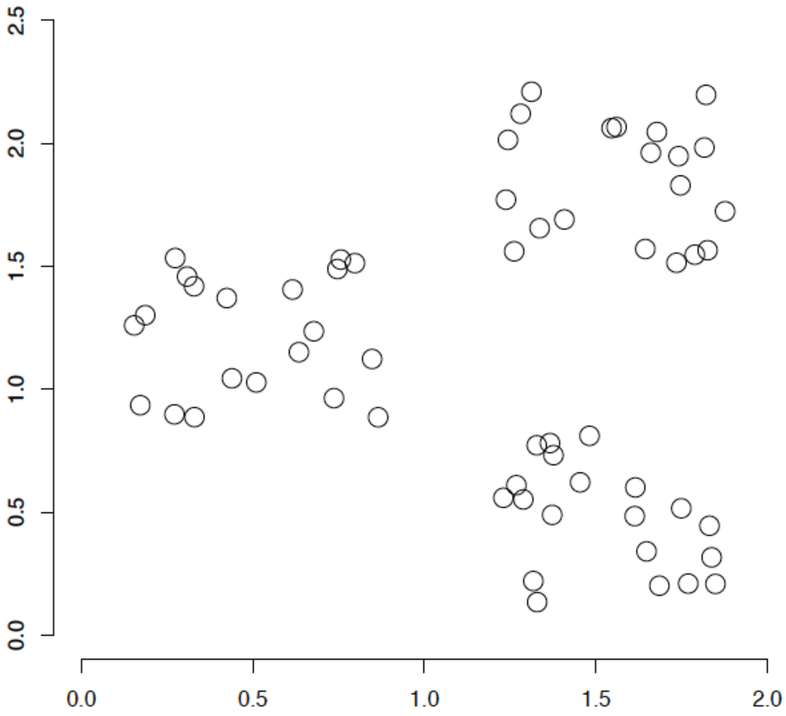
Data Clustering Techniques – Unsupervised Learning

a presentation by
B.Sivaselvan
Associate Professor, CSE
IIITD&M Kancheepuram

- Clustering - common form of Unsupervised Learning.
- Unsupervised learning

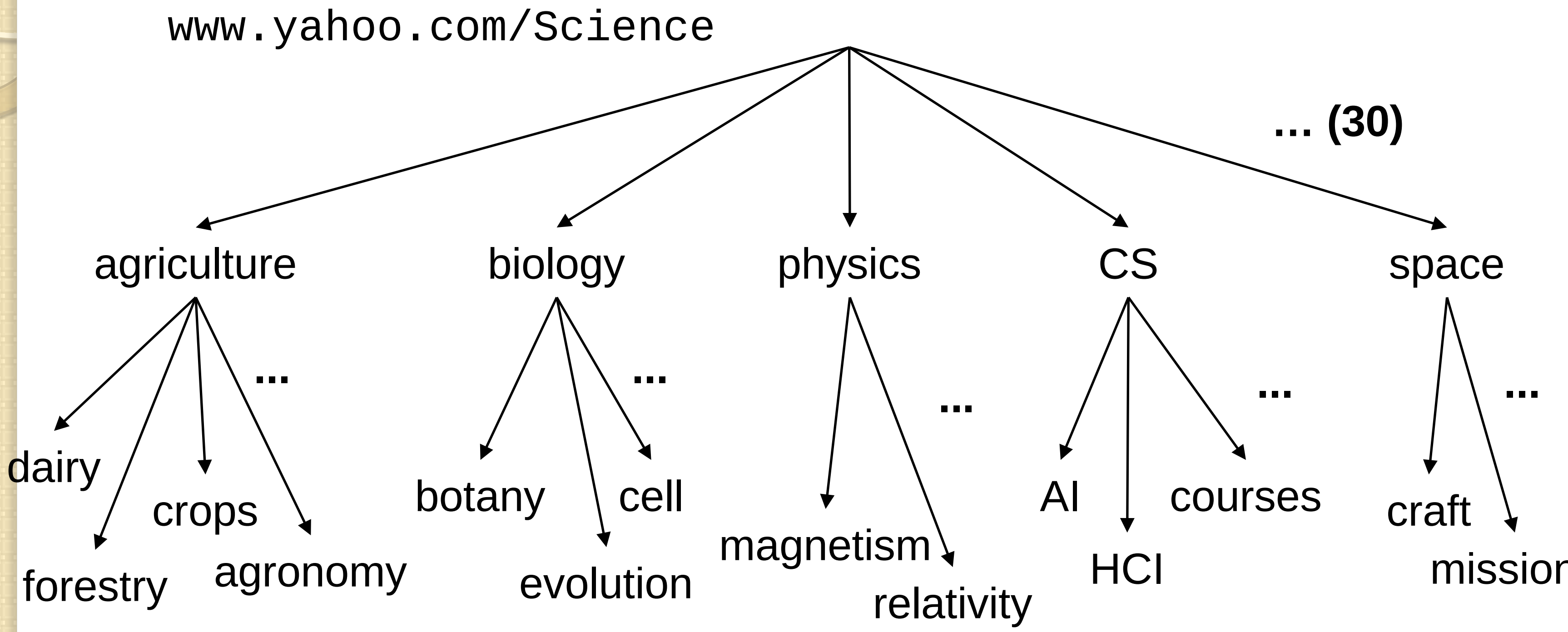
  learning from raw data, as opposed to supervised data where a classification of examples is given
- the process of grouping a set of objects into classes of similar objects
  - Documents within a cluster should be similar.
  - Documents from different clusters should be dissimilar.



Some classical apps of Clustering

- Whole corpus analysis/navigation
  - Better user interface: search without typing
- For improving recall in search applications
  - Better search results
- For better navigation of search results
  - Effective "user recall" will be higher
- Recommender Systems
- Image Segmentation , Customer Segmentation,…..

Yahoo! Hierarchy *isn't* clustering but *is* the kind of output you want from clustering



www.yahoo.com/Science

… (30)

agriculture

dairy

crops

forestry

agronomy

…

biology

botany

cell

evolution

…

physics

magnetism

relativity

…

CS

AI

HCI

courses

…

space

craft

mission

…

- *Cluster hypothesis* - Documents in the same cluster behave similarly with respect to relevance to information needs
- Therefore, to improve search recall:
  - Cluster docs in corpus a priori
  - When a query matches a doc *D*, also return other docs in the cluster containing *D*
- The query "car" will also return docs containing *automobile*
  - Because clustering grouped together docs containing *car* with those containing *automobile.*

Clustering Algorithms

- Flat algorithms
  - Usually start with a random (partial) partitioning
  - Refine it iteratively
    - *K* means clustering
    - (Model based clustering)
- Hierarchical algorithms
  - Bottom-up, agglomerative
  - (Top-down, divisive)

Partitioning Algorithms

- Partitioning method: Construct a partition of *n* documents into a set of *K* clusters
- Given: a set of documents and the number *K*
- Find: a partition of *K* clusters that optimizes the chosen partitioning criterion
- **k-Means: Step-By-Step Example**

Subject	A	B
1	1.0	1.0
2	1.5	2.0
3	3.0	4.0
4	5.0	7.0
5	3.5	5.0
6	4.5	5.0
7	3.5	4.5

- Initial Setup K =2 Clusters and representatives
- A & B values of the two individuals furthest apart (using the Euclidean distance measure),

	Individual	Mean Vector (centroid)
Group 1	1	(1.0, 1.0)
Group 2	4	(5.0, 7.0)

- Other data points examined in sequence and allocated to the cluster to which they are closest, in terms of Euclidean distance to the cluster mean
- Mean vector is recalculated each time a new member is added. This leads to the following series of steps:

	Cluster 1		Cluster 2	
Step	Individual	Mean Vector (centroid)	Individual	Mean Vector (centroid)
1	1	(1.0, 1.0)	4	(5.0, 7.0)
2	1, 2	(1.2, 1.5)	4	(5.0, 7.0)
3	1, 2, 3	(1.8, 2.3)	4	(5.0, 7.0)
4	1, 2, 3	(1.8, 2.3)	4, 5	(4.2, 6.0)
5	1, 2, 3	(1.8, 2.3)	4, 5, 6	(4.3, 5.7)
6	1, 2, 3	(1.8, 2.3)	4, 5, 6, 7	(4.1, 5.4)

Now the initial partition has changed, and the two clusters at this stage having the following characteristics:

	Individual	Mean Vector (centroid)
Cluster 1	1, 2, 3	(1.8, 2.3)
Cluster 2	4, 5, 6, 7	(4.1, 5.4)

Not sure that each individual has been assigned to the right cluster.
compare each individual's distance to its own cluster mean and to that of the opposite cluster. And we find:

Individual	Distance to mean (centroid) of Cluster 1	Distance to mean (centroid) of Cluster 2
1	1.5	5.4
2	0.4	4.3
3	2.1	1.8
4	5.7	1.8
5	3.2	0.7
6	3.8	0.6
7	2.8	1.1

- Only individual 3 is nearer to the mean of the opposite cluster (Cluster 2) than its own (Cluster 1).
- In other words, each individual's distance to its own cluster mean should be smaller than the distance to the other cluster's mean (which is not the case with individual 3).
- Thus, individual 3 is relocated to Cluster 2 resulting in the new partition:

	Individual	Mean Vector (centroid)
Cluster 1	1, 2	(1.3, 1.5)
Cluster 2	3, 4, 5, 6, 7	(3.9, 5.1)

Hierarchical methods - **agglomerative** and **divisive.**

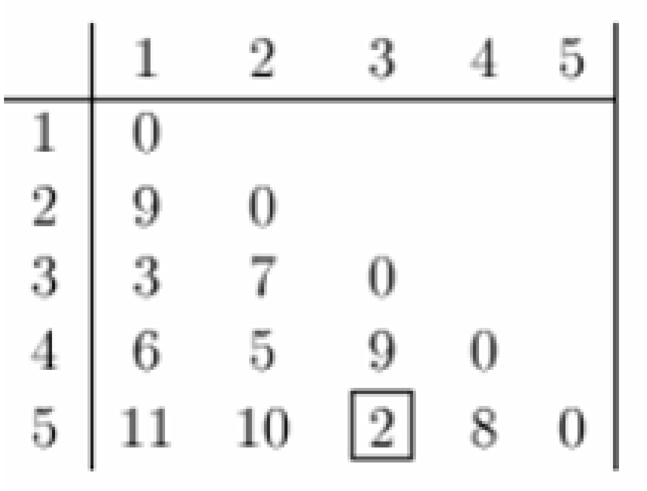
Agglomerative methods:

- Start with partition P_n, where each object forms its own cluster.
- Merge the two closest clusters, obtaining P_{n-1}.
- Repeat merge until only one cluster is left.

Divisive methods

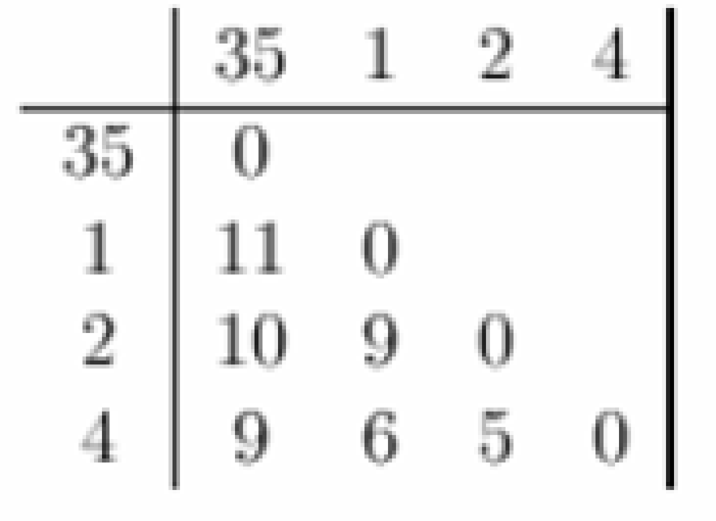
- Start with P_1.
- Split the collection into two clusters that are as homogenous (and as
  different from each other) as possible.
- Apply splitting procedure recursively to the clusters.

Only the lower triangle is shown, because the upper triangle can be filled in by reflection.



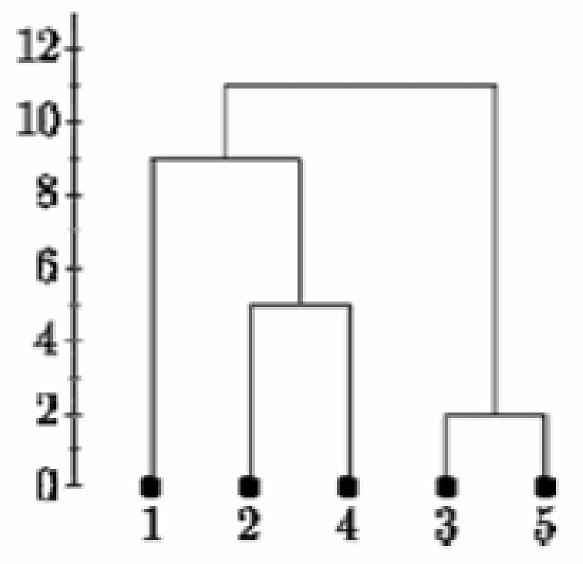
	1	2	3	4	5
1	0				
2	9	0			
3	3	7	0		
4	6	5	9	0	
5	11	10	$\boxed{2}$	8	0

- smallest distance is between three and five and they get linked up or merged first into a the cluster '35'.
- remove the 3 and 5 entries, and replace it by an entry "35" .
- distance between "35" and every other item is the maximum of the distance between this item and 3 and this item and 5.
- For example, d(1,3)= 3 and d(1,5)=11. So, D(1,"35")=11. This gives us the new distance matrix.



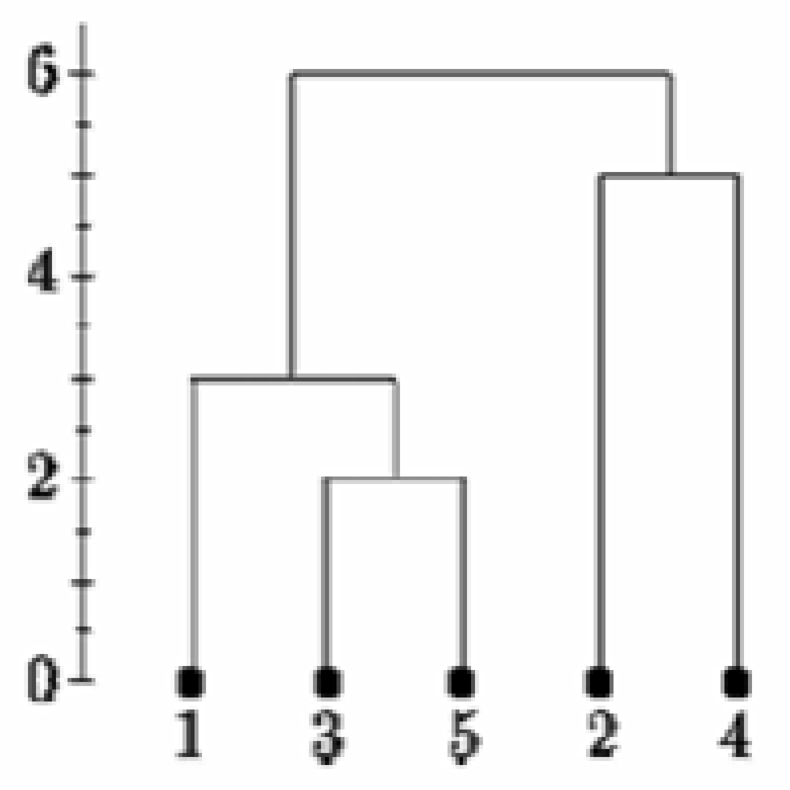
	35	1	2	4
35	0			
1	11	0		
2	10	9	0	
4	9	6	5	0

after 6 steps, everything is clustered. Below Plot (classically called as Dendrogram) On this plot, the y-axis shows the distance between the objects at the time they were clustered. This is called the cluster height. Different visualizations use different measures of cluster height.



Below is the single linkage dendrogram for the same distance matrix. It starts with cluster "35" but the distance between "35" and each item is now the minimum of d(x,3) and d(x,5). So c(1,"35")=3.

Single Linkage



Determining clusters

One of the problems with hierarchical clustering is that there is no objective way to say how many clusters there are.

If we cut the single linkage tree at the point shown below, we would say that there are two clusters.

However, if we cut the tree lower we might say that there is one cluster and two singletons.