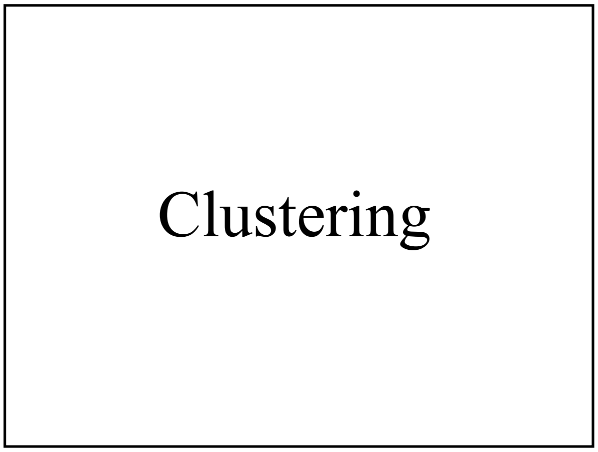
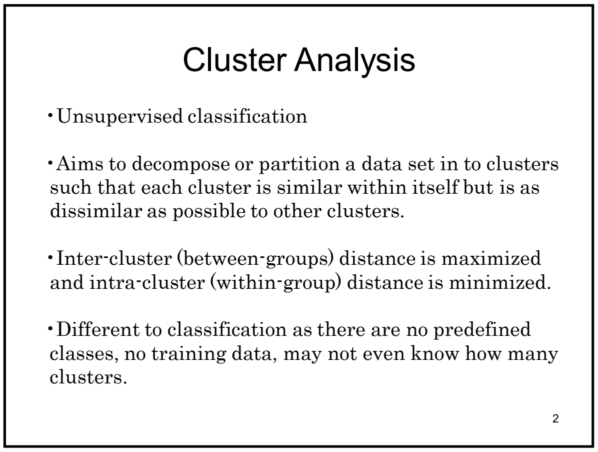
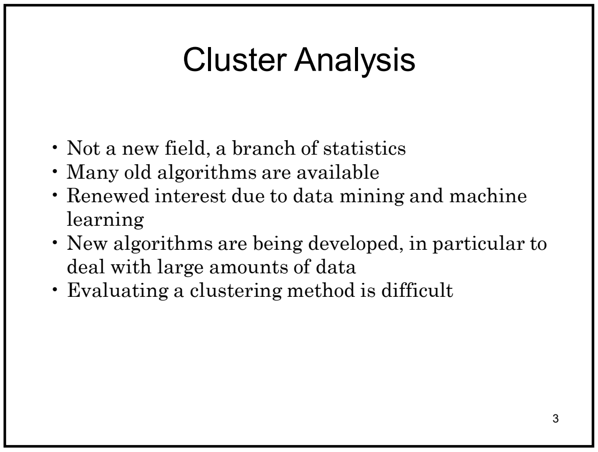


Clustering



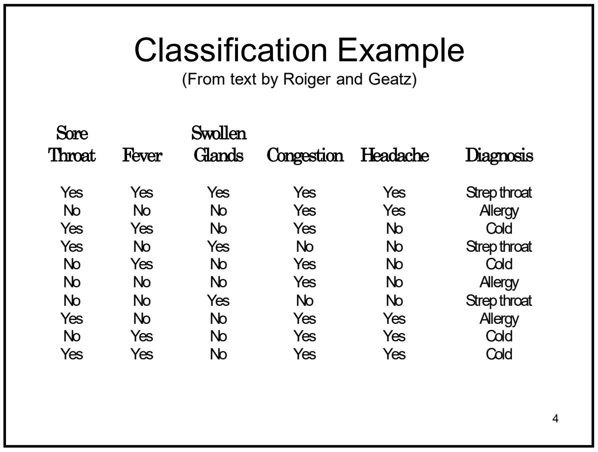
Cluster Analysis

- Unsupervised classification
- Aims to decompose or partition a data set in to clusters such that each cluster is similar within itself but is as dissimilar as possible to other clusters.
- Inter-cluster (between-groups) distance is maximized and intra-cluster (within-group) distance is minimized.
- Different to classification as there are no predefined classes, no training data, may not even know how many clusters.



Cluster Analysis

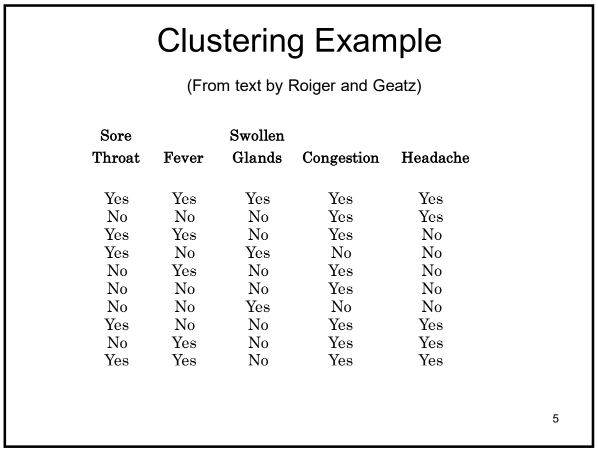
- Not a new field, a branch of statistics
- Many old algorithms are available
- Renewed interest due to data mining and machine learning
- New algorithms are being developed, in particular to deal with large amounts of data
- Evaluating a clustering method is difficult



Classification Example

(From text by Roiger and Geatz)

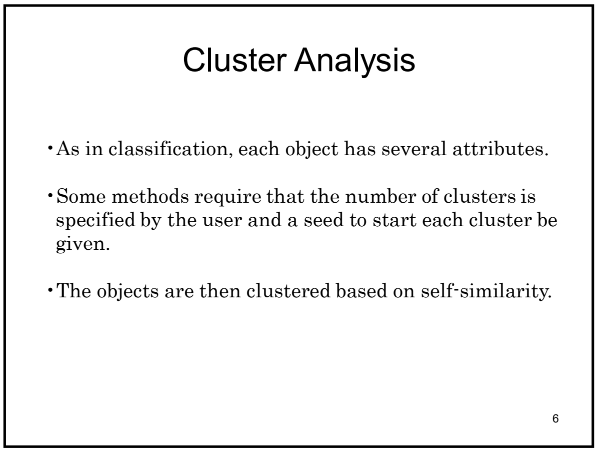
Sore		Swollen			
Throat	Fever	Glands	Congestion	Headache	Diagnosis
Yes	Yes	Yes	Yes	Yes	Strep throat
No	No	No	Yes	Yes	Allergy
Yes	Yes	No	Yes	No	Cold
Yes	No	Yes	No	No	Strep throat
No	Yes	No	Yes	No	Cold
No	No	No	Yes	No	Allergy
No	No	Yes	No	No	Strep throat
Yes	No	No	Yes	Yes	Allergy
No	Yes	No	Yes	Yes	Cold
Yes	Yes	No	Yes	Yes	Cold



Clustering Example

(From text by Roiger and Geatz)

Sore		Swollen		
Throat	Fever	Glands	Congestion	Headache
Yes	Yes	Yes	Yes	Yes
No	No	No	Yes	Yes
Yes	Yes	No	Yes	No
Yes	No	Yes	No	No
No	Yes	No	Yes	No
No	No	No	Yes	No
No	No	Yes	No	No
Yes	No	No	Yes	Yes
No	Yes	No	Yes	Yes
Yes	Yes	No	Yes	Yes



Cluster Analysis

- As in classification, each object has several attributes.
- Some methods require that the number of clusters is specified by the user and a seed to start each cluster be given.
- The objects are then clustered based on self-similarity.

Questions

- How to define a cluster?
- How to find if objects are similar or not?
- How to define some concept of distance between individual objects and sets of objects?

Types of Data

- Interval-scaled data - continuous variables on a roughly linear scale e.g. marks, age, weight. Units can affect the analysis. Distance in metres is obviously a larger number than in kilometres. May need to scale or normalise data (how?).
- Binary data – many variables are binary e.g. gender, married or not, u/g or p/g. How to compute distance between binary variables?

Types of Data

- Nominal data - similar to binary but can take more than two states e.g. colour, staff position. How to compute distance between two objects with nominal data?
- Ordinal data - similar to nominal but the different values are ordered in a meaningful sequence
- Ratio-scaled data - nonlinear scale data

Distance

A simple, well-understood concept. Distance has the following properties (assume x and y to be vectors):

- distance is always positive
- distance x to x is zero
- distance x to y cannot be greater than the sum of distance x to z and z to y
- distance x to y is the same as from y to x.

Examples: absolute value of the difference $\sum|x-y|$ (the Manhattan distance) or ∑(x-y)**2 (the Euclidean distance).

Distance

Three common distance measures are;

1. Manhattan Distance or the absolute value of the difference
   $D(x, y) = \sum|x-y|$

2. Euclidean distance
   $D(x, y) = (\sum(x-y)^2)^{½}$

3. Maximum difference
   $D(x, y) = \max_i |x_i - y_i|$

Distance

(From text by Roiger and Geatz)

Sore Throat	Fever	Swollen Glands	Congestion	Headache
Yes	Yes	Yes	Yes	Yes
No	No	No	Yes	Yes
Yes	Yes	No	Yes	No
Yes	No	Yes	No	No
No	Yes	No	Yes	No
No	No	No	Yes	No
No	No	Yes	No	No
Yes	No	No	Yes	Yes
No	Yes	No	Yes	Yes
Yes	Yes	No	Yes	Yes

Clustering

It is not always easy to predict how many clusters to expect from a set of data.

One may choose a number of clusters based on some knowledge of data. The result may be an acceptable set of clusters. This does not however mean that a different number of clusters will not provide an even better set of clusters. Choosing the number of clusters is a difficult problem.

Types of Methods

- Partitioning methods – given n objects, make k (≤n) partitions (clusters) of data and use iterative relocation method. It is assumed that each cluster has at least one object and each object belongs to only one cluster.
- Hierarchical methods - start with one cluster and then split it in smaller clusters or start with each object in an individual cluster and then try to merge similar clusters.

Types of Methods

- Density-based methods - for each data point in a cluster, at least a minimum number of points must exist within a given radius
- Grid-based methods - object space is divided into a grid
- Model-based methods - a model is assumed, perhaps based on a probability distribution

The K-Means Method

Perhaps the most commonly used method.

The method involves choosing the number of clusters and then dividing the given n objects choosing k seeds randomly as starting samples of these clusters.

Once the seeds have been specified, each member is assigned to a cluster that is closest to it based on some distance measure. Once all the members have been allocated, the mean value of each cluster is computed and these means essentially become the new seeds.

The K-Means Method

Using the mean value of each cluster, all the members are now re-allocated to the clusters. In most situations, some members will change clusters unless the first guesses of seeds were very good.

This process continues until no changes take place to cluster memberships.

Different starting seeds obviously may lead to different clusters.

The K-Means Method

Essentially the algorithm tries to build cluster with high level of similarity within clusters and low level of similarity between clusters. Similarity measurement is based on the mean values and the algorithm tries to minimize the squared-error function.

This method requires means of variables to be computed.

The method is scalable and is efficient. The algorithm does not find a global minimum, it rather terminates at a local minimum.

Example

Student	Age	Marks1	Marks2	Marks3
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	65
S8	21	53	56	59
S9	19	82	82	60
S10	40	76	60	78

Example

Let the three seeds be the first three records:

Student	Age	Mark1	Mark2	Mark3
S1	18	73	75	57
S2	18	79	85	75
S3	23	70	70	52

Now we compute the distances between the objects based on the four attributes. K-Means requires Euclidean distances but we use the sum of absolute differences as well as to show how the distance metric can change the results. Next page table shows the distances and their allocation to the nearest neighbor.

Distances

Student	Age	Marks1	Marks2	Marks3	Manhattan		Euclidean	
					Distance		Distance	
S1	18	73	75	57	Seed 1	C1	Seed 1	C1
S2	18	79	85	75	Seed 2	C2	Seed 2	C2
S3	23	70	70	52	Seed 3	C3	Seed 3	C3
S4	20	55	55	55	42/76/36	C3	27/43/22	C3
S5	22	85	86	87	57/23/67	C2	34/14/41	C2
S6	19	91	90	89	66/32/82	C2	40/19/47	C2
S7	20	70	65	60	23/41/21	C3	13/24/14	C1
S8	21	53	56	59	44/74/40	C3	28/42/23	C3
S9	19	82	82	60	20/22/36	C1	12/16/19	C1
S10	47	75	76	77	60/52/58	C2	33/34/32	C3

Example

Note the different allocations by the two different distance metrics. K-Means uses the Euclidean distance.

The means of the new clusters are also different as shown on the next slide. We now start with the new means and compute Euclidean distances. The process continues until there is no change.

Cluster Means

	Cluster	Age	Mark1	Mark2	Mark3
Manhattan	C1	18.5	77.5	78.5	58.5
	C2	24.8	82.8	80.3	82
	C3	21	62	61.5	57.8
	Seed 1	18	73	75	57
	Seed 2	18	79	85	75
	Seed 3	23	70	70	52
Euclidean	C1	19.0	75.0	74.0	60.7
	C2	19.7	85.0	87.0	83.7
	C3	26.0	63.5	60.3	60.8

Distances

Student	Age	Marks1	Marks2	Marks3	Distance	Cluster
C1	18.5	77.5	78.5	58.5		
C2	24.8	82.8	80.3	82		
C3	21	62	61.5	57.8		
S1	18	73	75	57	58/2/36	C1
S2	18	79	85	75	30/18/63	C2
S3	23	70	70	52	22/67/28	C1
S4	20	55	55	55	46/91/26	C3
S5	22	85	86	87	51/7/78	C2
S6	19	91	90	89	60/15/93	C2
S7	20	70	65	60	19/56/22	C1
S8	21	53	56	59	44/89/22	C3
S9	19	82	82	60	16/32/48	C1
S10	47	75	76	77	52/63/43	C3

C1= s1, s3, s7, s9 C2=s2, s5, s6 C3=s4, s8, s10

Comments

- Euclidean Distance used in the last slide since that is what K-Means requires.
- The results of using Manhattan distance and Euclidean distance are quite different
- The last iteration changed the cluster membership of S3 from C3 to C1. New means are now computed followed by computation of new distances. The process continues until there is no change.

Distance Metric

Most important issue in methods like the K-Means method is the distance metric although K-Means considersEuclidean distance.

There is no real definition of what metric is a good one but the example shows that different metrics, used on the same data, can produce different results making it very difficult to decide which result is the best.

Whatever metric is used is somewhat arbitrary although extremely important.

Hierarchical Clustering

Quite different than partitioning. Involves gradually merging different objects into clusters (called agglomerative) or dividing large clusters into smaller ones (called divisive).

We consider one method.

Agglomerative Clustering

The algorithm normally starts with each cluster consisting of a single data point. Using a measure of distance, the algorithm merges two clusters that are nearest, thus reducing the number of clusters. The process continues until all the data points are in one cluster.

The algorithm requires that we be able to compute distances between two objects and also between two clusters.

Distance between Clusters

Single Link method – nearest neighbour - distance between two clusters is the distance between the two closest points, one from each cluster.

Complete Linkage method – furthest neighbour – distance between two clusters is the distance between the two furthest points, one from each cluster.

Distance between Clusters

Centroid method – distance between two clusters is the distance between the two centroids or the two centres of gravity of the two clusters.

Unweighted pair-group average method –distance between two clusters is the average distance between all pairs of objects in the two clusters. This means p*n distances need to be computed if p and n are the number of objects in each of the clusters

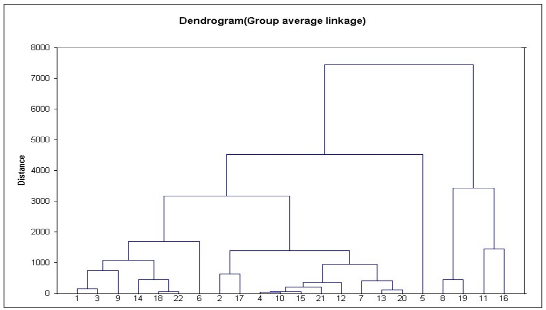
Algorithm

- Each object is assigned to a cluster of its own
- Build a distance matrix by computing distances between every pair of objects
- Find the two nearest clusters
- Merge them and remove them from the list
- Update matrix by including distance of all objects from the new cluster
- Continue until all objects belong to one cluster

The next slide shows an example of the final result of hierarchical clustering.

Hierarchical Clustering

http://www.resample.com/xlminer/help/HClst/HClst_ex.htm



Example

We now consider the simple example about students' marks and use the distance between two objects as the Manhattan distance.

We will use the centroid method for distance between clusters.

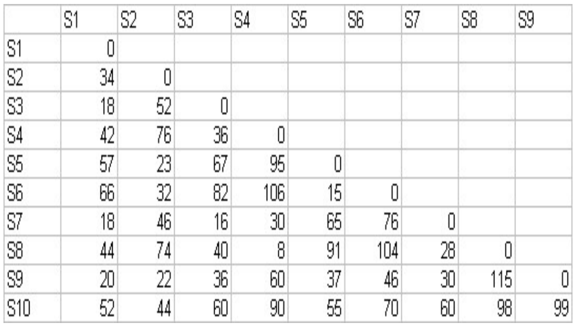
We first calculate a matrix of distances.

20.5 54 55.5 57

Example

Student	Age	Marks1	Marks2	Marks3
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

Distance Matrix

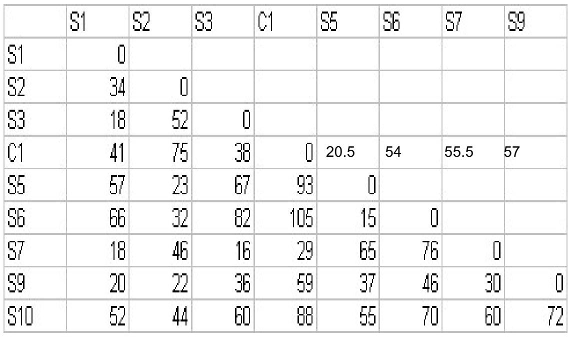


	S1	S2	S3	S4	S5	S6	S7	S8	S9
S1	0								
S2	34	0							
S3	18	52	0						
S4	42	76	36	0					
S5	57	23	67	95	0				
S6	66	32	82	106	15	0			
S7	18	46	16	30	65	76	0		
S8	44	74	40	8	91	104	28	0	
S9	20	22	36	60	37	46	30	115	0
S10	52	44	60	90	55	70	60	98	99

Example

- The matrix gives distance of each object with every other object.
- The smallest distance of 8 is between object 4 and object 8. They are combined and put where object 4 was.
- Compute distances from this cluster and update the distance matrix.

Updated Matrix



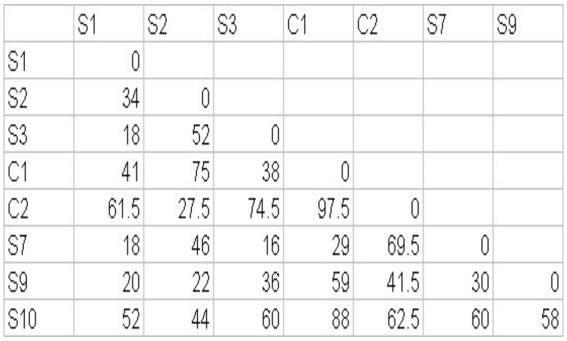
	S1	S2	S3	C1	S5	S6	S7	S9
S1	0							
S2	34	0						
S3	18	52	0					
C1	41	75	38	0	20.5	54	55.5	57
S5	57	23	67	93	0			
S6	66	32	82	105	15	0		
S7	18	46	16	29	65	76	0	
S9	20	22	36	59	37	46	30	0
S10	52	44	60	88	55	70	60	72

20.5 88 88 88

Note

- The smallest distance now is 15 between the objects 5 and 6. They are combined in a cluster and 5 and 6 are removed.
- Compute distances from this cluster and update the distance matrix.

Updated Matrix

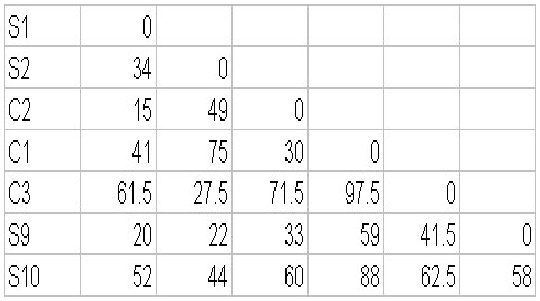


	S1	S2	S3	C1	C2	S7	S9
S1	0						
S2	34	0					
S3	18	52	0				
C1	41	75	38	0			
C2	61.5	27.5	74.5	97.5	0		
S7	18	46	16	29	69.5	0	
S9	20	22	36	59	41.5	30	0
S10	52	44	60	88	62.5	60	58

Next

- Look at shortest distance again.
- S3 and S7 are at a distance 16 apart. We merge them and put C3 where S3 was.
- The updated distance matrix is given on the next slide. It shows that C2 and S1 have the smallest distance and are then merged in the next step.
- We stop short of finishing the example.

Updated matrix



S1	0					
S2	34	0				
C2	15	49	0			
C1	41	75	30	0		
C3	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

Divisive Clustering

Opposite to Agglomerative. Less commonly used.

Start with one cluster that has all the objects and seek to split the cluster into two which themselves are then split into even smaller clusters.

The split may be based on using one variable at a time or using all the variables together.

The algorithm terminates when a termination condition is met or when each cluster has only one object.

Divisive Clustering

There are two types of divisive methods.

The split may be based on using one variable at a time or using all the variables together.

Monothetic - split a cluster using only one variable at a time. How does one choose the variable?

Polythetic - split a cluster using all of the attributes together. How to allocate objects to clusters?

Example

Student	Age	Marks1	Marks2	Marks3
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

Algorithm

A typical (polythetic) divisive algorithm works like the following:

1. Decide on a method of measuring the distance between two objects and decide a threshold distance.
2. Create a distance matrix by computing distances between all pairs of objects within the cluster. Sort these distances in ascending order.
3. Find the two most dissimilar objects (i.e. the two objects that have the largest distance between them).

Algorithm

4. If the distance is smaller than a pre-specified threshold and there is no other cluster that needs to be divided then stop.
5. Use the pair of objects identified in Step 3 as seeds of a K-means type algorithm to create two new clusters. Examine all objects. Place each object in the cluster which has the seed with a smaller distance.
6. If there is only one object in each cluster then stop otherwise continue with Step 2.

Divisive Method

Two major issues that need resolving are:

- Which cluster to split next?
- How to split the cluster?

Which cluster to split?

Number of possibilities:

- Split the clusters in some sequential order
- Split the cluster that has the largest number of objects
- Split the cluster that has the largest variation within it.

The first two approaches are clearly very simple but the third approach is better since it is based on splitting a cluster that has the most variation which is a sound criterion.

How to split a cluster?

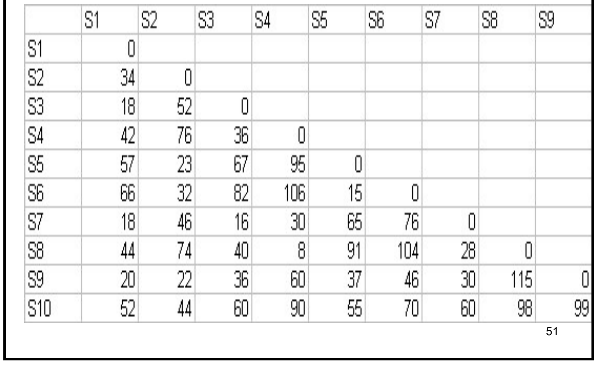
A simple approach for splitting a cluster is to split the cluster based on distances between the objects in the cluster as outlined in the algorithm.

A distance matrix is created and the two most dissimilar objects are selected as seeds of two new clusters. A method like the K-Means method may then be used to split the cluster.

Example

Student	Age	Marks1	Marks2	Marks3
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

Distance Matrix



	S1	S2	S3	S4	S5	S6	S7	S8	S9
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

Example

- The matrix gives distance of each object with every other object.
- The largest distance is between S8 and S9. They become the seeds of two new clusters. Use K-means to split the group in two clusters.

Example

- Distances of other objects from S8 and S9 are:

	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10
C1-S8	44	74	40	8	91	104	28	0	115	98
C2-S9	20	22	36	60	37	46	30	115	0	99

The two Clusters are:

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

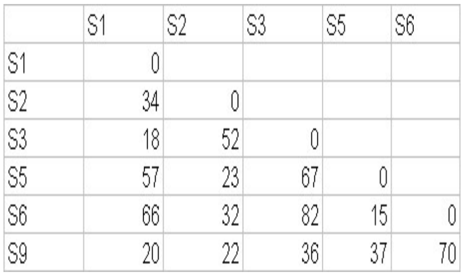
Next split

- We now decide which cluster to split next and then repeat the process.
- The two Clusters are:

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

- We may want to split the larger cluster first
- Find the largest distance in C2 first. The information is available in the distance matrix given earlier.

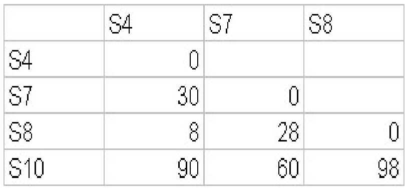
Distances in C2



	S1	S2	S3	S5	S6
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	70

In cluster C2 the largest distance is 82 between S3 and S6. C2 can be split with these as seeds.

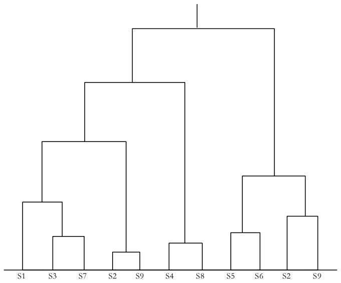
Distances in C1



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

In cluster C1 (distance matrix on next slide) the largest distance is 98 between S8 and S10. C1 can be split with these seeds.

Sample Result



Hierarchical Clustering

- The ordering produced can be useful in gaining some insight into the data.
- The major difficulty is that once an object is allocated to a cluster it cannot be moved to another cluster even if the initial allocation was incorrect
- Different distance metrics can produce different results