Cluster Analysis

Lucy Ng'ang'a

October 15, 2018

What is Cluster Analysis?

- Cluster Analysis is a collection of numerical and statistical techniques with a common goal of uncovering or discovering groups or clusters of observations that are homogeneous/similar and separated from other groups.
- The objective is to group the observation into clusters that share similar characteristics by some measure of similarity.

Clustering Methods

- Hierarchical techniques
- K-Means clustering
- Model based clustering

Clustering objectives

There are several ways to measure similarity

1. Measure of similarity between observations.We need to measure how similar two observations are to one another.(inter cluster similarity)
2. Measure of similarity between clusters or groups.We need to measure how similar two clusters are to one another.(intra cluster similarity)

Objectives

- Minimize the inter-cluster similarities
- Maximize the intra-cluster similarities

A good clustering algorithms can be evaluated on the above two objectives. i.e. Having a high intra-cluster similarity and a low inter-cluster similarity.

Applications

- **Marketing** a marketing department can use clustering to segment customers by personal attributes. As a result of this, different marketing campaigns targeting various types of customers can be designed.
- **Medical diagnosis** Medical symptoms and results clustering is useful in uncovering diagnosis.
- **Data and text mining and search machine results**
- **Pattern and Voice recognition**

Clustering Scenarios

1. Law enforcement stations of patrol vans so that high crime areas are in vicinity of the patrol vans. These can be found by finding the center of a high crime cluster.
2. Location of network towers can be found by a clustering algorithm so that all its users can receive optimum signal strengths.

Example

Consider the following data set *Iris* containing sepal and petal measurement of 3 species of iris plant.

```
> library(MASS)
> iris<-iris
```

How to measure of association

- **Euclidean Distance.** In two dimensions,it is simply measure the distances between the pairs of points. More generally we can use the following equation.

$$d(x_i, x_j) = \sqrt{\sum_{i=k}^{p} (x_{ij} - x_{ik})^2}$$

  is the distance between x_{ik} and x_{jk}. p represents the count of variables.The distance between the cluster center and the data points is the Euclidean distance

- **Manhattan distance** is the distance between the cluster center and the data points is the sum of the absolute values of the distances.

Why Euclidean Distance?

- Euclidean distance are preferred than Manhattan distances because they are non negative.
- Euclidean distances between each pair of individuals can be be arranged in a matrix that is symmetric because $d_{ij} = d_{ji}$ and has zeros on the main diagonal.such a matrix is the starting point of many cluster analysis.
- Calculations of euclidean distances from the raw data may not be sensible when the variables are on different scales. The variables are first standardized before calculating the distances.

```
> scp=plot(iris$Petal.Length,iris$Petal.Width)#a scatter plot
> # To calculate distances the factor variable in the data has to be
> data=iris
> data$Species=NULL
> edd=dist(iris)
> summary(edd)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  0.000   1.167   2.639   2.845   4.337   7.921
```

Standardize the data

To Standardize the data

```
> apply(data,2,var) #calculates variance of each variable if signifi

Sepal.Length  Sepal.Width Petal.Length  Petal.Width
   0.6856935    0.1899794    3.1162779    0.5810063

> a=apply(data,2,mean)
> b=apply(data,2,sd)
> iris_std=scale(data,a,b)
> ed=dist(iris_std)
> summary(ed)

   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  0.000   1.332   2.489   2.502   3.561   6.508
```

Agglomerative hierarchical clustering

- This class of clustering methods produces a hierarchical classification of data.Classifications consist of a series of partitions that may run from a single cluster containing all individuals to n clusters each containing a single individual.
- That is, we start by defining each data point to be a cluster and combine existing clusters at each step and eventually into a single cluster that contains all individuals.
- It is therefore necessary for the investigator to decide the number of clusters.The problem is to deciding the "correct" number of clusters.

Cluster dendrogram

Hierarchieal clustering is represented by a two way dimensional diadram known as **dendrogram**. It illustrates fusions made at each stage of statistical analysis.

```
> hc=hclust(ed)
> plot(hc, hang = -1)#warning not appropriate for large n
```

Methods of combining clusters

- **Single Linkage;**we define the distance between two clusters to be the minimum distance between any single data point in the first cluster and any single data point in the second cluster. On the basis of this definition of distance between clusters, at each stage of the process we combine the two clusters that have the smallest single linkage distance.
- **Complete linkage;** we define the distance between two clusters to be the maximum distance between any single data point in the first cluster and any single data point in the second cluster. On the basis of this definition of distance between clusters, at each stage of the process we combine the two clusters that have the smallest complete linkage distance.
- **Average Linkage;**we define the distance between two clusters to be the average distance between data points in the first cluster and data points in the second cluster. On the basis of this definition of distance between clusters, at each stage of the process we combine the two clusters that have the smallest average linkage distance.

Decision on the number of clusters

Choice on number of clusters

One informal approach to determine the number of the clusters is to examine the changes in the height in the dendrogram. A large change indicates the appropriate number of clusters in the data.

```
> plot(hclust(ed,method="complete"))
```

Assigning Membership

Assigning membership based on 3 clusters

```
> members=cutree(hc,3)
```

Calculating cluster means

```
> std_mean=aggregate(iris_std,list(members),mean)
> aggregate(data,list(members),mean)
```

	Group.1	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width
1	1	5.016327	3.451020	1.465306	0.244898
2	2	5.512500	2.466667	3.866667	1.170833
3	3	6.472727	2.990909	5.183117	1.815584

Comparing the Results

Comparisons

- Bind the clusters membership with the data
- Ploting a scatter plot using two variables
- Compare with the three known categories

```
> new=cbind(iris,members)
> scp1=plot(new$Petal.Length,new$Petal.Width,col=new$members)
> scp2=plot(iris$Petal.Length,iris$Petal.Width, col=iris$Species)
> table(iris$Species,new$members)
```

```
            1  2  3
setosa     49  1  0
versicolor  0 21 29
virginica   0  2 48
```

K-means Clustering

Definition

- K-means clustering is a non-hierarchical approach thus we do not have to calculate the distance measures between all pairs of subjects.Therefore, this procedure seems much more efficient or practical when you have very large datasets.
- Under this k-means clustering you need to pre-specifiy how many clusters you want to consider. The commonly used implementation is one that tries to find the partion of n individuals into k groups that minimize the *within group sum of squares* over all variables.

Implementation on iris data

```
> km=kmeans(iris_std,3)
> km$size
[1] 50 47 53
> km$withinss
[1] 47.35062 47.45019 44.08754
```

Interpretation of the results

```
> km
K-means clustering with 3 clusters of sizes 50, 47, 53

Cluster means:
  Sepal.Length Sepal.Width Petal.Length Petal.Width
1  -1.01119138  0.85041372   -1.3006301  -1.2507035
2   1.13217737  0.08812645    0.9928284   1.0141287
3  -0.05005221 -0.88042696    0.3465767   0.2805873

Clustering vector:
  [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 [38] 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 3 3 3 2 3 3 3 3 3 3 3 3 2 3 3
 [75] 3 2 2 2 3 3 3 3 3 3 3 2 2 3 3 3 3 3 3 3 3 3 3 3 3 3 2 3 2 2 2
[112] 2 2 3 3 2 2 2 2 3 2 3 2 3 2 2 3 2 2 2 2 2 2 3 3 2 2 2 3 2 2 2
[149] 2 3

Within cluster sum of squares by cluster:
[1] 47.35062 47.45019 44.08754
 (between_SS / total_SS =  76.7 %)
```

Evaluating and determining K

```
> table(iris$Species,km$cluster)

              1  2  3
  setosa     50  0  0
  versicolor  0 11 39
  virginica   0 36 14

> wss=(nrow(iris_std)-1)*sum(apply(iris_std,2,var))
> for (i in 2:10)
+  wss[i]=sum(kmeans(iris_std,centers = i)$withinss)
> plot(1:10,wss,type = "b",xlab = "Number of Clusters",
+      ylab = "Within group ss")
```

- To determine the number of groups k, one needs to examine a scree plot. A scree plot is a simple line segment plot that shows the fraction of total variance in the data as explained or represented by number of clusters. To plot one one needs to calculate the within group sum of squares

Model Based Clustering

- The hierarchical and k-means clustering methods are based largely on heuristic but intuitively reasonable procedures but not formal models for clustering structure in the data. Thus problems such as deciding between methods and estimating the number of clusters difficult.
- The key advantage of model-based approach, compared to the standard clustering methods (k-means, hierarchical clustering, etc.), is the suggestion of the number of clusters and an appropriate model.
- Model based clustering assumes that the population consists of a number of sub populations each having variables with different multivariate pdf, resulting in what is known as **finite mixture density** for the population as whole.
- Each cluster k is centered at the means, with increased density for points near the mean. Geometric features (shape, volume, orientation) of each cluster are determined by the covariance matrix.

Model Based Clustering

Implementation in R

```
> library(mclust)
> clPairs(data,iris$Species)
> BIC=mclustBIC(data)
> plot(BIC)
> summary(BIC)

Best BIC values:
             VEV,2         VEV,3      VVV,2
BIC      -561.7285 -562.5522369 -574.01783
BIC diff    0.0000   -0.8237748  -12.28937
```

Mclust uses an identifier for each possible parametrization of the covariance matrix that has three letters: E for "equal", V for "variable" and I for "coordinate axes". The first identifier refers to volume, the second to shape and the third to orientation.

Model Based Clustering

```
> summary(BIC)

Best BIC values:
             VEV,2         VEV,3      VVV,2
BIC      -561.7285 -562.5522369 -574.01783
BIC diff    0.0000   -0.8237748  -12.28937

> mod1=Mclust(data,x=BIC)
> a=summary(mod1)
> plot(mod1, what="classification")
> plot(mod1, what="density")
> table(iris$Species,mod1$classification)

               1  2
  setosa      50  0
  versicolor   0 50
  virginica    0 50
```

Model Based Clustering

Bayesian information criterion (BIC) is a criterion for model selection among a finite set of models; the model with the lowest BIC is preferred. It is based, in part, on the likelihood function

$$BIC = \frac{1}{n}(RSS - log(n)d\hat{\sigma}^2) \quad (1)$$

Calculate the residual sum of squares and then add an adjustment term which is the log of the number of observations times d, which is the number of parameters in the model.

```
> mod2=mclustICL(data)
> summary(mod2)
```

```
Best ICL values:
              VEV,2       VEV,3      VVV,2
ICL      -561.7289 -566.467287 -574.01910
ICL diff    0.0000   -4.738411  -12.29022
```

```
> plot(mod2)
```

Model Based Algorithms

- Integrated Complete-data Likelihood(ICL) is another criterion of selecting the best model for model-based hierarchical clustering.
- Convex Clustering is another alogorithm that perform k-means clustering and other machine learning algorithms

```
> library(flexclust)
> library(mvtnorm)
> cc=cclust(iris_std,k=3,dist= "euclidean",
+           method = "kmeans",save.data=TRUE)
> plot(cc,hull=FALSE,col=rep("black",3))
> cc
kcca object of family 'kmeans'

call:
cclust(x = iris_std, k = 3, dist = "euclidean", method = "kmeans",
    save.data = TRUE)

cluster sizes:
```

Comparing clustering algorithms

After fitting data into clusters using different clustering methods, you may wish to measure the accuracy of the clustering. In most cases, you can use either intracluster or intercluster metrics as measurements. The higher the intercluster distance, the better it is, and the lower the intracluster distance, the better it is.

- The within.cluster.ss measurement stands for the within clusters sum of squares and the smaller the value, the more closely related objects are within the cluster.
- The avg.silwidth is a measurement that considers how closely related objects are within the cluster and how clusters are separated from each other. The silhouette value usually ranges from 0 to 1; a value closer to 1 suggests the data is better clustered.

Comparing clustering algorithms

```
> library(fpc)
> cs1= cluster.stats(dist(iris[1:4]), mod1$classification)
> cs1[c("within.cluster.ss","avg.silwidth")]

$within.cluster.ss
[1] 154.947


$avg.silwidth
[1] 0.6867351

> cs2= cluster.stats(dist(iris[1:4]), km$cluster)
> cs2[c("within.cluster.ss","avg.silwidth")]

$within.cluster.ss
[1] 86.42692


$avg.silwidth
[1] 0.5061527
```