Clustering

(for classification)

●●

USC Viterbi
School of Engineering

Clustering

••

USC Viterbi School of Engineering

Today's Topic: Clustering

- **Document clustering**
  - Motivations
  - Document representations
  - Success criteria
- **Clustering algorithms**
  - Partitional
  - Hierarchical

●●

USC Viterbi School of Engineering

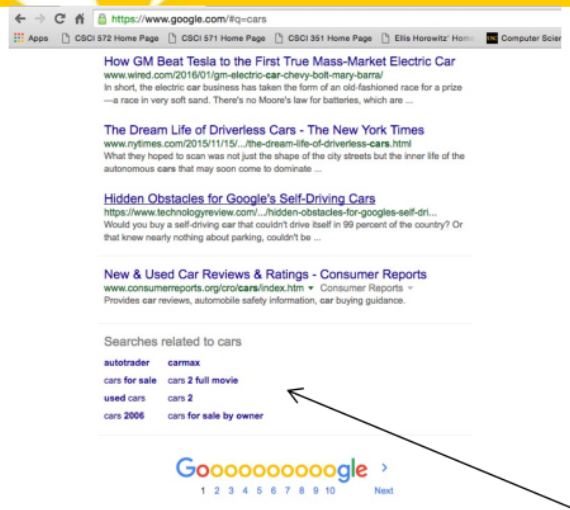
What is Clustering?

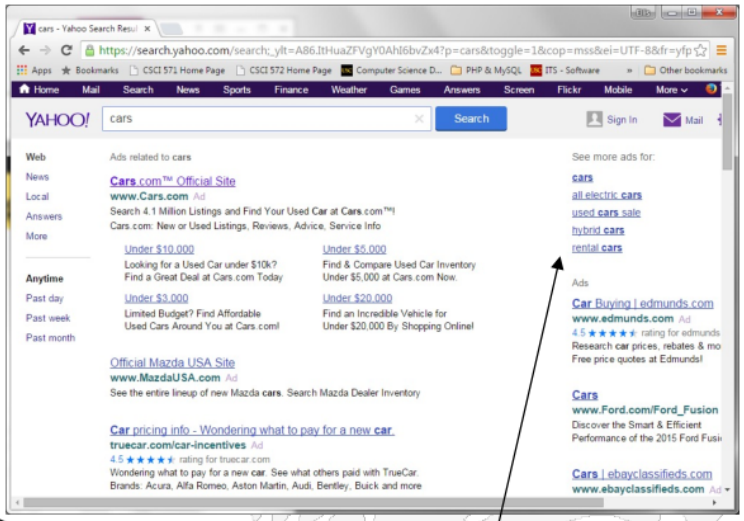
- **Clustering: the process of grouping a set of objects into classes of similar objects**
  - Documents within a cluster should be similar.
  - Documents from different clusters should be dissimilar.
- **Clustering is the most common form of *unsupervised learning***
  - Unsupervised learning = learning from raw data, as opposed to supervised learning where a classification of examples is given a priori
- Clustering is a common and important task that finds many applications in IR and other places

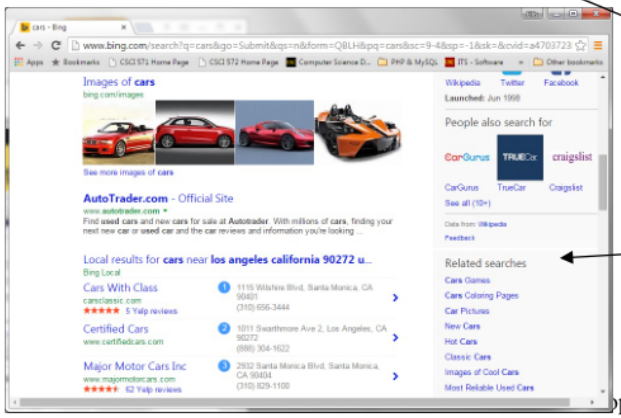
●●

USC Viterbi School of Engineering

Related Searches are a Form of Clustering







Google related searches

Yahoo does some clustering via alternate queries

Bing does a little better

●●

USC Viterbi School of Engineering

yippy.com Search Engine

- **Yippy** (formerly **Clusty**) is a metasearch engine developed by Vivísimo which emphasizes clusters of results.

initial screen
with query "cars"

clustered results appear
on the left column: e.g.
sale
reviews
dealers
rentals

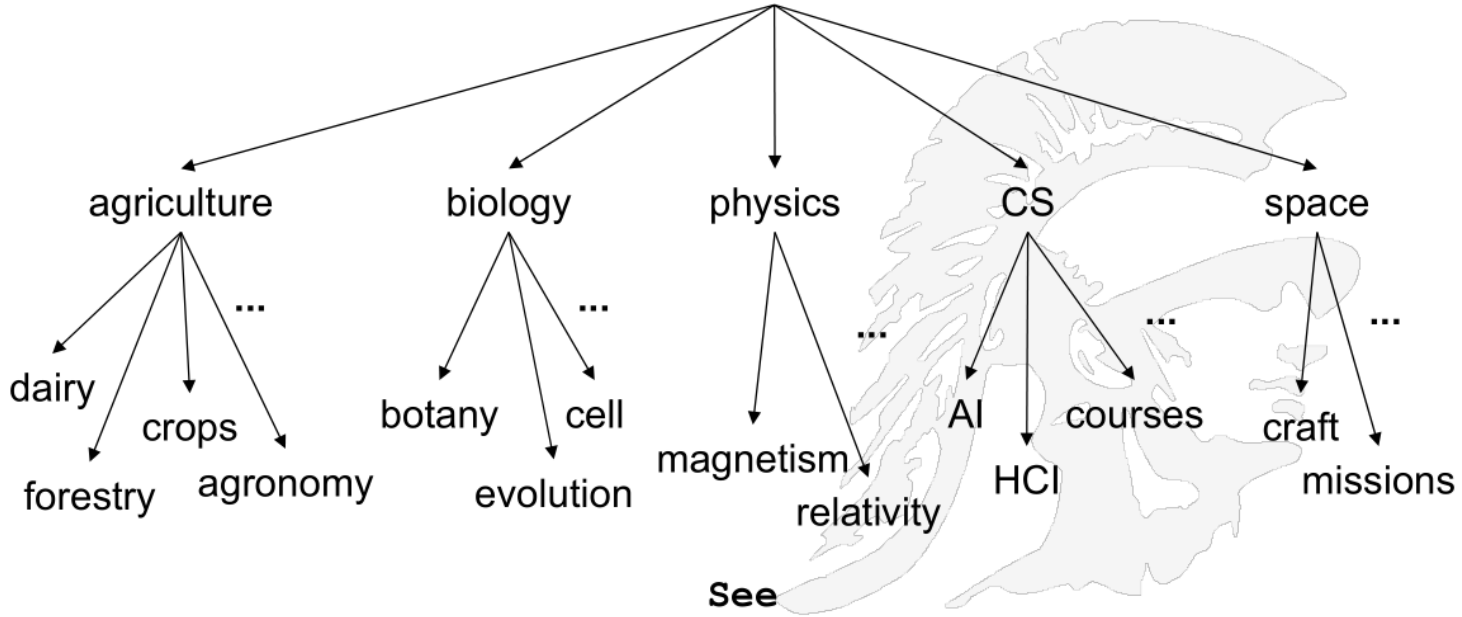
multiple level clusters:
car dealers
trucks
ebay

●●

USC Viterbi School of Engineering

Yahoo's Name Derives from *Yet Another Hierarchical Officious Oracle*

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



See

https://searchengineland.com/yahoo-directory-close-204370

●●

USC Viterbi School of Engineering

What Is A Good Clustering?

- **Internal criterion: A good clustering will produce high quality clusters in which:**
  - the **intra-class** (that is, intra-cluster) similarity is high
  - the **inter-class** similarity is low
  - The measured quality of a clustering depends on both the document representation and the similarity measure used

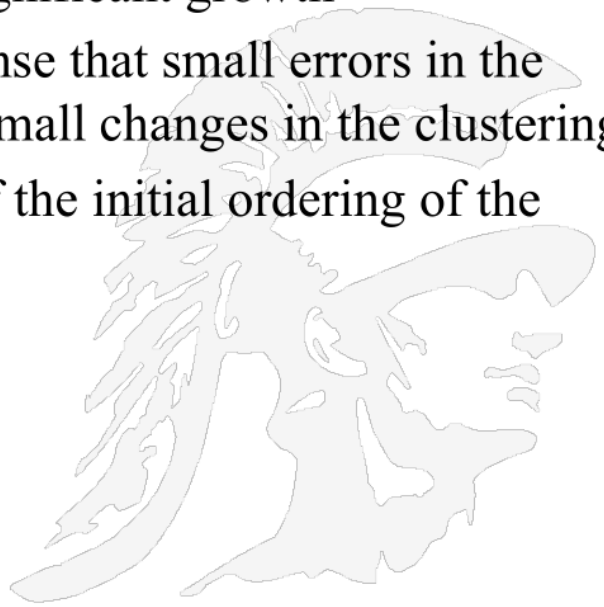
 12

●●

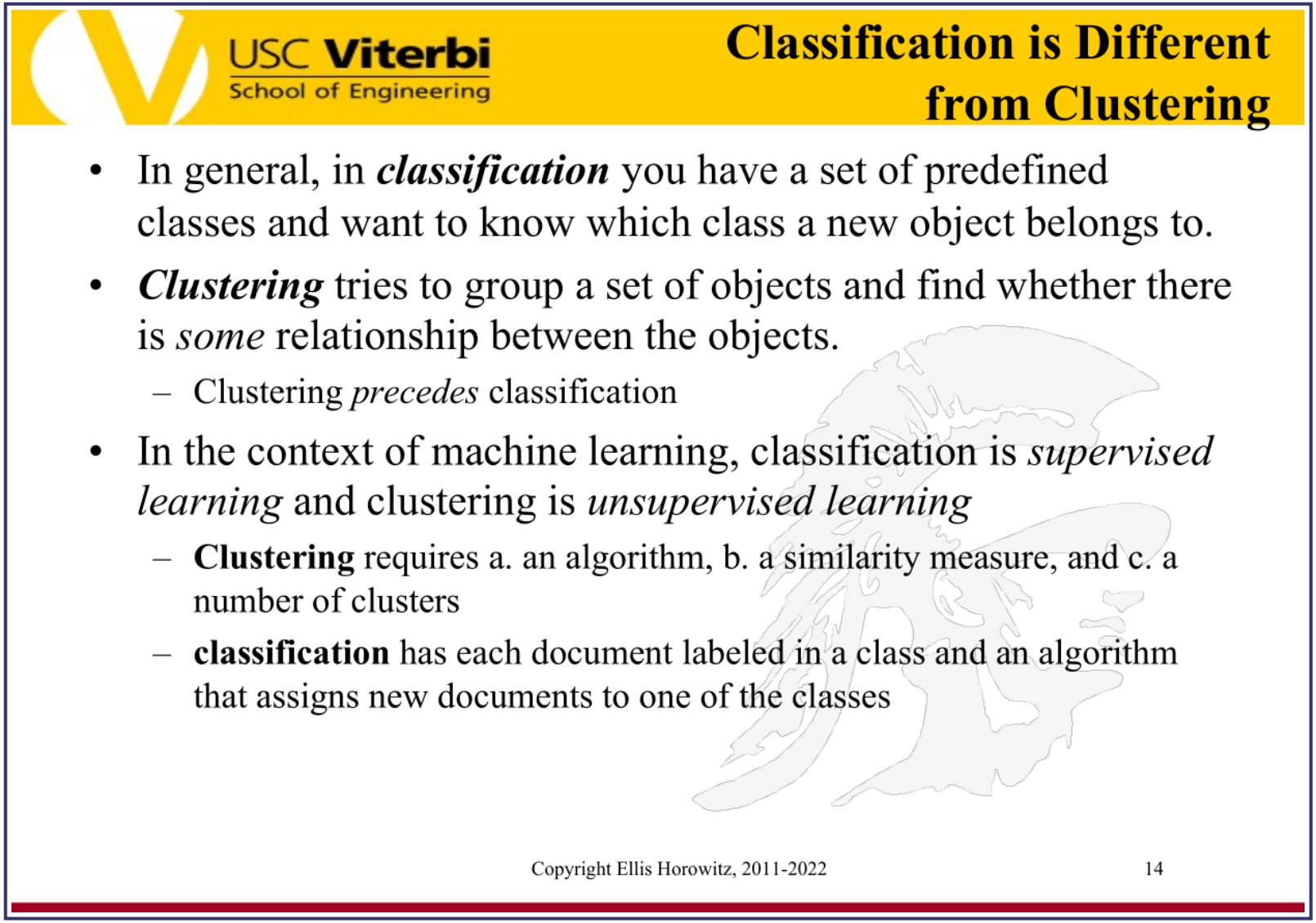
USC Viterbi School of Engineering

Three Criteria of Adequacy for Clustering Methods

1. The method produces a clustering which is **unlikely to be altered drastically** when further objects are incorporated
   - i.e. it is stable even under significant growth
2. The method is **stable** in the sense that small errors in the description of objects lead to small changes in the clustering
3. The method is **independent** of the initial ordering of the objects



●●

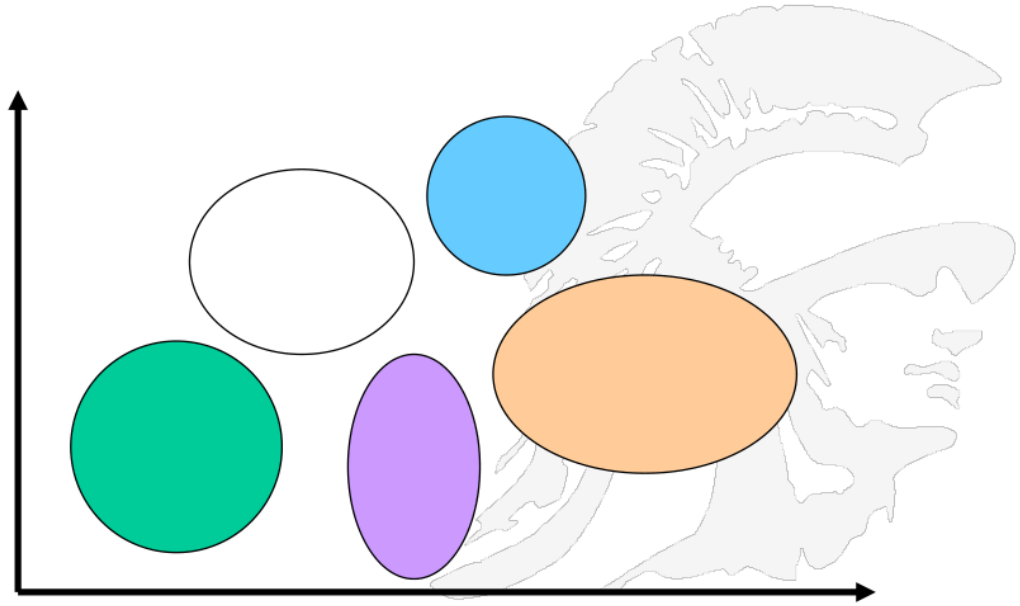

USC Viterbi
School of Engineering
Classification is Different from Clustering
• In general, in ***classification*** you have a set of predefined classes and want to know which class a new object belongs to.
• ***Clustering*** tries to group a set of objects and find whether there is *some* relationship between the objects.
– Clustering *precedes* classification
• In the context of machine learning, classification is *supervised learning* and clustering is *unsupervised learning*
– **Clustering** requires a. an algorithm, b. a similarity measure, and c. a number of clusters
– **classification** has each document labeled in a class and an algorithm that assigns new documents to one of the classes
Copyright Ellis Horowitz, 2011-2022
14

●●

USC Viterbi School of Engineering

Begin with Clustering

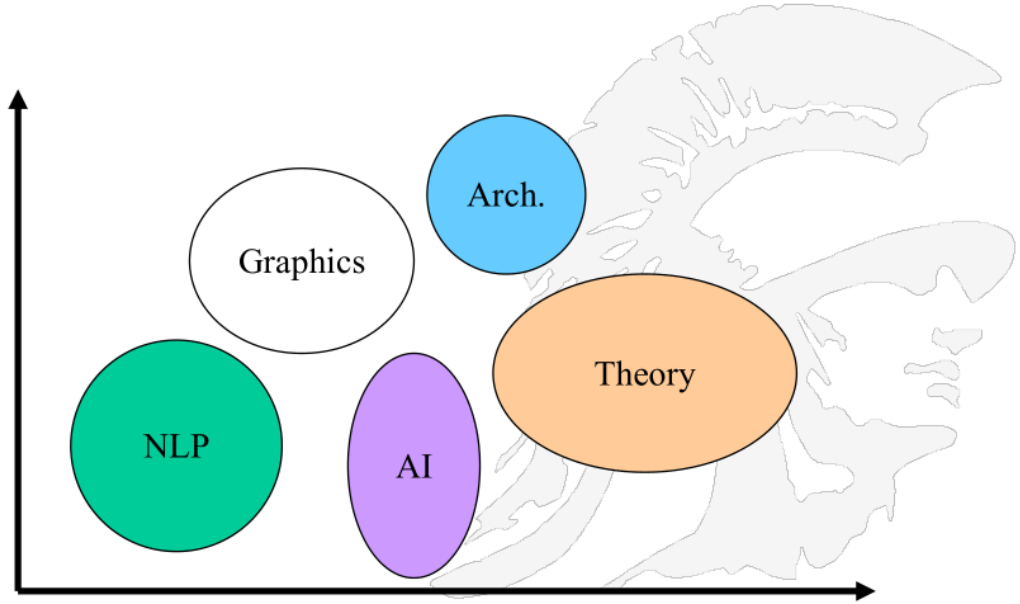
- **Step 1: Given a large set of computer science documents, first we cluster them using some algorithm (to be presented)**



USC Viterbi School of Engineering

Then We Name the Clusters

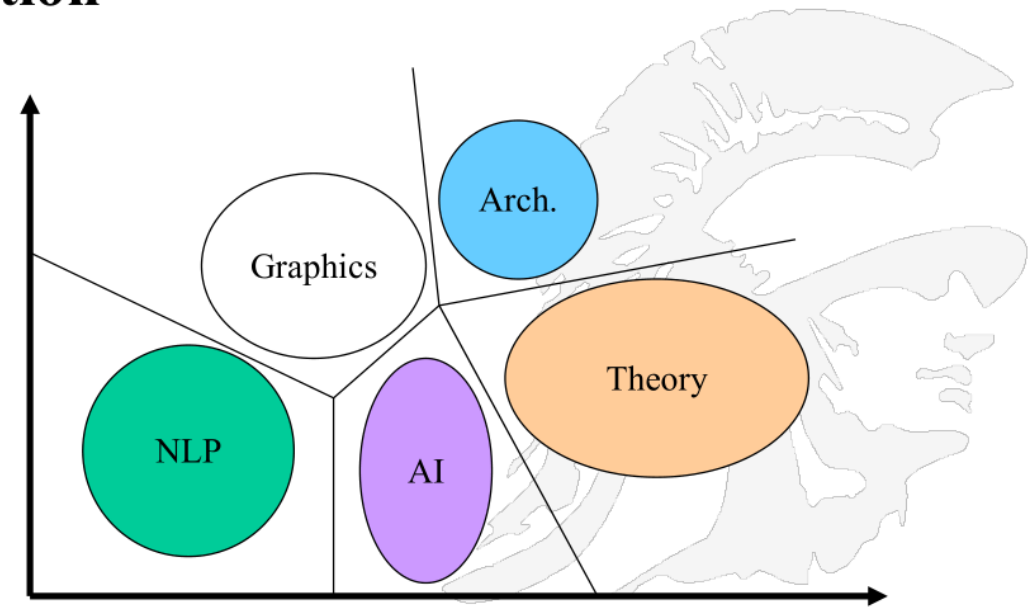
- **Step 2: we label the clusters**
  - **choosing a popular name from each document cluster**



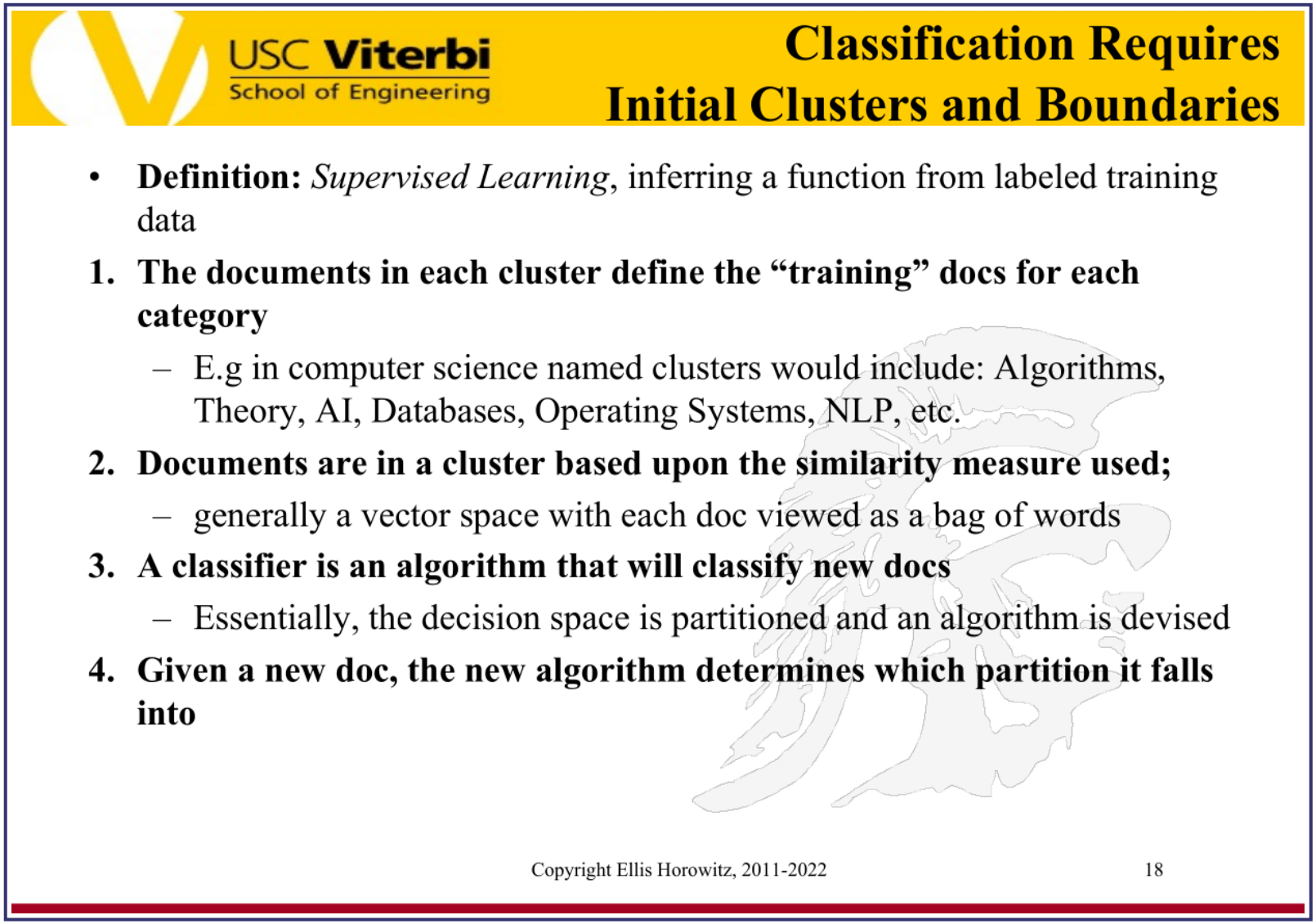
USC Viterbi School of Engineering

Still Clustering: Determine Decision Boundaries

- **Step 3: we compute boundaries for the clusters that can be used as new documents appear; i.e. classification**



●●


USC Viterbi
School of Engineering
Classification Requires Initial Clusters and Boundaries
• Definition: *Supervised Learning*, inferring a function from labeled training data
1. The documents in each cluster define the "training" docs for each category
– E.g in computer science named clusters would include: Algorithms, Theory, AI, Databases, Operating Systems, NLP, etc.
2. Documents are in a cluster based upon the similarity measure used;
– generally a vector space with each doc viewed as a bag of words
3. A classifier is an algorithm that will classify new docs
– Essentially, the decision space is partitioned and an algorithm is devised
4. Given a new doc, the new algorithm determines which partition it falls into
Copyright Ellis Horowitz, 2011-2022
18

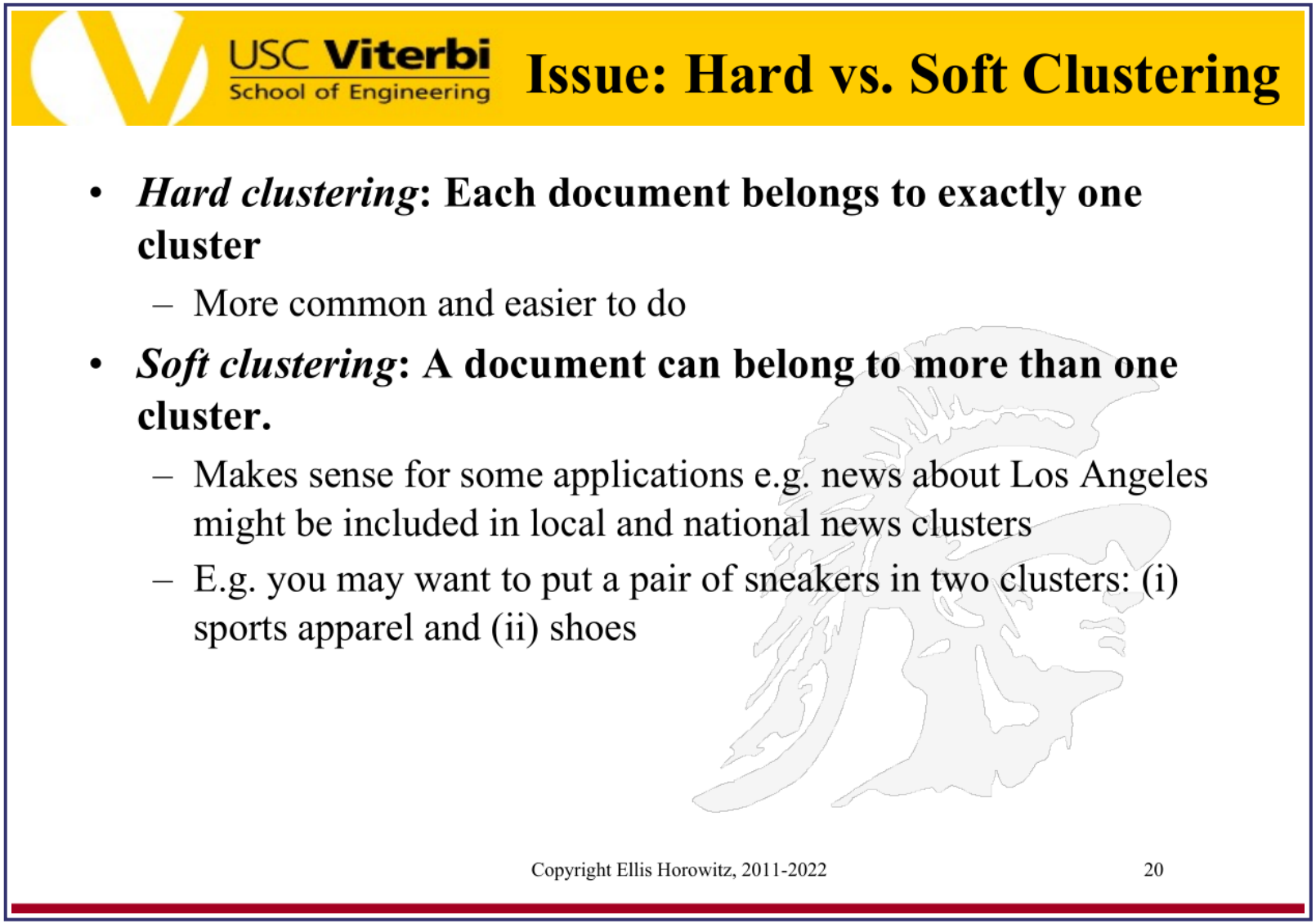
●●

USC Viterbi School of Engineering

Now Let's Return to the Earlier Problem: Clustering

- **Questions to consider when clustering**
  - How do we represent the document?
    - *Usually as a vector space*
  - How do we compute similarity/distance?
    - *Using cosine similarity*
  - How many clusters?
    - *will it be a fixed a priori number? or*
    - *completely data driven?*
  - Be careful to avoid "trivial" clusters - too large or small
    - *If a cluster is too large, then for navigation purposes you've wasted an extra user click without whittling down the set of documents much*

●●


USC Viterbi
School of Engineering
Issue: Hard vs. Soft Clustering
• *Hard clustering*: Each document belongs to exactly one cluster
– More common and easier to do
• *Soft clustering*: A document can belong to more than one cluster.
– Makes sense for some applications e.g. news about Los Angeles might be included in local and national news clusters
– E.g. you may want to put a pair of sneakers in two clusters: (i) sports apparel and (ii) shoes
Copyright Ellis Horowitz, 2011-2022
20

●●

USC Viterbi School of Engineering

What Definition of Similarity/Distance Will Be Used

- **Once again we will treat documents as vectors**
  - **Cosine similarity (seen before many times)**
    - Cosine similarity is a measure of similarity between two vectors of an inner product space that measures the cosine of the angle between them. Range from 0 (dissimilar) to 1 (exactly similar)

$$\text{similarity} = \cos(\theta) = \frac{A \cdot B}{\|A\|\|B\|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} (A_i)^2} \times \sqrt{\sum_{i=1}^{n} (B_i)^2}}$$

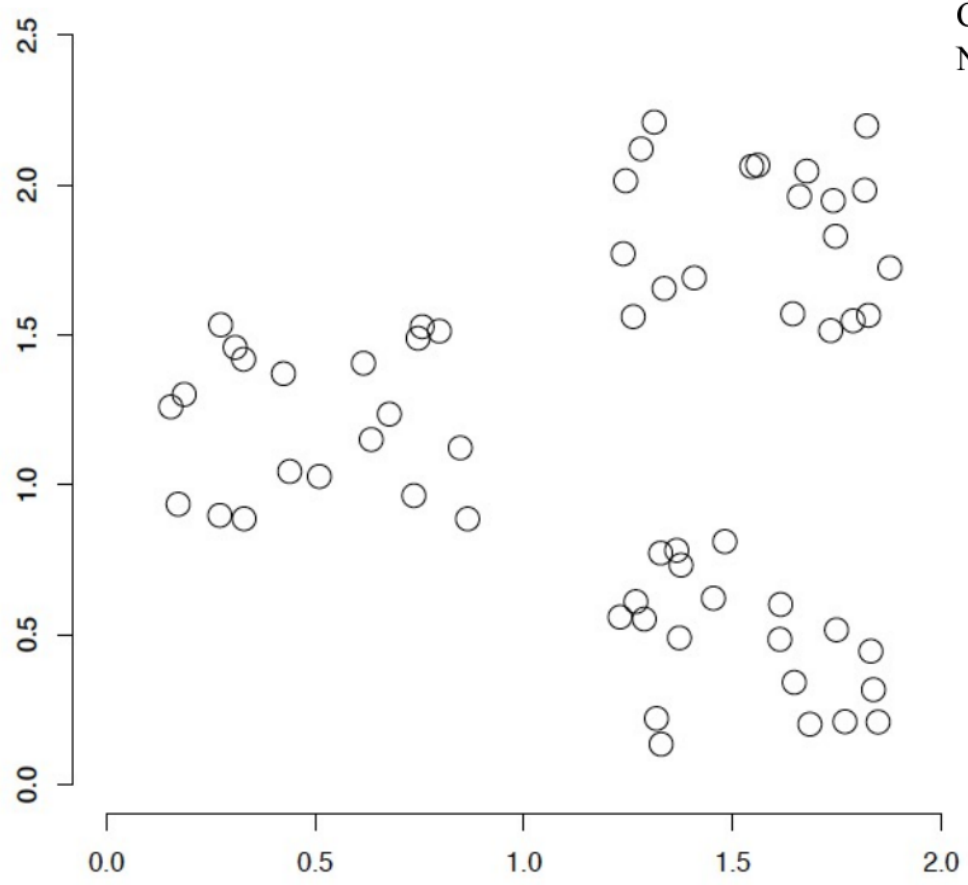
  - **Most clustering implementations use cosine similarity**
  - **Euclidean distance is a close alternative that is also popular**

••

USC Viterbi School of Engineering

A Data Set with Clear Cluster Structure

Circles represent documents as N-vectors



- **How would you design an algorithm for finding the three clusters in this case?**
- **Hint: use a distance measure**

22

●●

USC Viterbi School of Engineering

Clustering Algorithms

- **Two general methodologies**
  - Partitioning Based Algorithms
  - Hierarchical Algorithms
- **Partitioning Based**
  - Choose K and then divide a set of N items into K clusters
- **Hierarchical – Bottom Up/Top Down**
  - agglomerative: pairs of items or clusters are successively linked to produce larger clusters (hierarchy produced bottom-up)
  - divisive: start with the whole set as a cluster and successively divide sets into smaller partitions (hierarchy produced top-down)

 23

●●

USC Viterbi School of Engineering

A Partitioning Algorithm
K-Means Clustering Algorithm

- **Clustering algorithm strategy**
  - Choose *k* random data items out of the *n* items; call these items the *means*; they designate the prototype or name of the cluster
  - **Refine it iteratively**
    - Associate each of the *n-k* items with one of the ***k* clusters** choosing the **cluster** that it is nearest to**;**
    - **This is called *K*-means clustering**
- **Recall**
  - The "***mean***" is the "average" where you add up all the numbers and then divide by the number of numbers.
  - The "***median***" is the "middle" value in the list of numbers. To find the median, you may have to sort
  - The "***mode***" is the value that occurs most often. If no number is repeated, then there is no mode for the list

 24

USC Viterbi School of Engineering

Different Ways of Clustering the Same Set of Points

(a) Original points. (b) Two clusters.

(c) Four clusters. (d) Six clusters.

K-means clustering critically depends upon the value of k

●●

USC Viterbi School of Engineering

K-Means Clustering Algorithm Mathematical Formulation

(*stated mathematically*)

Given an initial set of k means $m_1^{(1)},\ldots,m_k^{(1)}$, the algorithm proceeds by alternating between two steps:

Assignment step: Assign each observation to the cluster whose mean yields the least within-cluster sum of squares. Since the sum of squares is the squared Euclidean distance, this is intuitively the "nearest" mean

$$S_i^{(t)} = \{x_p : \|x_p - m_i^{(t)}\|^2 \leq \|x_p - m_j^{(t)}\|^2 \ \forall j, 1 \leq j \leq k\},$$

where each x_p is assigned to exactly one $S^{(t)}$, even if it could be assigned to two or more of them.

Update step: Calculate the new means to be the centroids of the observations in the new clusters.

$$m_i^{(t+1)} = \frac{1}{|S_i^{(t)}|} \sum_{x_j \in S_i^{(t)}} x_j$$

- The algorithm has converged when the assignments no longer change.
- The algorithm will converge to a (local) optimum.
- There is no guarantee that the global optimum is found using this algorithm.

USC Viterbi School of Engineering

An Approximation Clustering Algorithm

1. **Select K points as initial centroids**
2. **repeat**
   - form K clusters by assigning each remaining point to its closest centroid
   - re-compute the centroid of each cluster
3. **until centroids do not change or M iterations reached**

- **the algorithm will always terminate, however it does not always find the optimal solution**
- **this is an example of a greedy algorithm**

USC Viterbi School of Engineering

K-Means Depends on Centroids

- **Assumes instances are real-valued vectors**
  - **Let $\vec{x}$ represent the vectors in a cluster *c***
- **Then we define the *centroids, (or center of gravity)*, of the cluster to be the mean of the vectors in the cluster; we write this in the following way**

$$\vec{\mu}(c) = \frac{1}{|c|}\sum_{\vec{x}\in c}\vec{x}$$

- **Reassignment of instances to clusters is based on distance to the current cluster centroids.**

 29

Here is a demo of k-means clustering; this is a copy where you can modify the code (eg. alter the number of clusters, data points, etc.).

••

USC Viterbi School of Engineering

There are Several Possible Distance Metrics

- **Euclidean distance (L_2 norm):**

$$L_2(\vec{x},\vec{y}) = \sqrt{\sum_{i=1}^{m}(x_i - y_i)^2}$$

- **L_1 norm:**

$$L_1(\vec{x},\vec{y}) = \sum_{i=1}^{m}|x_i - y_i|$$

- **Cosine Similarity (transform to a distance by subtracting from 1):**

$$1 - \frac{\vec{x} \bullet \vec{y}}{|\vec{x}| \cdot |\vec{y}|}$$

●●

USC Viterbi School of Engineering

K-means Clustering – Summary Details

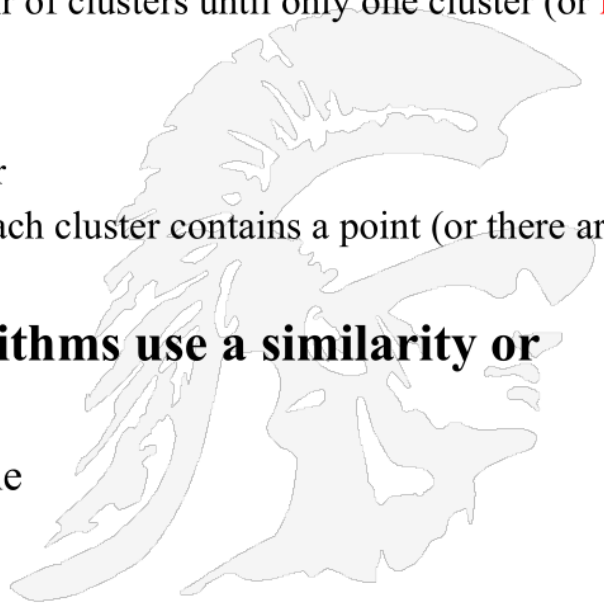
- **Initial centroids are often chosen randomly**
  - Clusters produced vary from one run to another
- **The centroid is (typically) the mean of the points in the cluster**
- **'Closeness' is measured by cosine similarity, a variation of Euclidean distance**
- **Most of the convergence happens in the first few iterations.**
  - Often the stopping condition is changed to 'Until relatively few points change clusters
- **Complexity is *O(i * k * n * m)***
  - n = number of points, k = number of clusters, i = number of iterations, m = number of attributes

●●

USC Viterbi School of Engineering

Hierarchical Clustering Algorithms

- **Two main types of hierarchical clustering**
  - **Agglomerative:**
    - Start with the points as individual clusters
    - At each step, merge the closest pair of clusters until only one cluster (or k clusters) left (bottom-up)
  - **Divisive:**
    - Start with one, all-inclusive cluster
    - At each step, split a cluster until each cluster contains a point (or there are k clusters), (top-down)
- **Traditional hierarchical algorithms use a similarity or distance matrix**
  - Merge or split one cluster at a time



●●

USC Viterbi School of Engineering

How Can We Compute the Distance Between Two Clusters

- As before, the **Centroid** of a cluster is the component-wise average of the vectors in a cluster, which is itself a vector
- Example, the Centroid of (1,2,3); (4,5,6); (7,2,6); is (4,3,5)
- **4 possible ways to compute the distance between two clusters**

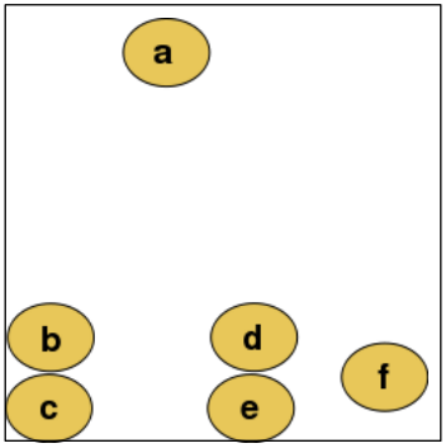
1. **Center of Gravity**
   - Compute the distance between the two centroids of the cluster
2. **Average Link**
   - Compute the average distance between all pairs of points across the two clusters
3. **Single Link**
   - Compute the distance between the two closest points in the two clusters, i.e. the most cosine similar
4. **Complete Link**
   - Compute the distance between the furthest points in the two clusters, i.e. the least cosine similar

●●

USC Viterbi School of Engineering

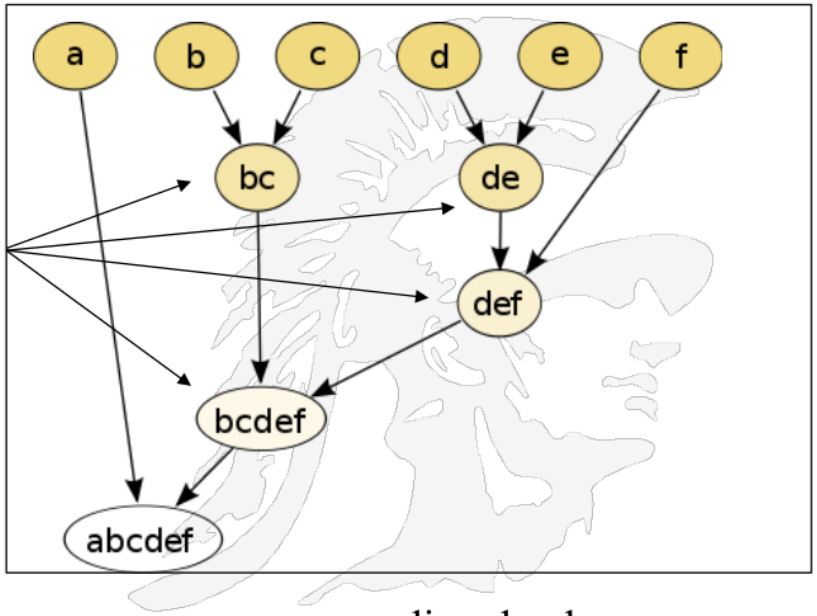
A Dendrogram is Used to Display Clusters

- A **dendrogram** is a tree diagram frequently used to illustrate the arrangement of the clusters produced by hierarchical clustering



original input

clusters



corresponding dendrogram

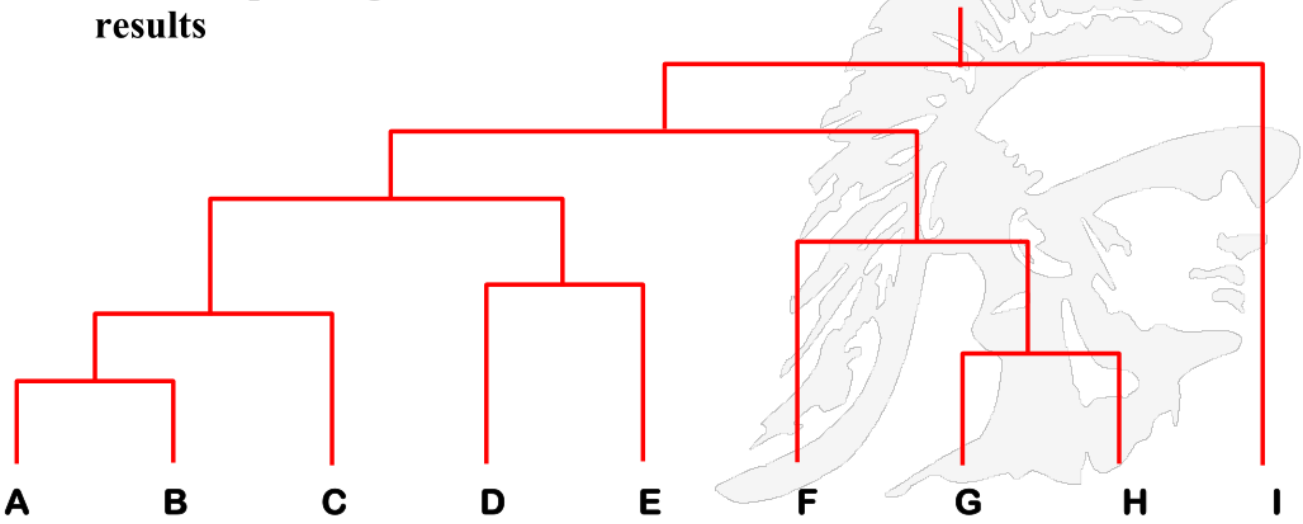
second row clusters are: {a}, {b c}, {d e} {f}
third row clusters are: {a}, {b c} {d e f}

●●

USC Viterbi School of Engineering

Hierarchical Agglomerative Clustering

- HAC starts with unclustered data and performs successive pairwise joins among items (or previous clusters) to form larger ones
  - this results in a hierarchy of clusters which can be viewed as a dendrogram
  - Dendrograms are usually drawn as shown below
  - The height of an edge can sometimes refer to the degree of similarity
  - useful in pruning search in a clustered item set, or in browsing clustering results



●●

USC Viterbi School of Engineering

Divisive Clustering Algorithm

1. Start at the top with all documents in one cluster.
2. The cluster is split using a partitioning clustering algorithm.
   - Use the k-means clustering algorithm, which is linear in computing time whereas HAC (hierarchical agglomerative clustering) algorithms are quadratic
3. Apply the procedure recursively until each document is in its own singleton cluster

- Studies show that the divisive algorithms produce more accurate hierarchies than bottom up
  - Bottom-up methods make clustering decisions based on local patterns without initially taking into account the global distribution. These early decisions cannot be undone.
  - Top-down clustering benefits from complete information about the global distribution when making top-level partitioning decisions.

 54