Machine Learning with Python

Session 14: Agglomerative Clustering

Arghya Ray

Measuring Distance Between Clusters:

- **Single Linkage**
  - Minimum Distance (Cluster A to Cluster B)
  - Distance between two clusters is the distance between the pair of records Ai and Bj that are closest.
- **Complete Linkage**
  - Maximum Distance (Cluster A to Cluster B)
  - Distance between two clusters is the distance between the pair of records Ai and Bj that are farthest from each other
- **Average Linkage**
  - Distance between two clusters is the average of all possible pair-wise distances
- **Centroid**
  - Distance between two clusters is the distance between the two cluster centroids.
  - Centroid is the vector of variable averages for all records in a cluster

(x1, y1, z1) (x2, y2, z2) (x3, y3, z3)

C1 = ((x1+x2+x3)/3, (y1+y2+y3)/3, (z1+z2+z3)/3)

Q. Consider the following three clusters, each with four members:

Cluster 1: {(1,5), (2,4), (3,3), (2,1)} Centroid- ((1+2+3+2)/4, (5+4+3+1)/4) (2,3.25)

Cluster 2: {(5,4), (6,6), (7,5), (8,8)}

Cluster 3: {(4,1), (3,0), (5,1), (6,2)}

Distance Between	Single Link	Complete-Link	Centroid	Average-Link
Cluster 1 & 2	2.24	9.22	5.15	5.43
Cluster 2 & 3	2.24	9.43	5.15	5.38
Cluster 3 & 1	1.41	5.83	3.36	3.76

- Whichever distance algorithm is applied, two nearby clusters can be merged if an agglomerative approach is being used.
- It has been reported that the ***complete link algorithm*** generally produces compact and more useful clusters.
- The ***single link algorithm*** tends to suffer from chaining effects and elongated clusters in some situations. However, the single link algorithm is found to be effective in some applications.
- Both the complete link algorithm and single link algorithm can suffer from the presence of outliers.

Agglomerative Method:

The basic idea of the agglomerative method is to start out with n-clusters for ‘n’ data points and keep on combining points.

Steps involved:

1. Allocate each point to a cluster of its own. Thus we start with *n* clusters for *n* objects.
2. Create a distance matrix by computing distances between all pairs of clusters using one of the distance measuring methods (e.g. single link metric or complete link metric). Sort these distances in ascending order.
3. Find the two clusters that have the smallest distance between them.
4. Remove the pair of objects and merge them. When you are merging, take the average value of the two cluster distances.
5. If there is only one cluster left, then stop.
6. Compute all distances from the new cluster and update the distance matrix after the merger and go to step 3.

Use agglomerative clustering method for clustering the data (use centroid method for calculating distance between clusters).

Student	Age	Mark1	Mark2	Mark3
S1	18	73	75	57
S2	18	79	85	75
S3	23	70	70	52
S4	20	55	55	55
S5	22	85	86	87
S6	19	91	90	89
S7	20	70	65	60
S8	21	53	56	59
S9	19	82	82	60
S10	47	75	76	77

S1→ (18,73,75,57)
S2→ (18,79,85,75)

S1 and S2→ 34
S2 and S3→ 52
S1 and S3→ 18
S2 and S4 → 76

Step 1 and 2: Allocate each point to a cluster and compute the distance matrix using the centroid method. The distance matrix is symmetric.

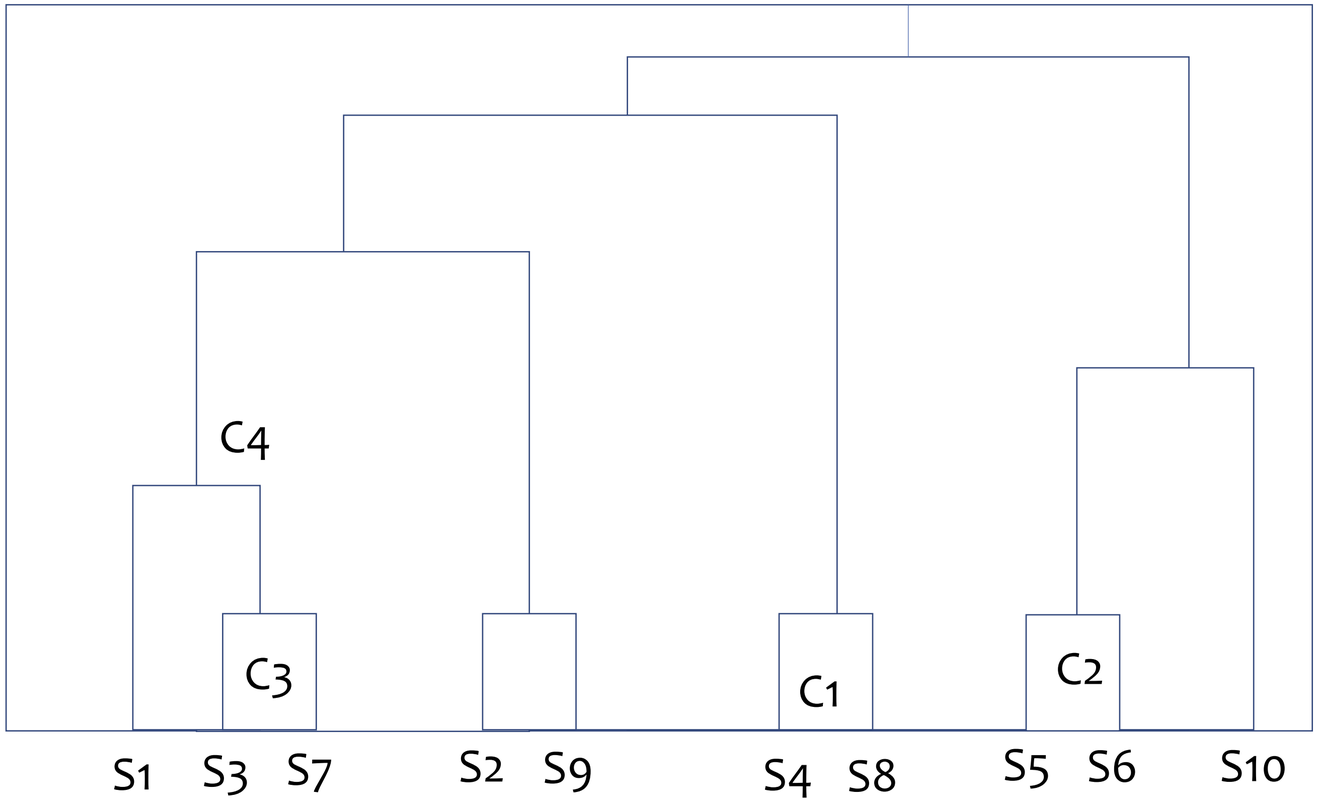
	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10
S1	0									
S2	34	0								
S3	18	52	0							
S4	42	76	36	0						
S5	57	23	67	95	0					
S6	66	32	82	106	15	0				
S7	18	46	16	30	65	76	0			
S8	44	74	40	8	91	104	28	0		
S9	20	22	36	60	37	46	30	58	0	
S10	52	44	60	90	55	70	60	86	58	0

Step 3 and 4: The smallest distance is 8 between objects S4 and S8. We combine this to cluster (C1) and put it where S4 was.

	S1	S2	S3	C1	S5	S6	S7	S9	S10
S1	0								
S2	34	0							
S3	18	52	0						
C1	41	75	38	0					
S5	57	23	67	95	0				
S6	66	32	82	106	15	0			
S7	18	46	16	29	65	76	0		
S9	20	22	36	59	37	46	30	0	
S10	52	44	60	88	55	70	60	58	0

Step 5 and 6: The smallest distance is now 15 between objects S5 and S6. We combine this to cluster (C2) and put it where S5 was.

	S1	S2	S3	C1	C2	S7	S9	S10
S1	0							
S2	34	0						
S3	18	52	0					
C1	41	75	38	0				
C2	61.5	27.5	74.5	97.5	0			
S7	18	46	16	29	69.5	0		
S9	20	22	36	59	41.5	30	0	
S10	52	44	60	88	62.5	60	58	0



	S1	S2	S3	C1	C2	S9	S10
S1	0						
S2	34	0					
C3	15	49	0				
C1	41	75	30	0			
C2	61.5	27.5	71.5	97.5	0		
S9	20	22	33	59	41.5	0	
S10	52	44	60	88	62.5	58	0

Divisive Hierarchical Method:

The basic idea of the divisive method is that it starts with the whole dataset as one cluster and then proceeds to recursively divide the cluster into two sub-clusters and continues until each cluster has only one object or some other stopping criterion has been reached. There are two types of divisive methods:

- ***Monothetic***: It splits a cluster using only one attribute at a time.
- ***Polythetic***: It splits a cluster using all attributes together.

Steps involved in a polythetic divisive method:

1. Decide a method of measuring the distance between two objects. Also decide a threshold distance.
2. Create a distance matrix by computing distance between all pairs of objects within the cluster. Sort these distances in ascending order.
3. Find the two objects that have the largest distance between them. They are most dissimilar.
4. If the distance between the two objects is smaller than the pre-specified threshold and there is no other cluster that needs to be divided then stop, otherwise continue.
5. Use the pair of objects as seeds of a K-means method to create two new clusters
6. If there is only one object in each cluster, then stop otherwise continue with Step 2.

Use divisive clustering method for clustering the data (use centroid method for calculating distance between clusters).

Student	Age	Mark1	Mark2	Mark3
S1	18	73	75	57
S2	18	79	85	75
S3	23	70	70	52
S4	20	55	55	55
S5	22	85	86	87
S6	19	91	90	89
S7	20	70	65	60
S8	21	53	56	59
S9	19	82	82	60
S10	47	75	76	77

Step 1 and 2: Allocate each point to a cluster and compute the distance matrix using the centroid method. The distance matrix is symmetric.

	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10
S1	0									
S2	34	0								
S3	18	52	0							
S4	42	76	36	0						
S5	57	23	67	95	0					
S6	66	32	82	106	15	0				
S7	18	46	16	30	65	76	0			
S8	44	74	40	8	91	104	28	0		
S9	20	22	36	60	37	46	30	115	0	
S10	52	44	60	90	55	70	60	98	99	0

The largest distance is 115 between the objects S8 and S9. They becomes the seed for to new clusters. K-Means is used to split the group into two clusters.

	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10
S8	44	74	40	8	91	104	28	0	115	98
S9	20	22	36	60	37	46	30	115	0	99

Cluster C1: S4, S7, S8, S10
Cluster C2: S1, S2, S3, S5, S6, S6, S9

Cluster C1: S4, S7, S8, S10

Cluster C2: S1, S2, S3, S5, S6, S6, S9

If the stopping criteria is not met, we can follow the previous steps and divide these two clusters again one by one.

For Cluster C2:

	S1	S2	S3	S5	S6	S9
S1	0					
S2	34	0				
S3	18	52	0			
S5	57	23	67	0		
S6	66	32	82	15	0	
S9	20	22	36	37	46	0

The largest distance is 82 between the objects S3 and S6. They becomes the seed for to new clusters.

For Cluster C1:

	S4	S7	S8	S10
S4	0			
S7	30	0		
S8	8	28	0	
S10	90	60	98	0

The largest distance is 98 between the objects S8 and S10. They becomes the seed for to new clusters.

This continues until one of the stopping criteria is met.

The content of the slides are prepared from different textbooks.

References:

- Data Mining and Predictive Analytics, By Daniel T. Larose. Copyright 2015 John Wiley & Sons, Inc.
- Predictive Analytics for Dummies, By Anasse Bari, Mohamed Chaouchi, & Tommy Jung, Copyright 2016, John Wiley & Sons, Inc.
- Introduction to Data Mining with Case Studies, By G.K. Gupta. Copyright 2014 by PHI Learning Private Limited.


Thank you..