step 2: Cluster observations #2 & #4

Step 3: Cluster observation #1 with #(24)

Step 4: Last cluster – cluster observations #(124) with #(35)

* Cluster #(12345)

In R…

mymatrix <- matrix(c(0,9,3,6,11,9,0,7,5,10,3,7,0,9,2,6,5,9,0,8,11,10,2,8,0),nrow=5,ncol=5)

mydist <- as.dist(mymatrix)

mycluster <- hclust(mydist,method='comp')

plot(mycluster)

**Dendrogram for Complete Linkage**

Chart, box and whisker chart

Description automatically generated

**Other Linkage Methods:**

* + **Average**
  + **Centroid**
  + **Wards Method – based on variation, i.e. Sums-of-Squares Error**

A picture containing diagram

Description automatically generated

Example 5.2: Illicit drug use and binge drinking in the U.S.  
These data, which look at illicit drug use and binge drinking for the 50 states, come from the U.S. Census Bureau.

The variables in the data frame IllDrug are:

* DrugUse = estimate of the % of the population in the state that use illicit drugs.
* BingeDrink = estimate of the % of the population in the state that binge drink.
* Poverty = estimate of the % of the population in the state living in poverty.
* HSdrop = estimate of the % of the population that has dropped out of high school.
* Income = estimate of the per capita income ($).

> attach(IllDrug)

> names(IllDrug)

[1] "State" "DrugUse" "BingeDrink" "Poverty" "HSdrop" "Income"   
  
> head(IllDrug)

State DrugUse BingeDrink Poverty HSdrop Income

Alabama Alabama 3.3 15.6 14.5 12.6 22946

Alaska Alaska 8.5 19.8 9.4 10.9 28523

Arizona Arizona 5.4 17.4 16.6 14.4 25307

Arkansas Arkansas 2.8 16.8 14.8 11.4 22114

California California 6.2 16.7 15.4 14.2 29819

Colorado Colorado 6.5 19.8 9.2 9.8 31678

> X <- as.matrix(IllDrug[,2:6])

> head(X)

DrugUse BingeDrink Poverty HSdrop Income

Alabama 3.3 15.6 14.5 12.6 22946

Alaska 8.5 19.8 9.4 10.9 28523

Arizona 5.4 17.4 16.6 14.4 25307

Arkansas 2.8 16.8 14.8 11.4 22114

California 6.2 16.7 15.4 14.2 29819

Colorado 6.5 19.8 9.2 9.8 31678

If we use the Euclidean metric (or any other for that matter) to construct the distances between the states using the raw data what will happen?

What do we need to do?

As with PCA if the variables are on vastly different scales, as is the case with income in this example, we need to standardize the data first!

> X <- scale(X)

> head(X)

DrugUse BingeDrink Poverty HSdrop Income

Alabama -0.9150748 -1.2127521 0.6098588 0.9105382 -1.0569412

Alaska 3.0829126 0.4407902 -0.8767792 0.2063128 0.2459105

Arizona 0.6994970 -0.5040911 1.2220039 1.6561886 -0.5053843

Arkansas -1.2994966 -0.7403114 0.6973081 0.4134379 -1.2513061

California 1.3145720 -0.7796815 0.8722067 1.5733385 0.5486712

Colorado 1.5452251 0.4407902 -0.9350788 -0.2493624 0.9829551

To perform *hierarchical cluster analysis* in R we can use the hclust function which is a base function in R. To use hclust we first need to decide two things:

1) What metric will be used to measure distance between observations?

2) What method will be used to measure distance between clusters?

The clusters can differ greatly depending on which metric is used to measure distance and to a greater degree which linkage is used when clustering.

The dist function in R allows you to choose the following for distance metrics:

'euclidean': Usual square distance between the two vectors   
 (L2 norm).

'maximum': Maximum distance between two components of x and y

(supremum norm).

'manhattan': Absolute distance between the two vectors (1 norm).

'canberra': sum(|x\_i - y\_i| / |x\_i + y\_i|). Terms with zero

numerator and denominator are omitted from the sum and

treated as if the values were missing.

'binary': (aka \_asymmetric binary\_): The vectors are regarded as

binary bits, so non-zero elements are 'on' and zero elements

are 'off'. The distance is the \_proportion\_ of bits in which

only one is on amongst those in which at least one is on.

'minkowski': The p norm, the pth root of the sum of the pth powers

of the differences of the components.

If we have loaded the proxy library we have many more choices for measuring distance between observations as we have seen previously.

The hclust function in R allows you to choose from the following linkages:

method: the agglomeration method to be used. This should be (an unambiguous abbreviation of) one of '"ward"', '"single"', '"complete"', '"average"', '"mcquitty"', '"median"' or

'"centroid"'.

How do you decide which distance metric and linkage to use? There is no easy/standard answer! Usually we try different combinations and choose which is “best” subjectively. For some methods of clustering we will examine there are more objective measures for the “goodness” of the clustering achieved.

We begin by setting up some distance matrices using different metrics for the standardized numeric variables in our data matrix .

> dman <- dist(X,method="manh")

> dcan <- dist(X,method="canb")

> de <- dist(X,method="euclid")

> dmax <- dist(X,method="max")

Using Euclidean perform clustering using different linkages...  
  
> desing <- hclust(de,method=”sing”)

> decomp <- hclust(de,method=”comp”)

> deave <- hclust(de,method=”ave”)

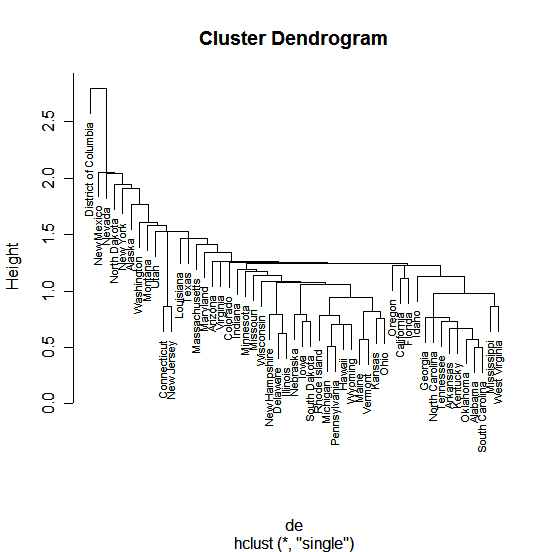
> deward <- hclust(de,method=”ward.D”)

> demcq <- hclust(de,method=”mcqu”)

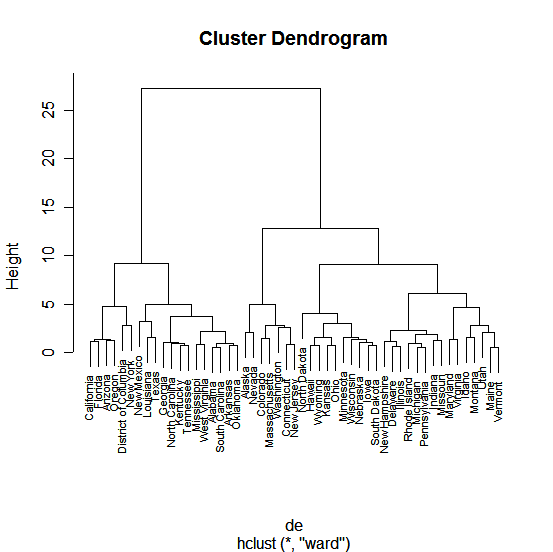
Plot cluster dendograms/trees...

> state <- IllDrug$State 🡨 Extract the state names from the IllDrug data frame.

> plot(desing,labels=state,cex=.7)



> plot(deward,labels=state,cex=.7)

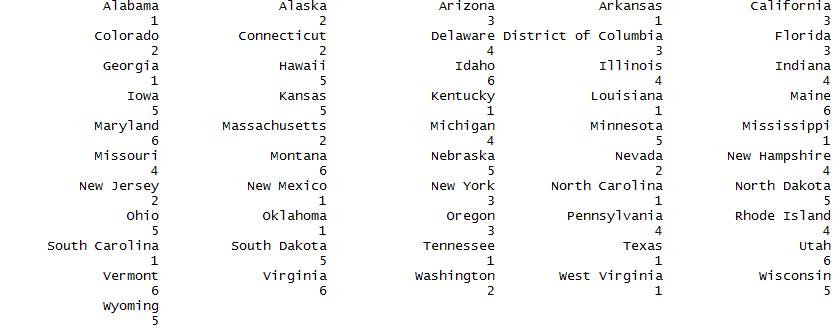


How are clusters formed? To form clusters we need to cut the tree at a height of our choosing, which then determines our clusters by looking at observations that are on the same branch. You can form any number of clusters you want by cutting the tree at the appropriate height. The function cutree allows you to either cut the tree at a certain height to determine the clusters or to specify the number of clusters you want and the height required for that many clusters will be used. The latter approach is generally easier.

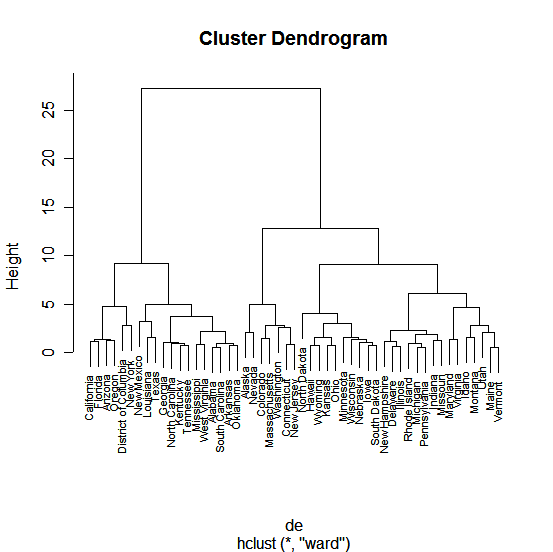
We now consider using the tree from the Ward’s linkage to form clusters for these data.

> drug.clust <- cutree(deward,k=6)

> drug.clust



Below you can see where the Ward’s linkage dendogram is cut in order form groups.



Cutting the tree here will produce the six clusters above.

Which states are in each cluster? The function clust.grps (code shown below) the results from cutting the tree and the original data matrix as arguments and prints the clusters.  
  
> library(MASS)  
> X.unscaled = as.matrix(IllDrug[,2:6])

> clust.grps(X.unscaled,drug.clust,parcoord=T)

Cluster 1 consists of:

=========================================================

Alabama Arkansas Georgia Kentucky Louisiana Mississippi New Mexico North Carolina Oklahoma South Carolina Tennessee Texas West Virginia

Variables means in this cluster are:

DrugUse BingeDrink Poverty HSdrop Income

3.507692 16.869231 15.530769 12.246154 23560.846154

Cluster 2 consists of:

=========================================================

Alaska Colorado Connecticut Massachusetts Nevada New Jersey Washington

Variables means in this cluster are:

DrugUse BingeDrink Poverty HSdrop Income

6.585714 19.285714 9.271429 10.514286 33121.857143

Cluster 3 consists of:

=========================================================

Arizona California District of Columbia Florida New York Oregon

Variables means in this cluster are:

DrugUse BingeDrink Poverty HSdrop Income

5.416667 17.100000 16.516667 13.083333 30409.666667

Cluster 4 consists of:

=========================================================

Delaware Illinois Indiana Michigan Missouri New Hampshire Pennsylvania Rhode Island

Variables means in this cluster are:

DrugUse BingeDrink Poverty HSdrop Income

4.7875 19.7250 10.4000 10.4250 28923.3750

Cluster 5 consists of:

=========================================================

Hawaii Iowa Kansas Minnesota Nebraska North Dakota Ohio South Dakota Wisconsin Wyoming

Variables means in this cluster are:

DrugUse BingeDrink Poverty HSdrop Income

3.98 21.90 10.88 7.14 26738.20

Cluster 6 consists of:

=========================================================

Idaho Maine Maryland Montana Utah Vermont Virginia

Variables means in this cluster are:

DrugUse BingeDrink Poverty HSdrop Income

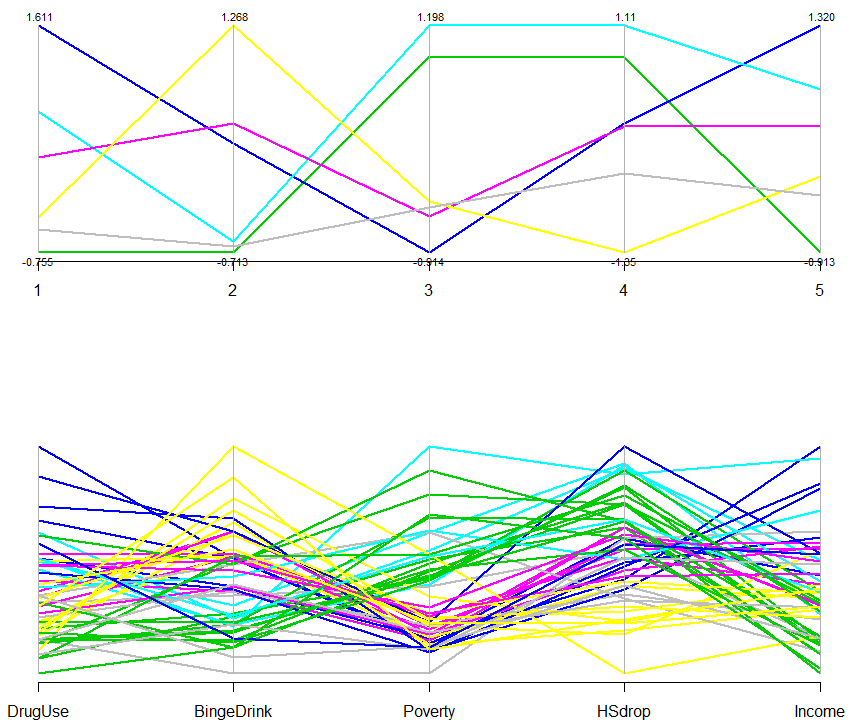
3.814286 17.000000 10.700000 9.200000 25945.285714

The code for the clust.grps function is given below:

|  |
| --- |
| Custom R Function |
| clust.grps = function(X,grps,parcoord=F,suppress=F) {  require(MASS)  k = length(unique(grps))  p = dim(X)[[2]]  Xmeans = matrix(0,nrow=length(unique(grps)),ncol=p+1)  X = as.data.frame(X)  for (i in 1:k){  cat("\n")  cat(paste("Cluster",i,"\n"))  cat("=======================================================================\n")  if (suppress==F){  cat(row.names(X)[grps==i])  cat("\n\n")}  cat("Variable means in this cluster are:\n")  cat("----------------------------------------------------------------------\n")  print(apply(X[grps==i,],2,mean))  Xmeans[i,]=c(apply(X[grps==i,],2,mean),as.numeric(i))  cat("\n\n")    }  if (parcoord) {  par(mfrow=c(2,1)) parcoord(Xmeans[,-(p+1)],col=as.numeric(Xmeans[,(p+1)])+2,  lwd=2,var.label=T)  parcoord(X,col=as.numeric(grps)+2,lwd=2)  }  } |

The top parallel coordinate plot shows the mean for each of the scaled variables within each of the clusters. You will need to look at the tables produced by this function to identify the clusters in this plot.

The bottom parallel coordinate plot shows the scaled values for each of the individuals in the data colored by cluster.



The first parallel coordinate plot displays the mean for each variable for each cluster. The clusters are color coded, but are not labelled. The plot is simply a visualization of the means reported for each of the clusters in the first part of the output. This can aid in identifying what makes the clusters distinct. The second parallel coordinate plot shows the individual profiles within each cluster.