VICTORIA UNIVERSITY OF WELLINGTON
TE HERENGA WAKA

AIML 231/DATA 302— Week 7

Clustering

Dr Bach Hoai Nguyen

School of Engineering and Computer Science

Victoria University of Wellington

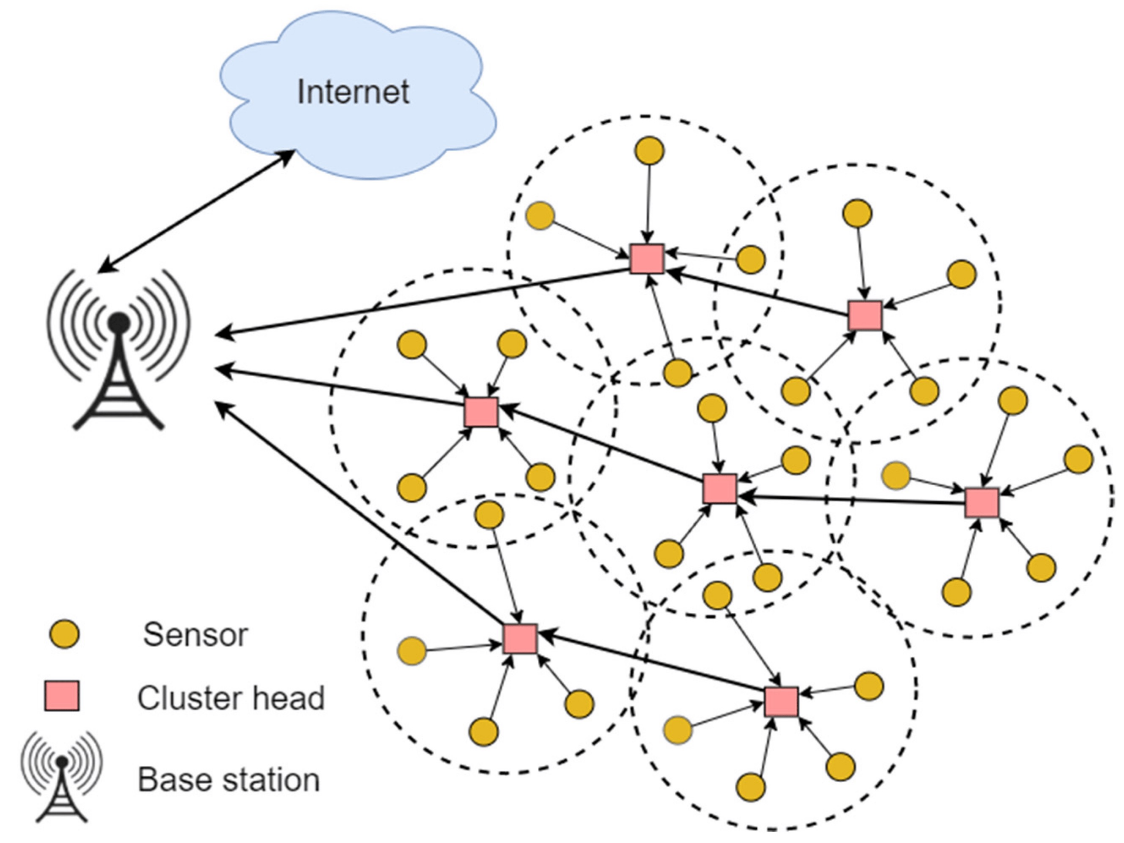
Bach.Nguyen@vuw.ac.nz

Week Overview

- ★ What is clustering?
- ★ Distance measures
- ★ Clustering Models: K-Means, Agglomerative Clustering
- ★ Clustering metric

Clustering in Wireless Networks

- Information transferred from Sensors to a Base Station
- Lifetime of sensor batteries are limited



- Divide sensors into groups (clusters) by distance
  - Each cluster is managed by a cluster head (CH)
  - CH group gathers data from sensors and send data to the Base Station
  - Removes redundant data and reduces network energy consumption

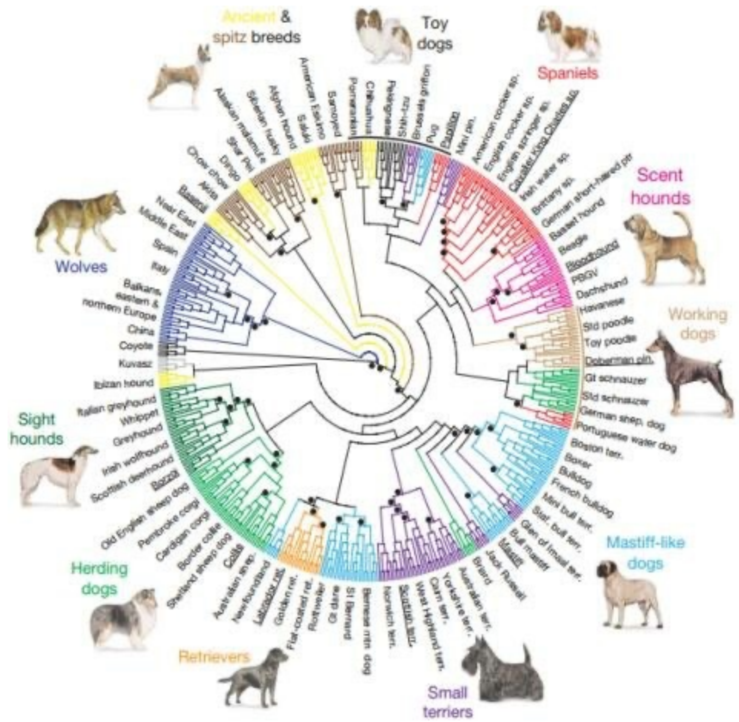
Jubair, A.M.; Hassan, R.; Aman, A.H.M.; Sallehudin, H.; Al-Mekhlafi, Z.G.; Mohammed, B.A.; Alsaffar, M.S. Optimization of Clustering in Wireless Sensor Networks: Techniques and Protocols. *Appl. Sci.* **2021**, *11*, 11448. https://doi.org/10.3390/app112311448

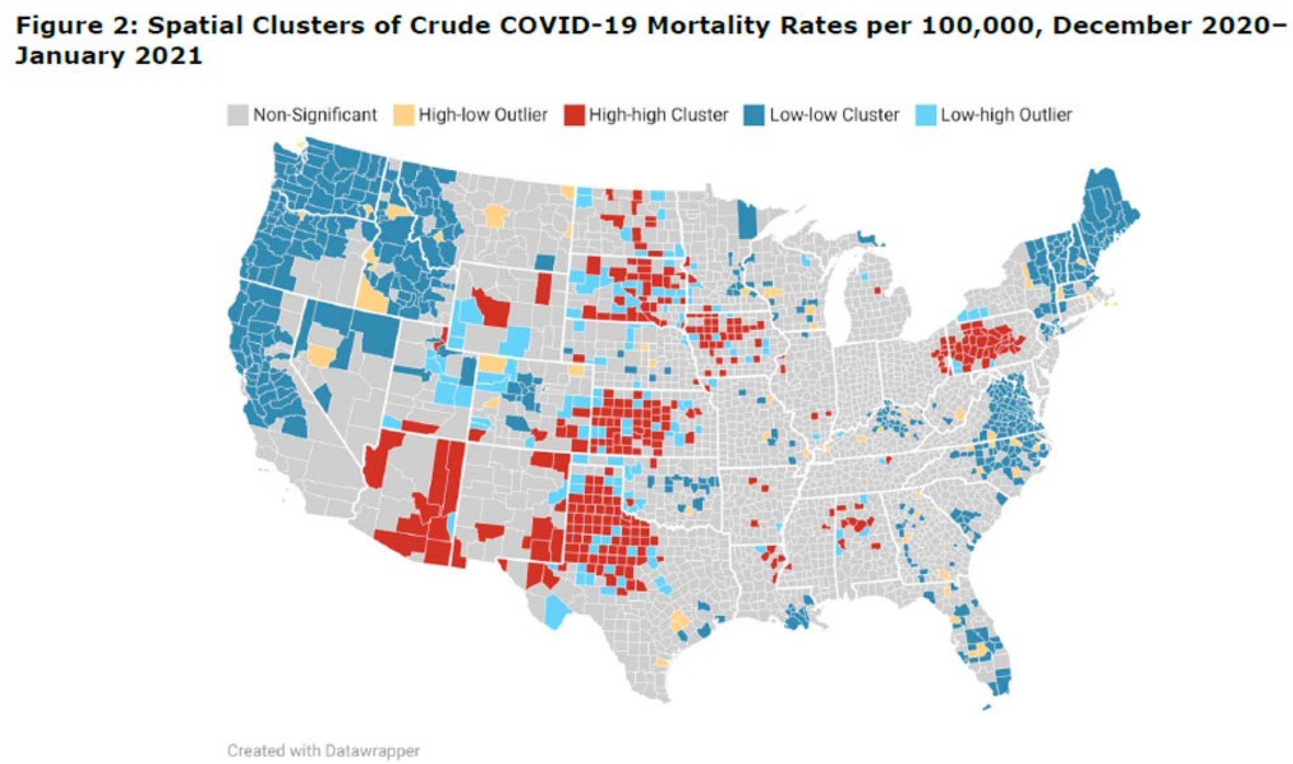
Customer Segmentation

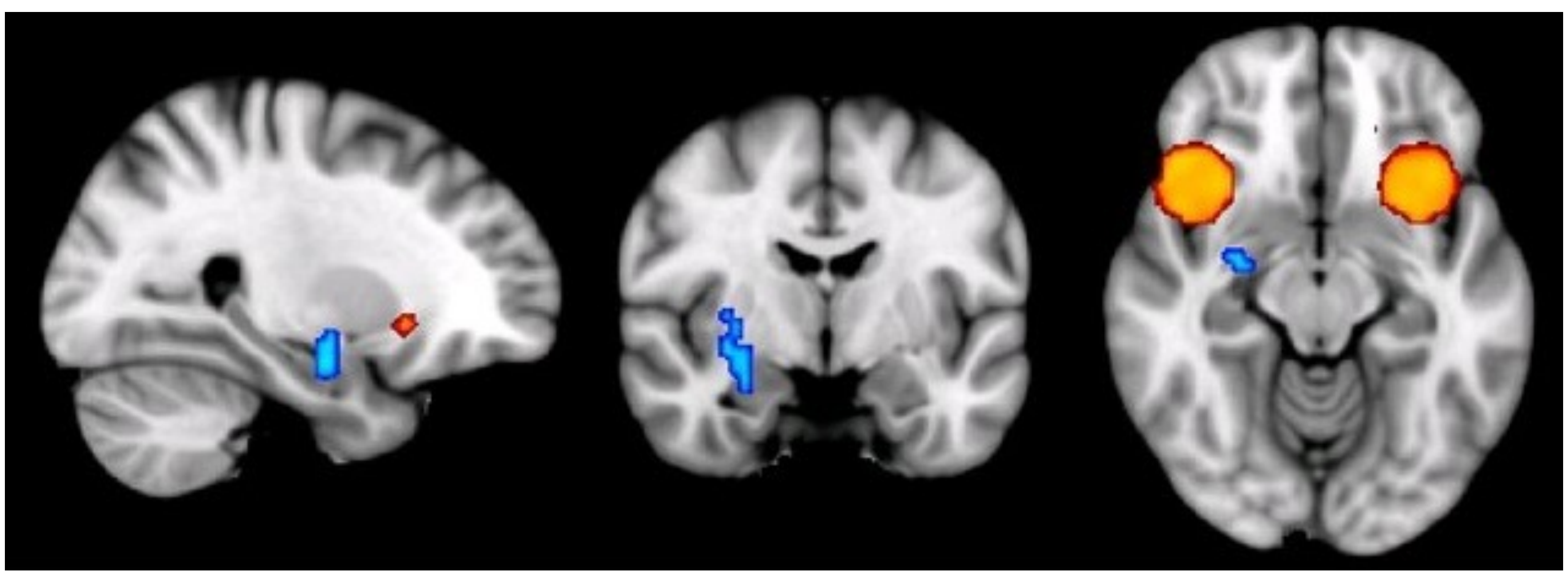
- Divide customers into similar groups based on common characteristics:
  - Historical purchases
  - Geographical locations
  - Products and services
  - Socio-economic: income, education
- Each group has its own effective marketing strategies
- Effective communication
- Better customer supports
- Increase revenue

https://www.linkedin.com/pulse/8020-rule-customer-segmentation-how-grow-your-revenue-alex-zbooker/

Other applications



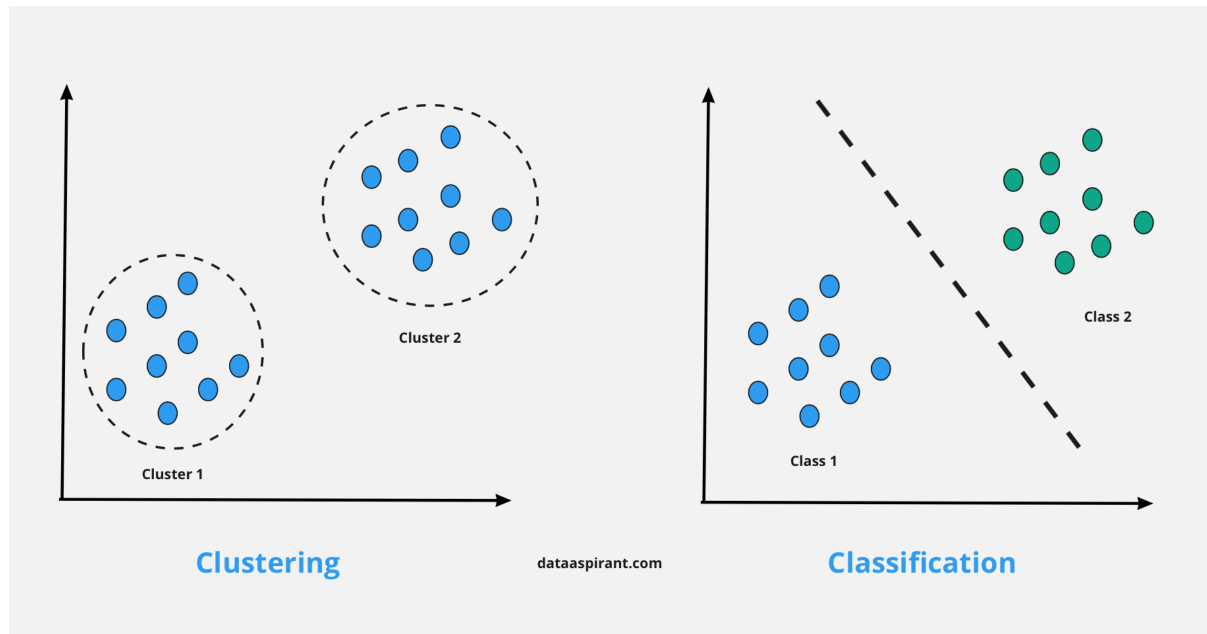




Chavez, Robert S., and Dylan D. Wagner. "Mass univariate testing biases the detection of interaction effects in whole-brain analysis of variance." *BioRxiv* (2017): 130773.

Clustering

Clustering: the task of grouping a set of objects in such a way that objects in the same group (called a cluster) are more similar to each other than to those in other groups



https://dataaspirant.com/4-difference-between-clustering-and-classification/

	Clustering	*Classification*
Number of Classes	Unknown	Known
Training Data	No required	Required
Aim	Work on existing data	Classify future instances into classes

What is similarity?

- Similarity is hard to define
- Typically measured by a distance or similarity measure
- Different measures lead to different clusters -> clustering is subjective

Distance measures

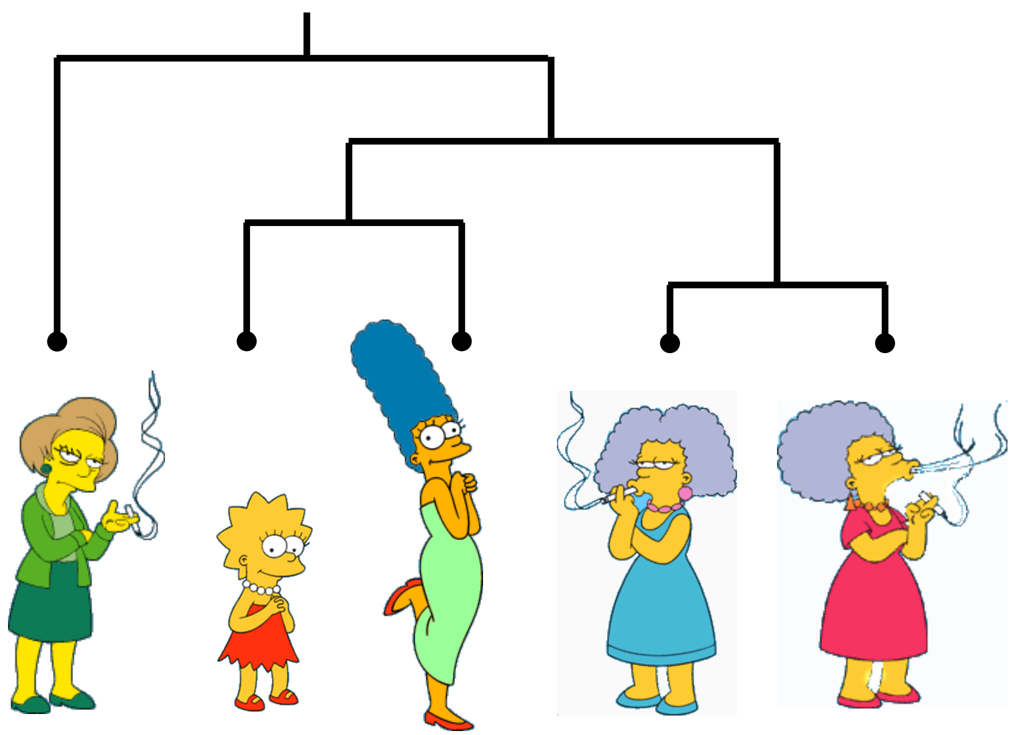
- Let O_1 and O_2 be two objects from the universe of possible objects. The distance (dissimilarity) between O_1 and O_2 is a real number denoted by $D(O_1,O_2)$

- Depends on the data types
  - Numerical features: Euclidian distance, Manhattan distance, Cosine distance
  - Categorical features: Hamming distance

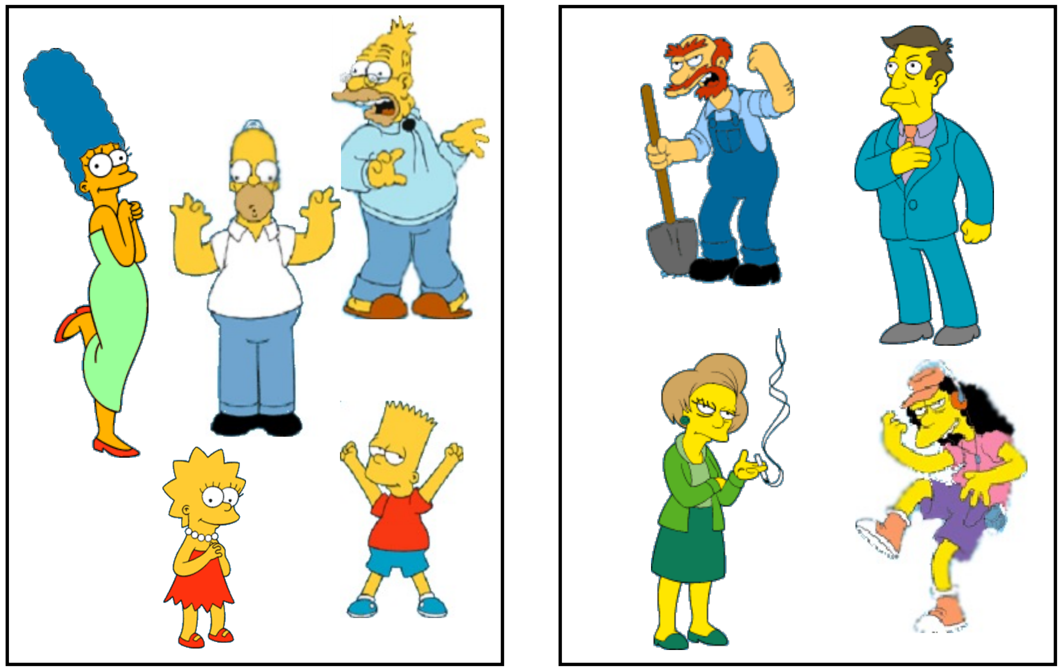
Clustering methods

- Two main types of clustering
- **Partitional algorithms**: Construct various partitions and then evaluate them by some criterion
- **Hierarchical algorithms**: Create a hierarchical decomposition of the set of objects using some criterion

Hierarchical



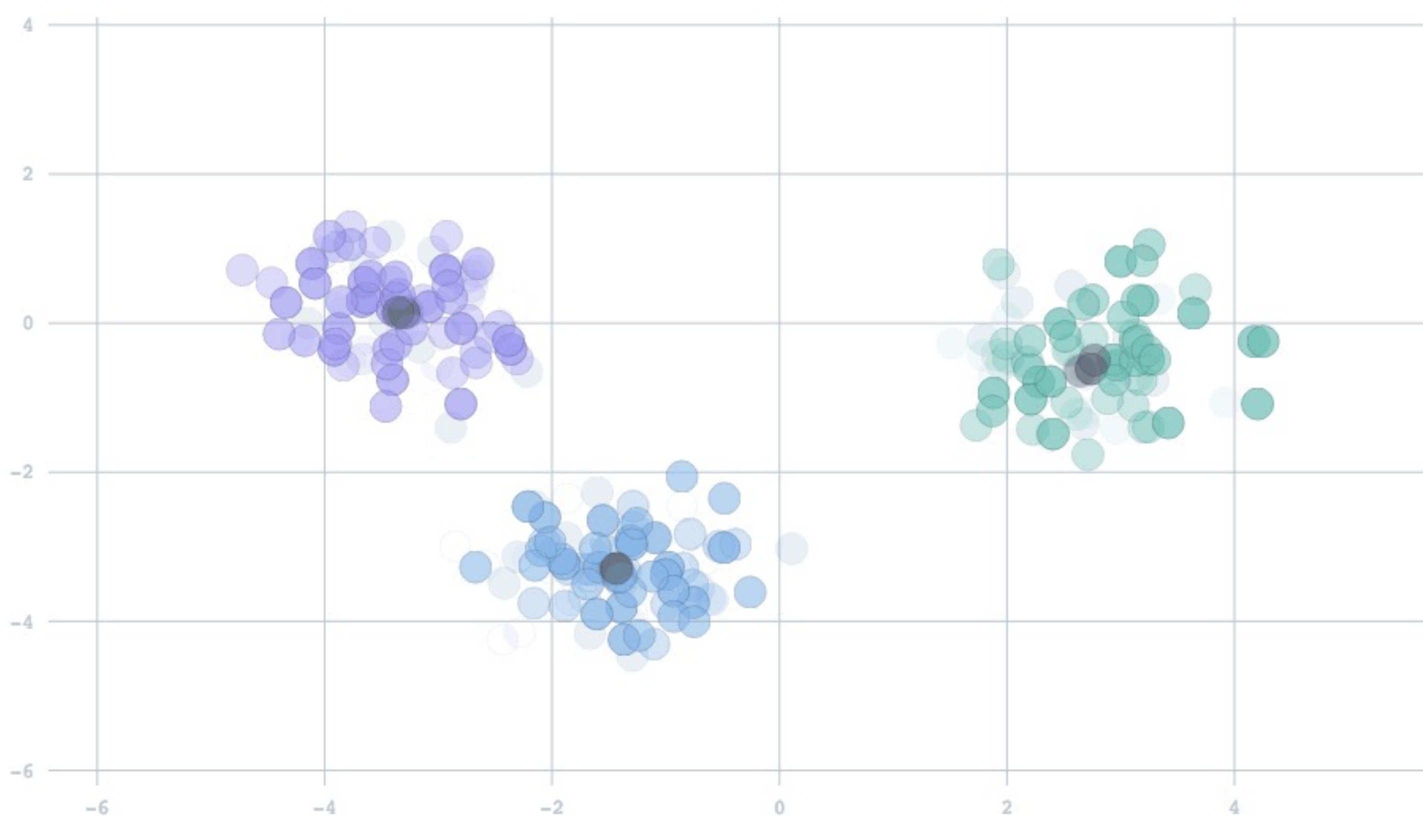
Partitional



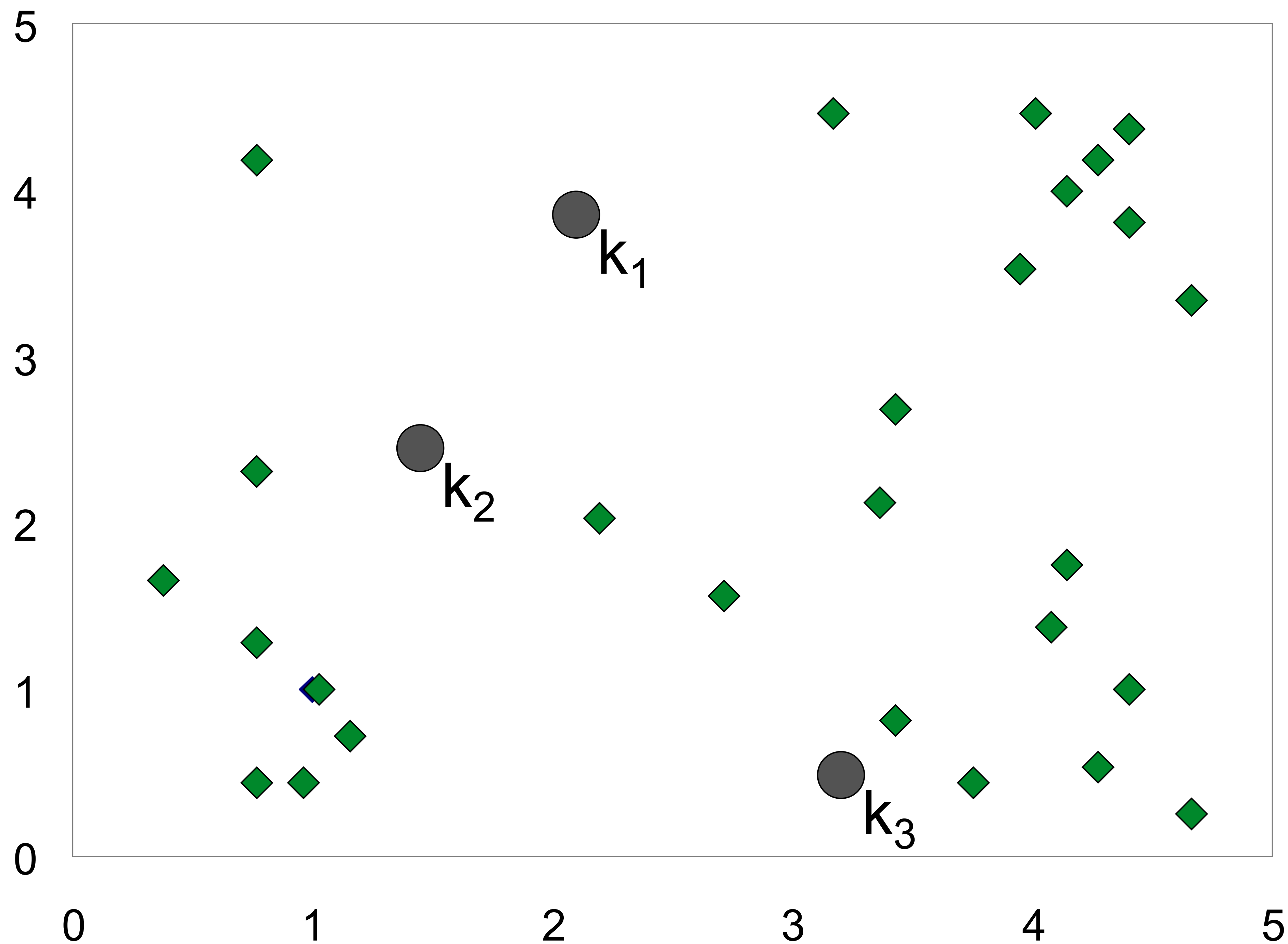
K-Means

- A partitional method

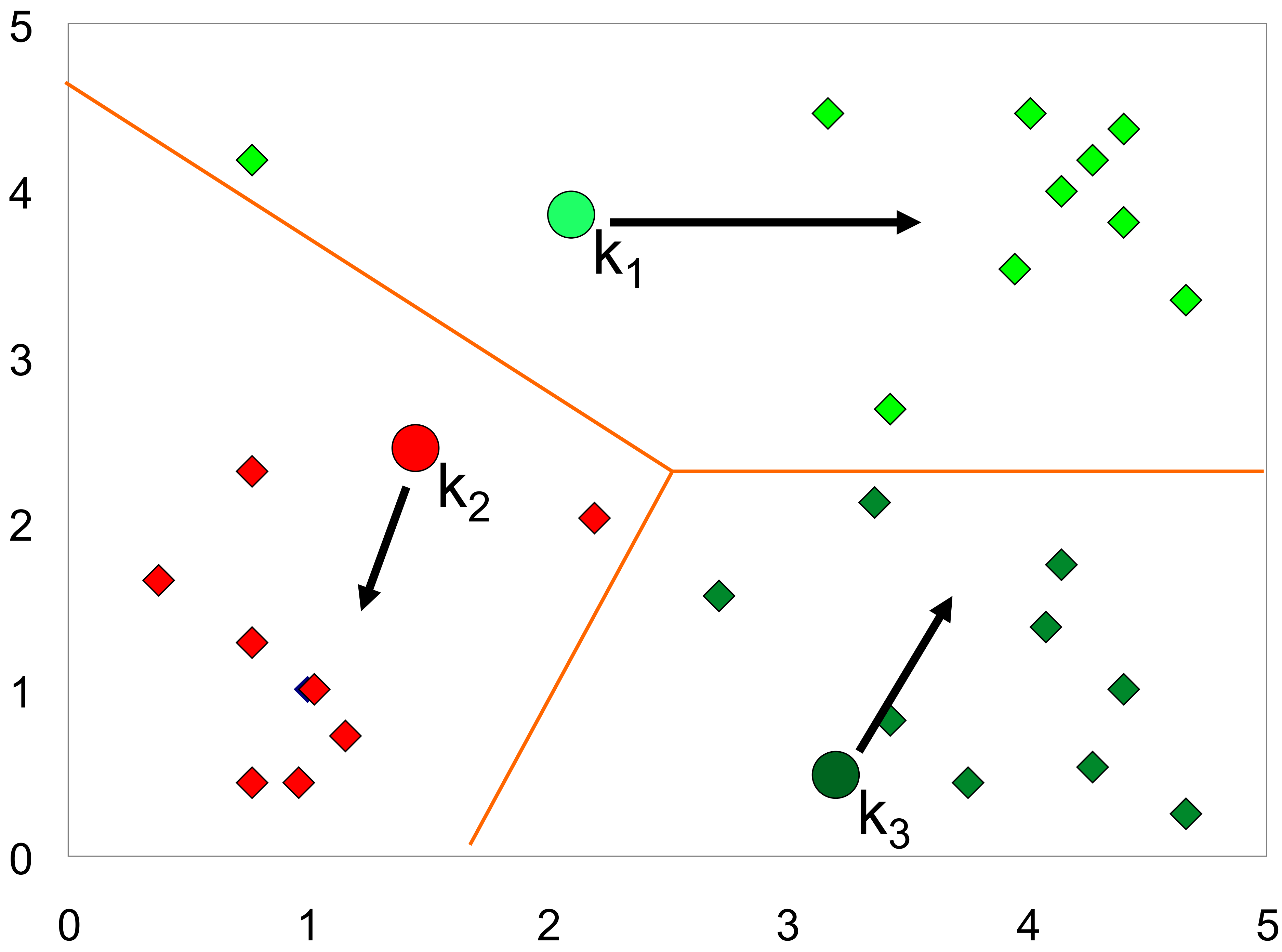
1. Start with *K* random cluster centres, aka centroids
2. Reassign: assign each instance/object to the nearest centroids
3. Updating: compute the new centroid for each cluster as the mean of the objects assigned to the cluster
4. Repeat step 2 until no change to the centroids



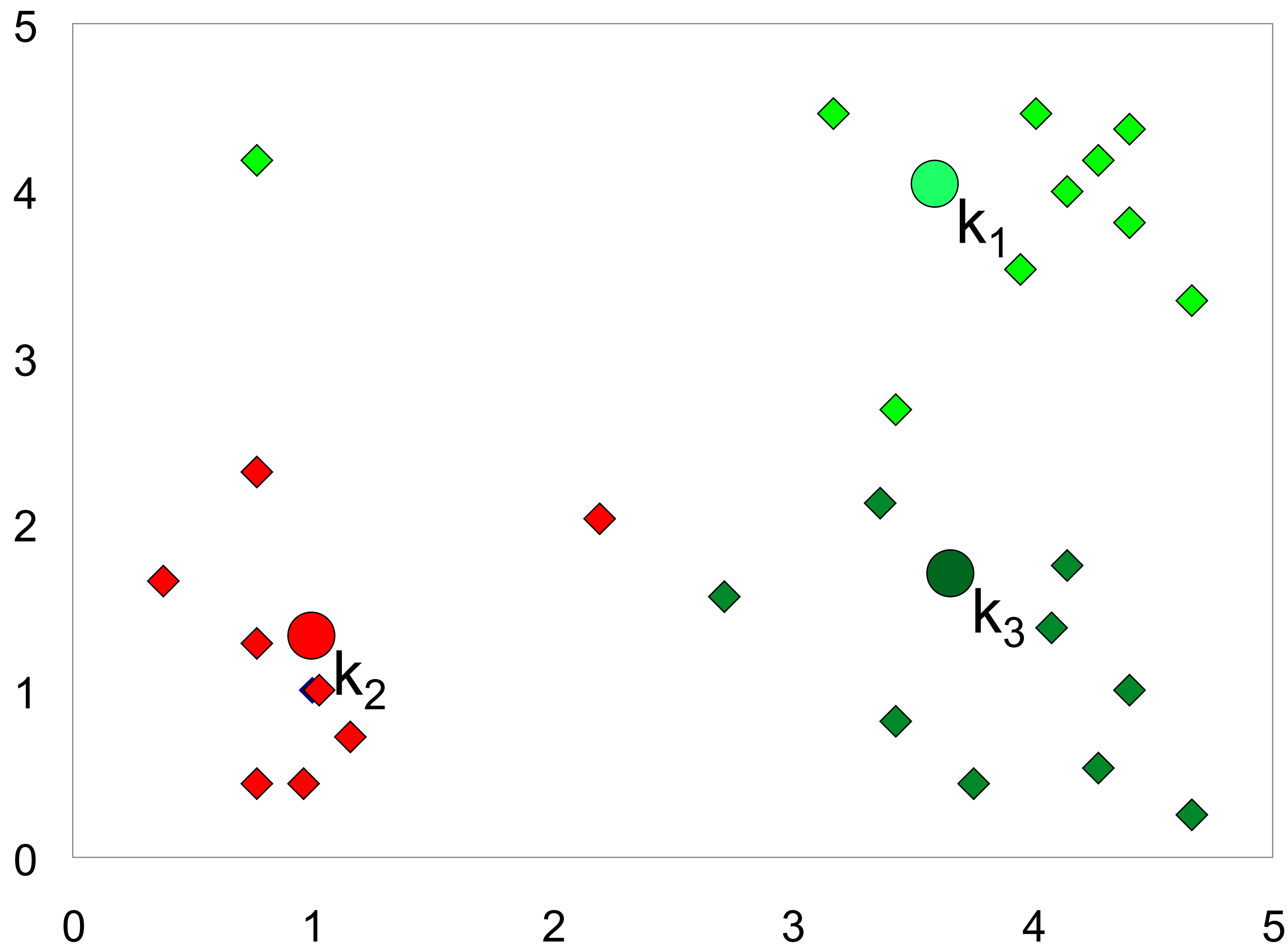
K-means Clustering: Step 1



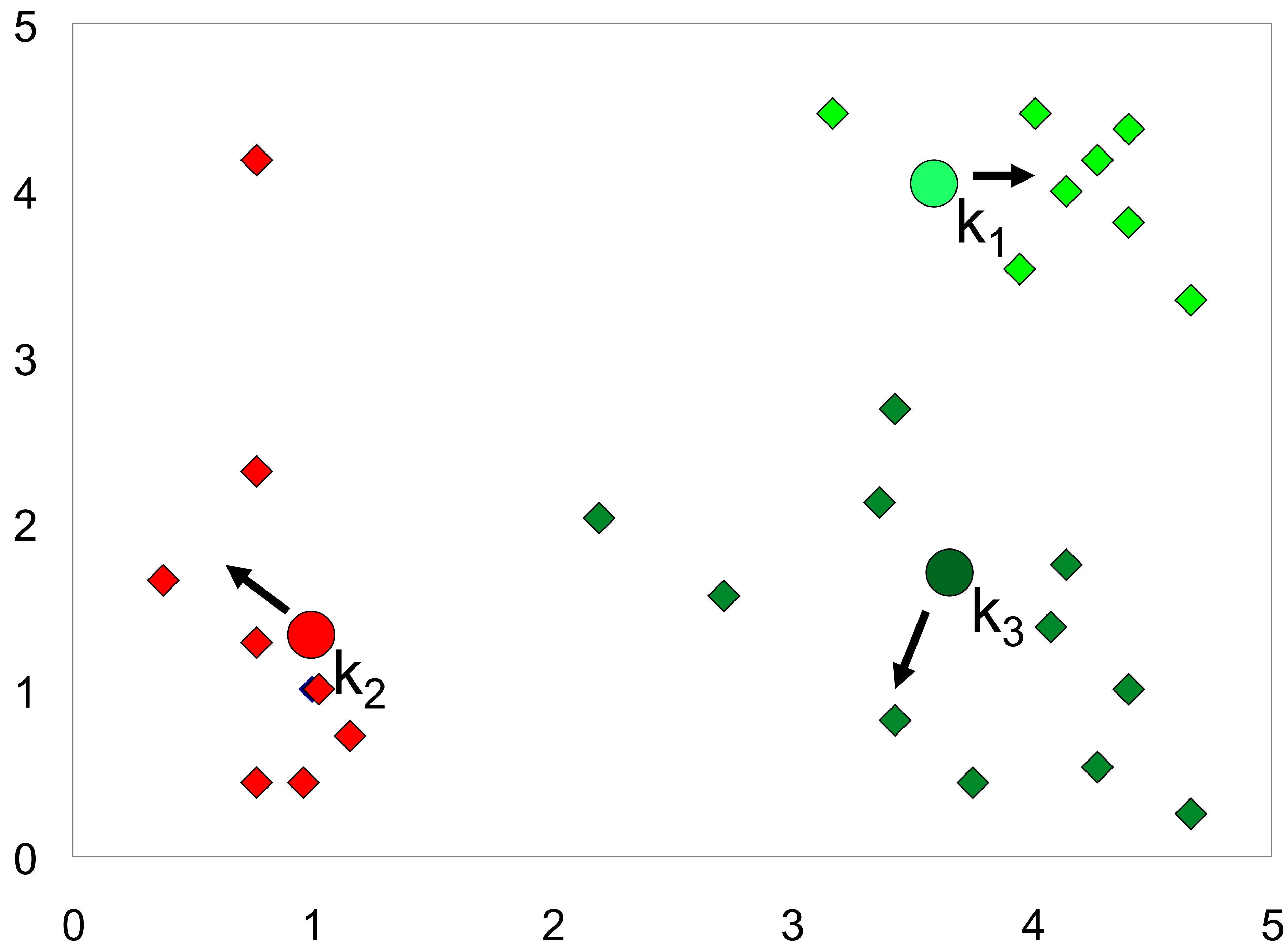
K-means Clustering: Step 2



K-means Clustering: Step 3



K-means Clustering: Step 4



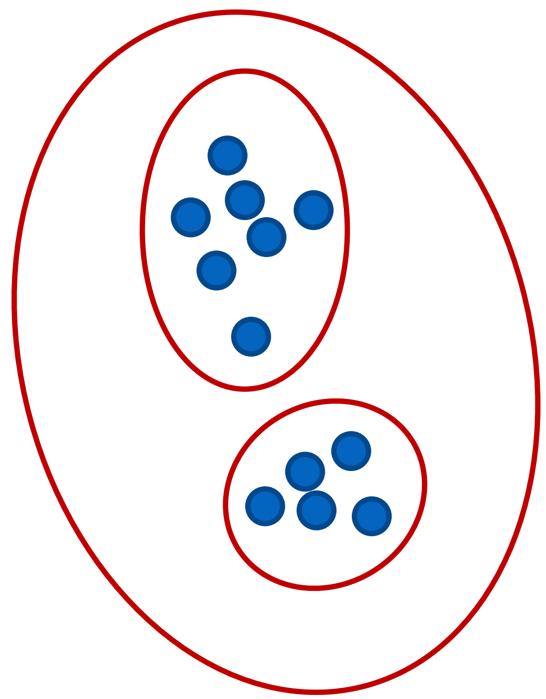
Comments on K-Means

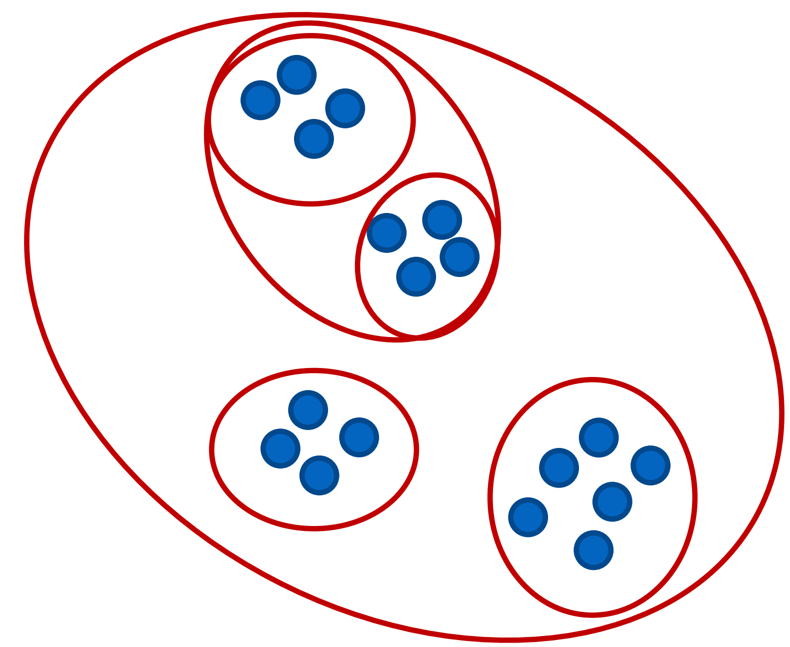
- Very simple and flexible algorithms
- Scale well with large numbers of samples and features

- Limitations:
  - Need to specify *K* in advance
  - Need to re-run to obtain clustering with different numbers of clusters
  - Applicable when mean is defined, what about categorical data?
  - Stochastic algorithm: different initialised centroids -> different clusters
  - Usually convert to local optima

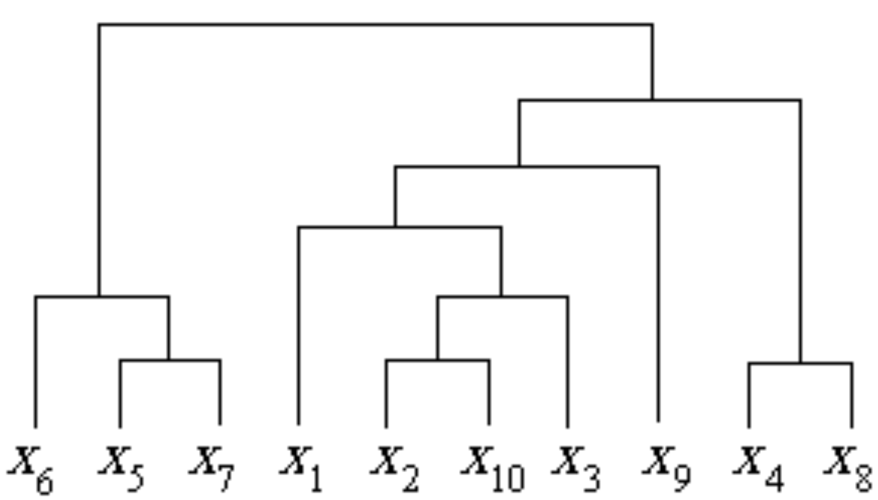
Hierarchical Clustering (1)

- How would you describe data like this?





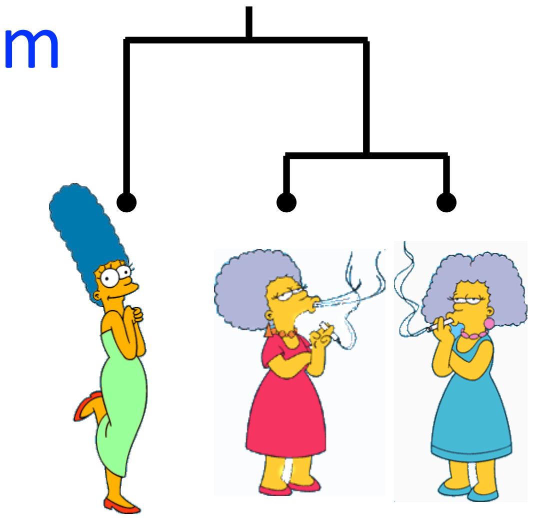
maybe it's "really" a tree?

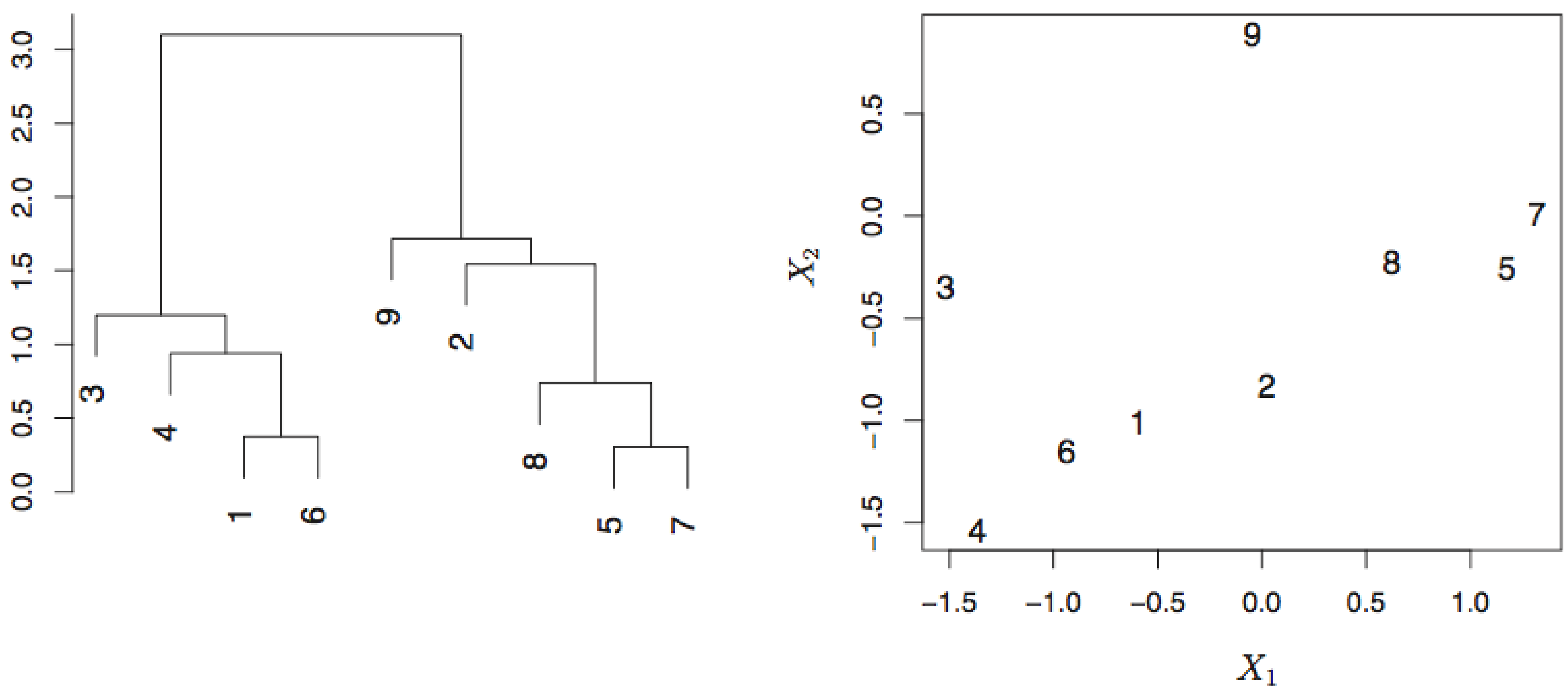


k=2, 5, 6, …

Dendrogram

- The hierarchy of clusters is represented as a tree/dendrogram
- The dissimilarity between two observations is related to the vertical height at which they first get merged into the same cluster. The greater the height, the greater the dissimilarity

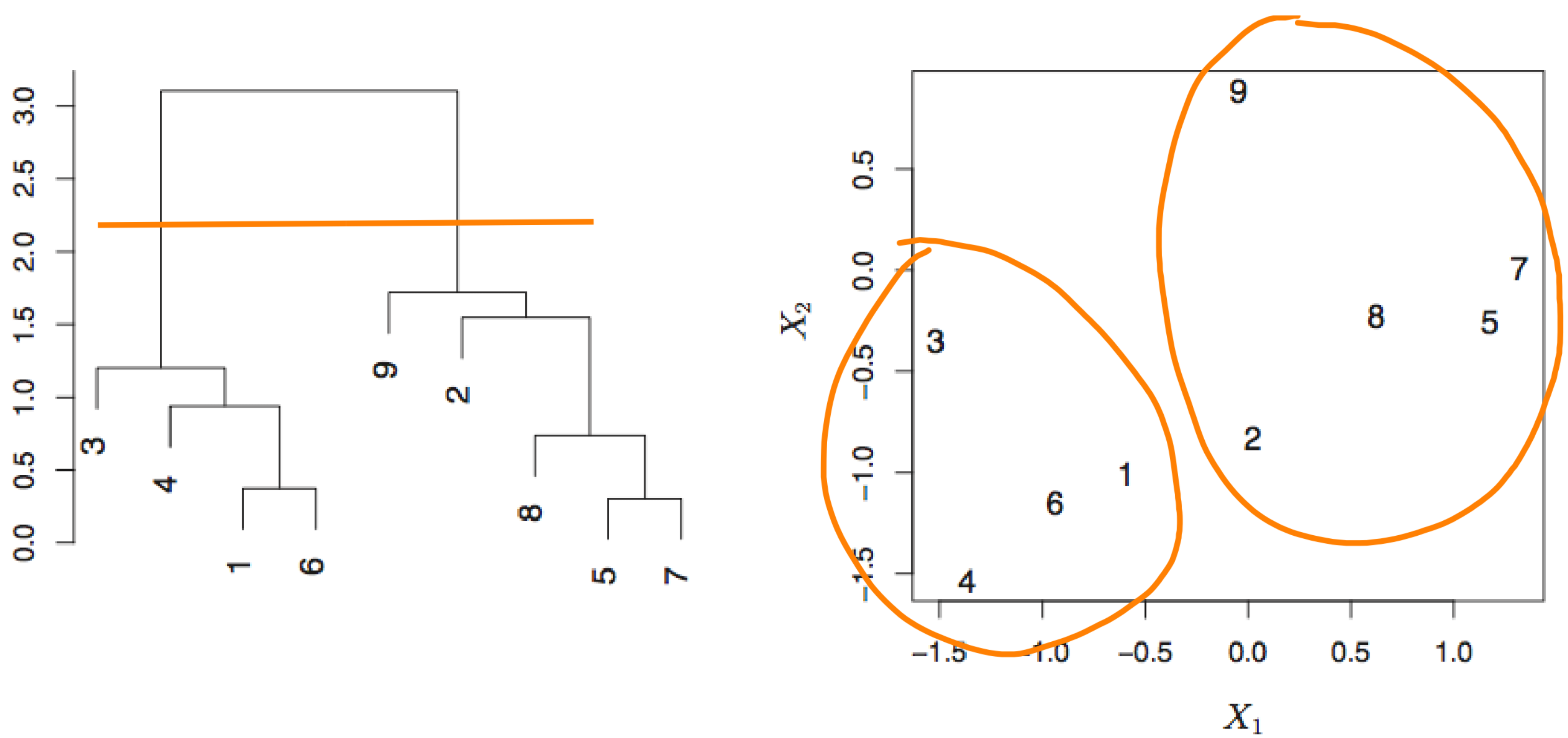




ISLR Figure 10.10: $n = 9$ and $p = 2$

Cutting a dendrogram

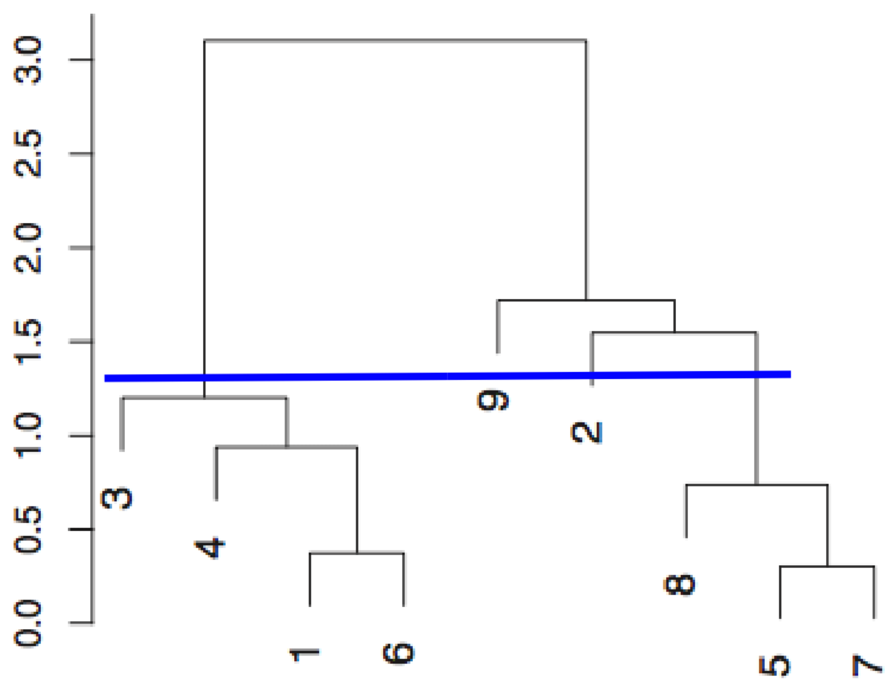
- Cutting a dendrogram horizontally gives a natural clustering. The height of the cut determines the number of clusters
- No need to re-run to get different number of clusters

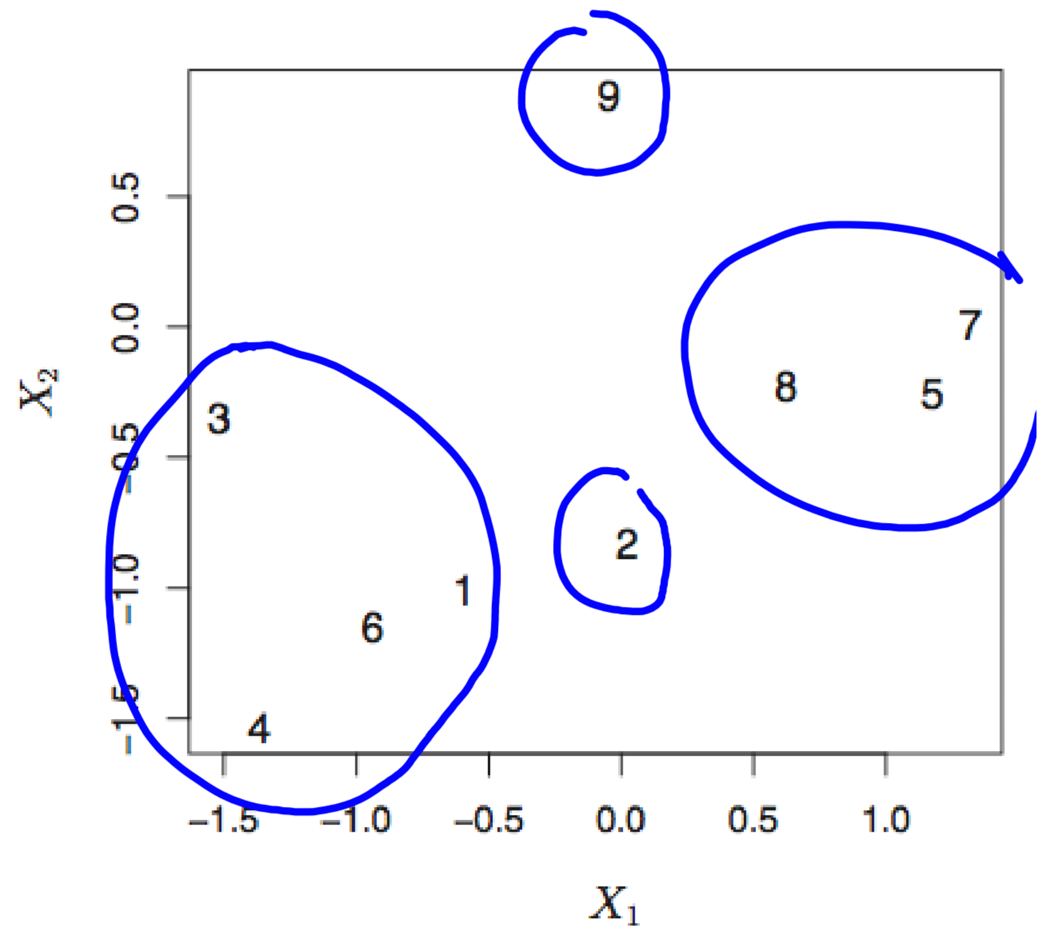


2 clusters

Cutting a dendrogram

- 4 clusters

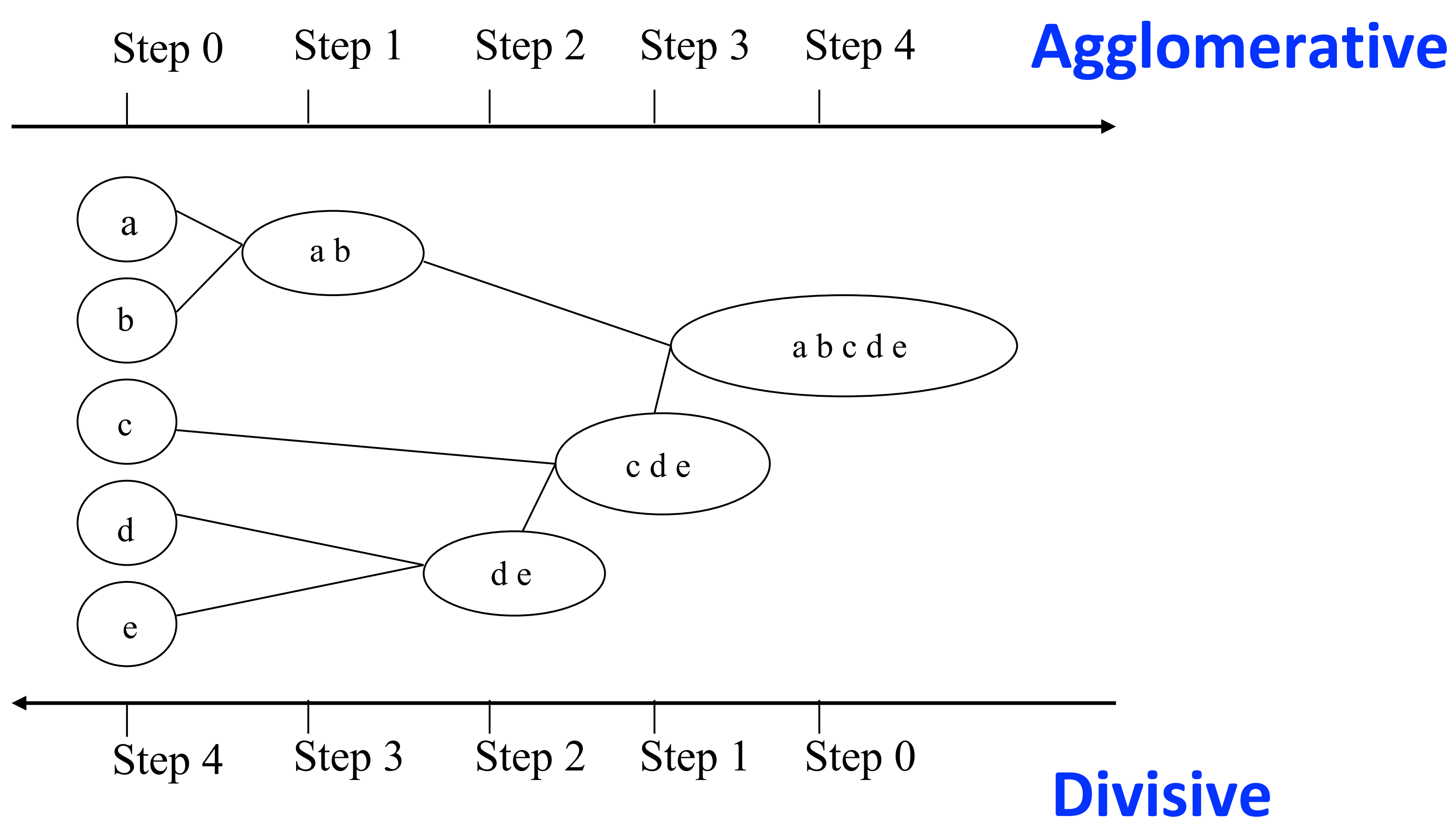




4 clusters

Hierarchical Clustering (2)

- There are two ways to do hierarchical clustering:
  - **Agglomerative** or **bottom-up** clustering where we start with the observations in *n* clusters – the leaves of the tree – and then merge clusters – forming branches – until there is only 1 cluster, the trunk of the tree
  - **Divisive** or **top-down** clustering where we start with the observations in 1 cluster and then split clusters until we reach the leaves
- We will focus on agglomerative clustering as it is generally much more efficient than divisive clustering

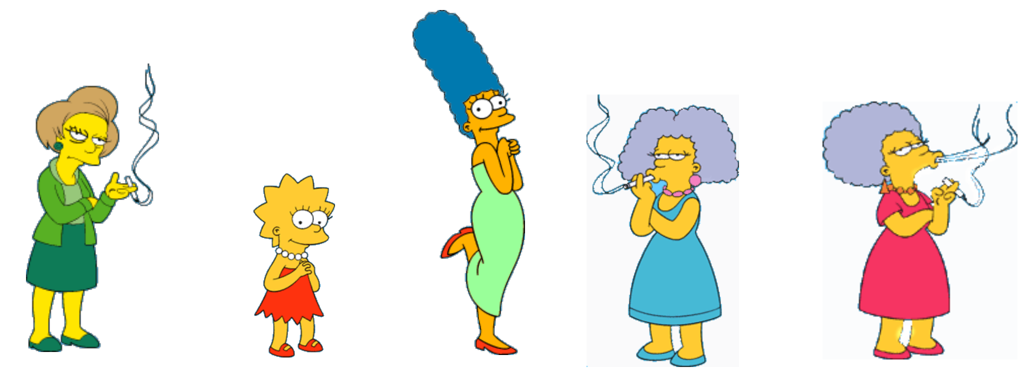


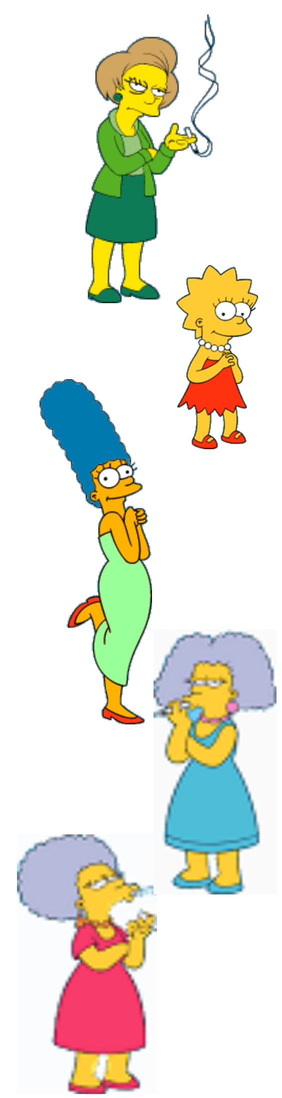
Agglomerative Clustering (1)

We begin with a distance matrix which contains the distances between every pair of objects in our database.

D( , ) = 8

D( , ) = 1



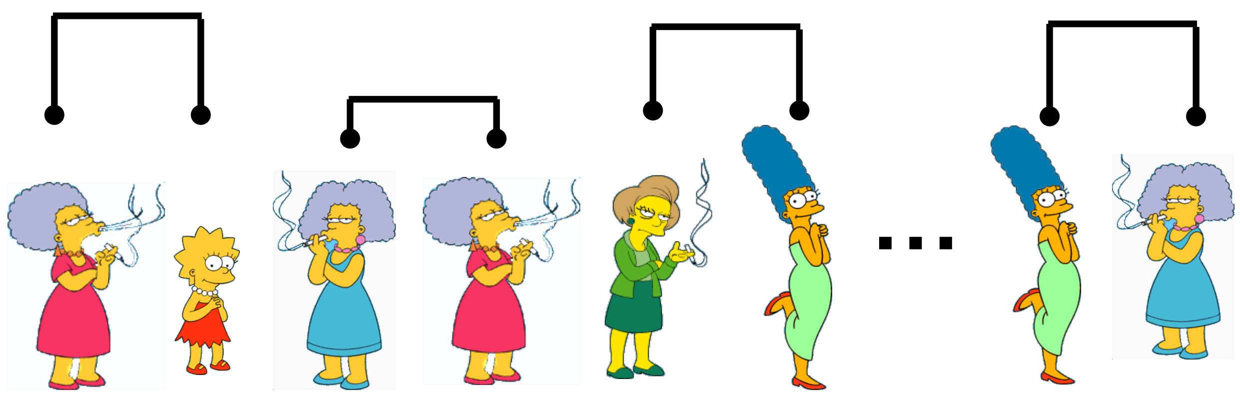


0	8	8	7	7
	0	2	4	4
		0	3	3
			0	1
				0

Agglomerative Clustering (2)

Starting with each item in its own cluster, find the best pair to merge into a new cluster. Repeat until all clusters are fused together.

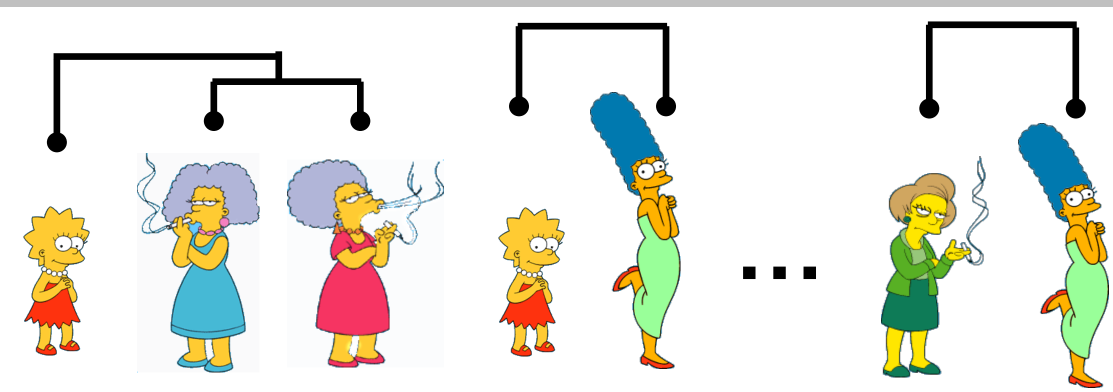
Consider all possible merges…



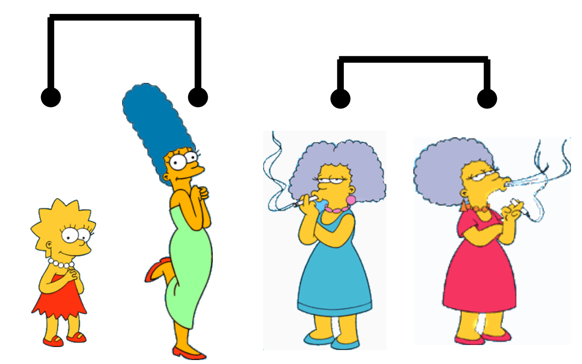
Choose the best

Agglomerative Clustering (3)

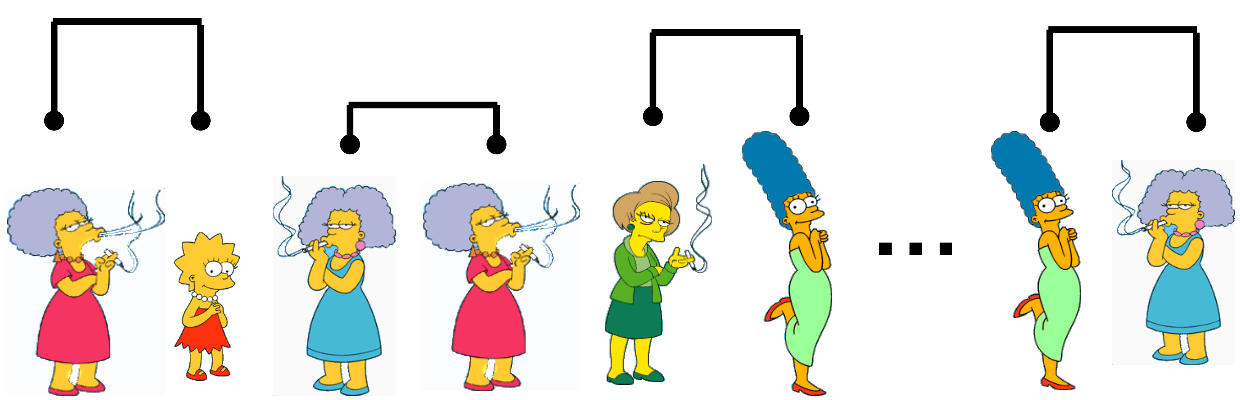
Consider all possible merges…



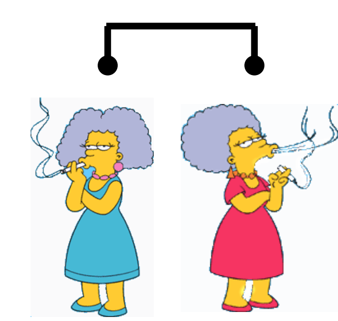
Choose the best



Consider all possible merges…

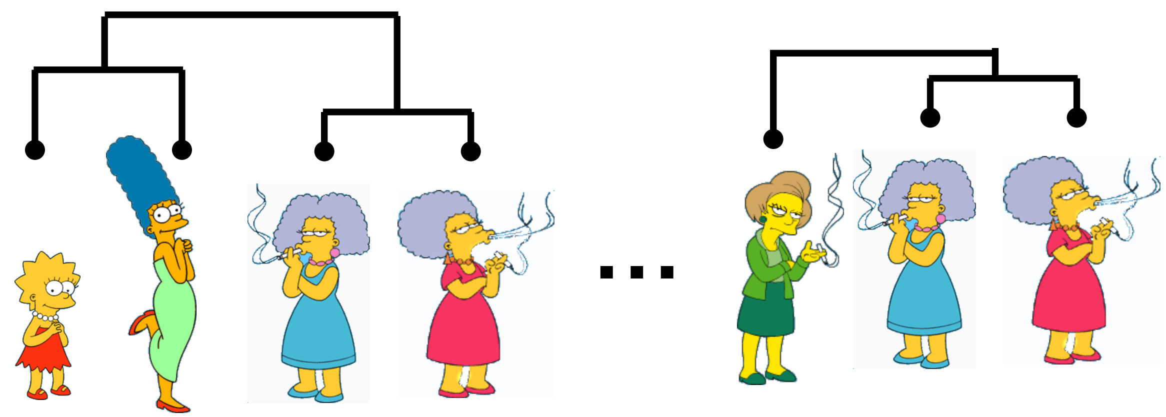


Choose the best



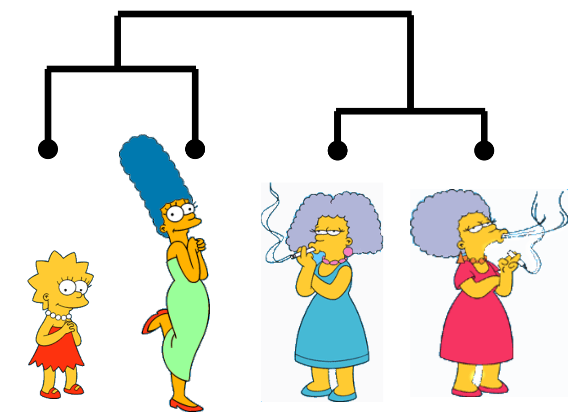
Agglomerative Clustering (4)

Consider all possible merges…

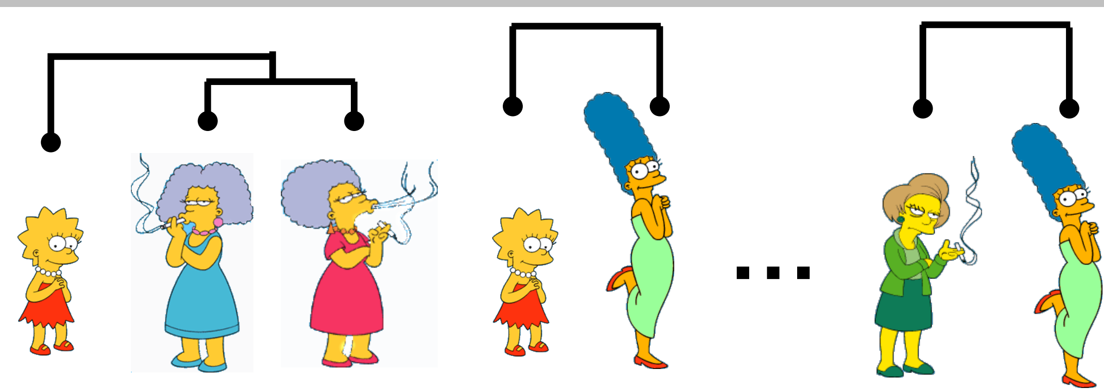


…

Choose the best

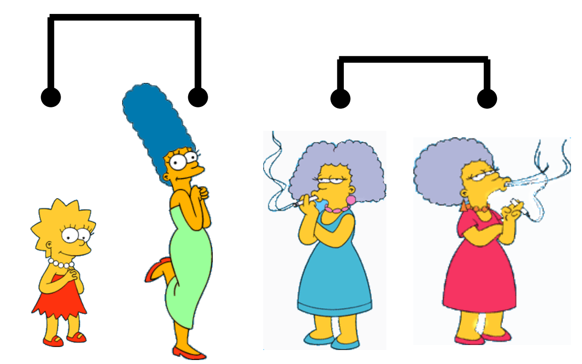


Consider all possible merges…

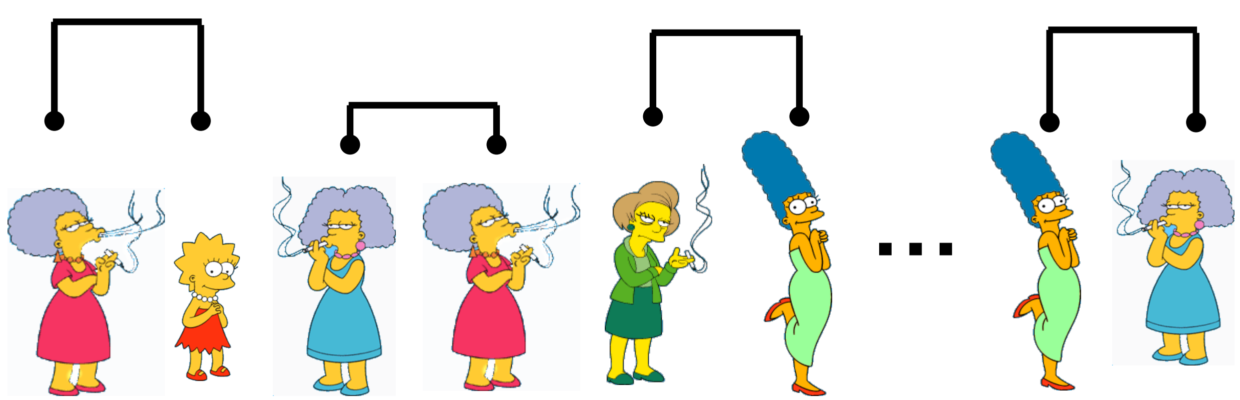


…

Choose the best

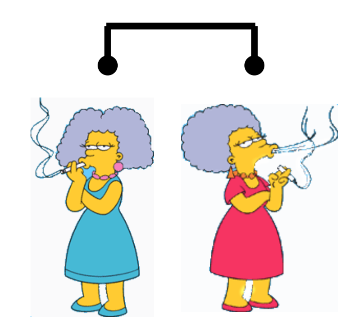


Consider all possible merges…

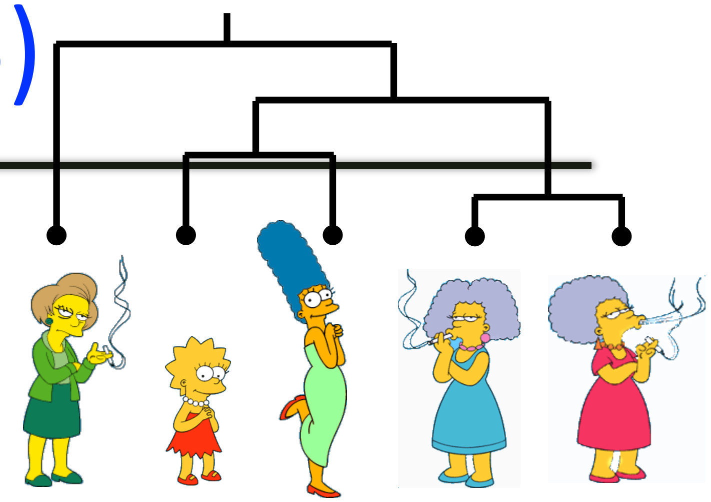


…

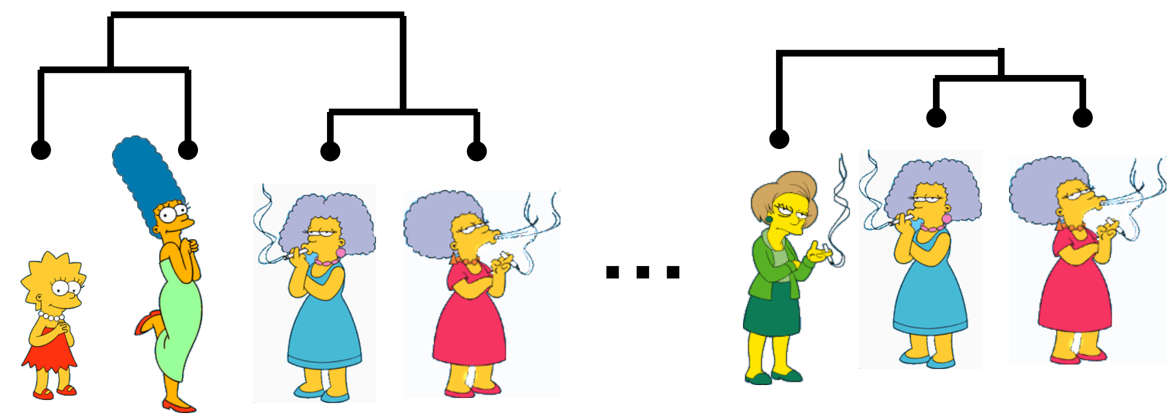
Choose the best



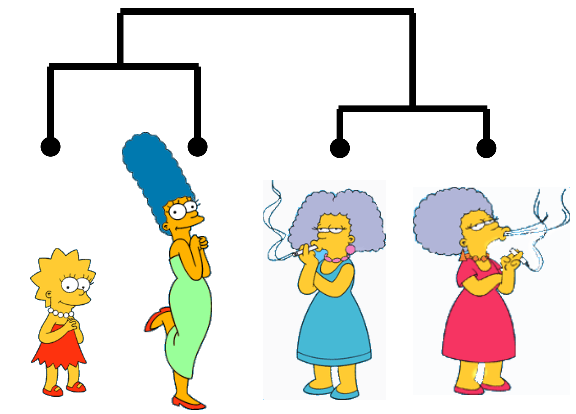
Agglomerative Clustering (5)



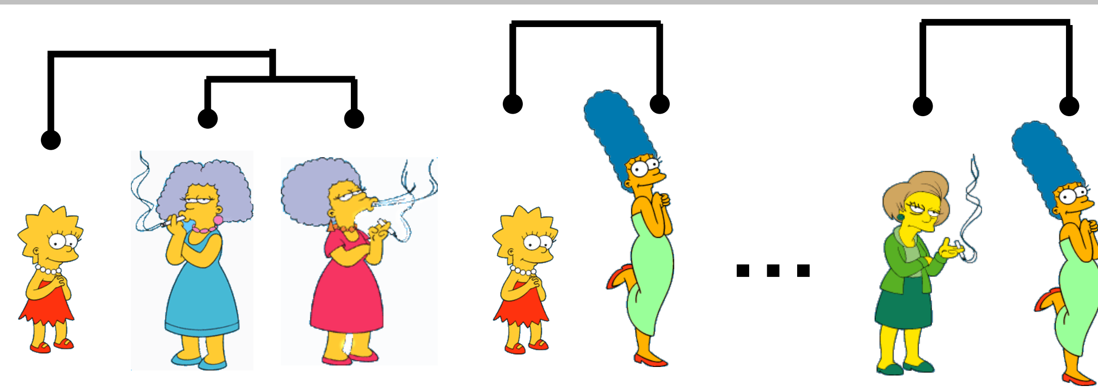
Consider all possible merges…



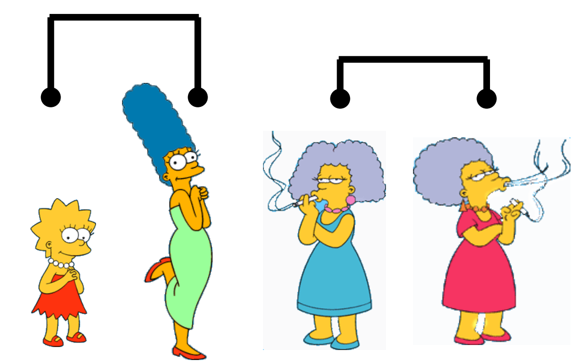
Choose the best



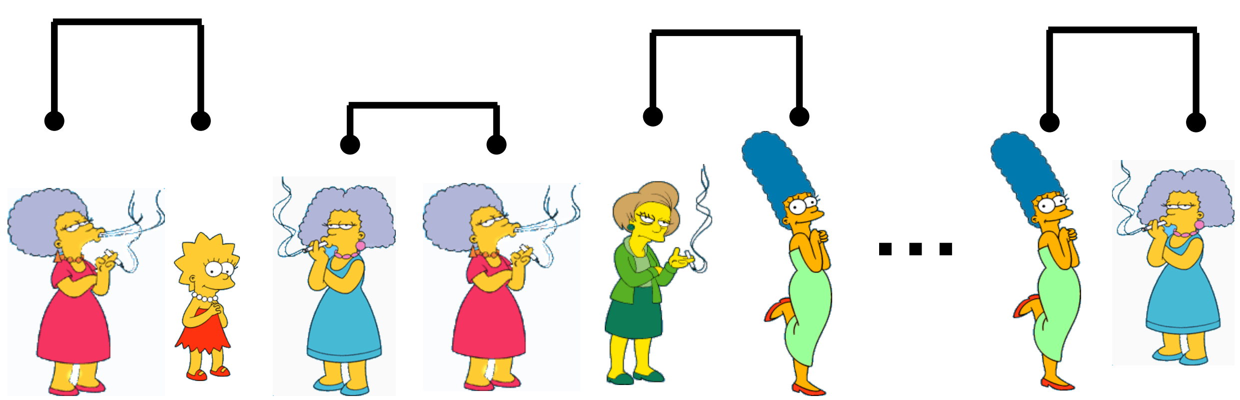
Consider all possible merges…



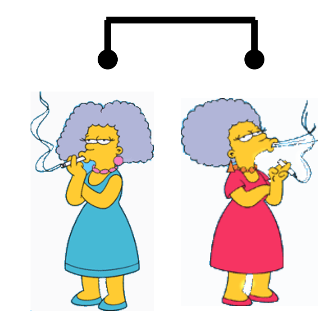
Choose the best



Consider all possible merges…



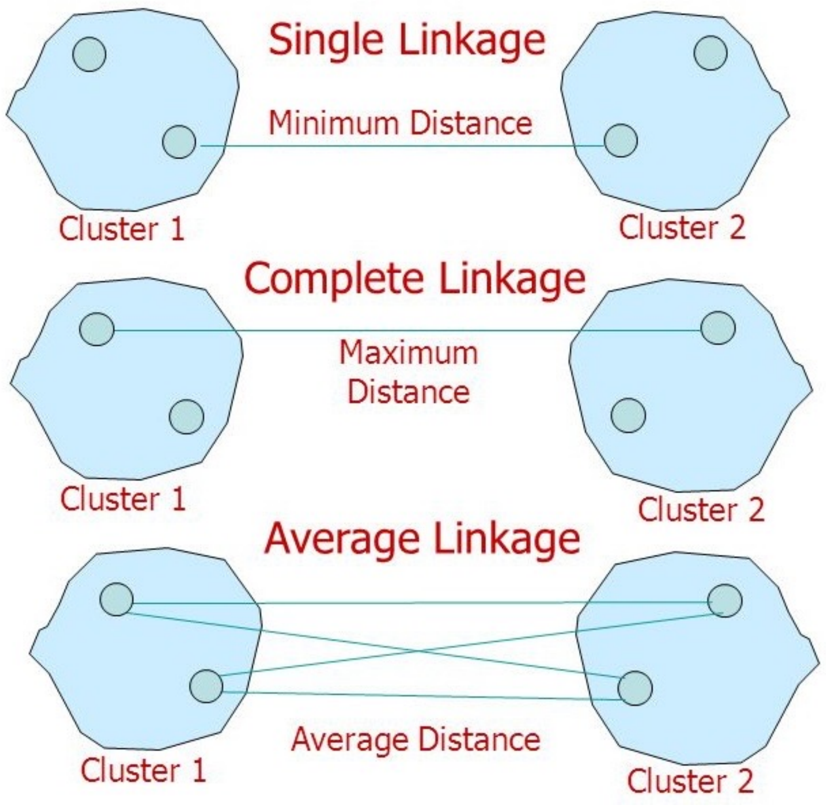
Choose the best



Linkage Methods (1)

- Agglomerative Clustering merges clusters based on distance between clusters – defined by linkage method

- **Single**: compute the minimum pairwise dissimilarity where one observation is in cluster A and the other is in cluster B.
- **Complete**: compute the maximum pairwise dissimilarity where one observation is in cluster A and the other is in cluster B
- **Average**: compute the average pairwise dissimilarity where one observation is in cluster A and the other is in cluster B



Linkage Methods (2)

Generally,

- Complete and average linkage produce more balanced dendrograms;
- Single linkage can produce trailing clusters in which single observations are merged one-at-a-time.

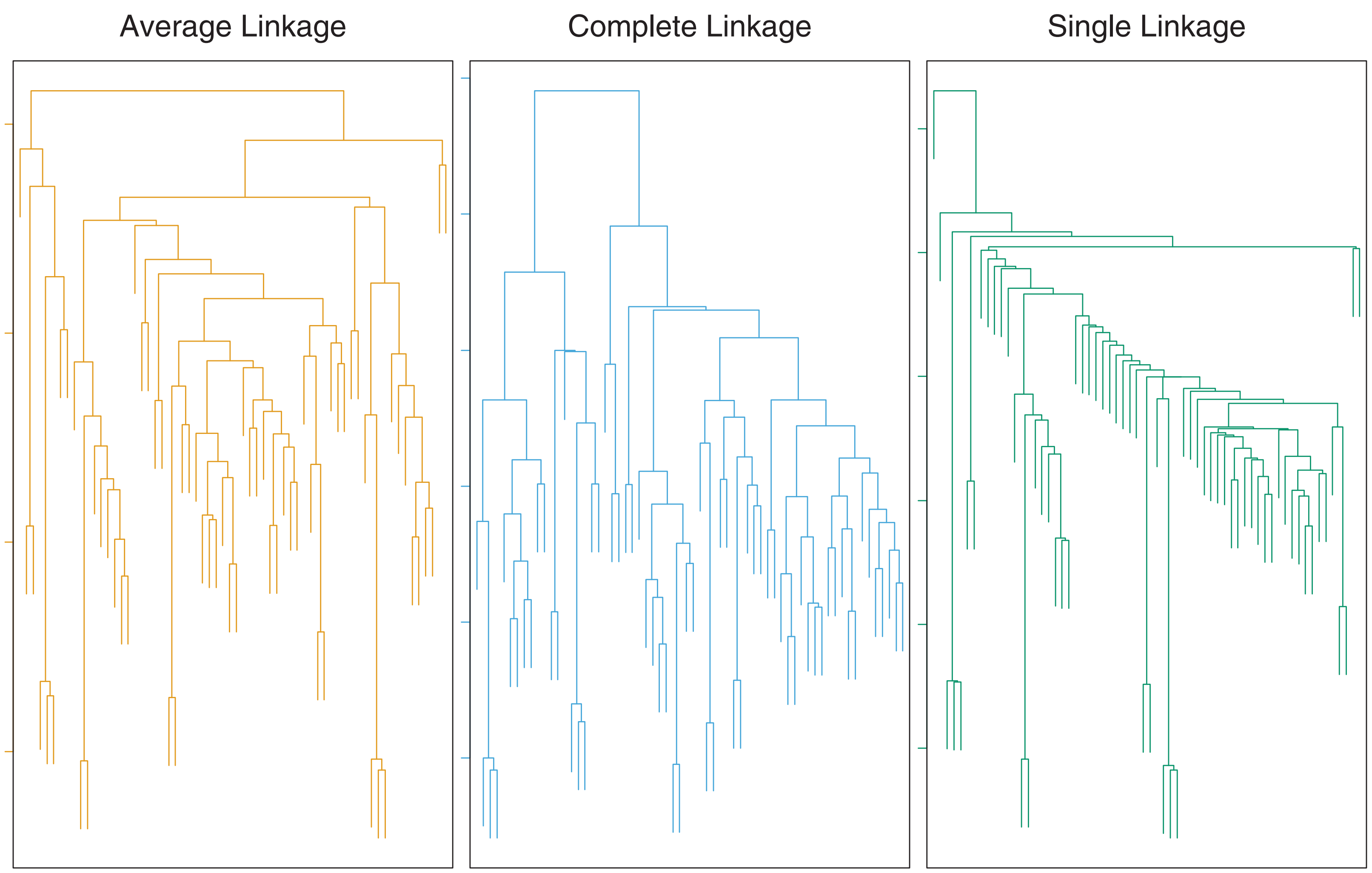


FIGURE 10.12. *Average, complete, and single linkage applied to an example data set. Average and complete linkage tend to yield more balanced clusters.*

Agglomerative clustering algorithm

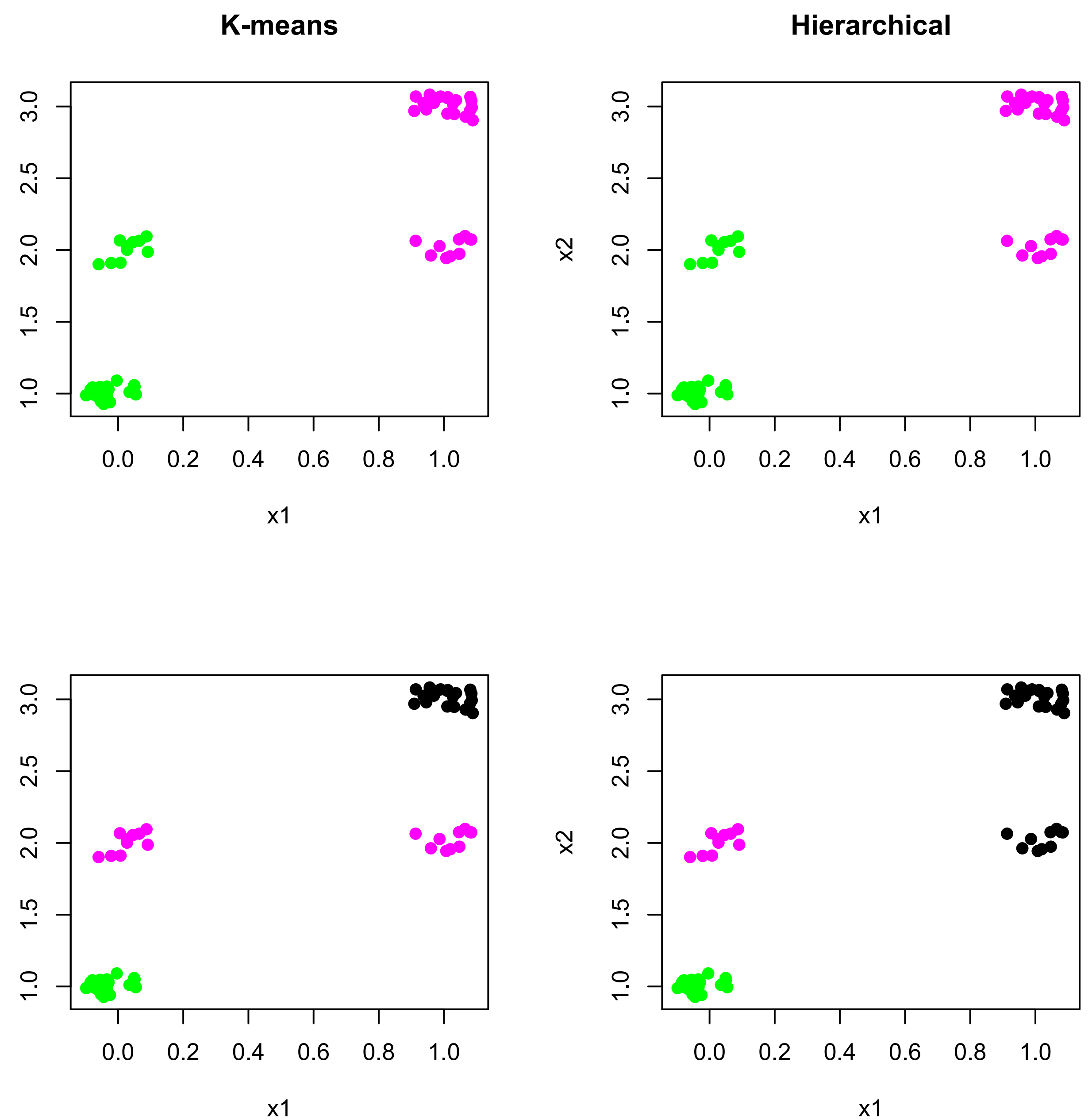
With the choice of dissimilarity measure and linkage method, **agglomerative clustering proceeds** as follows:

- Treat each observation as its own cluster, *n* clusters. Compute all pairwise dissimilarities (such as Euclidean distance) of all the $\binom{n}{2} = \frac{n(n-1)}{2}$ pairwise dissimilarities.
- For $i = n, n-1, \dots 2$
  - (a) Find the pair of clusters that are the least dissimilar and merge them
    - The dissimilarity between these two clusters indicates the height on the dendrogram where the merge is shown.
  - (b) Compute all pairwise dissimilarities between the *(i-1)* remaining clusters

Note that there is no random initialisation, so agglomerative clustering is a deterministic algorithm

Comments on Agglomerative clustering

- Do not need to specify *K* in advance
- Do no need to re-run to obtain clustering with different numbers of clusters
- Applicable to categorical data?
- Deterministic algorithm

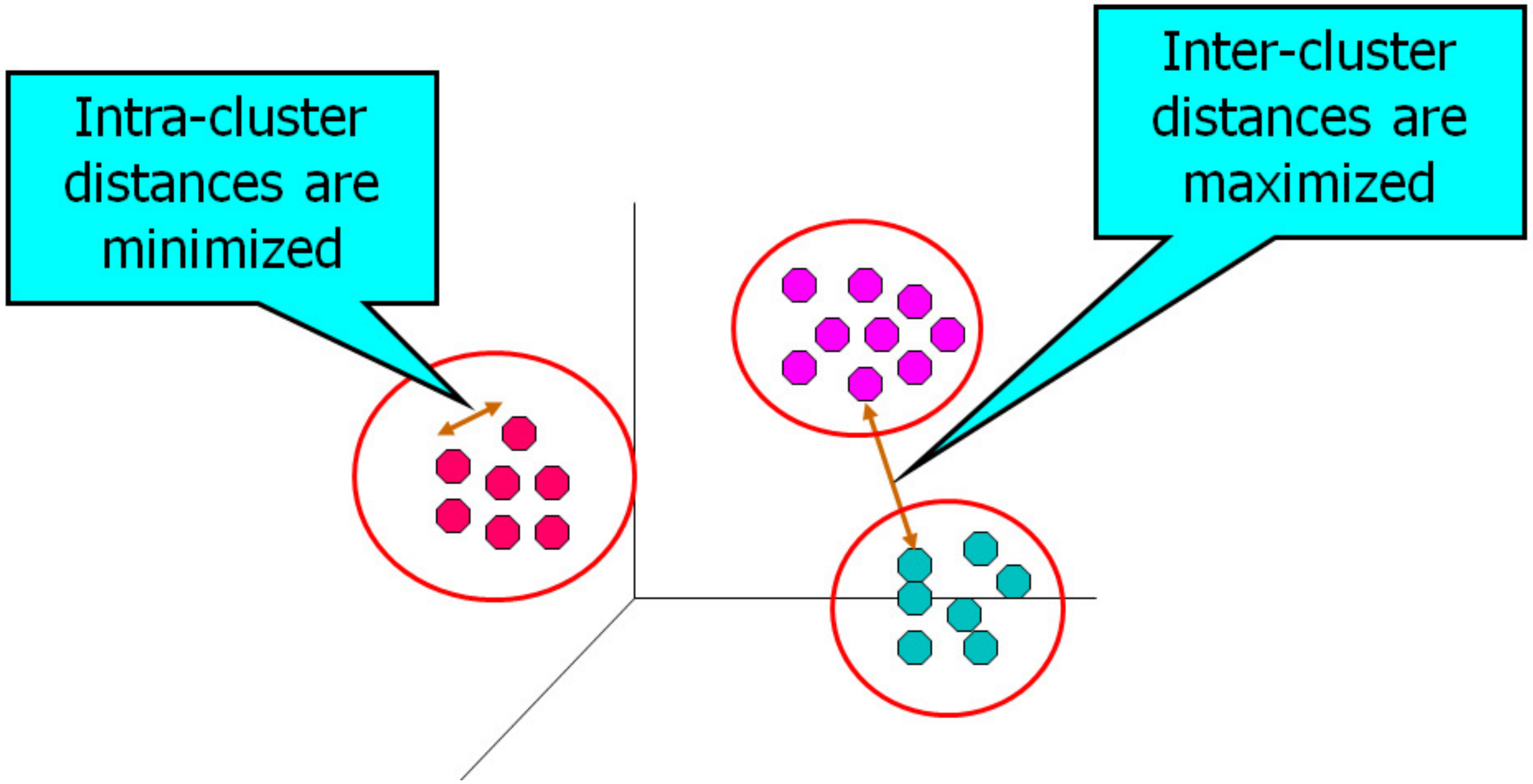


- A *potential drawback* of hierarchical clustering is that clustering obtained by cutting the dendrogram at a certain height is necessarily ***nested*** within the clustering obtained by cutting at a greater height
- Computationally expensive given a large number of samples due to pair-wise distances

Clustering performance

- **Compactness**: how tightly-packed a cluster is.
  - Clusters should be as compact as possible, so as to ensure that only the most related/similar instances have been grouped together.
  - Measured by intra-cluster distance - minimised

- **Separability**: how well neighbouring clusters are separated in the feature space.
  - Measured by inter-cluster distance - maximised



Silhouette Score

$$Silhouette(i) = \frac{b(i) - a(i)}{max\{a(i), b(i)\}}$$

- *a(i)* is the *average* distance between instance *i* and all other instances in its cluster;
- *b(i)* is the *minimum* average distance between instance *i* and the instances in each other cluster.

- Measures how well a given instance is matched to its cluster
- The average silhouette computed across all instances in a partition gives a measure of how good the partition is
- Implicitly balances both the intra- and inter-cluster metrics.

- 1 indicates an instance is perfectly clustered
- −1 indicates it should be in a neighbouring cluster;
- 0 indicates it is on the border of two clusters

Other metrics

- Davies-Bouldin index
- Dunn index
- Calinski-Harabasz Index
- ....

- https://scikit-learn.org/stable/modules/clustering.html#clustering-performance-evaluation