Data Mining for Business

Cluster Analysis

Dr. Shipra Maurya
Department of Management Studies
IIT (ISM) Dhanbad
Email: shipra@iitism.ac.in

Cluster Analysis

- It is an unsupervised learning method

- Divides a large group (called cluster) of observations into smaller group in such a manner that-
  - Within cluster – homogeneity
  - Between clusters - heterogeneity

Applications of Cluster Analysis

- Market segmentation
- For creating balanced portfolios based on the financial performance variables such as return, volatility, market capitalization etc.
- Search engines use this to cluster queries that users submit
- Clustering of investors according to their attribute preferences

Types of Clustering Algorithms

- **Hierarchical Methods:**
  - Agglomerative methods
  - Divisive methods

- **Non-hierarchical Methods:**
  - K-means clustering

Hierarchical Methods

- **Agglomerative methods** – start each object in its own separate cluster i.e. n cluster of size 1. At each process, find closest clusters and join them until a single cluster is obtained.

- **Divisive methods** - start from 1 cluster, to get to n clusters, The observation with the highest average dissimilarity (farthest from the cluster) is reassigned to its own cluster.

Non-Hierarchical method - K-means Clustering

- Using a pre-specified number of clusters, the method assigns records to each cluster.

- Less computationally intensive and are therefore preferred with large datasets.

Distance measures

- We need to group the objects together on the basis of mutual proximity or similarity and this can measured by distance

- In both Hierarchical and Non-hierarchical clustering, we need to define two types of distances:
  - Distance between two records
  - Distance between two clusters

Distance measures for two Records

- **For Numerical data:**
  - Euclidean Distance
  - Statistical distance or Mahalanobis distance
  - Manhattan distance
- **For Categorical data (similarity measures not distance measure):**
  - Matching coefficient
  - Jaquard's coefficient
- **For Mixed data:**
  - Gower's similarity measure

Euclidean distance

- Most popular distance measure

$$d_{ij} = \sqrt{(x_{i1} - x_{j1})^2 + (x_{i2} - x_{j2})^2 + \cdots + (x_{ip} - x_{jp})^2}.$$

Company	Fixed	RoR	Cost	Load	Demand	Sales	Nuclear	Fuel Cost
Arizona Public Service	1.06	9.2	151	54.4	1.6	9077	0.0	0.628
Boston Edison Co.	0.89	10.3	202	57.9	2.2	5088	25.3	1.555

$$d_{12} = \sqrt{(1.06 - 0.89)^2 + (9.2 - 10.3)^2 + (151 - 202)^2 + \cdots + (0.628 - 1.555)^2}$$
$$= 3989.408.$$

Euclidean distance Features

- Euclidean distance is scale dependent and hence data has to be normalized else variables with larger scales will have a much greater influence over the total distance

- It ignores the relationship between the measurements. If the measurements are highly correlated then Mahalanobis distance must be used

- It is sensitive to outliers. If the data contains outliers and careful removal is not possible, Manhattan distance can be used

Similarity measures for Categorical data

		Record j: 0	Record j: 1	
Record i	0	a	b	$a+b$
	1	c	d	$c+d$
		$a+c$	$b+d$	p

where a denotes the number of variables for which records i and j do not have that attribute (they each have value 0 on that attribute), d is the number of variables for which the two records have the attribute present, and so on. The most useful similarity measures in this situation are:

Matching coefficient: $(a + d)/p$.

Jaquard's coefficient: $d/(b + c + d)$. This coefficient ignores zero matches. This is desirable when we do not want to consider two people to be similar simply because a large number of characteristics are absent in both. For example, if *owns a Corvette* is one of the variables, a matching "yes" would be evidence of similarity, but a matching "no" tells us little about whether the two people are similar.

Gower's Similarity measure for Mixed data

- It is a weighted average of the distances computed for each variable, after scaling each variable to a [0,1] scale. It is defined as:

$$s_{ij} = \frac{\sum_{m=1}^{p} w_{ijm} s_{ijm}}{\sum_{m=1}^{p} w_{ijm}},$$

where s_{ijm} is the similarity between records i and j on measurement m, and w_{ijm} is a binary weight given to the corresponding distance.

The similarity measures s_{ijm} and weights w_{ijm} are computed as follows:

1. For continuous measurements, $s_{ijm} = 1 - \frac{|x_{im} - x_{jm}|}{\max(x_m) - \min(x_m)}$ and $w_{ijm} = 1$ unless the value for measurement m is unknown for one or both of the records, in which case $w_{ijm} = 0$.

Gower's Similarity measure for Mixed data

2. For binary measurements, $s_{ijm} = 1$ if $x_{im} = x_{jm} = 1$ and 0 otherwise. $w_{ijm} = 1$ unless $x_{im} = x_{jm} = 0$.

3. For nonbinary categorical measurements, $s_{ijm} = 1$ if both records are in the same category, and otherwise $s_{ijm} = 0$. As in continuous measurements, $w_{ijm} = 1$ unless the category for measurement m is unknown for one or both of the records, in which case $w_{ijm} = 0$.

Distance measures for two Clusters

- Minimum distance
- Maximum distance
- Average distance
- Centroid Linkage

Distance measures for two Clusters

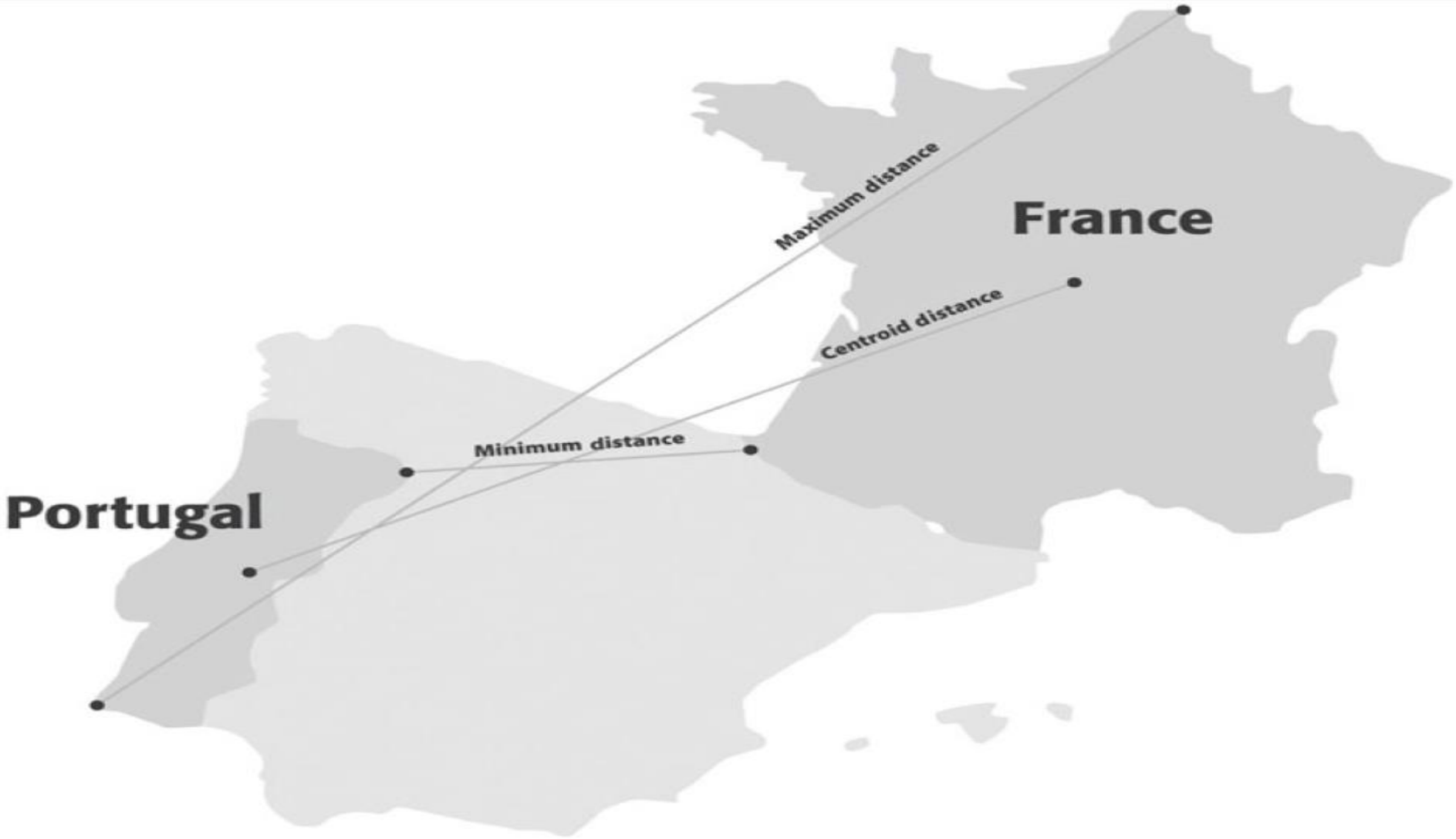


Figure 15.2 Two-dimensional representation of several different distance measures between Portugal and France

K-means Clustering

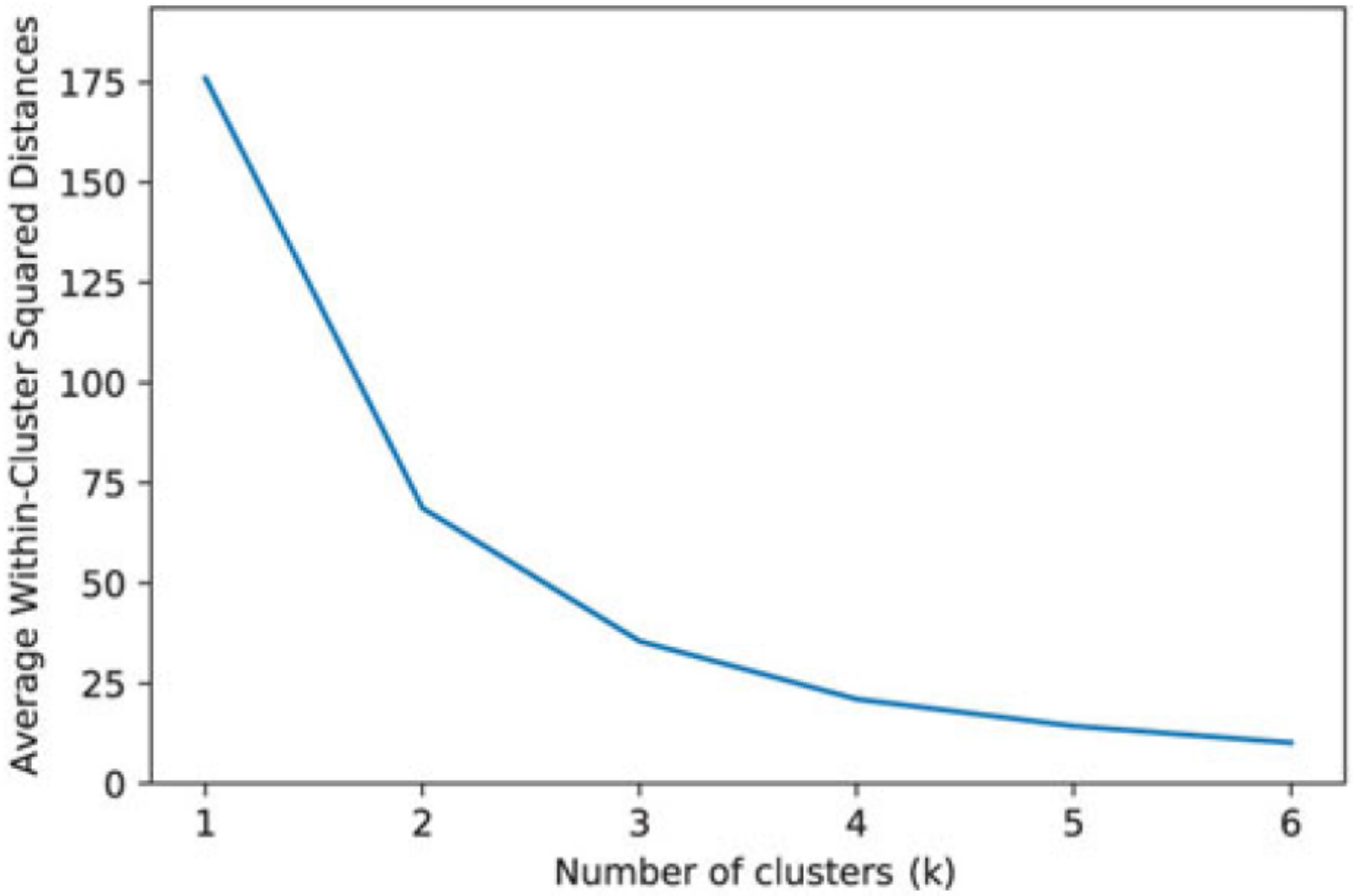
- Using a pre-specified number of clusters, the method assigns records to each cluster.

- Less computationally intensive and are therefore preferred with large datasets.

K-means Clustering Steps

1. Select an initial partition of the data into K clusters
2. Calculate the centroid for each clusters
3. Calculate the sum of squared distance of each object to its cluster centroid
4. Reassign each object to the cluster whose centroid is closest
5. Re-compute the centroid for each cluster
6. Repeat until centroid doesn't change

K-means Clustering – How to Choose k?



Hierarchical Agglomerative Clustering

- Start each object in its own separate cluster i.e. n cluster of size 1. At each process, find closest clusters and join them until a single cluster is obtained.

- **Linkage methods:**
  - Single linkage (minimum distance)
  - Complete linkage (maximum distance)
  - Average linkage (Average Distance)
  - Centroid linkage
  - Ward's method

Single Linkage

SINGLE-LINKAGE CLUSTERING

Steps of the Iterative Process

Step 0. Start with all objects in separate clusters (i.e., n clusters with one object in each). Denote these clusters $C_1, C_2, C_3, \ldots, C_n$. In this initial step, the distance between two clusters is defined to be the distance between the two objects they contain; that is,

$$d_{C_i C_j} = d_{ij}$$

Let $t = 1$ be an index of the iterative process.

Step 1. Find the smallest distance between any two clusters. Denote these two closest clusters C_i and C_j.

Step 2. Amalgamate clusters C_i and C_j to form a new cluster denoted C_{n+t}.

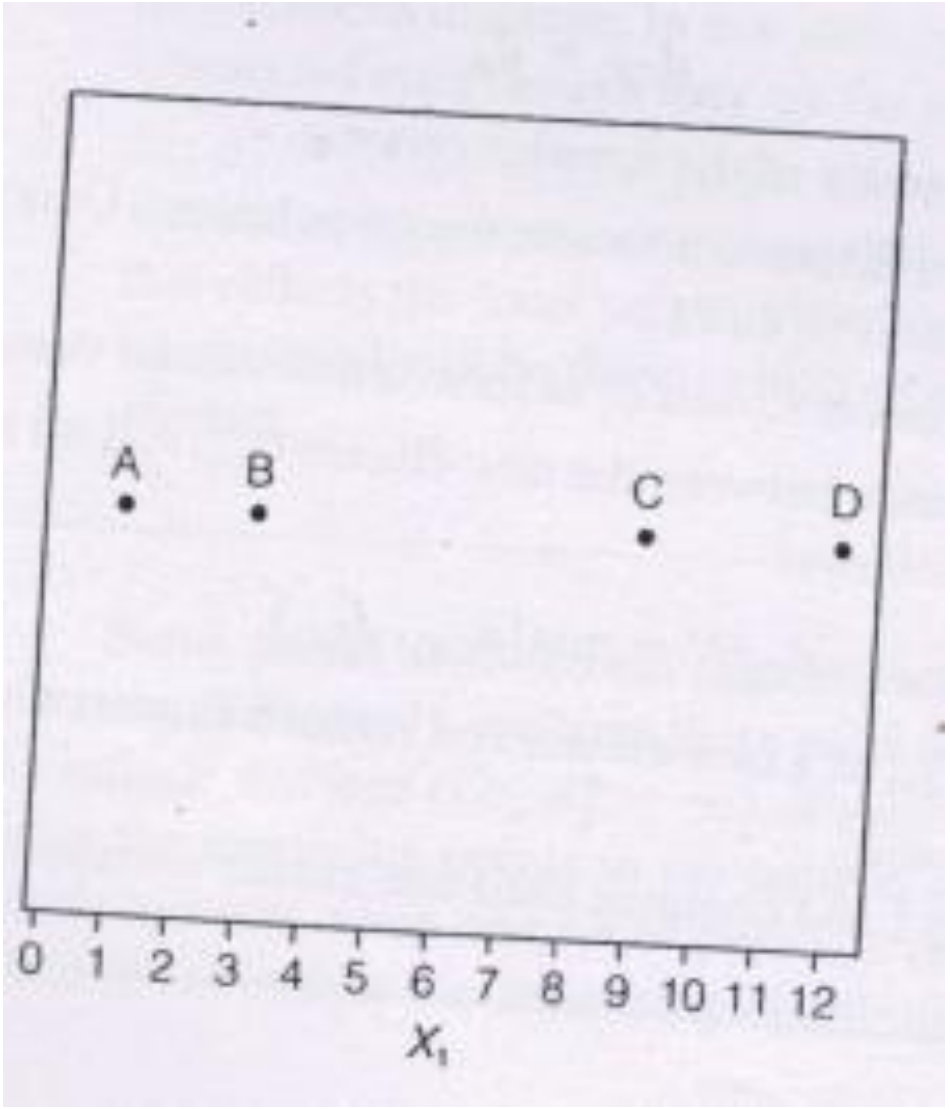
Step 3. Define the distance between the new cluster C_{n+t} and all remaining clusters C_k as follows:

$$d_{C_{n+t} C_k} = \min\{d_{C_i C_k}, d_{C_j C_k}\}.$$

Step 4. Add cluster C_{n+t} as a new cluster and remove clusters C_i and C_j. Let $t = t + 1$.

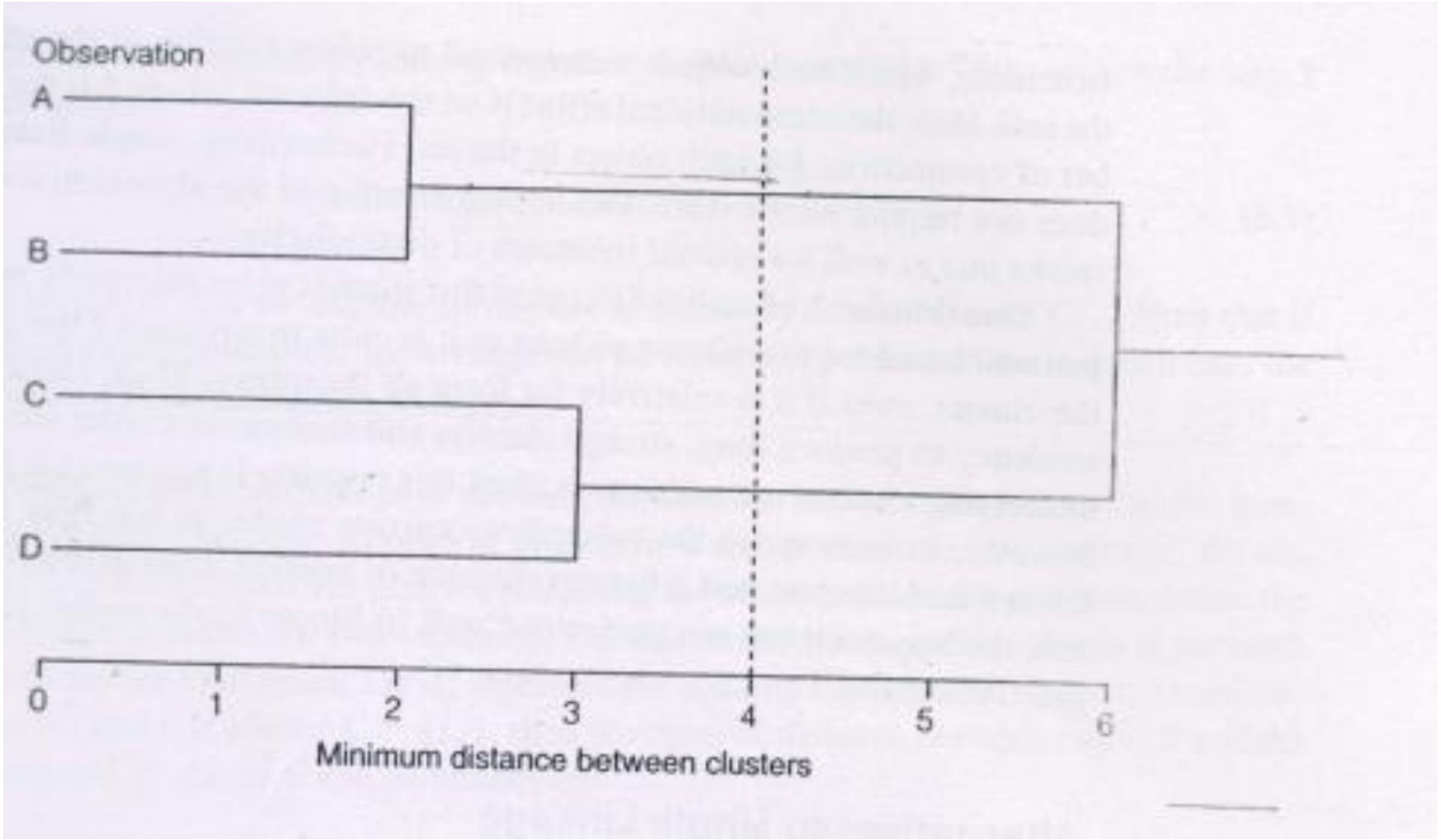
Step 5. Return to step 1 and continue until one cluster remains.

Single Linkage clustering Example



Iteration 0: {A}, {B}, {C}, {D}
{A} joins {B} at distance $d = 2$
Iteration 1: {A, B}, {C}, {D}
{C} joins {D} at distance $d = 3$
Iteration 2: {A, B}, {C, D}
{A, B} joins {C, D} at distance $d = 6$
Iteration 3: {A, B, C, D}

Dendrogram for Choosing number of Clusters



Example (1/2)

Distance Matrix

	a	b	c	d	e
a	0	17	21	31	23
b	17	0	30	34	21
c	21	30	0	28	39
d	31	34	28	0	43
e	23	21	39	43	0

In this example, $D_1(a, b) = 17$ is the smallest value of D_1, so we join elements a and b.

Update Distance Matrix- max for complete and min for single linkage

$$D_2((a, b), c) = max(D_1(a, c), D_1(b, c)) = max(21, 30) = 30$$

$$D_2((a, b), d) = max(D_1(a, d), D_1(b, d)) = max(31, 34) = 34$$

$$D_2((a, b), e) = max(D_1(a, e), D_1(b, e)) = max(23, 21) = 23$$

Example (2/2)

Updated Distance Matrix

	(a,b)	c	d	e
(a,b)	0	**30**	**34**	**23**
c	**30**	0	*28*	*39*
d	**34**	*28*	0	*43*
e	**23**	*39*	*43*	0

Here, $D_2((a,b),e) = 23$ is the lowest value of D_2, so we join cluster (a,b) with element e.

$$D_3(((a,b),e),c) = max(D_2((a,b),c), D_2(e,c)) = max(30,39) = 39$$

$$D_3(((a,b),e),d) = max(D_2((a,b),d), D_2(e,d)) = max(34,43) = 43$$

Ward's Method

- Ward's method (sometimes referred to as minimum variance method) adopts a slightly different strategy. Instead of joining the two closest clusters, ward's method seeks to join the two clusters whose merger leads to the smallest with-in cluster sum of squares (i.e., minimum within-group variance).

- Aggregate two clusters that lead to the smallest with-in cluster sum of squares at each step

  - **Error Sum of Squares**: $ESS = \sum_i \sum_j \sum_k |X_{ijk} - \bar{x}_{i \cdot k}|^2$
  - **Total Sum of Squares**: $TSS = \sum_i \sum_j \sum_k |X_{ijk} - \bar{x}_{\cdot \cdot k}|^2$
  - **R-Square**: $r^2 = \frac{\text{TSS-ESS}}{\text{TSS}}$

Limitations of Hierarchical clustering

- Sensitive to outliers
- It requires lots of computational power and storage as each record is treated as a cluster to start with
- It has low stability as dropping records or re-ordering data might change the results

Thank you!