Machine Learning - Clustering

Week 10 – Part 1 – Introduction to Clustering

CS 457 - L1 Data Science

Zeehasham Rasheed

Chapter Objectives

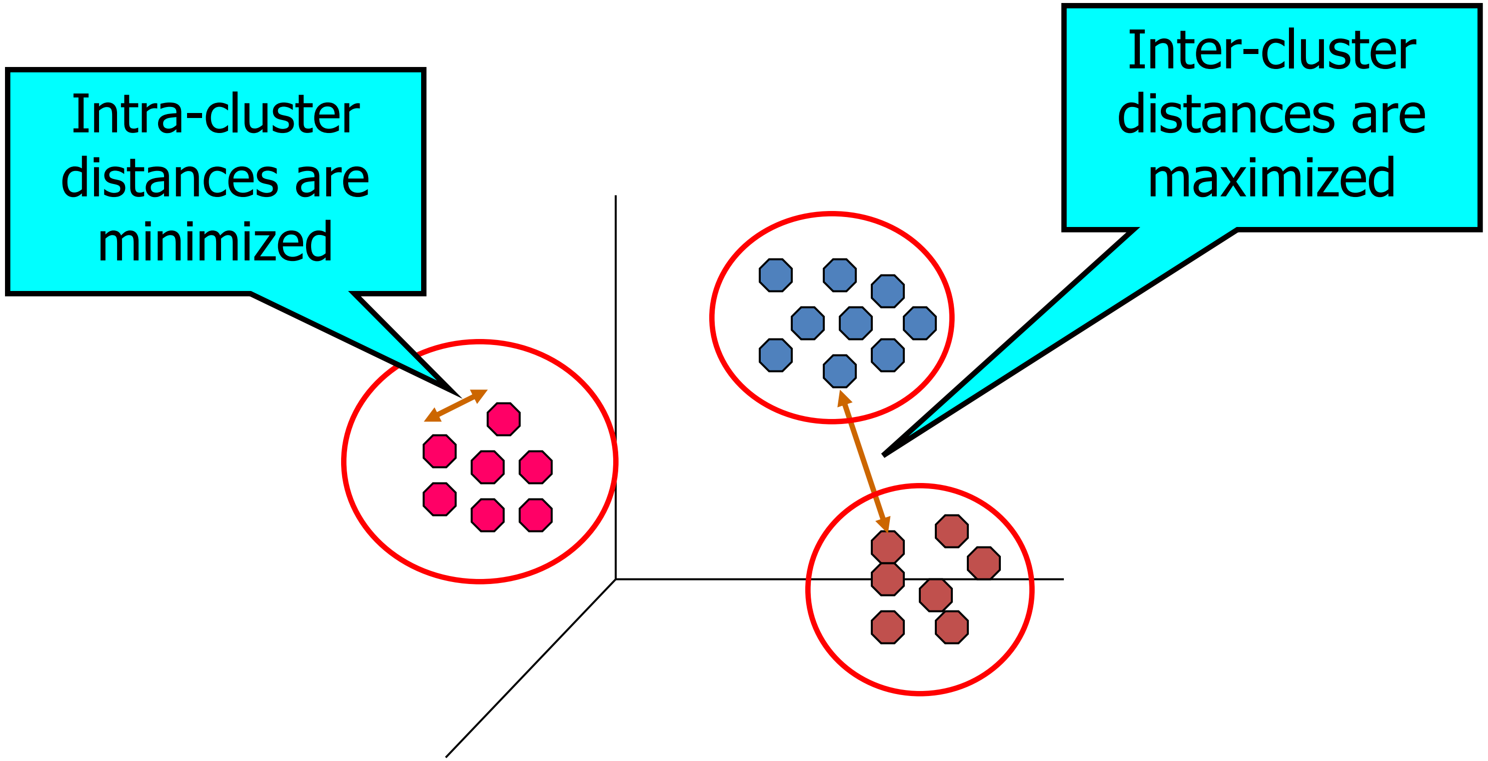
- Define and describe the Clustering process.
- Define and describe Clustering techniques.

Understanding Data Clustering

- One of the first steps a data analyst will perform when analyzing new data is to **decompose a large data set into smaller groups of related data elements** called clusters. The concept of clustering is that "an item in one cluster more closely resembles to the items in same cluster than items in another cluster".
- Clustering uses **unsupervised learning** in that it works with **unlabeled data** that has not been assigned to a category or group—the clustering process will form such groups. Business uses of clustering include:
  - Assigning customers to a market segment
  - Assigning website users into groups based upon their on-site behavior
  - Identifying healthcare risks and causes
  - Grouping sales by underlying product inventories
  - Clustering results for a search engine to better match user requests
  - Clustering housing and census data
  - And more…

What is Cluster Analysis?

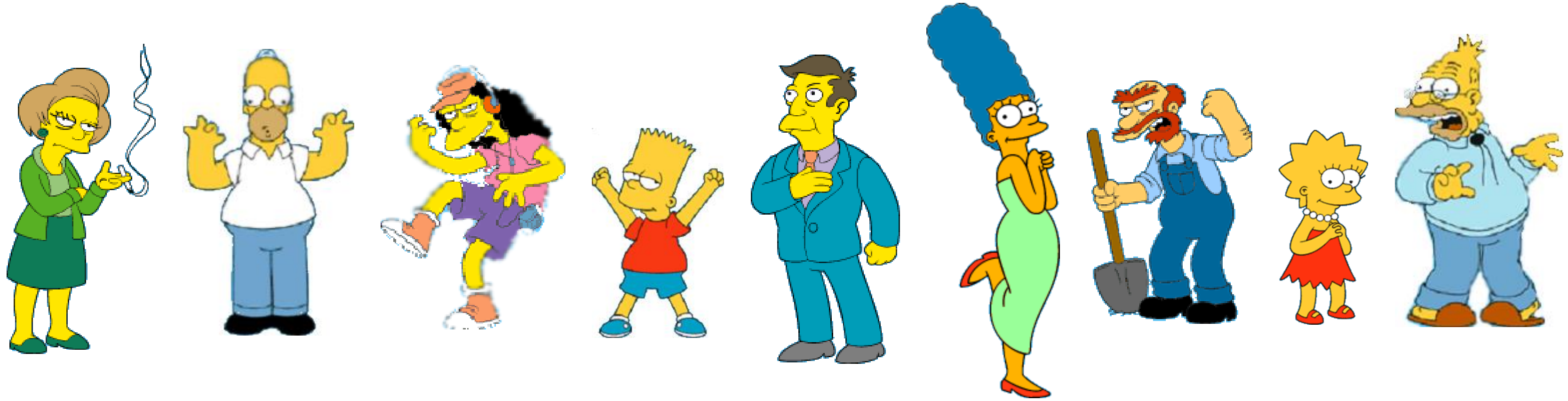
- Finding groups of objects such that the objects in a group will be similar (or related) to one another and
- Different from (or unrelated to) the objects in other groups



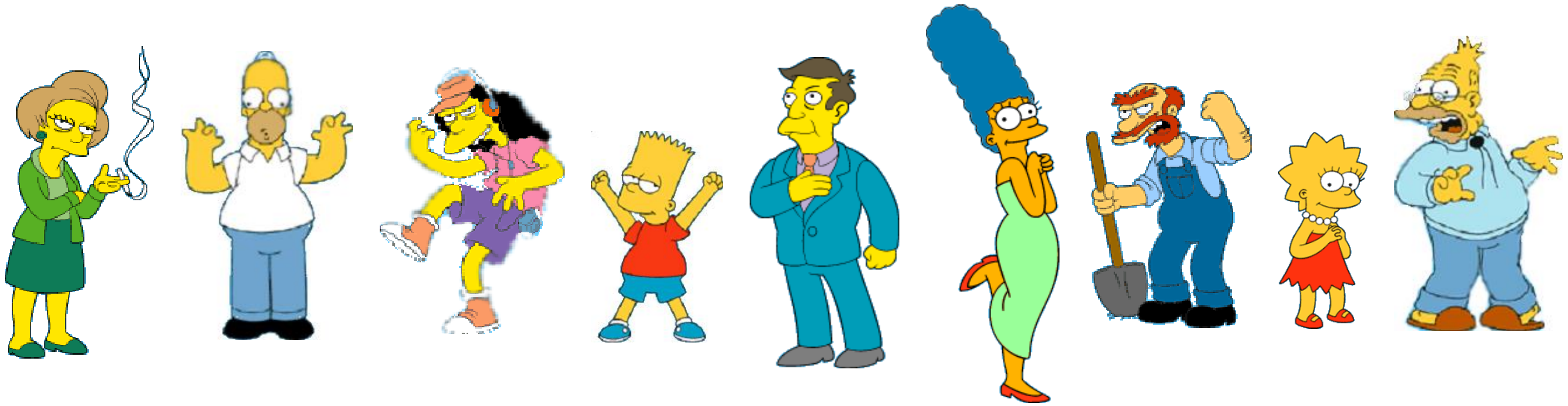
What is not Cluster Analysis?

- Supervised classification
  - Have class label information
- Simple Segmentation
  - Dividing students into different registration groups alphabetically, by last name
- Results of a query
  - Groupings are a result of an external specification

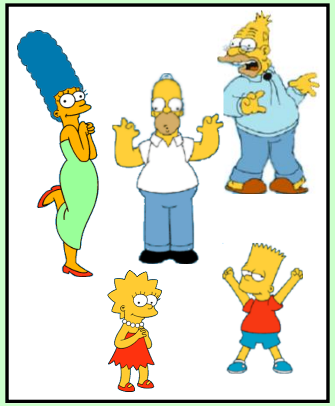
What is a natural grouping among these objects?



What is a natural grouping among these objects?

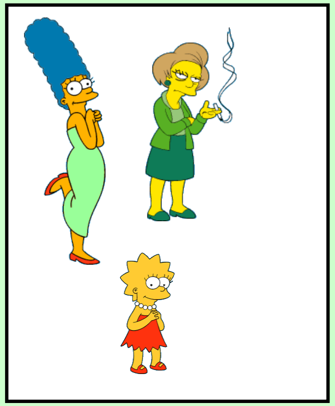


Clustering is subjective



Simpson's Family

School Employees



Females

Males

What is Similarity?

- The quality or state of being similar; likeness; resemblance; as, a similarity of features (**Webster's Dictionary)**

- Similarity is hard to define, but…
- “*We know it when we see it*”

- The real meaning of similarity is a philosophical question.

- Machine Learning takes a more pragmatic approach.

Similarity Measures (Distance)

For the moment assume that we can measure the similarity between any two objects.

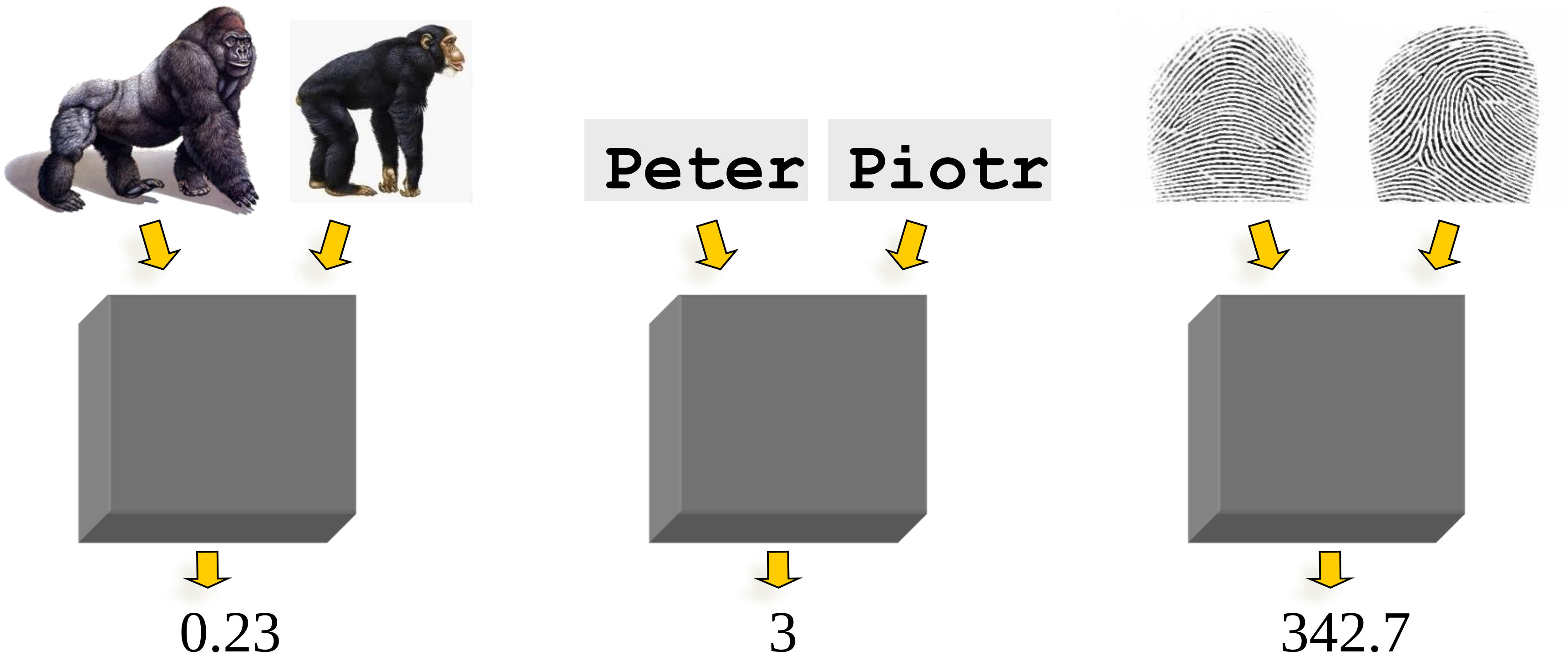
- One intuitive example is to measure the distance between two cities and call it the **similarity**.
  For example, we have *D*(LA,San Diego) = 110, and *D*(LA,New York) = 3,000.

This would allow use to make (subjectively correct) statements like

"LA is more similar to San Diego than it is to New York".

Defining Distance Measures

- **Definition**: 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 $D(O_1,O_2)$



Understanding Euclidian Distances

- In clustering algorithm, distances are normally defined in terms of the **Euclidian Distance** (or straight-line distance) between the points, which is calculated as shown (i.e., the Pythagorean Theorem)

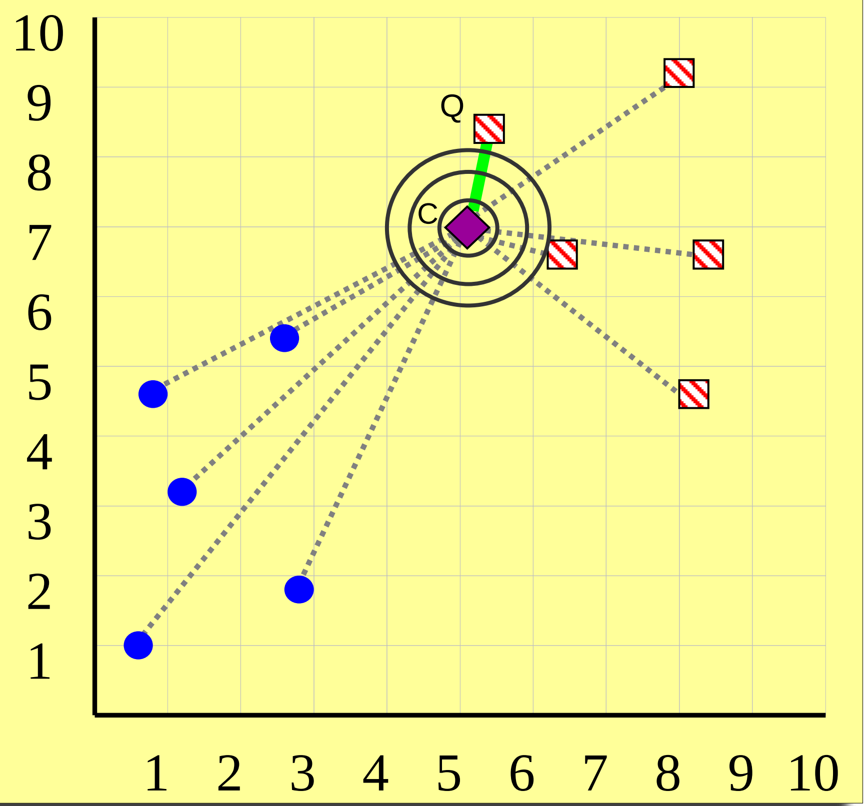
- Many of the clustering algorithms include or exclude a point within or from a cluster **based upon the point's distance from a cluster's center** (which is called the **centroid**).

$$\text{distance}(a, b) = \sqrt{(a_1 - b_1)^2 + (a_2 - b_2)^2}$$

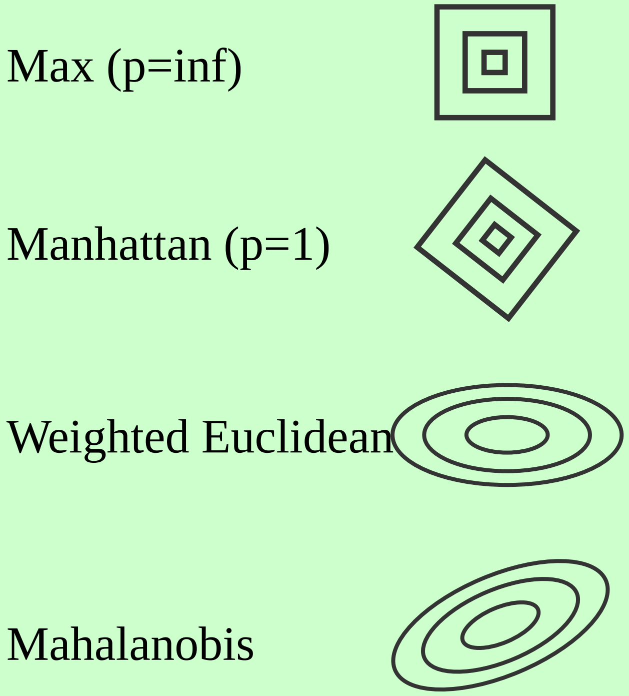
Different Distance Measures

- The most commonly used distance measure in data mining is the Euclidean Distance (and its variants)

$$D(Q,C) \equiv \sqrt{\sum_{i=1}^{n}(q_i - c_i)^2}$$



$$D(Q,C) \equiv \sqrt[p]{\sum_{i=1}^{n}(q_i - c_i)^p}$$



End of Part 1

Machine Learning - Clustering

Week 10 – Part 2 – Hierarchical Clustering

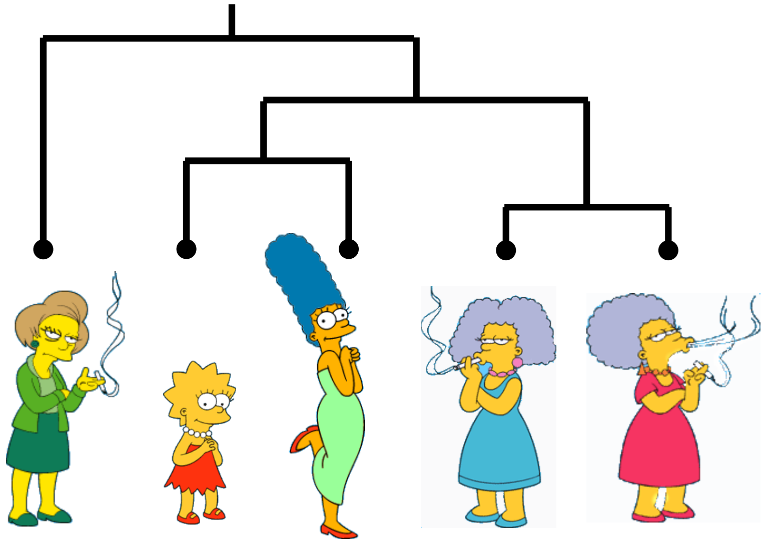
CS 457 - L1 Data Science

Zeehasham Rasheed

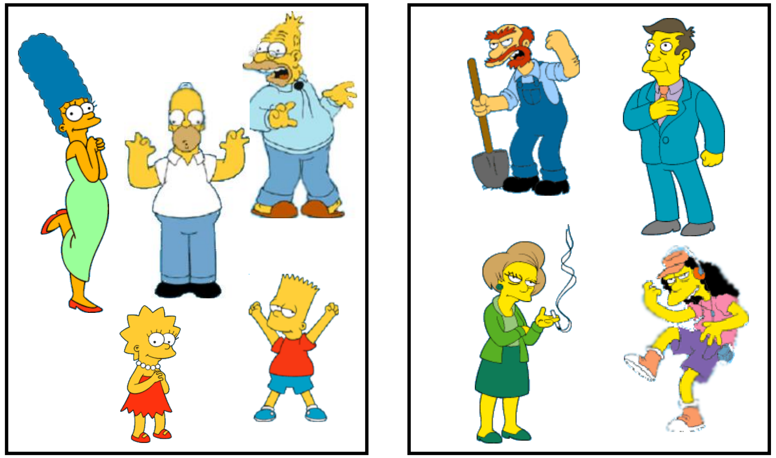
Two Types of Clustering

- **Hierarchical algorithms:** Create a hierarchical decomposition of the set of objects using some criterion (distance)
- **Partitional algorithms:** Construct various partitions and then evaluate them by some criterion (distance)

Hierarchical

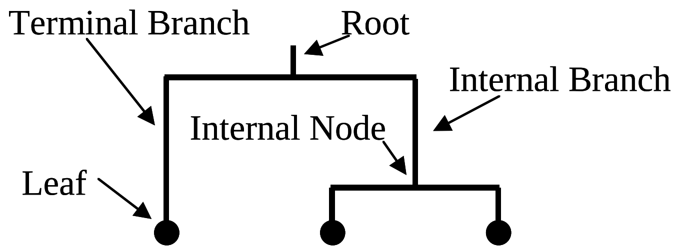


Partitional

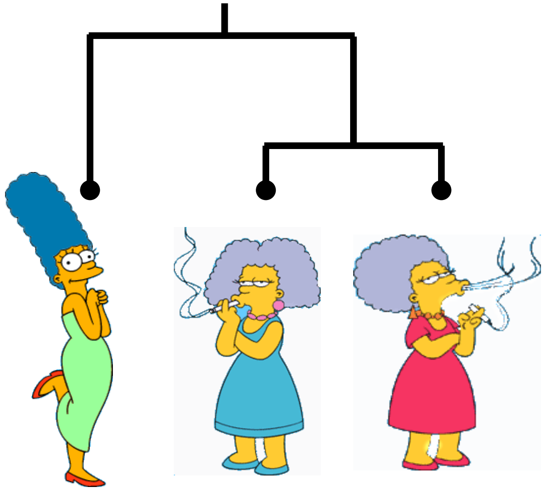


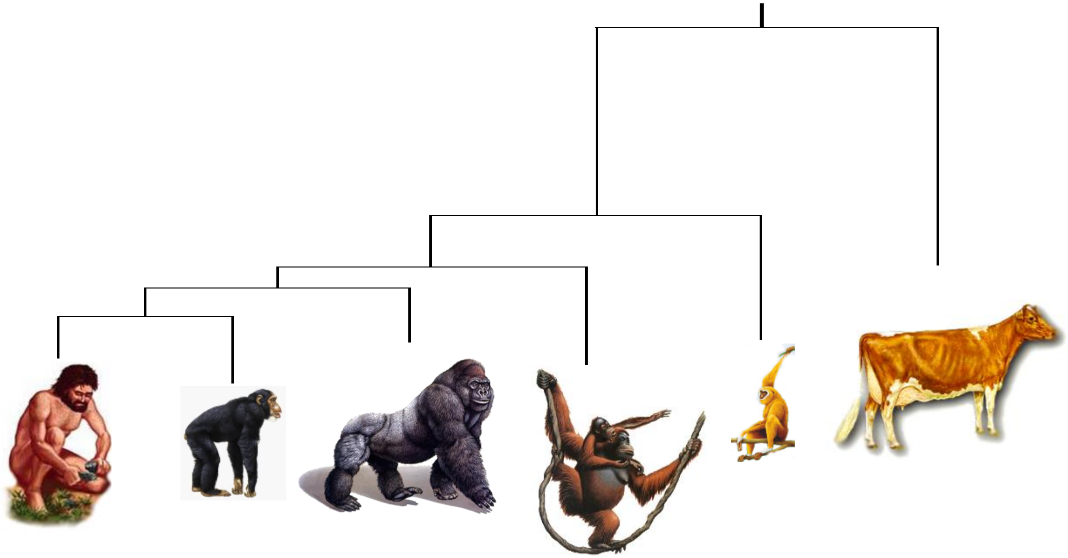
A Useful Tool for Summarizing Similarity Measurements

- In order to better appreciate and evaluate the examples given in the early part of this talk, we will now introduce the *dendrogram*.



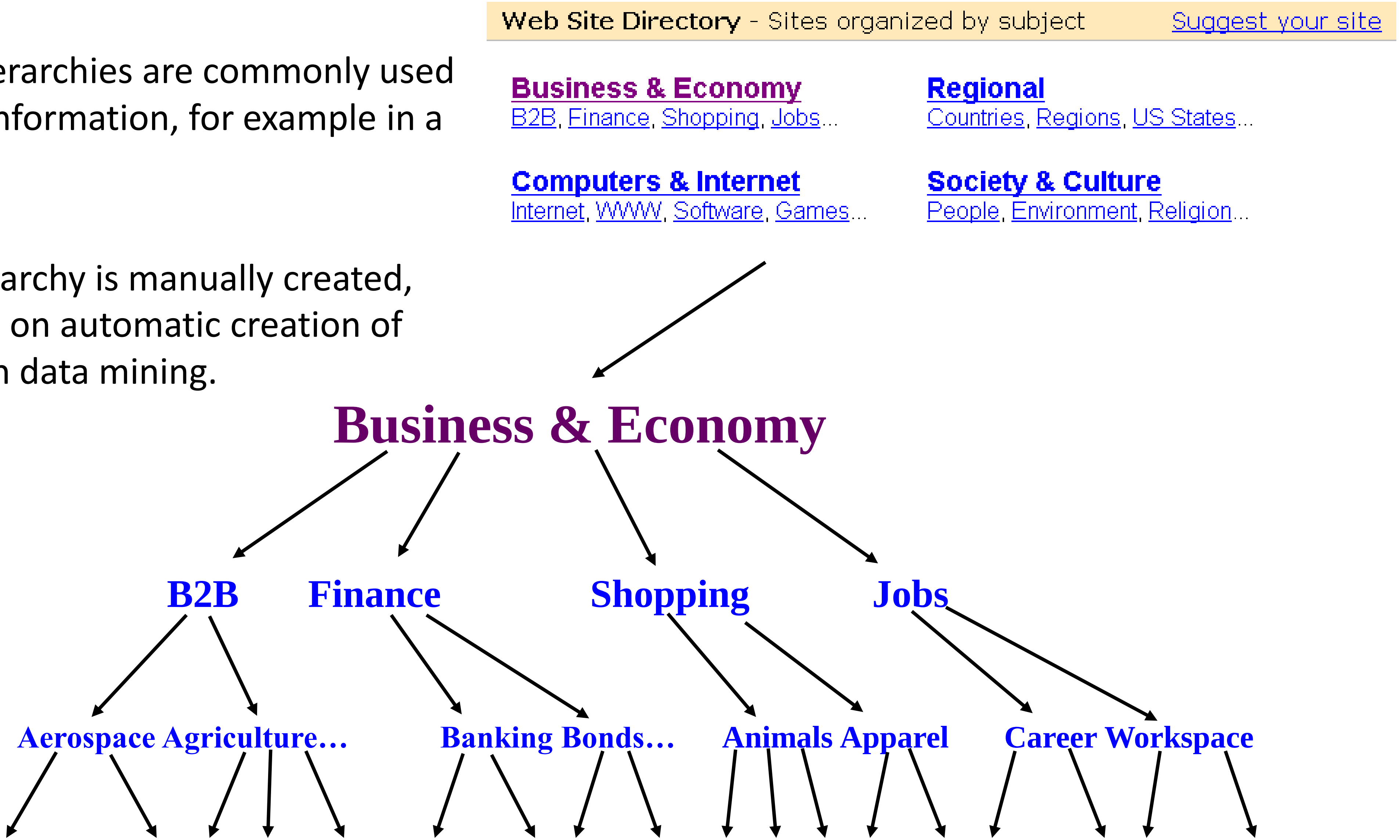
The similarity between two objects in a dendrogram is represented as the height of the lowest internal node they share.





More on Hierarchy

- Note that hierarchies are commonly used to organize information, for example in a web portal.
- Yahoo's hierarchy is manually created, we will focus on automatic creation of hierarchies in data mining.



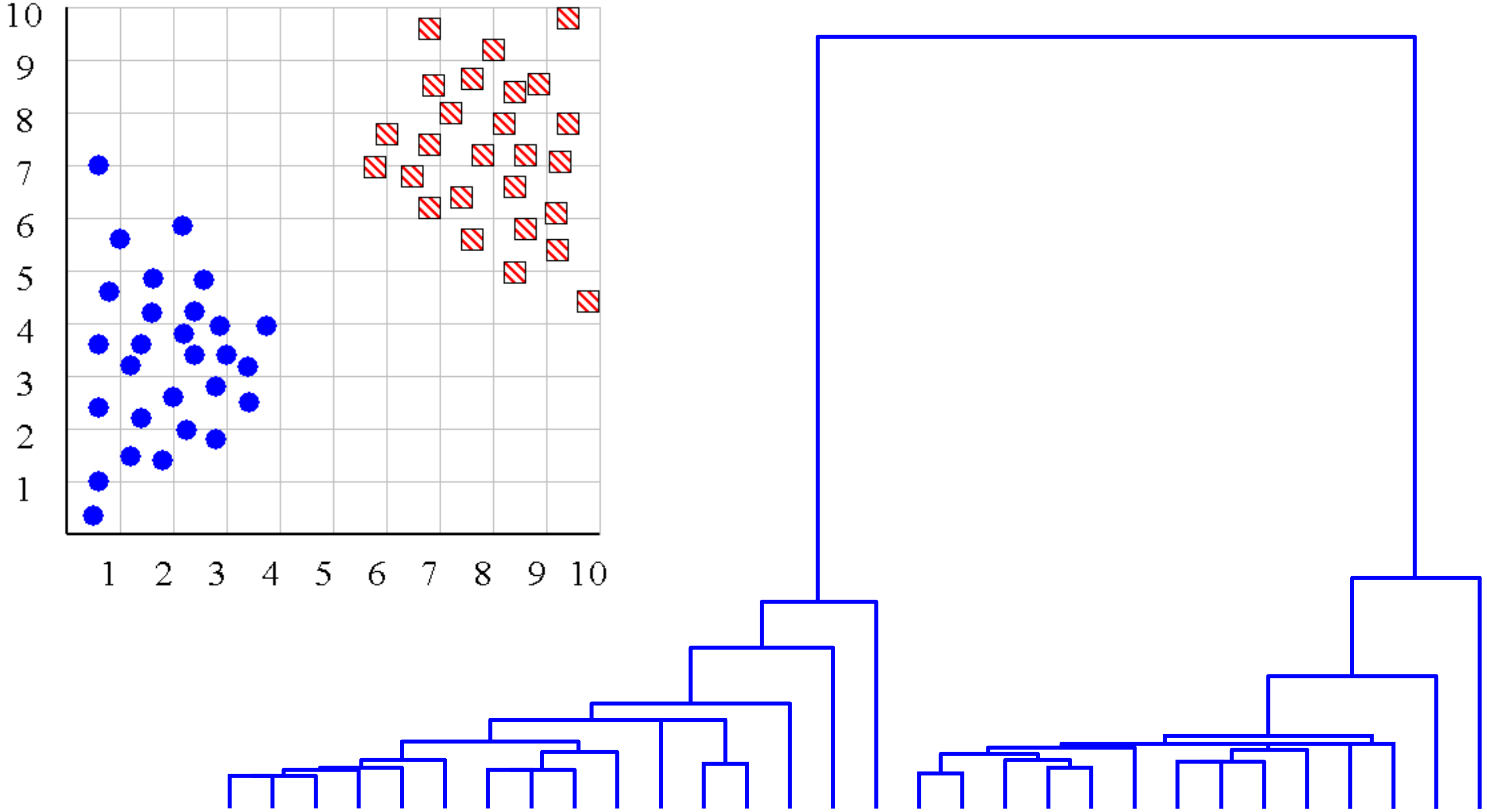
Desirable Properties of a Clustering Algorithm

- Scalability (in terms of both time and space)
- Ability to deal with different data types
- Minimal requirements for domain knowledge to determine input parameters
- Able to deal with noise and outliers
- Insensitive to order of input records
- Incorporation of user-specified constraints

- **Practical Importance**
  - **Interpretability and usability (each cluster showing unique behavior and attributes)**

Dendrogram Visualization

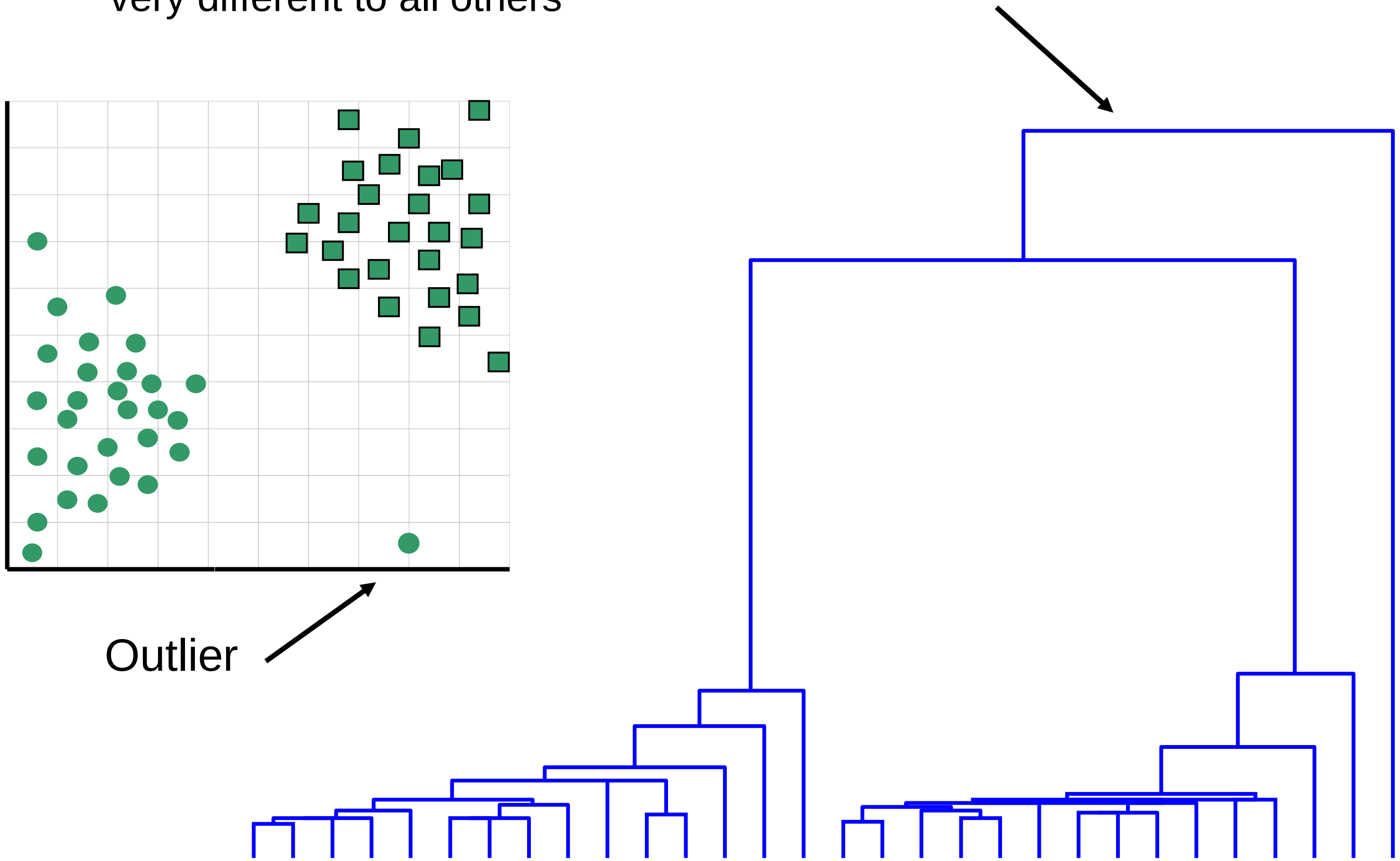
- We can look at the dendrogram to determine the “correct” number of clusters. In this case, the two highly separated subtrees are highly suggestive of two clusters. (Things are rarely this clear cut, unfortunately)



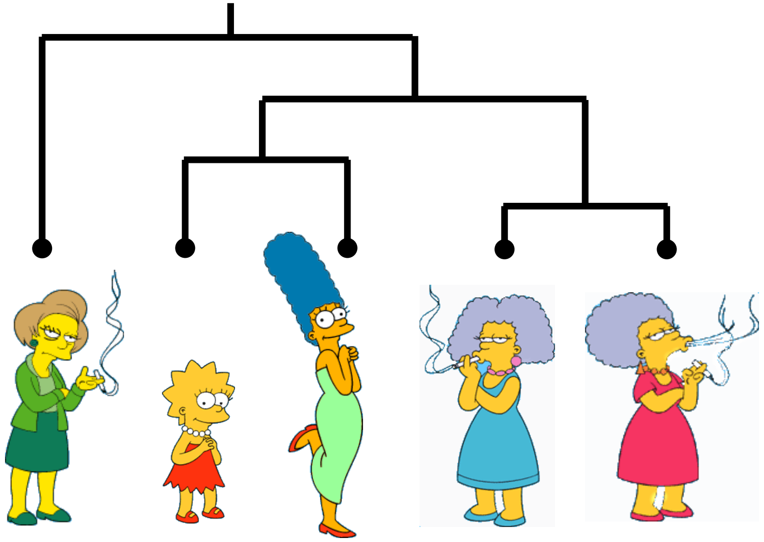
Outliers in Clustering

- One potential use of a dendrogram is to detect outliers

The single isolated branch is suggestive of a data point that is very different to all others



Hierarchical Clustering



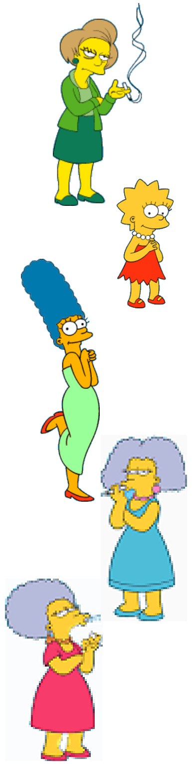
Since we cannot test all possible trees we will have to heuristic search of all possible trees. We could do this..

Bottom-Up (agglomerative): 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.

Top-Down (divisive): Starting with all the data in a single cluster, consider every possible way to divide the cluster into two. Choose the best division and recursively operate on both sides.

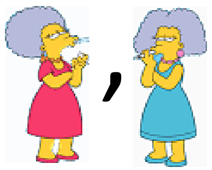
Distance Matrix

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



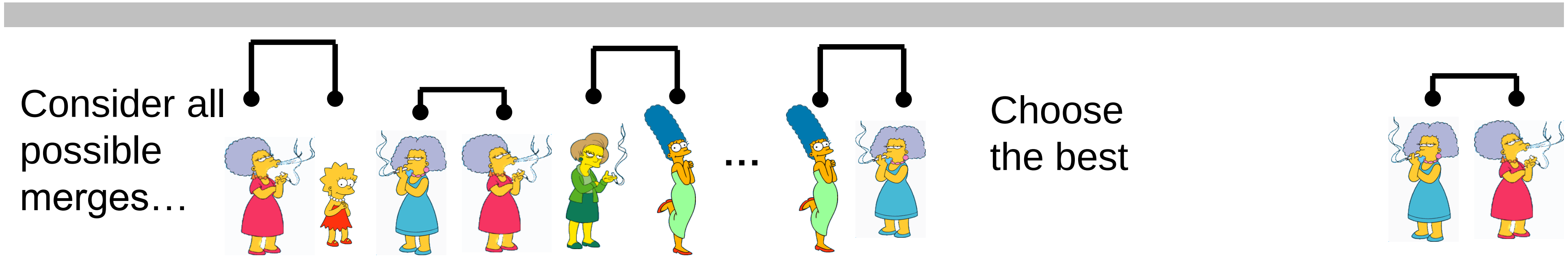
0	8	8	7	7
	0	2	4	4
		0	3	3
			0	1
				0

D()= 8

D() = 1

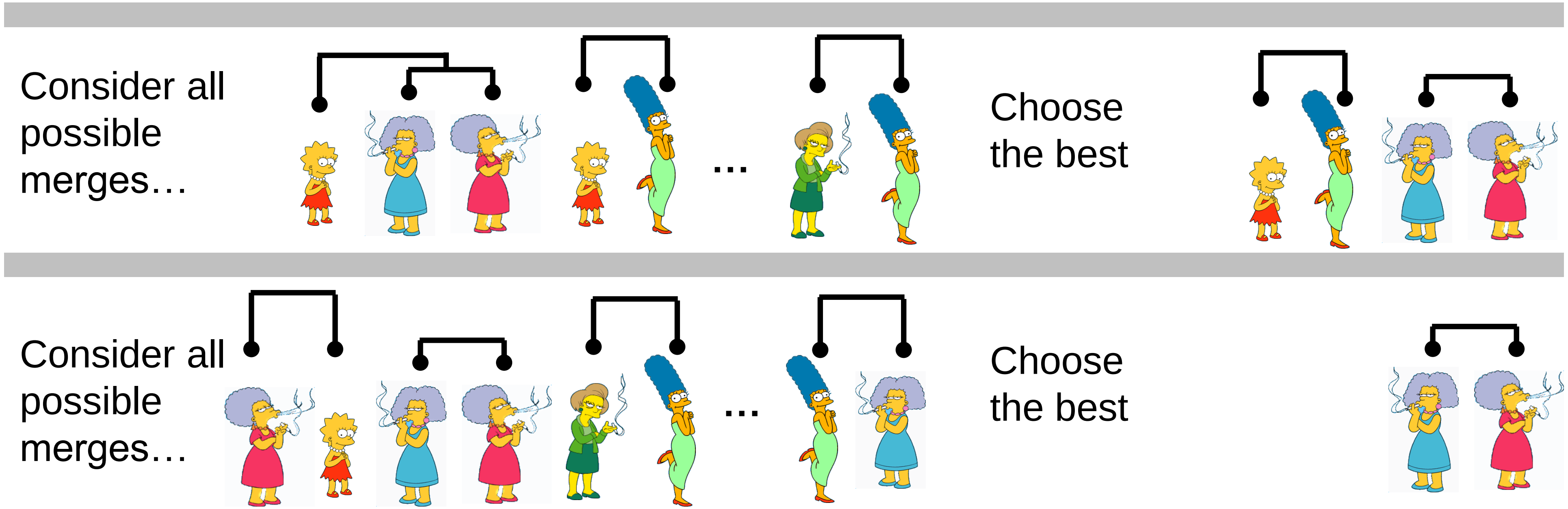
Bottom-Up (agglomerative):

- **Bottom-Up (agglomerative):** 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.



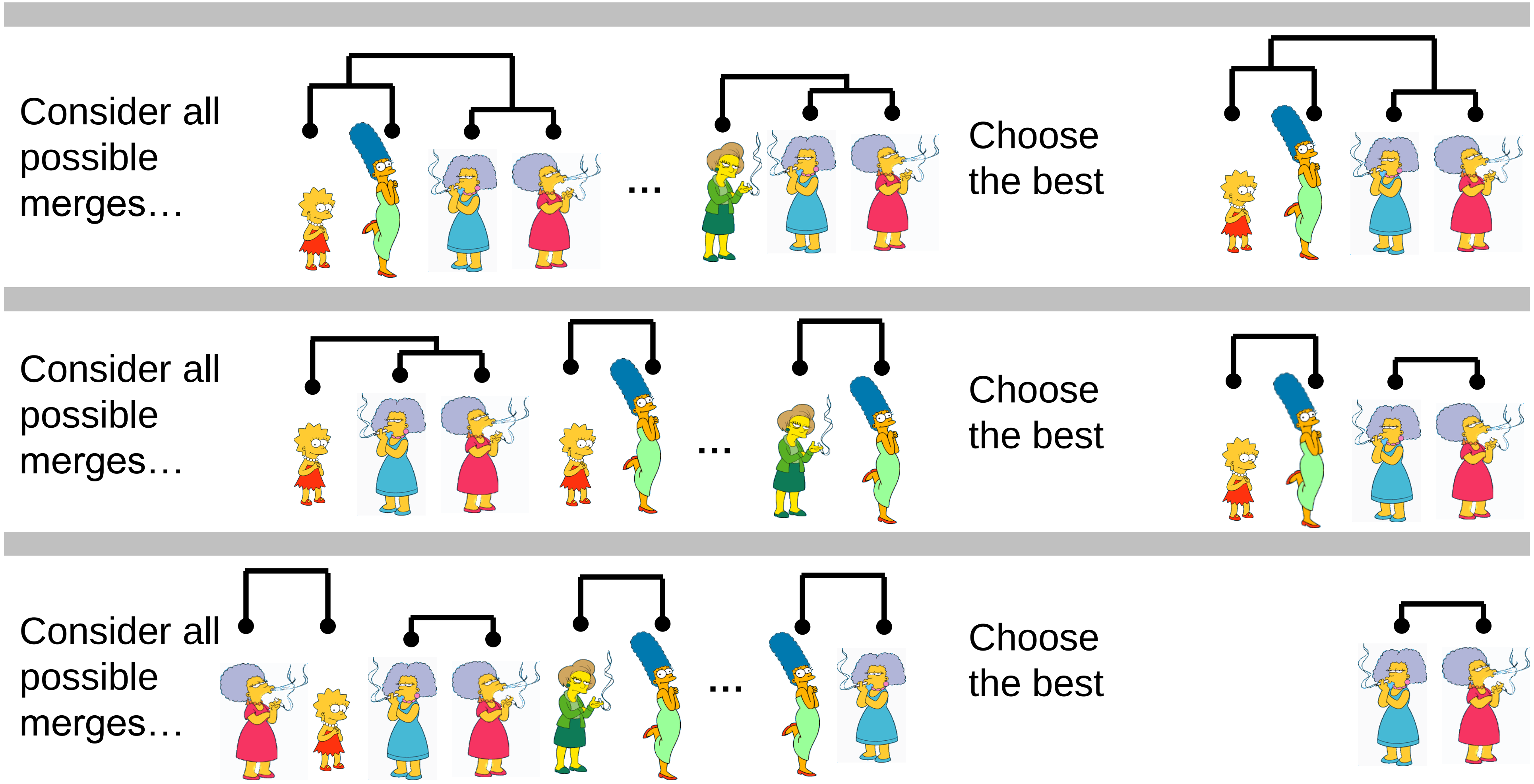
Bottom-Up (agglomerative) (2)

- **Bottom-Up (agglomerative):** 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.

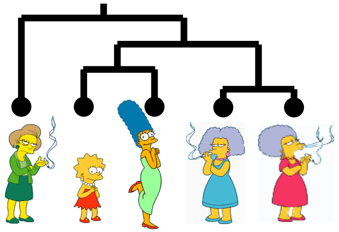


Bottom-Up (agglomerative) (3)

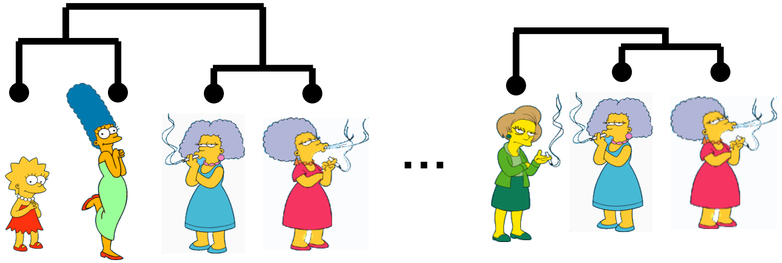
- **Bottom-Up (agglomerative):** Starting with each item in its own cluster, find the best pair to merge into a new cluster. Repeat until all clusters are fused together.



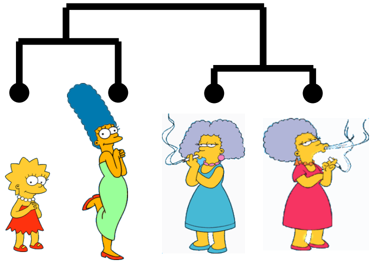
Bottom-Up (agglomerative) (Final)



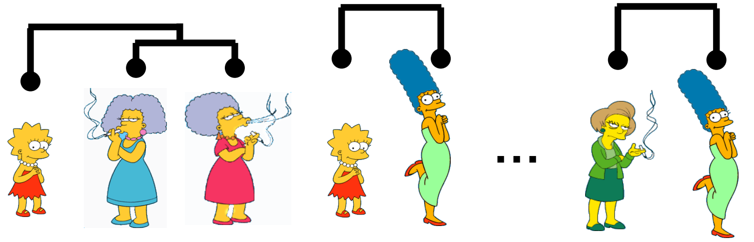
Consider all possible merges…



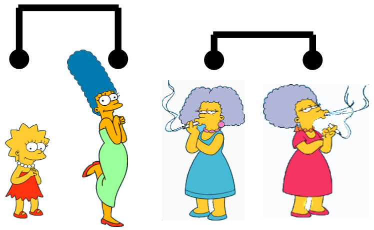
Choose the best



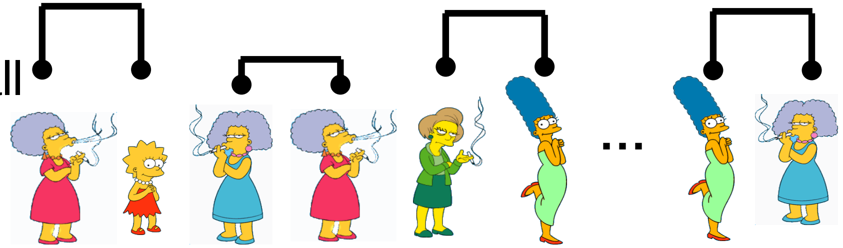
Consider all possible merges…



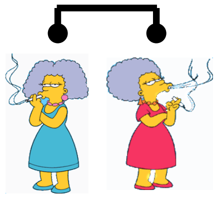
Choose the best



Consider all possible merges…



Choose the best



Distance Criteria

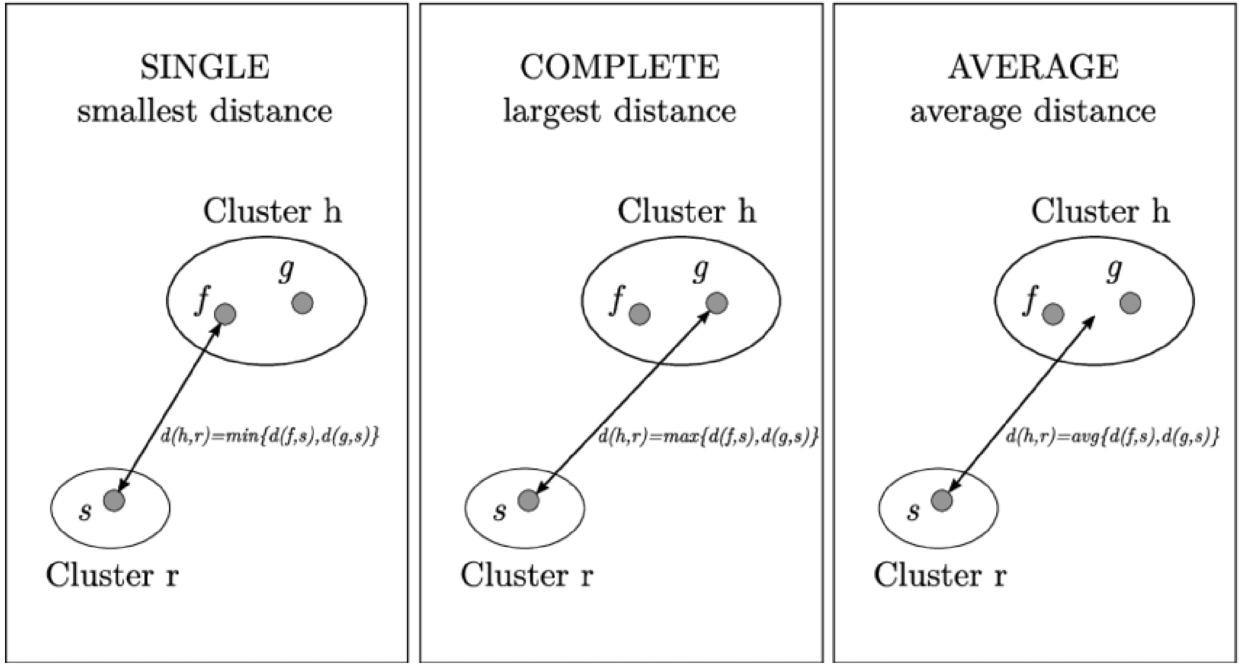
- We know how to measure the distance between two objects, but defining the distance between an object and a cluster, or defining the distance between two clusters is non obvious.

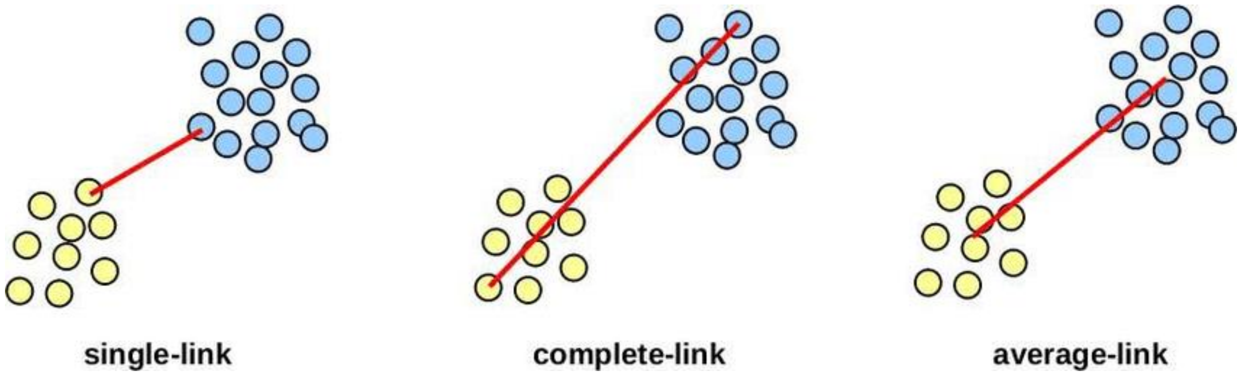
- **Single linkage (nearest neighbor):** In this method the distance between two clusters is determined by the distance of the two closest objects (nearest neighbors) in the different clusters.

- **Complete linkage (furthest neighbor):** In this method, the distances between clusters are determined by the greatest distance between any two objects in the different clusters (i.e., by the "furthest neighbors").

- **Group average linkage:** In this method, the distance between two clusters is calculated as the average distance between all pairs of objects in the two different clusters.

Distance Criteria Visual





Summary of Hierarchical Clustering

- No need to specify the number of clusters in advance.
- Hierarchal nature maps nicely onto human intuition for some domains
- They do not scale well (expensive to compute)
- Like any heuristic search algorithms, local optima are a problem (i.e. the solution found may not be the globally optimal solution).
- Interpretation of results is (very) subjective.

End of Part 2

Machine Learning - Clustering

Week 10 – Part 3 – Partitioned Clustering

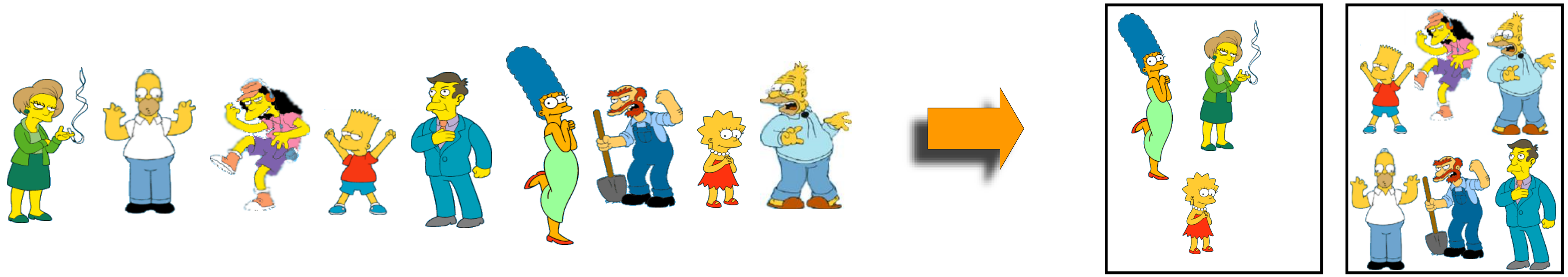
CS 457 - L1 Data Science

Zeehasham Rasheed

Partitioned Clustering

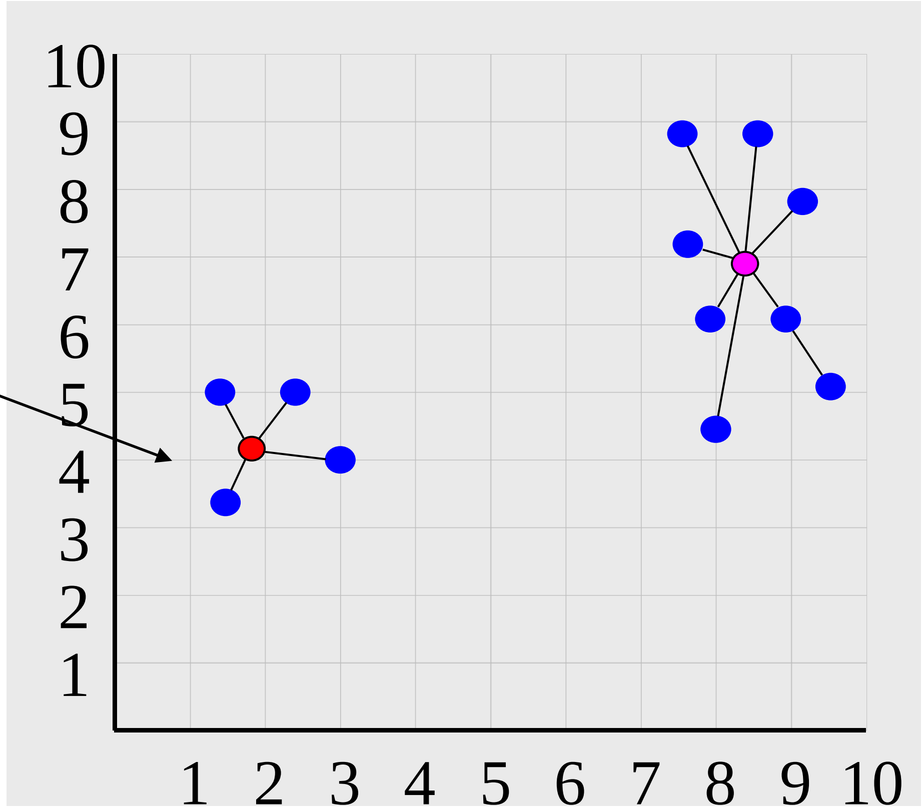
- Nonhierarchical, each instance is placed in exactly one of K non-overlapping clusters.

- Since only one set of clusters is output, the user normally has to input the desired number of clusters K.



Sum of Squared Error

- Lets assume we have two clusters in our data
- Compute the sum of squared Distances (average distance) for these 4 points.
- Equivalent to the **Residual Sum of squares (RSS).**
- This is called the objective function for this cluster.
- Do this for every cluster.
- We want the objective function to be **as small as possible**



- Different terminologies with same meaning
- **SSE (Sum of Squared Error)**
- **WSS (Weighted Sum of Squared Error)**

K-means Clustering

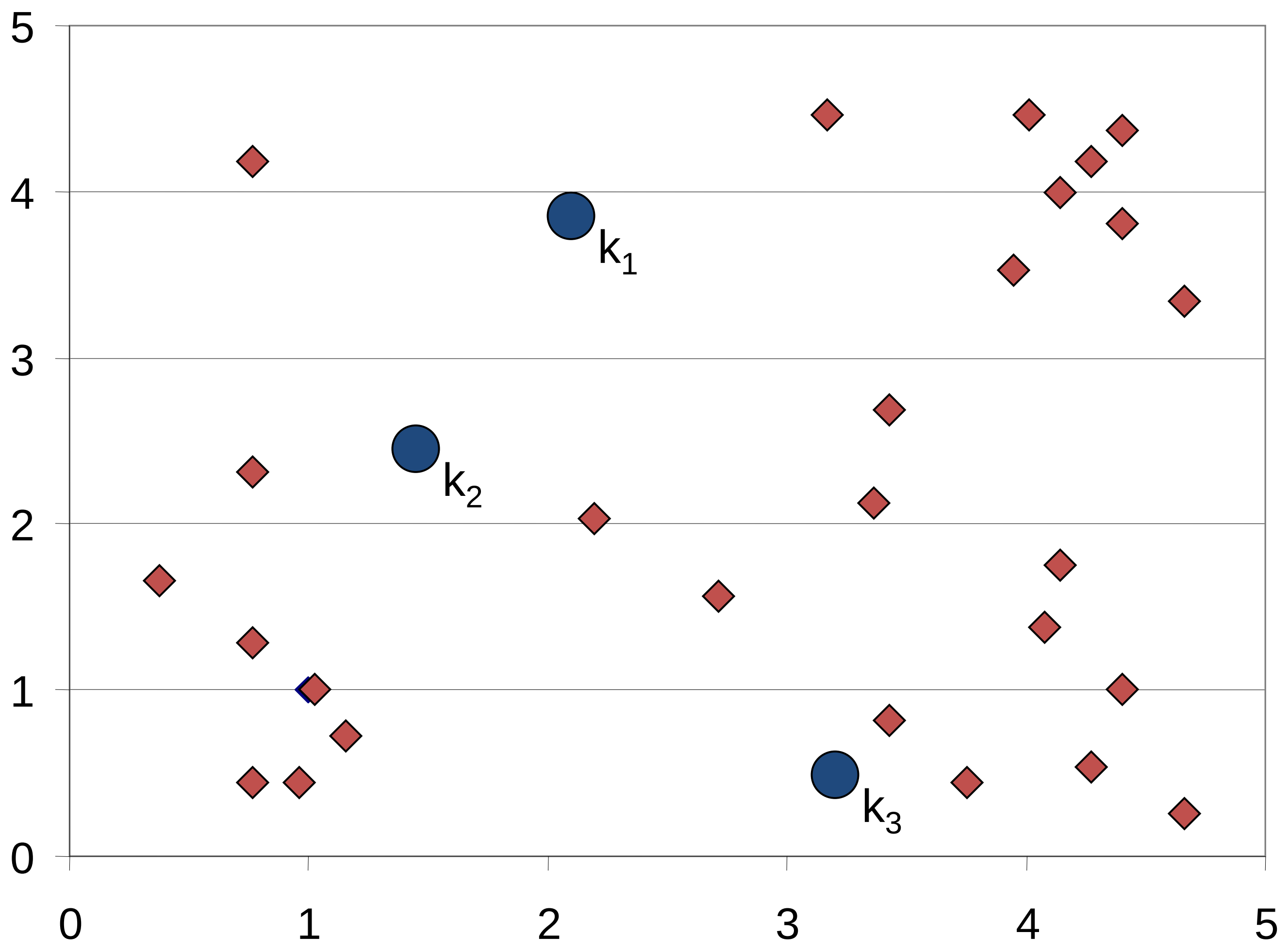
- K-Means is a Partitional clustering approach
- Each cluster is associated with a **centroid** (center point)
- Each point is assigned to the cluster with the closest centroid
- Number of clusters, K, must be specified in the beginning
- The basic algorithm is very simple

Understanding the Centroid in K-Means Clustering

- The “means” in K-means clustering corresponds to the average distance for each point (SSE/WSS) in the cluster to the cluster’s center (centroid).
- To start the K-means clustering process, you will specify the number of clusters, the maximum number of iterations, and the starting location for K centroids (cluster centers for which you will normally specify K-random values).
- The locations that you choose for the starting centroids can be random.
- The K-means algorithm will move the centroids as it performs its processing to the ideal locations (where each cluster has minimum SSE/WSS).
- K-means is an iterative algorithm that loops until either the maximum number of iterations is reached, or the clusters do not change.

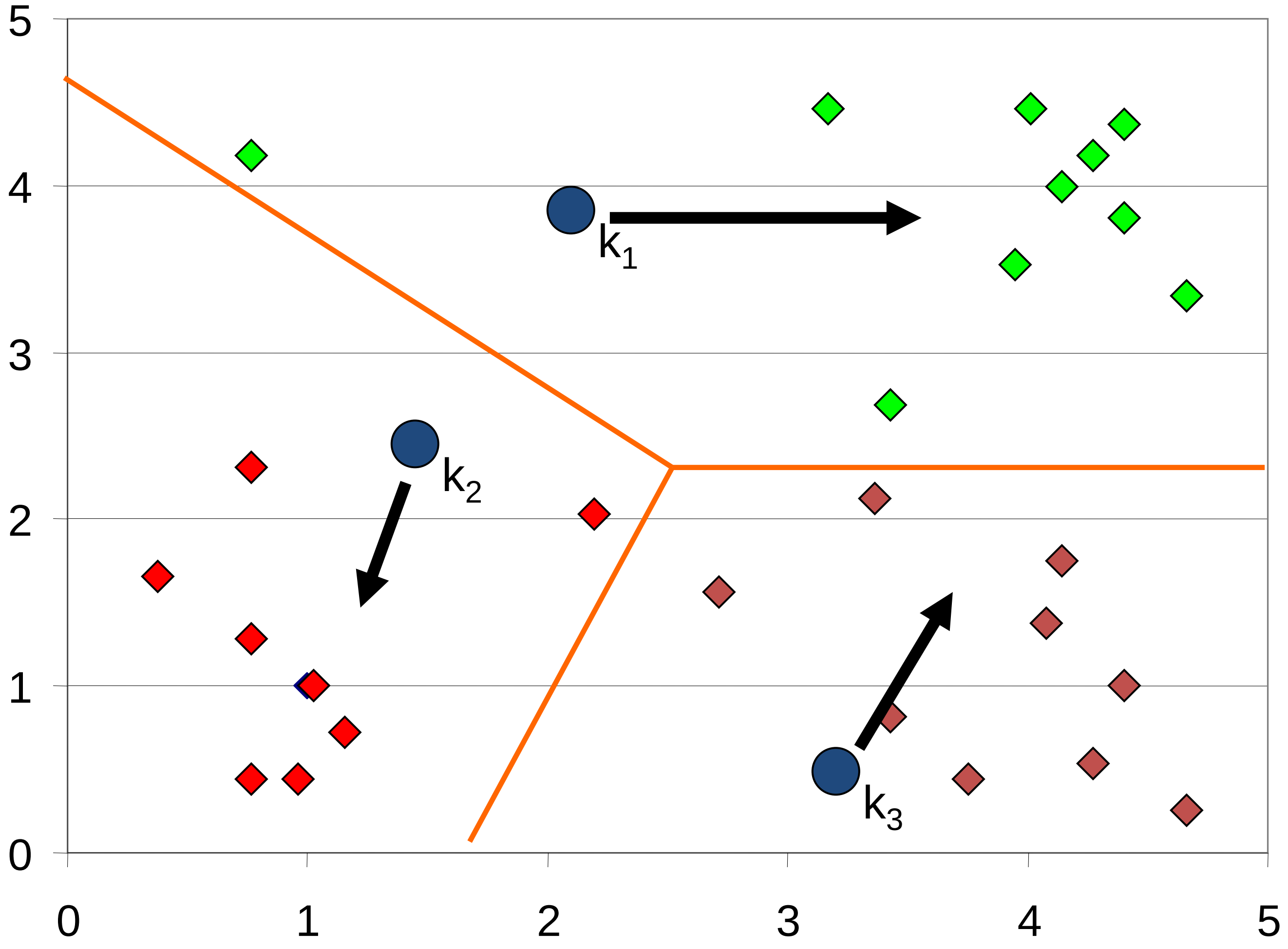
K-means Clustering: Step 1

Algorithm: k-means, Distance Metric: Euclidean Distance

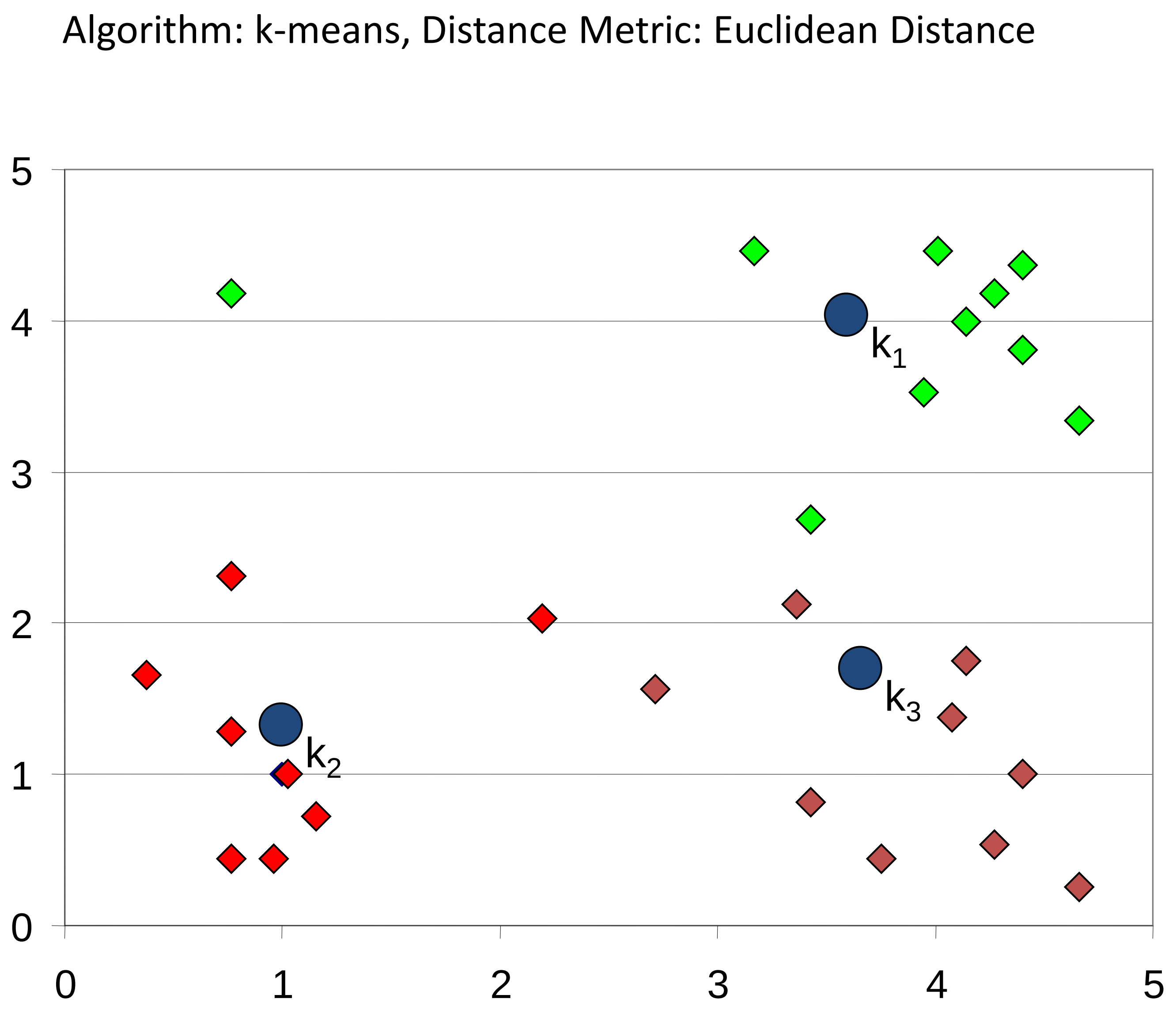


K-means Clustering: Step 2

Algorithm: k-means, Distance Metric: Euclidean Distance

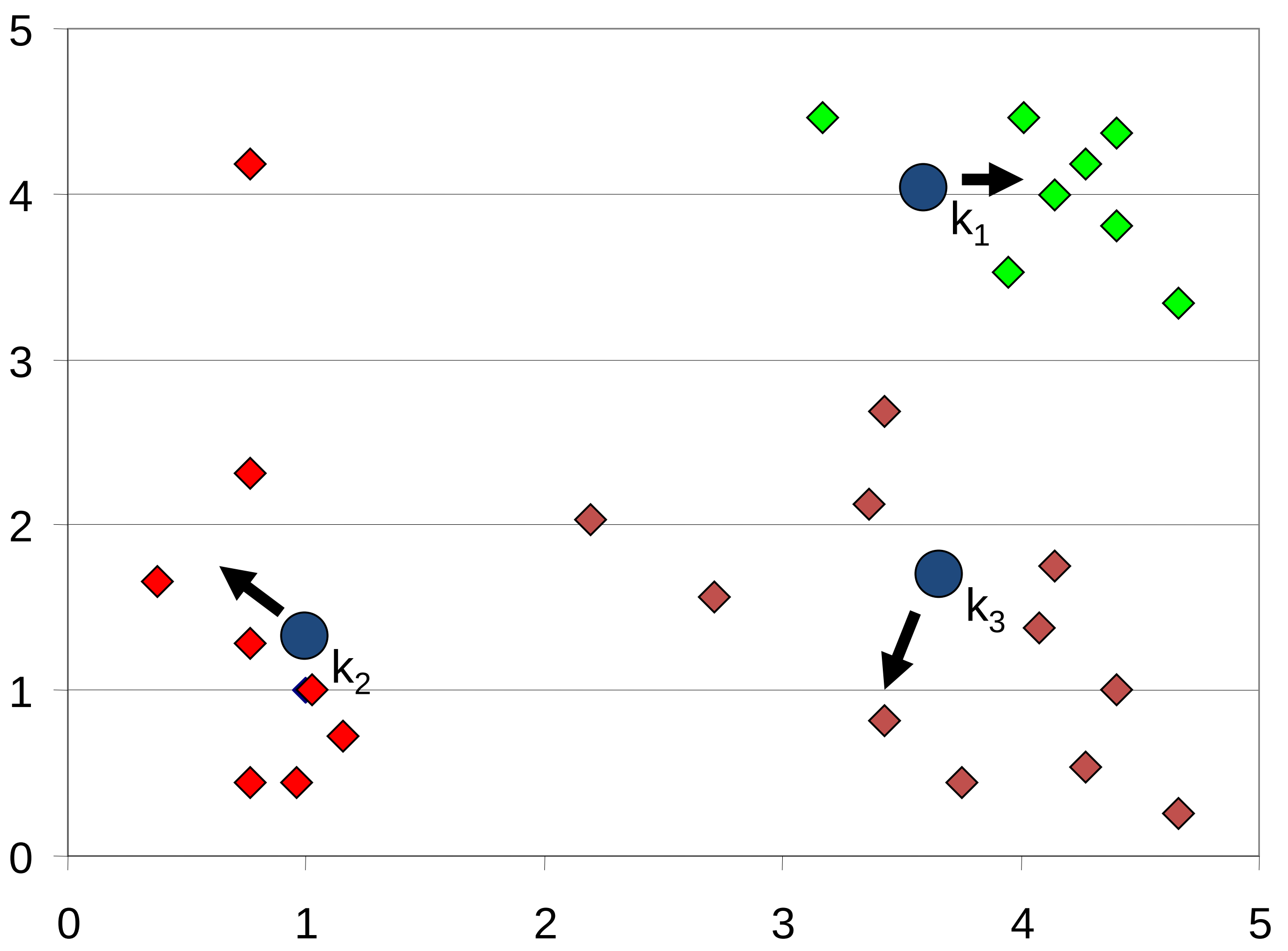


K-means Clustering: Step 3



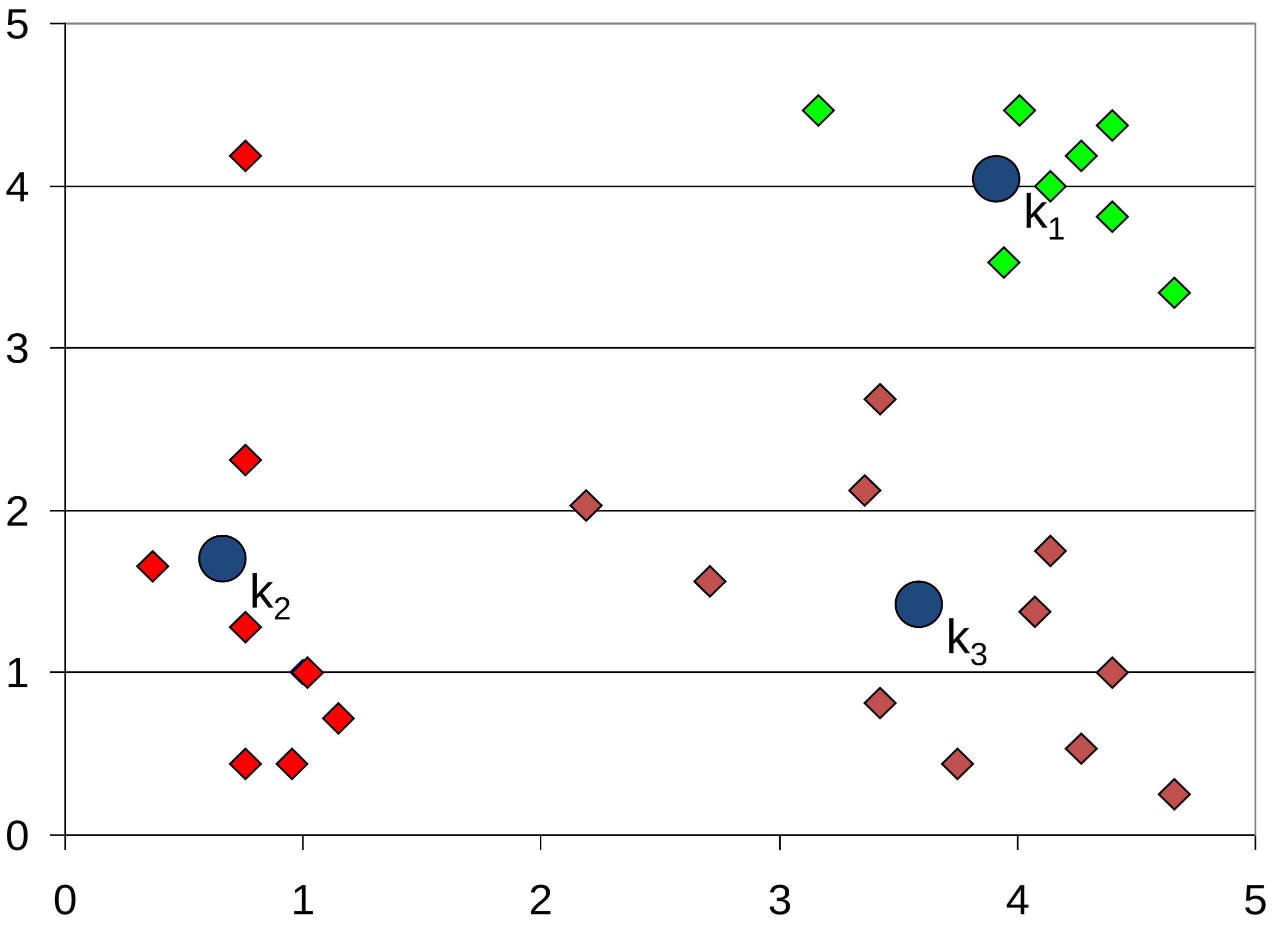
K-means Clustering: Step 4

Algorithm: k-means, Distance Metric: Euclidean Distance



K-means Clustering: Step 5

Algorithm: k-means, Distance Metric: Euclidean Distance



K-Means Demo

- http://his.anthropomatik.kit.edu/users/loesch/LaborWissRepr-DHBW-KA-2010SS/Clustering_K-means_demo.html

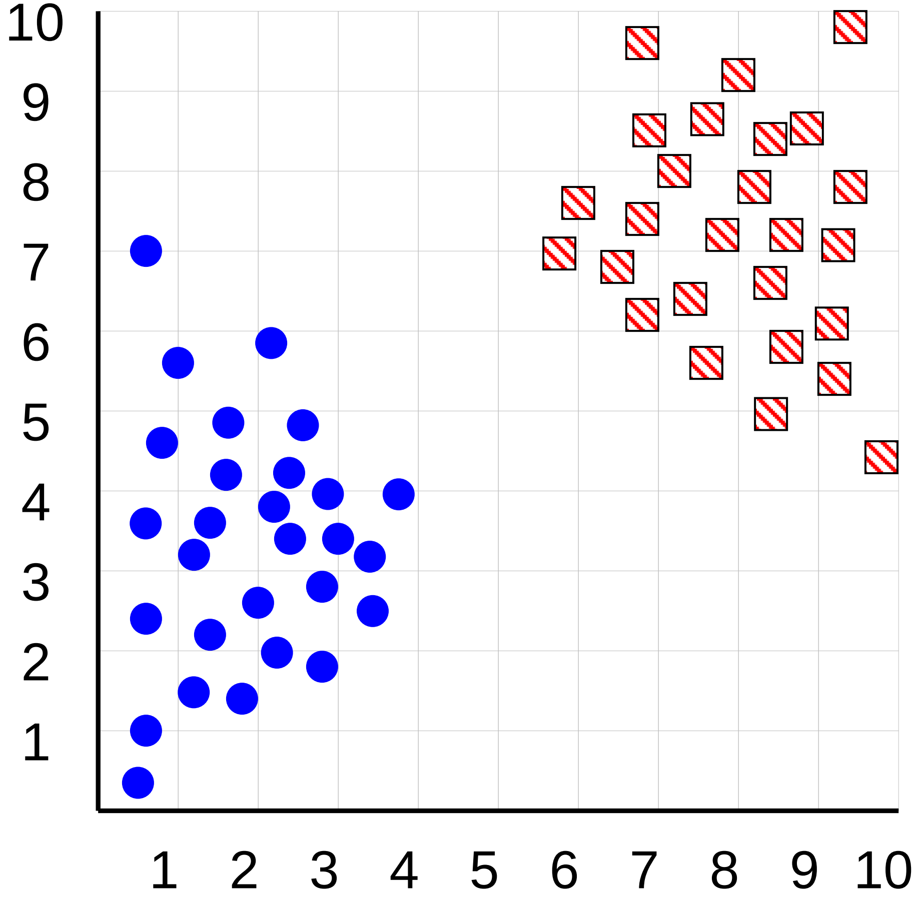
Comments on the K-Means Method

- Strength
  - *Relatively efficient.*
  - Easy to implement.

- Weakness
  - Applicable only when *mean* is defined, then what about categorical data?
  - Need to specify *k,* the *number* of clusters, in advance
  - Unable to handle noisy data and *outliers*
  - Clustering results greatly depend on the initial centroids.
    - Remedy: repeat the algorithm many times with different centroids

Katydid/Grasshopper Dataset

- How can we tell the right number of clusters (K)?
- In general, this is a unsolved problem. However there are many approximate methods. In the next few slides we will see an example.

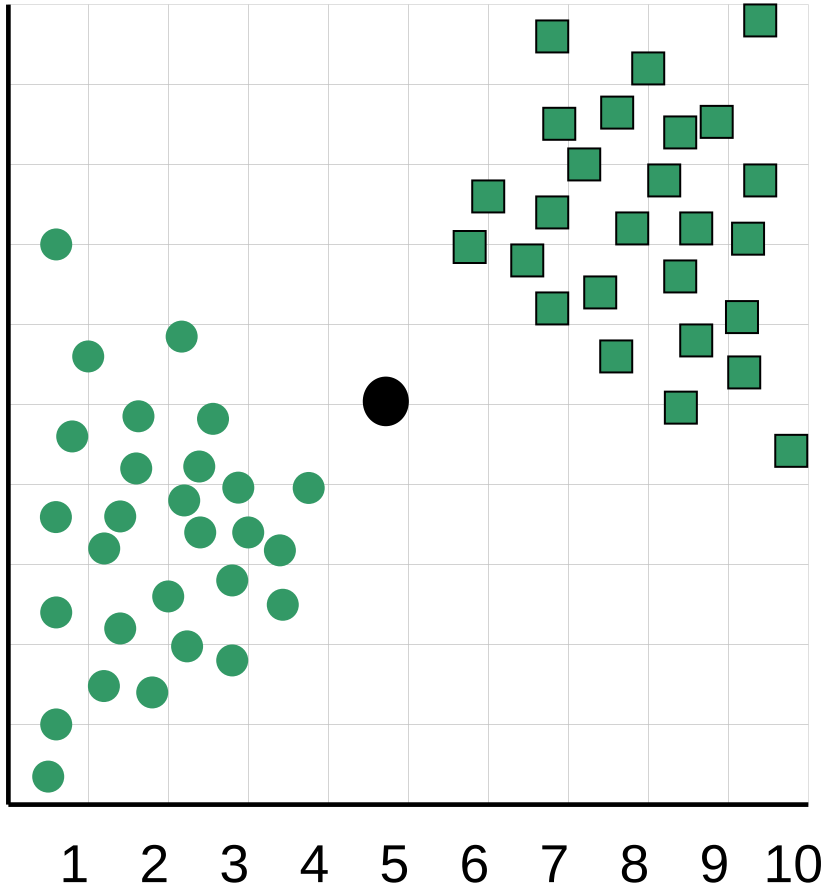


For our example, we will use the familiar katydid/grasshopper dataset.

However, in this case we are imagining that we do NOT know the class labels. We are only clustering on the X and Y axis values.

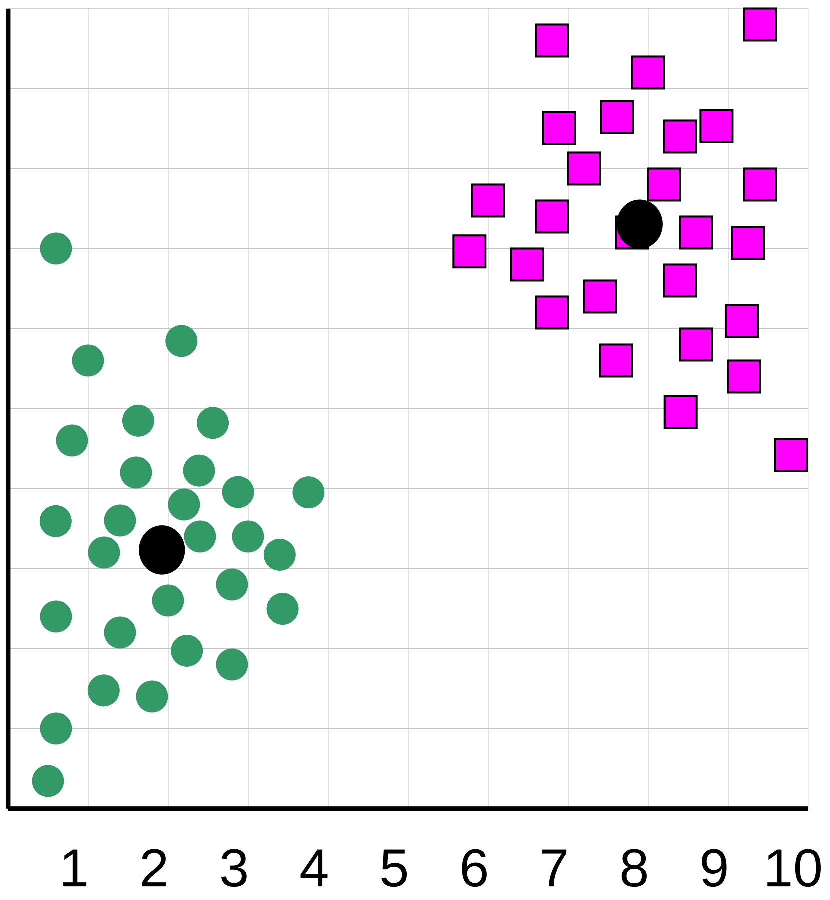
Katydid/Grasshopper Dataset (2)

- When k = 1, the objective function is 873.0



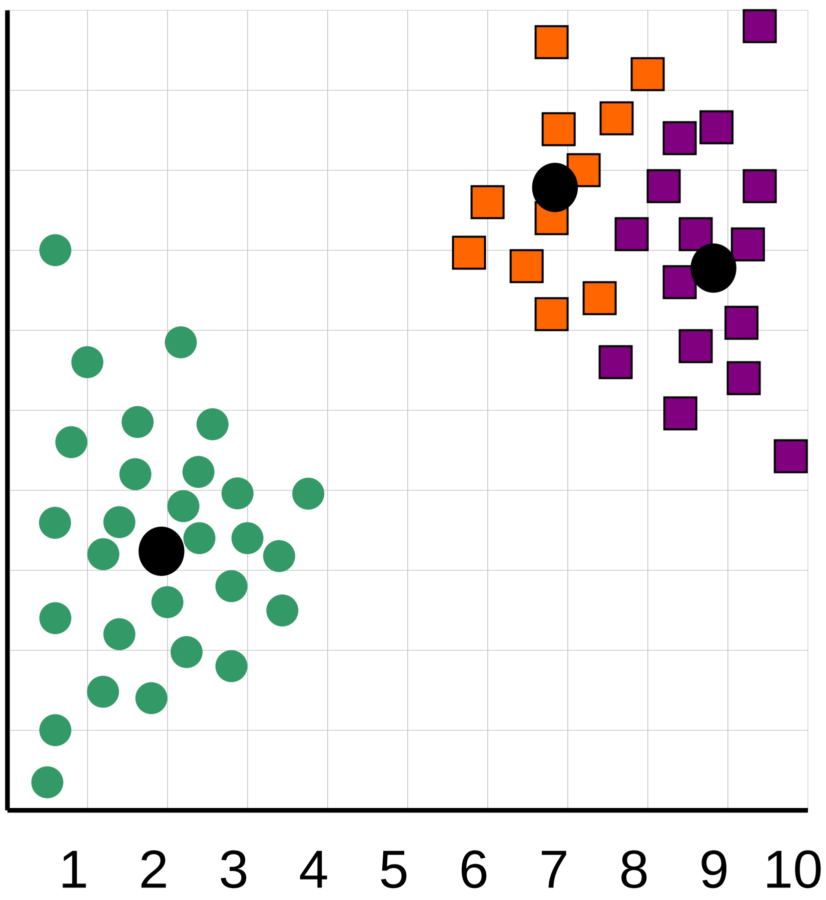
Katydid/Grasshopper Dataset (3)

- When k = 2, the objective function is 173.1



Katydid/Grasshopper Dataset (4)

- When k = 3, the objective function is 133.6



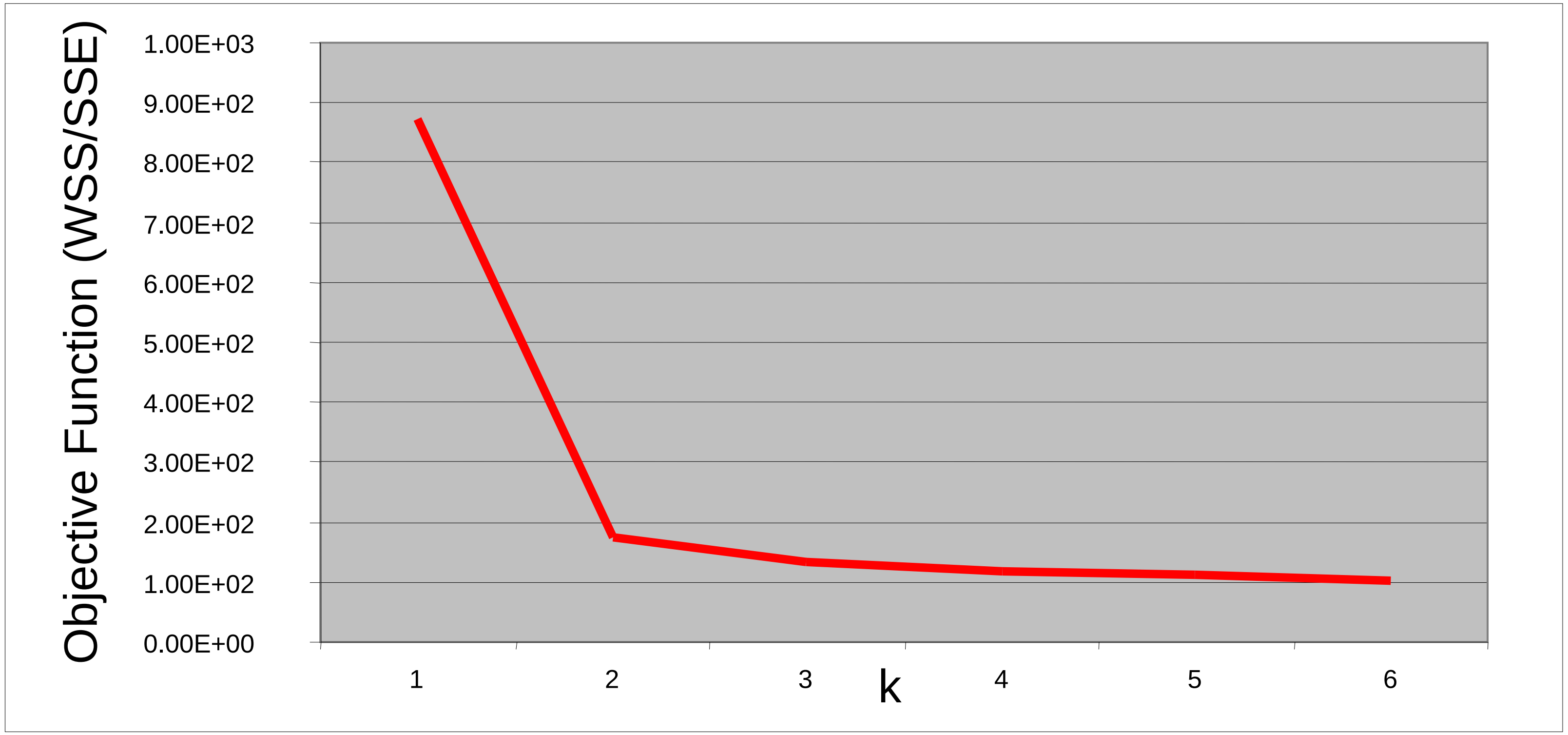
Determine Optimal Number of Clusters

- When you use the K-means algorithm, you must specify the value of K—the number of clusters you desire.
-
- If you specify too few clusters, you may lose valuable insights. Likewise, if you specify too many clusters, you will increase your processing time and you may not gain additional insights.
- You will need to determine and specify the number of clusters for each data set. Depending on the data-set values, you may find that for one set of values (possibly from the same data source), a cluster size of 3 is appropriate, whereas for others, a cluster size of 5 provides better grouping.
- The only way to determine the appropriate cluster size is to create clusters and the analyze/visualize the results (normally using the sum of squared distances).
- Algorithms exist to help you determine the proper number of clusters for your data. A common approach is called the **“Elbow Method” or “Knee Plot”** so named because the chart that it produces resembles the bend in an elbow.

Elbow/Knee Plot

We can plot the objective function values for k equals 1 to 6…

The abrupt change at k = 2, is highly suggestive of two clusters in the data. This technique for determining the number of clusters is known as "knee finding" or "elbow finding".



Note that the results are not always as clear cut as in this toy example

Scaling (Normalization)

- Clustering is all about calculating Distances
- Let's say that you have two features:
  - weight (in Lbs)
  - height (in Feet)
- and we are using them to make **'S' or 'L' size clusters**
- Lets say we have two people already in those clusters
  - Adam (175Lbs + 5.9ft) in L
  - Lucy (115Lbs + 5.2ft) in S
- We have a new person - Alan (140Lbs + 6.1ft.)
  - Your clustering algorithm will put it in the cluster which is nearest.
  - So, **if we do not scale** the features here, the height is not having much effect and Alan will be allotted in **'S'** cluster. (ideally it should be '**L**')

Adam/Alan = abs(175-140) + abs(5.9-6.1)
= 35.2

Lucy/Alan = abs(115-140) + abs(5.2-6.1)
= 25.9

End of Part 3

Machine Learning - Clustering

Week 10 – Part 4 – Cluster Validity/Evaluation

CS 457 - L1 Data Science

Zeehasham Rasheed

Cluster Validity

- For supervised classification we have a variety of measures to evaluate how good our model is
- – Accuracy, precision, recall

- For cluster analysis, the analogous question is **how to evaluate the “goodness” of the resulting clusters?**

Measures of Cluster Validity

- Numerical measures that are applied to judge various aspects of cluster Validity, are classified into the following three types.
- – **Internal Index:** Used to measure the goodness of a clustering structure *without* respect to external information (most common)
- Sum of Squared Error (SSE) or Weighted Sum of Square Error (WSS)
- – **External Index:** Used to measure the extent to which cluster labels match externally supplied class labels. (more of a supervised learning)
- Entropy
- Sometimes these are referred to as criteria instead of indices

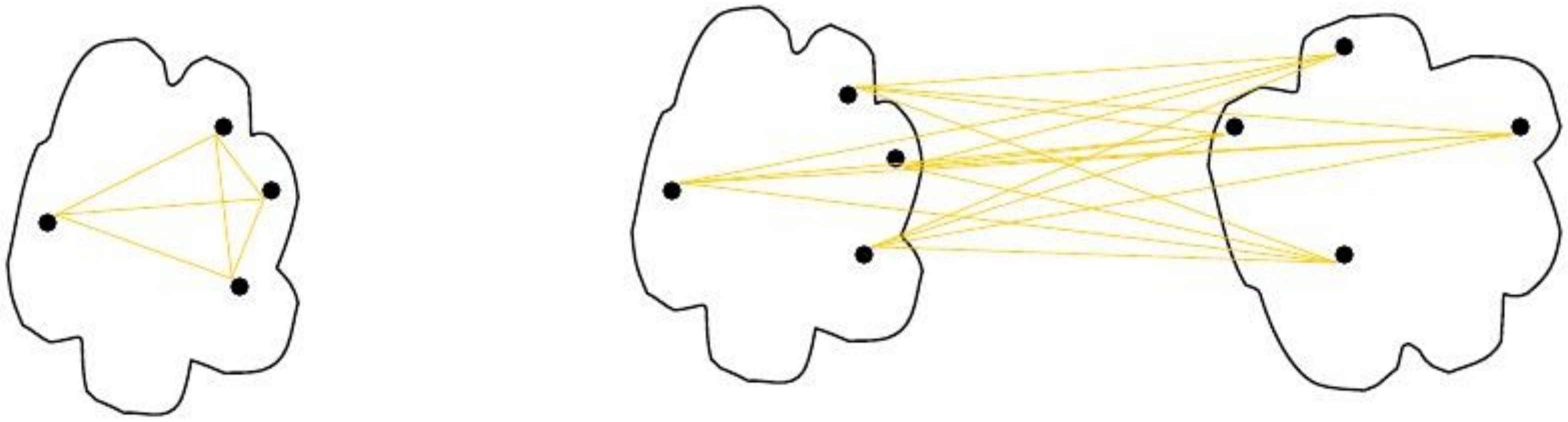
Internal Measures: Cohesion and Separation

- Cluster Cohesion:
  - Measures how closely related objects/points are in a cluster
- Cluster Separation:
  - Measure how distinct or well-separated a cluster is from other clusters
- Cohesion is measured by the within cluster sum of squares (SSE/WSS)
- Separation is measured by the between cluster sum of squares (SSE/WSS)

Internal Measures: Cohesion and Separation

- Cluster cohesion is the sum of the distance of all links within a cluster.

- Cluster separation is the sum of the distance between nodes in the cluster and nodes outside the cluster.



cohesion

within cluster sum of squares (SSE)

separation

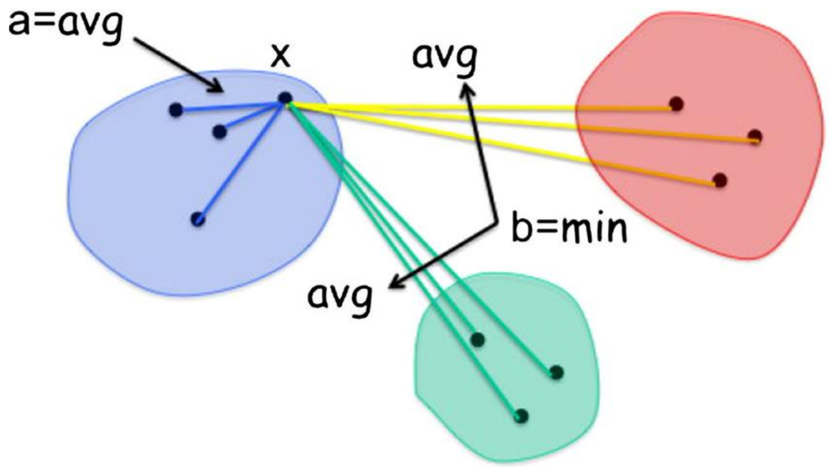
between cluster sum of squares (SSE)

Internal Measures: Silhouette Coefficient

- Silhouette Coefficient combine ideas of both cohesion and separation, but for individual points,
- as well as clusters and clustering
- For an individual point, i
  - Calculate a = average distance of i to the points in its cluster
  - Calculate b = min (average distance of i to points in another cluster)
  - The silhouette coefficient for a point is then given by

$$s = 1 - a/b \quad \text{if } a < b, \quad (\text{or } s = b/a - 1 \quad \text{if } a > b, \text{ not the usual case})$$

  - Typically between 0 and 1.
  - The closer to 1 the better.



Common Clustering Algorithms

TABLE 10.1 Common Clustering Approaches

Cluster Model	Example Algorithms	Notes
Centroid	K-means, K-means++	Selects cluster members based upon a mean optimization vector. Requires the data analyst to specify the number of clusters.
Connectivity	Hierarchical	Agglomerates (combines) related clusters to build a larger cluster. Illustrated using a chart called a dendrogram. Not well suited for large data sets.
Density	DBSCAN	Clusters are collected based on each point's proximity to dense regions in the data's coordinate space. Does not require the analyst to specify the number of clusters.

Key Terms You Should Know

- **Centroid**: the "center of a cluster". It does not have to correspond to a data point within the data set.
- **Clustering**: the process of "grouping related data". Clustering is an **unsupervised** learning process in that it works with unlabeled data.
- **Euclidian distance**: the straight-line distance between two points
- **K-means clustering**: a clustering technique that groups points based on minimizing the average distance of each point from its cluster's center (centroid)
- **Outlier**: a value that falls outside of the clusters

End of Part 4