

Hierarchical Clustering

Contents

1. Hierarchical Clustering

- i. Distance Measures
- ii. Agglomerative Clustering

1. Hierarchical Clustering in R

Hierarchical Methods

- Hierarchical clustering creates clusters which have a pre-determined, top to bottom ordering
- Hierarchical clustering method is further divided into two approaches
 - Agglomerative
 - Divisive

Agglomerative clustering starts with each object in a separate cluster, clusters are formed by grouping objects into bigger and bigger clusters, the process is continued until all objects are members of single cluster

Divisive clustering starts with all objects grouped in a single cluster, clusters are divided until each observation is in a separate cluster

Distance Measures – Euclidean Distance

- Squared Euclidean Distance

Object	X1	X2			XP
1	a1	a2			ap
2	b1	b2			bp

The sum of squared differences between values of each variable

$$d(x, y) = (a1-b1)^2 + (a2-b2)^2 + \dots + (ap-bp)^2$$

- The square root is defined as 'Euclidean Distance'
- 'Euclidean Distance' is the most widely used distance measure in cluster analysis

Case Study

Background

- An insurance company would like to explore its small town business and create strategies for different groups of small towns

Objective

- To form clusters of towns

Available Information

- Data for 8 towns is recorded
- Information fields for the towns are
 - Loss Ratio
 - Premium Rates
 - No. of Policies from that town


Data Snapshot

Town Insurance

Variables				
Observations	Town	x1	x2	x3
	A	1.06	9.2	151
	B	0.89	10.3	202
	C	1.43	15.4	113
	D	1.02	11.2	168
	E	1.49	8.8	192
	F	1.32	13.5	111
	G	1.22	12.2	175
Columns	Description	Type	Measurement	Possible values
Town	Towns under study	character	-	-
x1	Loss Ratio	numeric	-	Positive values
x2	Premium Rates	numeric	-	Positive values
x3	Number of Policies	numeric	-	Positive values

Euclidean Distance Matrix – Iteration 1

Distance between C and A = $\sqrt{(1.43-1.06)^2+(15.4-9.2)^2+(113-151)^2} = 38.50$

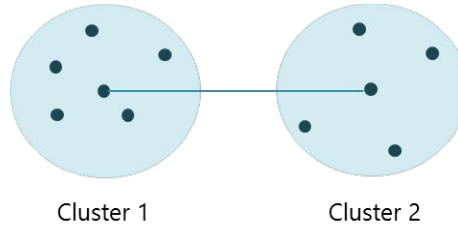


	A	B	C	D	E	F	G	H
A		51.01	38.50	17.12	41.00	40.23	24.19	94.00
B	51.01		89.15	34.01	10.13	91.06	27.07	43.01
C	38.50	89.15		55.16	79.28	2.76	62.08	132.15
D	17.12	34.01	55.16		24.12	57.05	7.07	77.03
E	41.00	10.13	79.28	24.12		81.14	17.34	53.00
F	40.23	91.06	2.76	57.05	81.14		64.01	134.07
G	24.19	27.07	62.08	7.07	17.34	64.01		70.06
H	94.00	43.01	132.15	77.03	53.00	134.07	70.06	

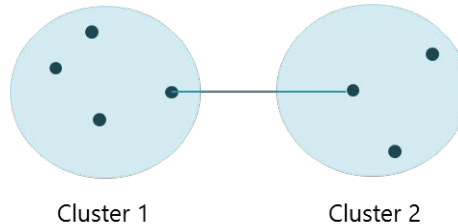
- Distance between Firm 'C' and 'F' is minimum hence ,Firms 'C' and 'F' are combined at the first iteration.
- The distance matrix is revised as per methods described in next 2 slides and subsequent merging is performed.

Agglomerative Clustering

- **Centroid method:** The distance between 2 clusters is measured as a Euclidean distance between centroids (average value) of each of the clusters

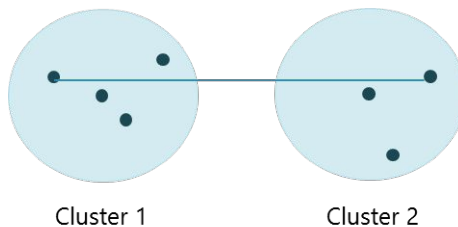


- **Single linkage:** The distance between 2 clusters is the distance between their two closest points. At any stage the clusters are merged by the single shortest distance

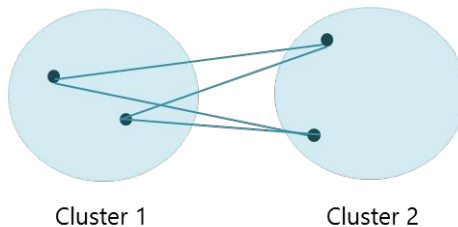


Agglomerative Clustering

- **Complete linkage:** The distance between 2 clusters is the distance between their two farthest points. At any stage the clusters are merged by the single farthest distance



- **Average linkage:** The distance between 2 clusters is defined as the average of the distances between all pairs of objects, where one member of the pair is from each of the cluster



Get an Edge!

More Linkage Methods

There are three more ways in which cluster linkages can be established

1. Median Linkage	Median distance between the observations in two clusters
2. McQuitty Linkage	Two clusters are to be joined, then the distance of the new cluster to any other cluster is the average of distance of the clusters to be joined to the other cluster
3. Ward Linkage	Sum of squared deviation from point to centroid. Objective is to minimize the within cluster sum of squares

Single Linkage Method

Iteration No	No of Cluster Formed	Cluster Membership							
		1	2	3	4	5	6	7	8
step1	8	A	B	C	D	E	F	G	H
1	7	A	B	C,F	D	E	G	H	
2	6	A	B	C,F	D,G	E	H		
3	5	A	B,E	C,F	D,G	H			
4	4	A	B,E,D,G	C,F	H				
5	3	A,B,E,D,G	C,F	H					
6	2	A,B,E,D,G, C,F	H						
7	1	A,B,E,D,G, C,F,H							

Hierarchical Clustering in R

#Importing and Readyng the Data

```
townins<-read.csv("Town Insurance.csv",header=T)
```

```
townins2<-subset(townins,select=c(-Town))
```

#Calculating Distance Matrix

```
d<-dist(townins2, method="euclidean")
```

↓
dist() computes and returns distance matrix.

#Hierarchical Clustering

```
fit <- hclust(d, method="single")
```

- ↓
- ☐ **hclust()** runs hierarchical clustering.
 - ☐ **d=** Distance Matrix
 - ☐ **method=** specifies linkage method.



method= can take the following arguments depending on which method is used for linkage. Available options are - "complete", "average", "centroid", "median", "mcquitty" and "ward.D2"

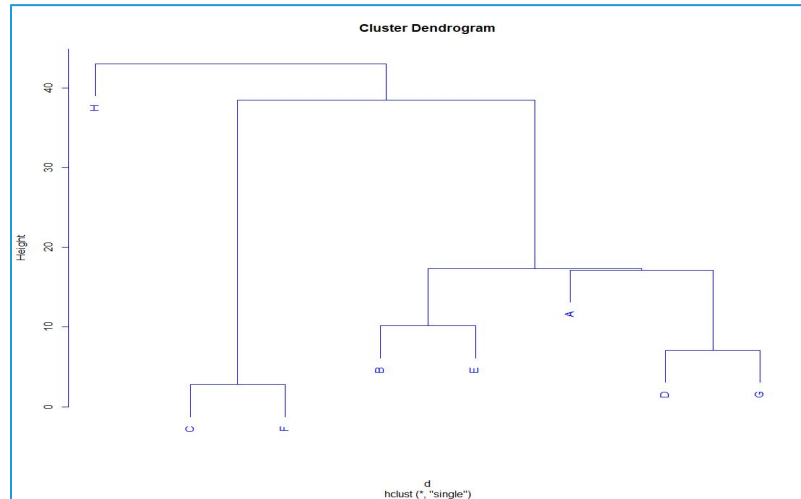
Hierarchical Clustering in R

#Display Dendrogram

```
plot(fit,label=townins$Town,col="blue")
```

Plotting **hclust()** output returns a dendrogram.

Output



- ❑ Towns C and F are close and form one cluster
- ❑ Towns A,B,D,E and G form second cluster
- ❑ Town H looks separate entity

Hierarchical Clustering in R

```
# Append Segment Variables  
townins$segment<-cutree(fit,k=3)  
  
townins
```

Output

	Town	x1	x2	x3	segment
1	A	1.06	9.2	151	1
2	B	0.89	10.3	202	1
3	C	1.43	15.4	113	2
4	D	1.02	11.2	168	1
5	E	1.49	8.8	192	1
6	F	1.32	13.5	111	2
7	G	1.22	12.2	175	1
8	H	1.10	9.2	245	3

- ❑ **cutree()** cuts a dendrogram tree into several groups by specifying the desired number of clusters $k(s)$, or cut height(s).
- ❑ **fit** is the dendrogram tree
- ❑ **k=** number of clusters the tree should be cut into.

Quick Recap

Hierarchical Clustering

- Hierarchical Clustering can be of two types –
 - **Agglomerative** – starts with each object in separate clusters and stepwise merging is carried out until a single cluster with all objects is formed
 - **Divisive** – All objects grouped in a single cluster, clusters are divided until each observation is in a separate cluster

Hierarchical Clustering in R

- **hclust()** function in package **stats** performs Hierarchical (Agglomerative) clustering.
- **cutree()** cuts a dendrogram tree into several groups by specifying the desired number of clusters $k(s)$, or cut height(s).