IFN509

Data Exploration and Mining

Week 7

Algorithms of Descriptive Data Mining (Unsupervised Learning) Clustering

Professor Richi Nayak
School of Computer Science
Centre for Data Science
Faculty of Science
https://research.qut.edu.au/adm

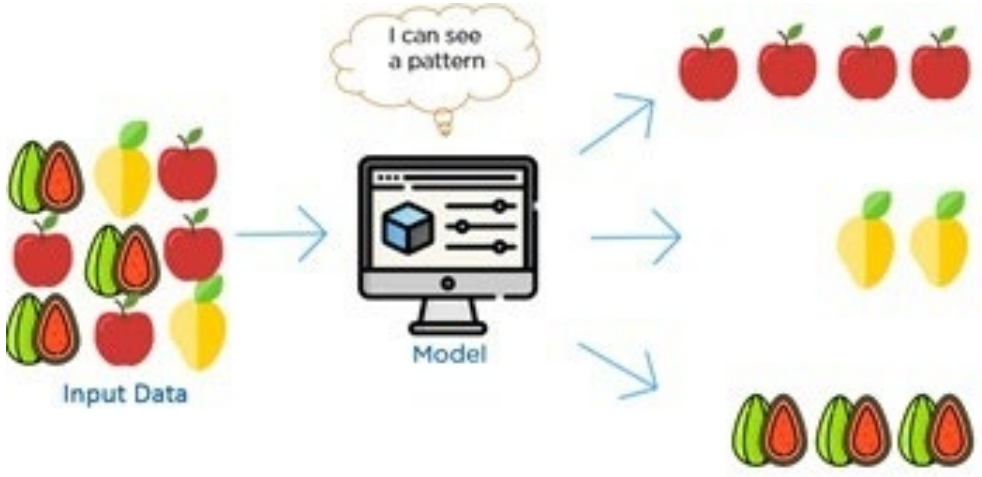
Week 7 Learning

- Lecture: Clustering
  - Clustering Basic
    - What is Clustering?
    - Clustering Centroid
    - Proximity Measures
  - Clustering Algorithms
    - K-means
    - Hierarchical
- Practical & Tutorial – Week 9
  - Part 1 - Reflective Pen-and-Paper exercises
    - Clustering: proximity measures and finding clustering solutions
  - Part 2 - Practical Exercises
    - Data processing for clustering
    - Finding clustering solutions
      - With k-means
      - With agglomerative clustering
      - With k-prototypes
    - Profiling clustering solutions

What Should You Do in Week 7?

- Review the **Clustering** lecture slides and reading materials.
- Be ready for the lab marking component of Assessment Item 1
- Consult the Lecturer/tutor for queries related to the subject.
- Assessment Item 2
  - The project report is due at the end of week 13 (June 4). An online quiz based on this assignment will be held on June 6 at 2 pm.
  - I will navigate through the assignment tasks in the Week 9 lecture.
  - What should you do?
    - You are allowed to form a new team if required, otherwise, continue with your current group.
    - Register your team (old or new) on Canvas by Week 8. No team movements will be allowed after Week 9.
    - You should **finish the association mining task by week 8**.

Clustering: Introduction



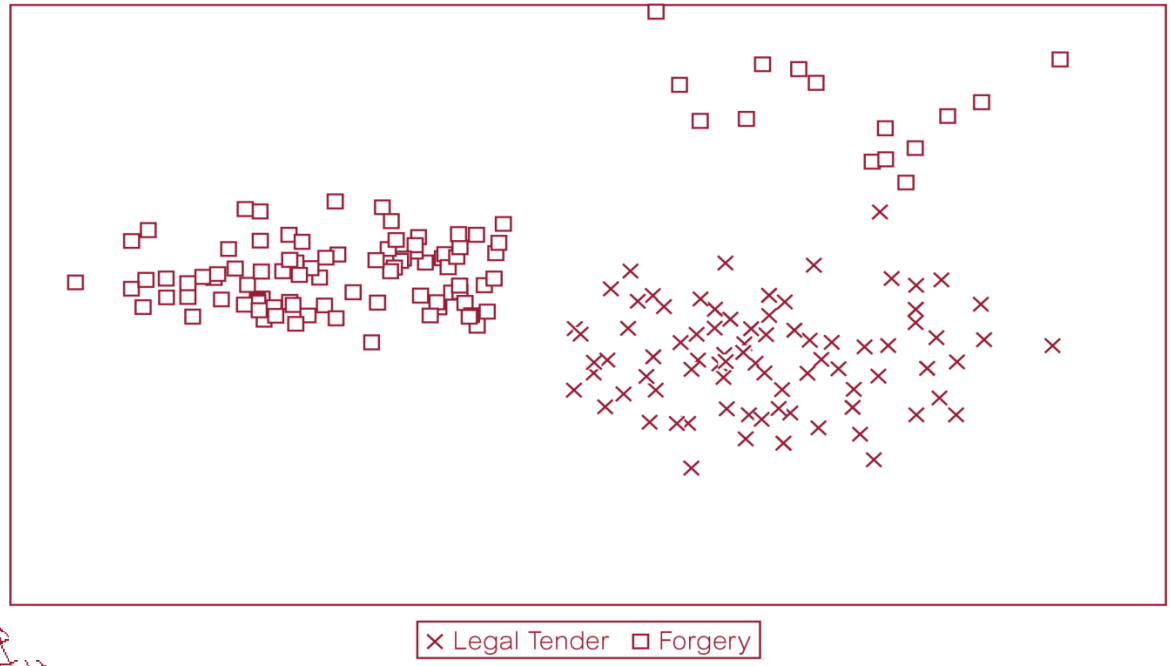
Clustering Applications

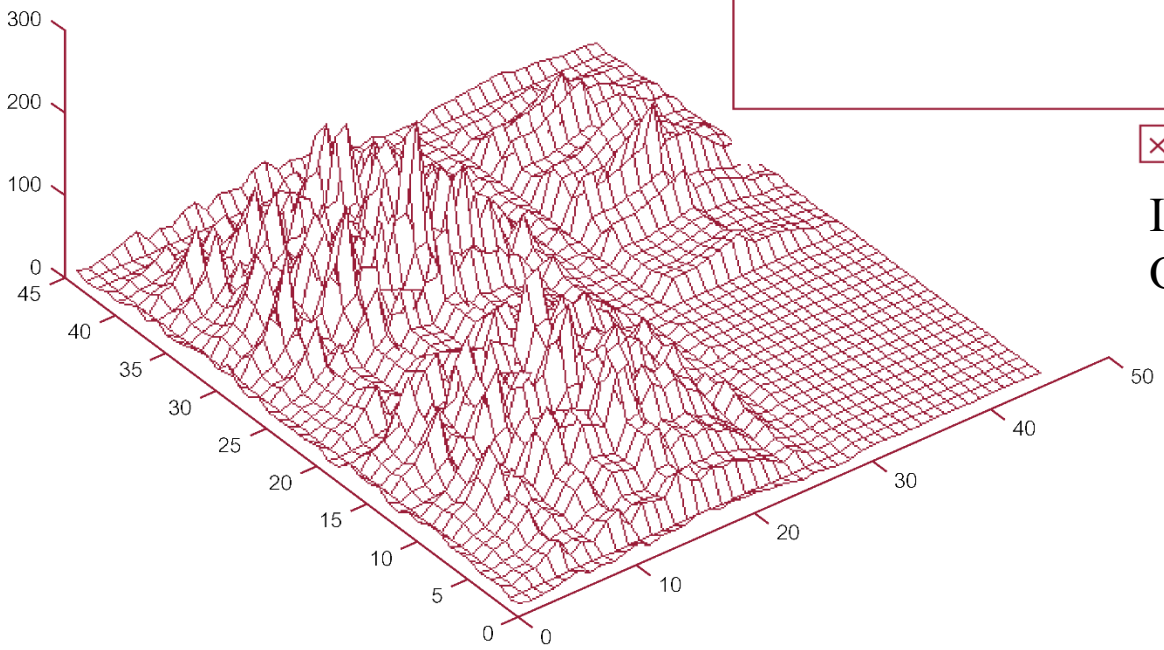
- Typical applications
  - Pattern Recognition - as a stand-alone tool to get insight into data distribution
  - As a preprocessing step for other algorithms
- Examples: customer profiling, fraud detection, direct marketing, document directory.
  - Identify micro-markets and develop policies for each.

Clustering Applications …(2)

- The Web or Internet or Social Networks
  - Document classification
  - Cluster Weblog data to discover groups of similar access patterns
  - Identify interesting groups in a customer base/online (social networks, Google etc) that may not have been recognised before.
  - Community detection on social networks
- Image Mining
  - Finding similar visual objects
- Spatial Data Analysis
  - create thematic maps in GIS by clustering feature spaces
  - detect spatial clusters and explain them in spatial data mining

Data Clustering and visualization using a Scatterplot: An Example

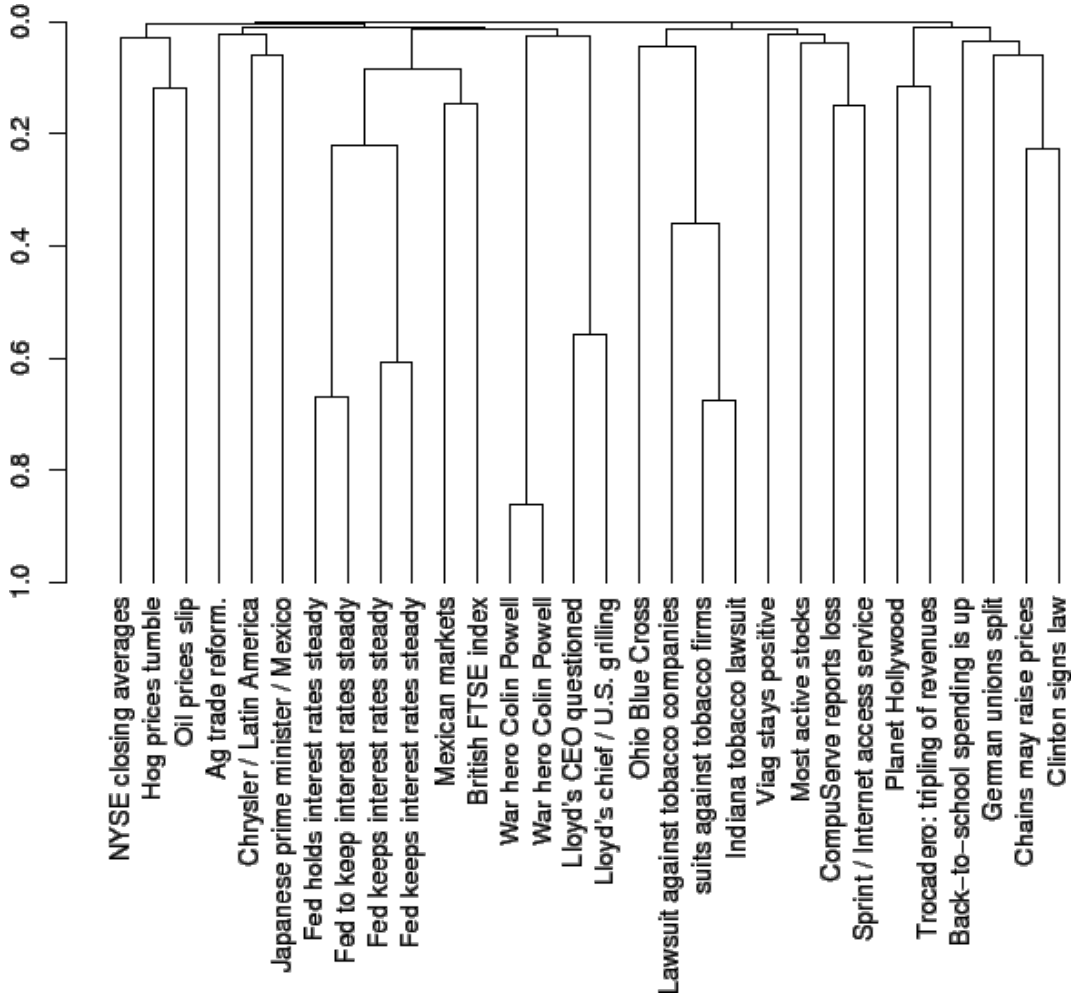




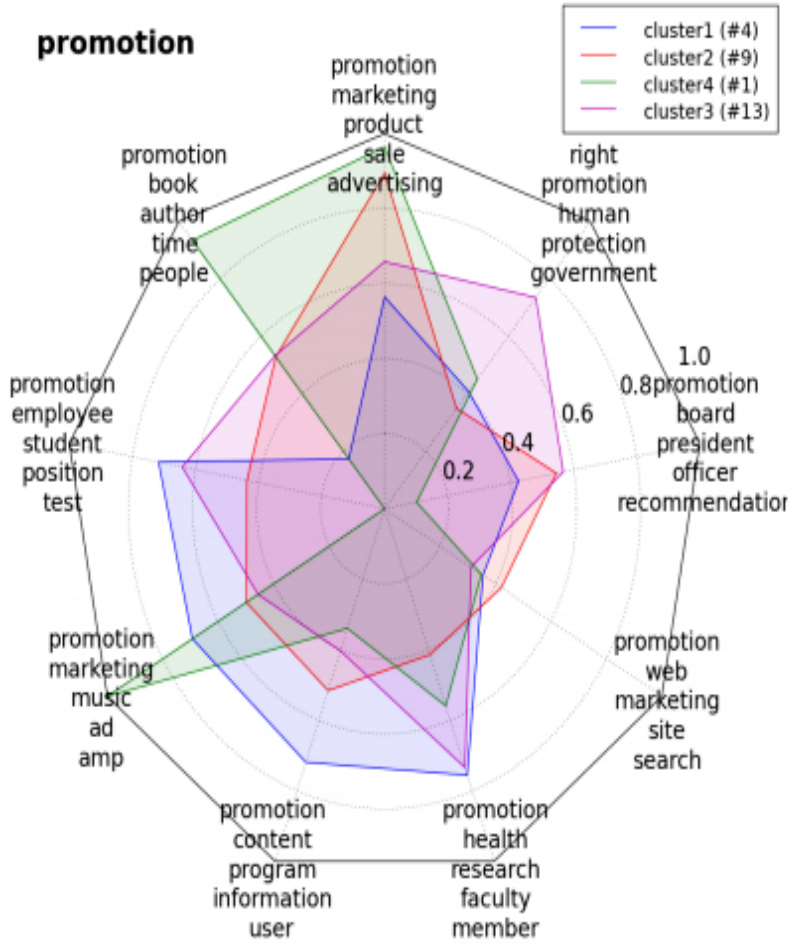
Input: 200 instances with 8 features
Output: 3 clusters

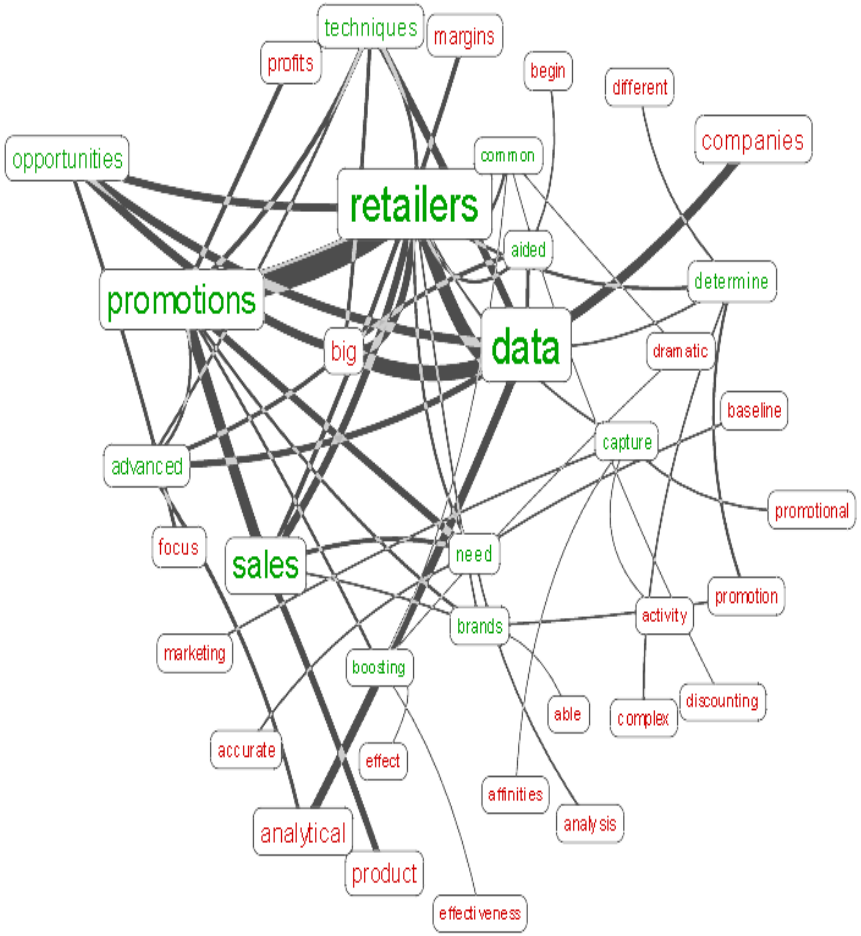
Application of text clustering: Example 1

- Organize document collections
  - Automatically identify hierarchical/topical relations among documents



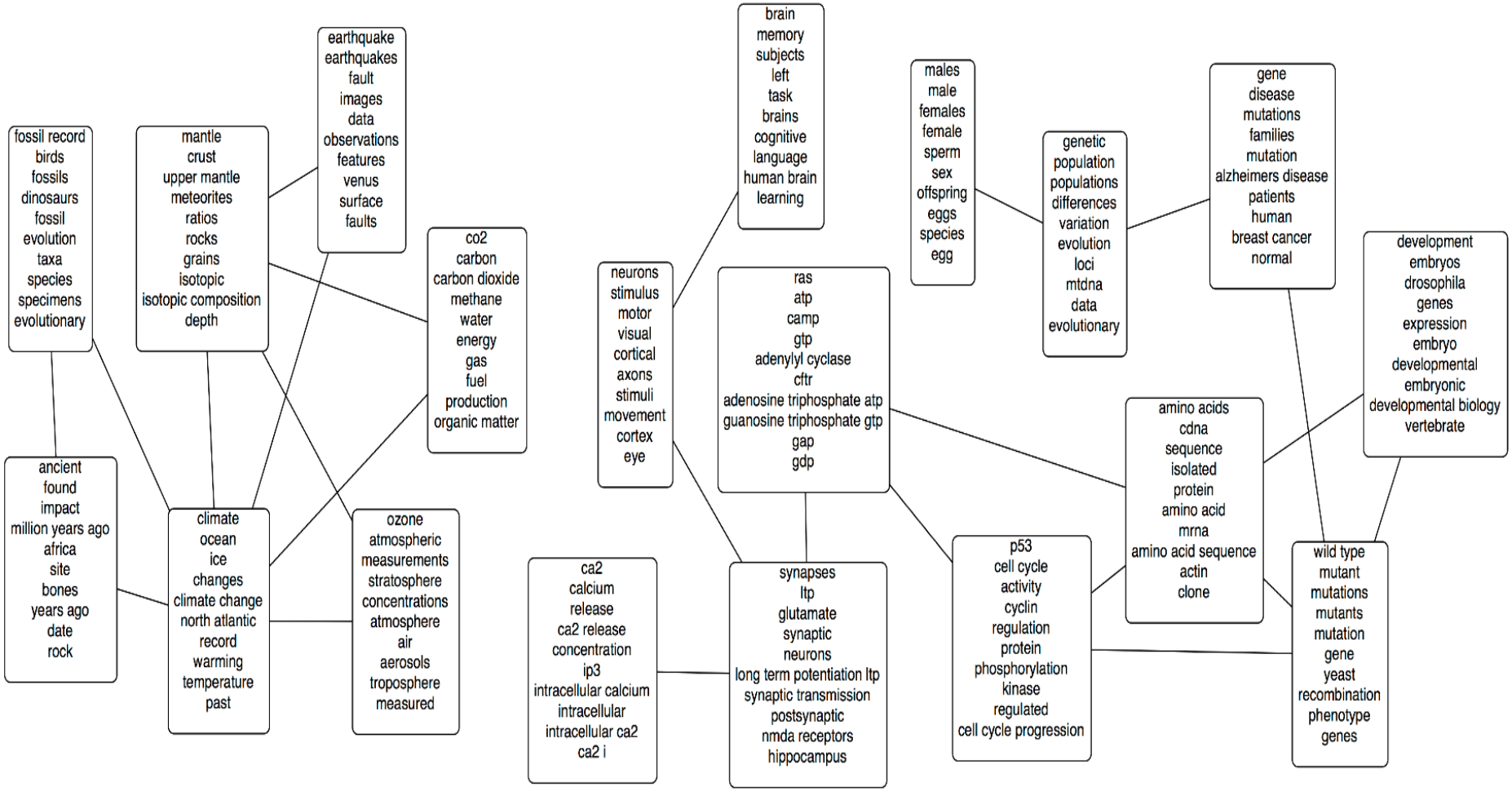
Application of text clustering: Example 2





Application of text clustering: Example 3

- Topic modeling: Grouping words into topics



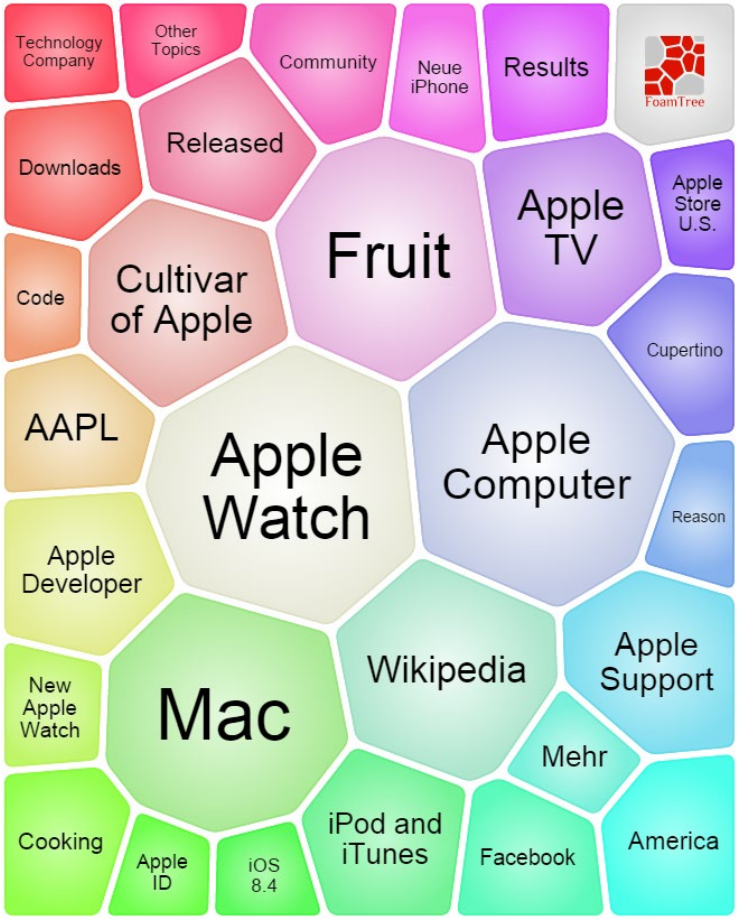
Application of text clustering: Example 4

- Grouping search results
  - Organize documents by topics
  - Facilitate user browsing

http://search.carrot2.org/stable/search

"Carrot^2 organizes search results into topics."

"With an instant overview of what's available, it is easier to look up."



Clustering: Process and Proximity Measures

Clustering Definition

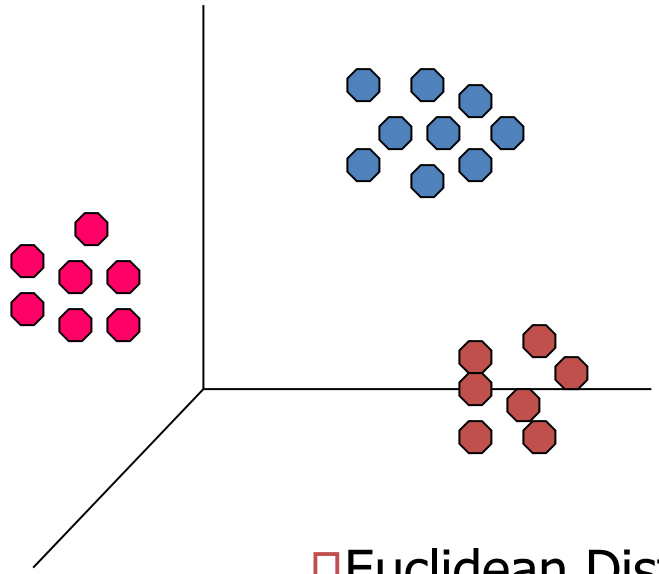
- Given a set of data points (or observations), each having a set of attributes, and a similarity measure among them, find clusters such that
  - Data points in one cluster are more similar to one another.
  - Data points in separate clusters are less similar to one another.
  - A cluster is a collection of data objects.
- Key requirement: A good measure of similarity between data points.
  - A metric or distance measure if attributes are continuous.
  - Other problem-specific Measures.
    - Cosine similarity for text documents

Basic Steps to perform Clustering

- Data processing and Feature selection
- Instances Comparison (using a Proximity measure)
  - Similarity/distance between two feature vectors (or data points)
- Clustering criterion
  - Expressed via a cost function or some rules
- Assigning Instances into groups
- Validation of the results
  - Measure of accuracy

Clustering Criteria

- Objects are similar to one another within the same cluster (measured as intra-cluster similarity)
- Objects are dissimilar to the objects in other clusters (measured as inter cluster similarity)
- Aim is to **Maximise intra-cluster similarity and minimise inter-cluster similarity**



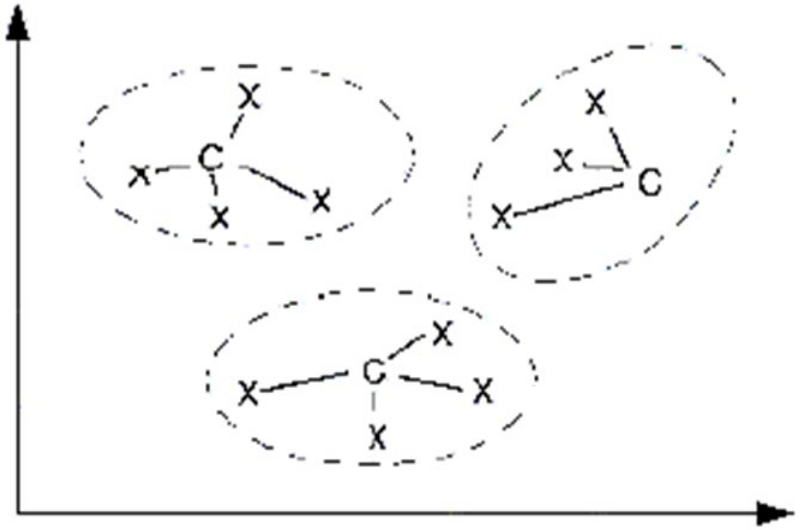
Euclidean Distance Based Clustering in 3-D space.

Proximity Measure

- How to find if data points are similar or not?
- Two common measures
- Distance based
  - assume metric space on attributes
  - K-means algorithm, Kohonen feature map neural networks
- Model based
  - estimate a density (mixture of gaussians)
  - Expectation Maximization (EM) algorithm, DBSCAN, OPTICS

Proximity (Distance Based) Measure

- This is typically measured as **distance** between each pair of points or between a point and the cluster centres within a multi-dimensional space,
  - where each dimension represents one of the variables being compared.
- Clusters are typically based around a "**centre**" value.
  - How centres are initially defined and adjusted vary between algorithms.



C = cluster centre or average values
X = instances to be clustered
—— = a distance measure

Distance measure

- Simplest case: one numeric attribute A
  - Simple Distance (X,Y) = A(X) – A(Y)
    - Let X and Y be two Instances with a single attribute
  - Euclidean Distance (X, Y) = $((A(X) - A(Y))^2)^{1/2}$
  - Dot product = A(X) * A(Y)
- Several numeric attributes:
  - Distance (X,Y) = Euclidean distance between each attribute of X and Y

$$Dist(X,Y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

Euclidean Distance

(X_2, Y_2)

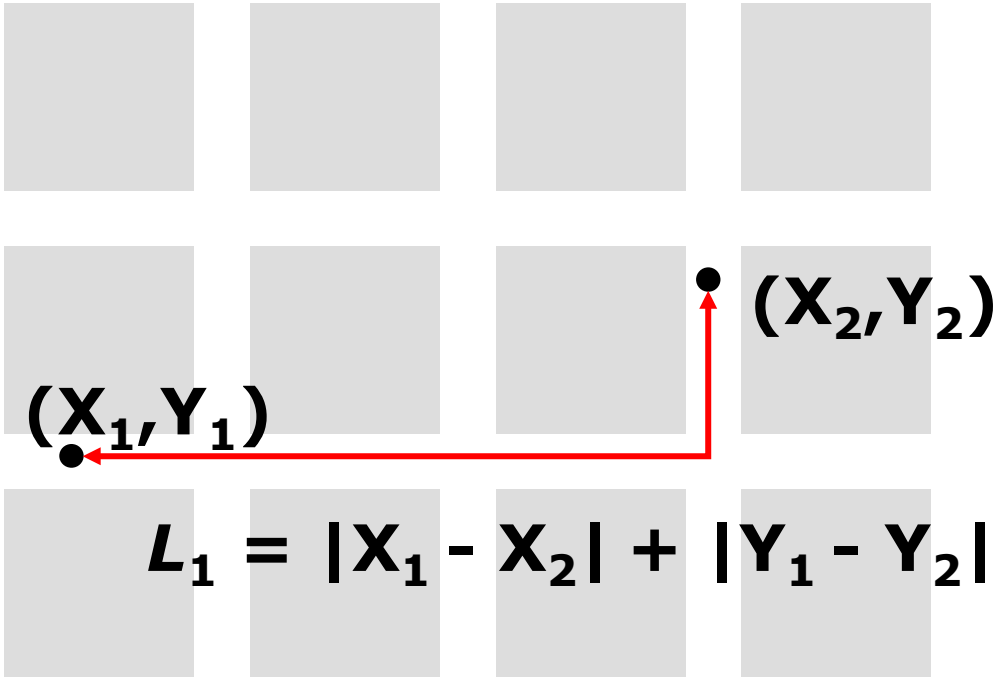
(X_1, Y_1)

$$L_2 = ((X_1 - X_2)^2 + (Y_1 - Y_2)^2)^{1/2}$$

- The Euclidean distance between two points is the length of the straight line that joins them.

* Clusters using this distance tend to be **spherical** in nature.

Manhattan Distance



* The Manhatton distance between two points is the length of the shortest axis-parallel connection between them.
* Clusters using this distance tend to be **cubical** in nature.
* Clusters using this distance are relatively **insensitive to outliers**.

Example: Distance between Samples

Dot product

Sim(S1, S2) = 1*1+0*1=**1**

Sim(S1, S3) = 1*0 + 1.*0 = **0**

Distance = (1 – sim)

Euclidean distance

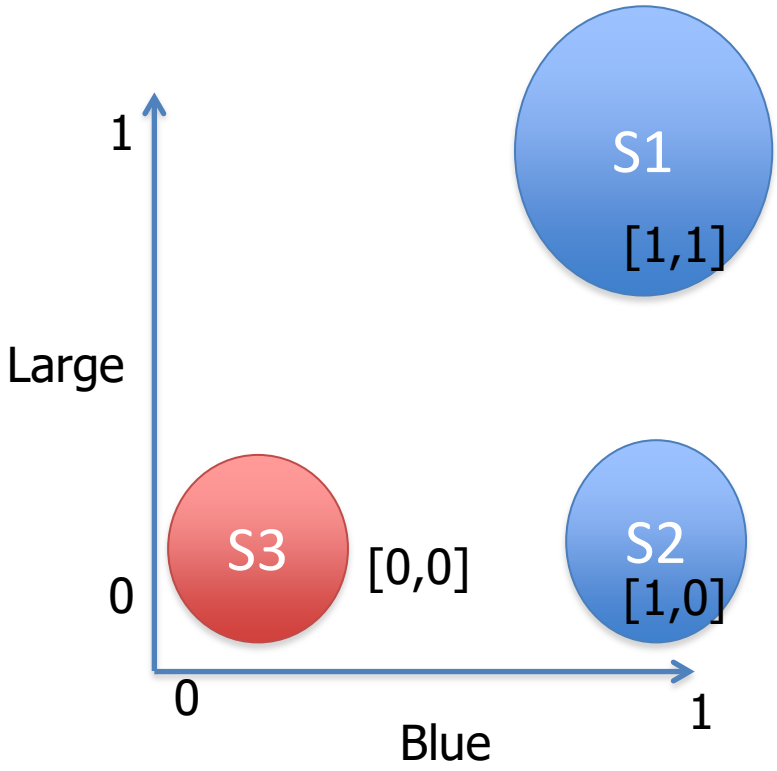
Dist(S1, S2) = $\sqrt{((1-1)^2+(1-0)^2)}$=sqrt(1)=**1**

Dist(S1, S3) = $\sqrt{((1-0)^2 + (1-0)^2)}$= $\sqrt{2}\approx$**1.4**

Manhattan distance

Dist(S1, S2) = |1-1|+|1-0| =**1**

Dist(S1, S3) = |1-0| + |1-0|= **2**



Minkowski Distance

$$d(\mathbf{x}, \mathbf{y}) = \left(\sum_{k=1}^{n} |x_k - y_k|^r \right)^{1/r}$$

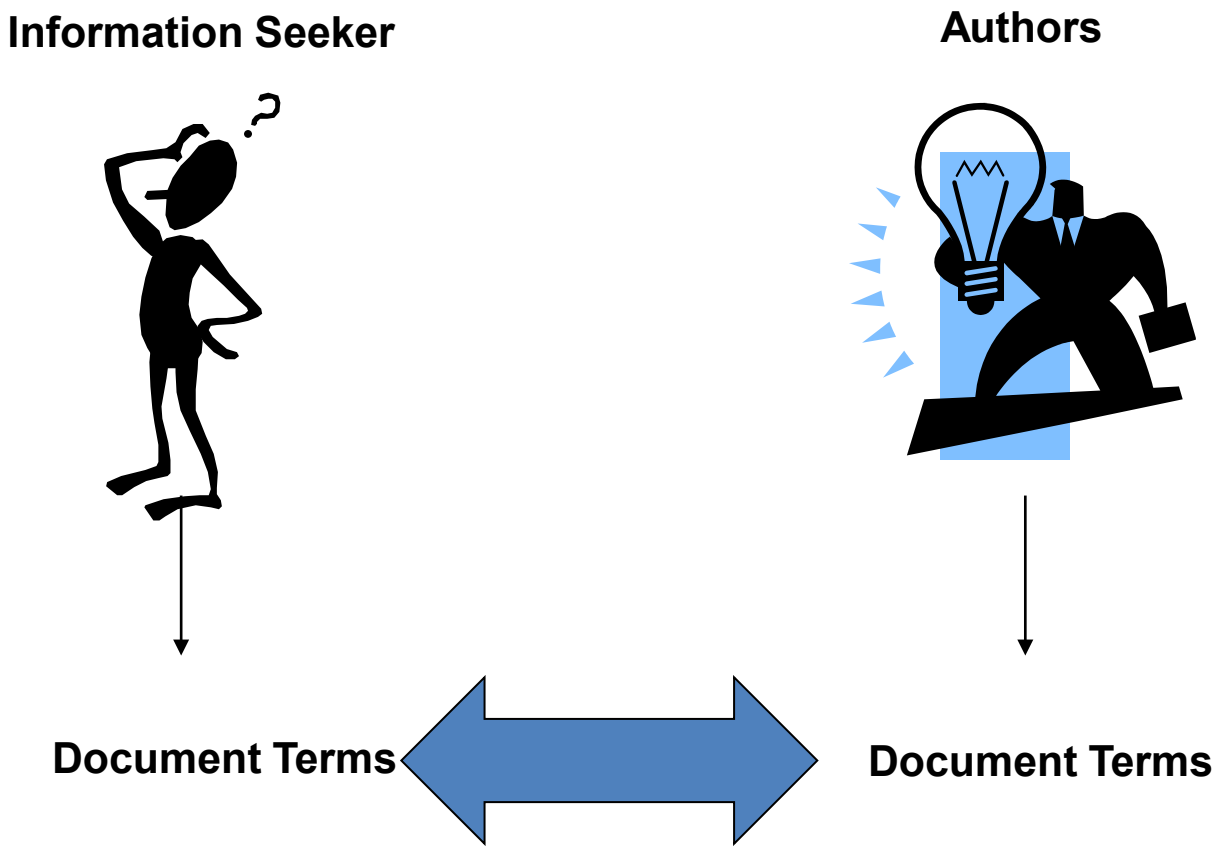
n is the number of dimensions (attributes) in the dataset

- *r* = 1. City block (Manhattan, taxicab, L_1 norm) distance.
  - A common example of this for binary vectors is the Hamming distance, which is just the number of bits that are different between two binary vectors
- *r* = 2. Euclidean distance (L_2 norm)
- $r \to \infty$. "supremum" (L_{max} norm, L_∞ norm) distance.
  - This is the maximum difference between any component of the vectors

Distance Measures: Categorical Data

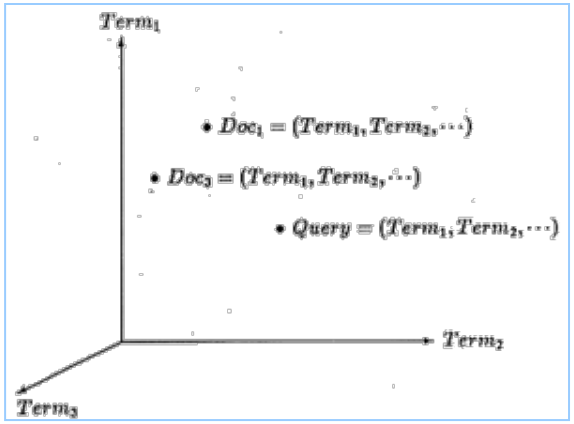
- Categorical Data
  - Is it sensible to measure the distance between male and female, or between MIT and BIT ?
  - Convert all categorical data into numeric data
    - Does this make any sense?
  - Treat distance between categorical values as a function that has values only 1 and 0
- Can we compare two objects that have all categorical attributes?
  - Essentially, counting how many attribute values are the same.
  - distance is set to 1 if values are different, 0 if they are equal

Text Data: Document Similarity or Query Processing



Text Similarity Computation

- Document → feature vector = ($Term_1$, $Term_2$,…,)
- Query → feature vector = ($Term_1$, $Term_2$,…,)
- Documents and queries can both be represented by feature vectors



- Similarity Computation
  - Cosine (Doc1, Doc2) or Cosine (Doc1, Query)

Distance measures: summary

- Numeric data: Minkowski Distance
- Categorical data: 0/1 to indicate presence/absence followed by
  - Hamming distance (# dissimilarity)
  - Jaccard coefficients: #similarity in 1s/(# of 1s)
- Combined numeric and categorical data
  - weighted normalized distance
- Text or Image (vector) data
  - Cosine similarity

Distance Measures: Problems

- Scales of different attribute values
  - Example: GPA varies 0-4, Age varies 18-60
  - Total Distance (Sum of the distance between individual attribute values) will be dominated by the age attribute and will be affected very little by the variations in GPA
  - Need to **normalise or standardise**
    - E.g. All data varies from 0 to 1 or data to have a mean of zero and a standard deviation of 1
- Missing values
  - Assumed to be maximally distant (given normalized attributes)
- Are all attributes equally important?
  - Weighting the attributes might be necessary

Distance Based Clustering Methods

Two Major Distance-based Clustering Methods

- Partitioning approach:
  - determine various partitions, guess the central point in each cluster, assign points to the cluster of their nearest centroid, and then evaluate clusters by some criterion, e.g., minimizing the sum of square errors
  - Common methods: k-means, k-medoids, CLARANS
- Hierarchical approach:
  - Create a hierarchical decomposition of the set of data (or objects) using some criterion
  - Agglomerative: a sequence of partitions in which repeatedly merge nearby clusters
  - Divisive: a sequence of all objects in one partition in which repeatedly divide the cluster into smaller pieces
  - Common methods: Diana, Agnes, BIRCH, ROCK, CAMELEON

Distance Based Clustering Methods: Partitional Algorithm: K-means

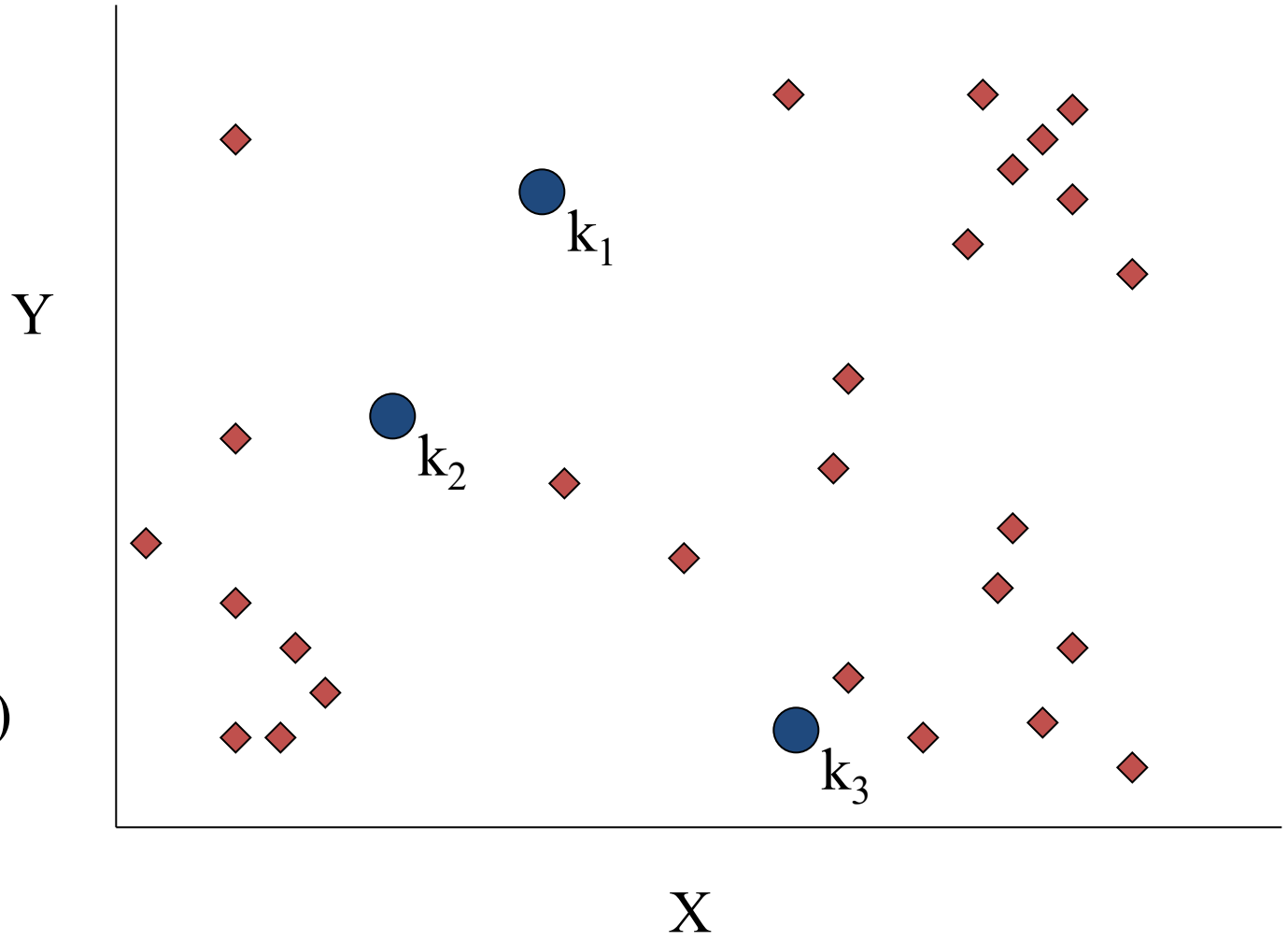
Partitioning or Centroid Approach: k-Means Clustering

1. Pick k random points as the initial cluster centres (or farthest k points)
2. Repeat
   - Assign each point to the cluster to which the point is closest, based on the mean value of the objects in the cluster;
   - Update the cluster means, i.e., calculate the mean value of the objects for each cluster;
4. Until no change;

✓ Guaranteed to give locally optimum answer,

✓ Most commonly used

K-means example, step 1

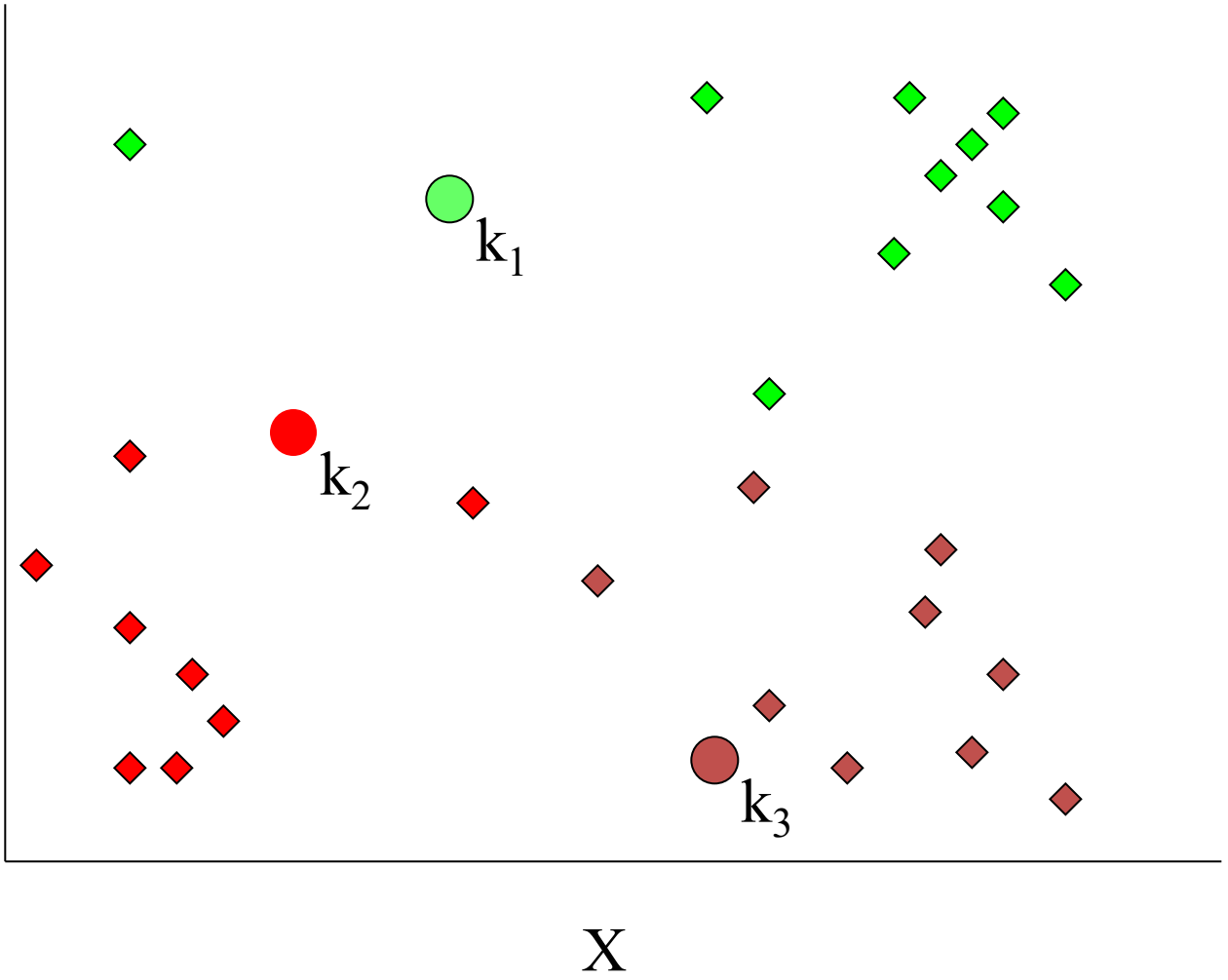


Pick 3 initial cluster centers (randomly)

Note that these initial center points can also be selected from data points.

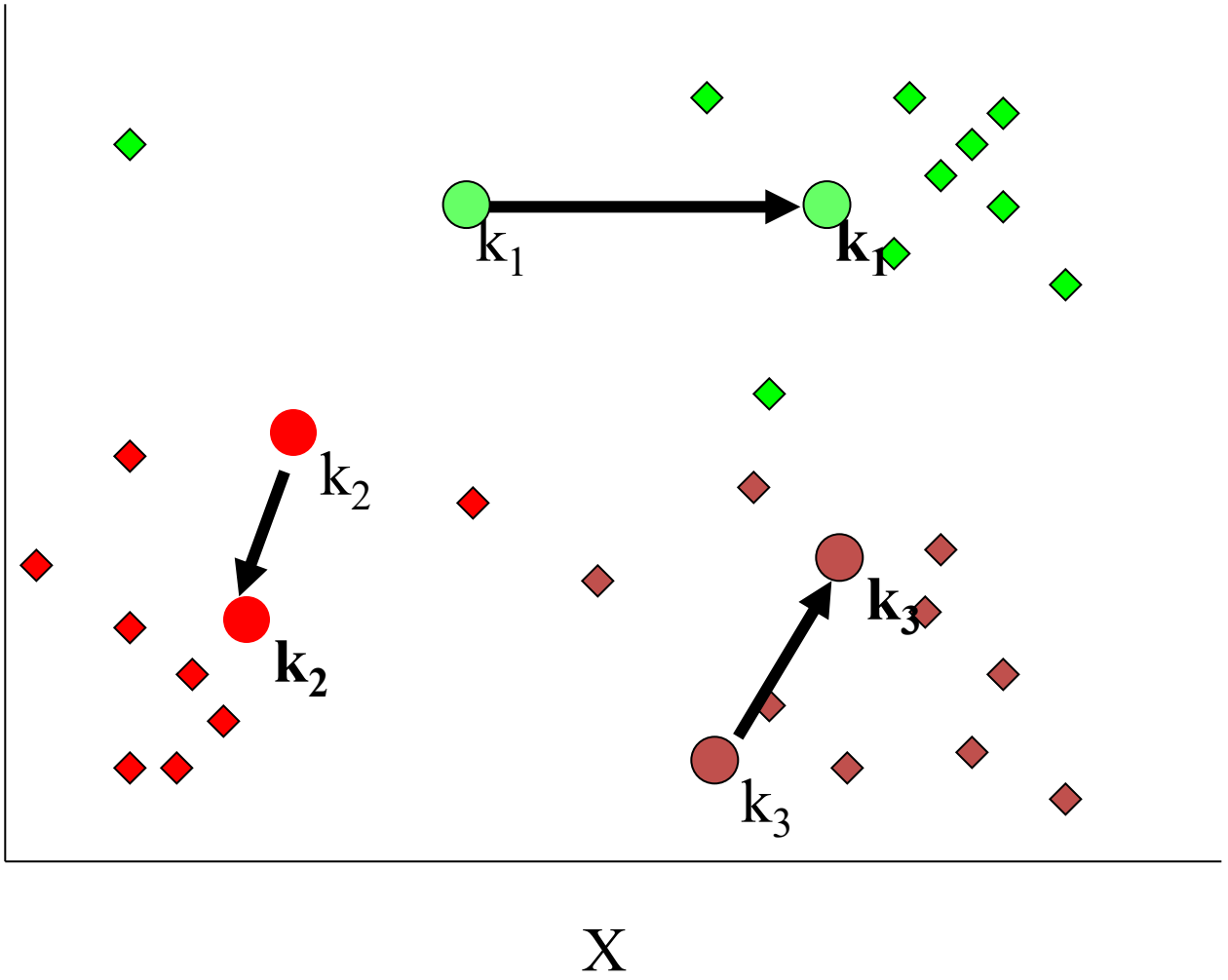
K-means example, step 2

Assign each point to the closest cluster center



K-means example, step 3

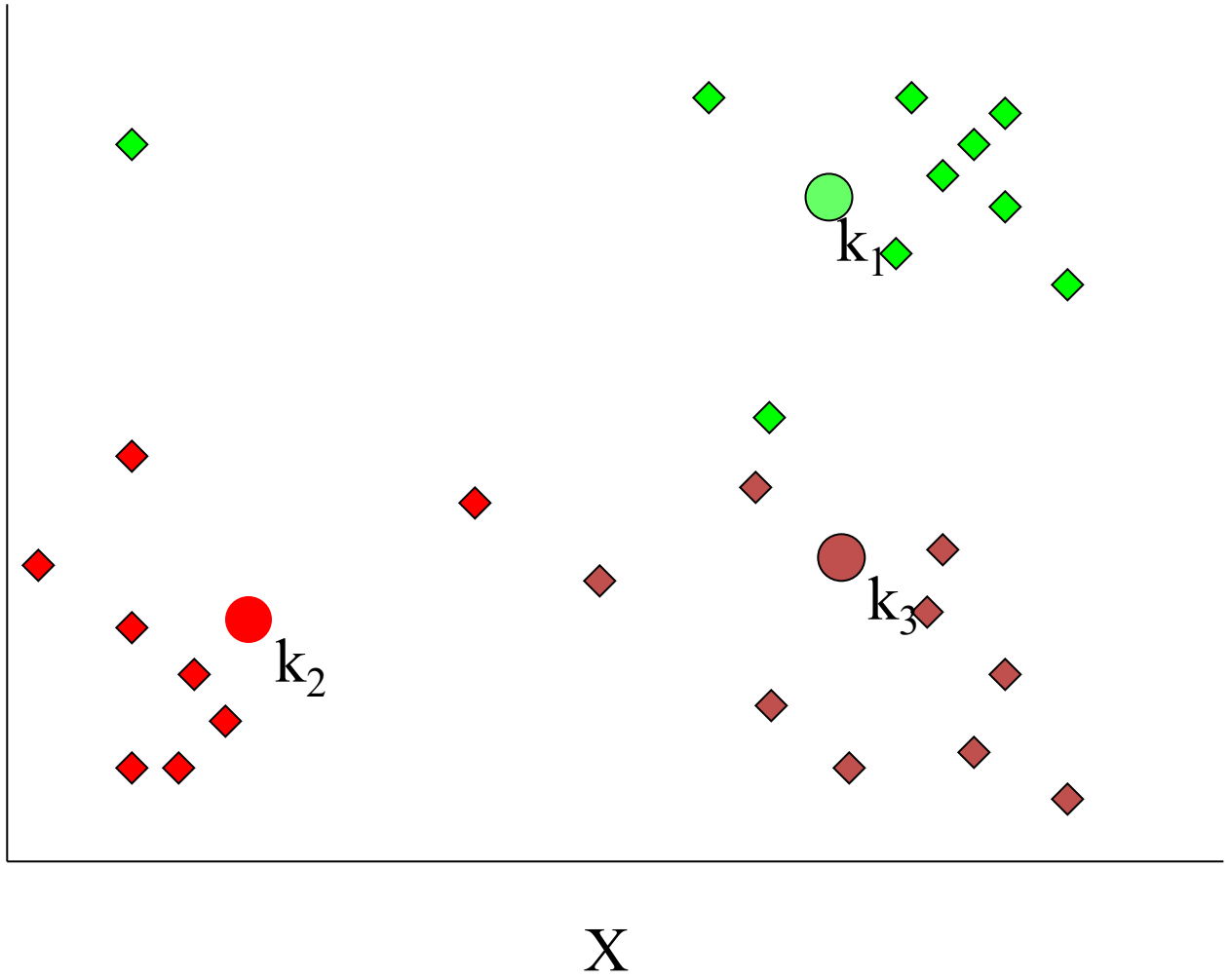
Move each cluster center to the mean of each cluster



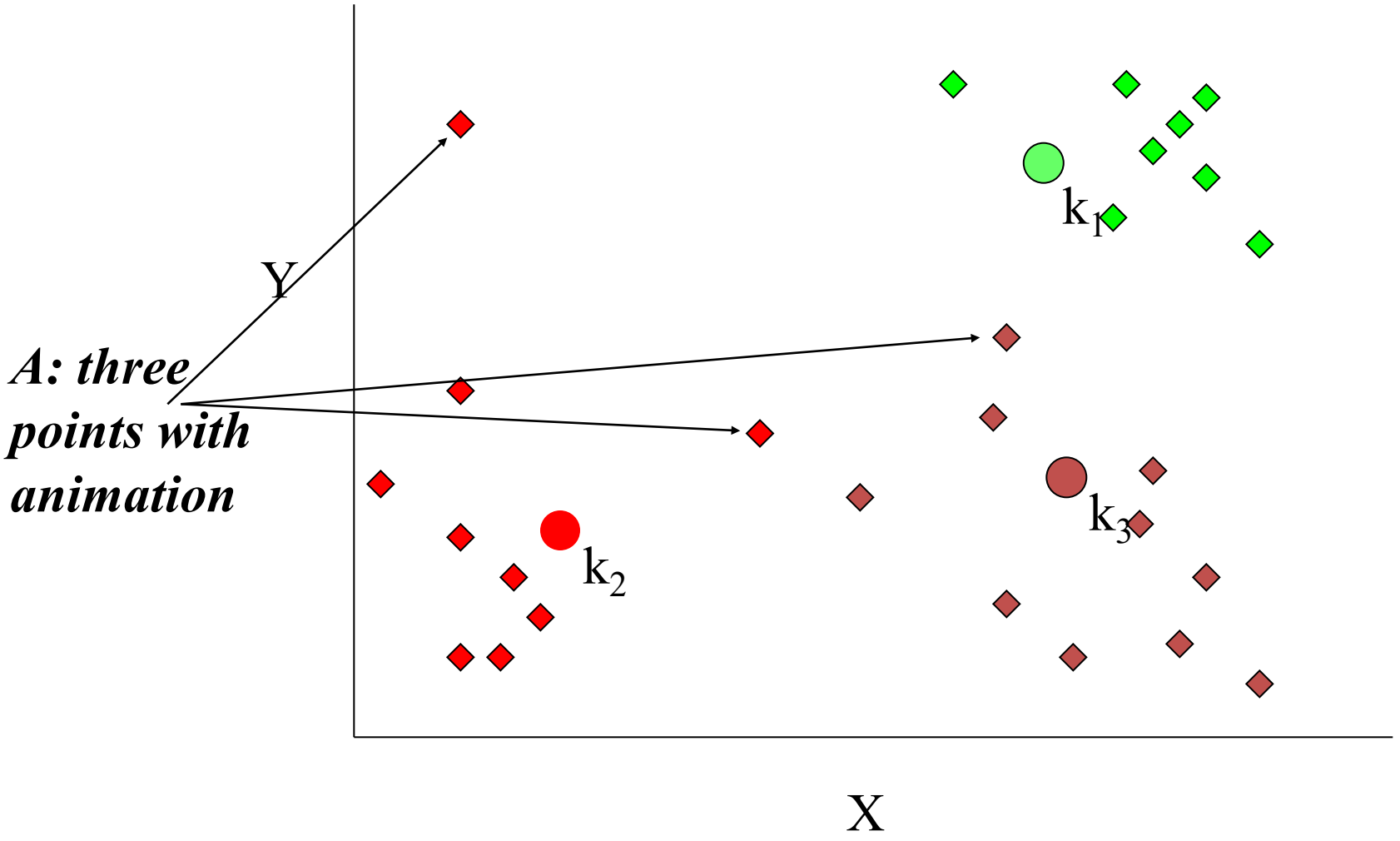
K-means example, step 4

Reassign points closest to a different new cluster center

Q: Which points are reassigned?

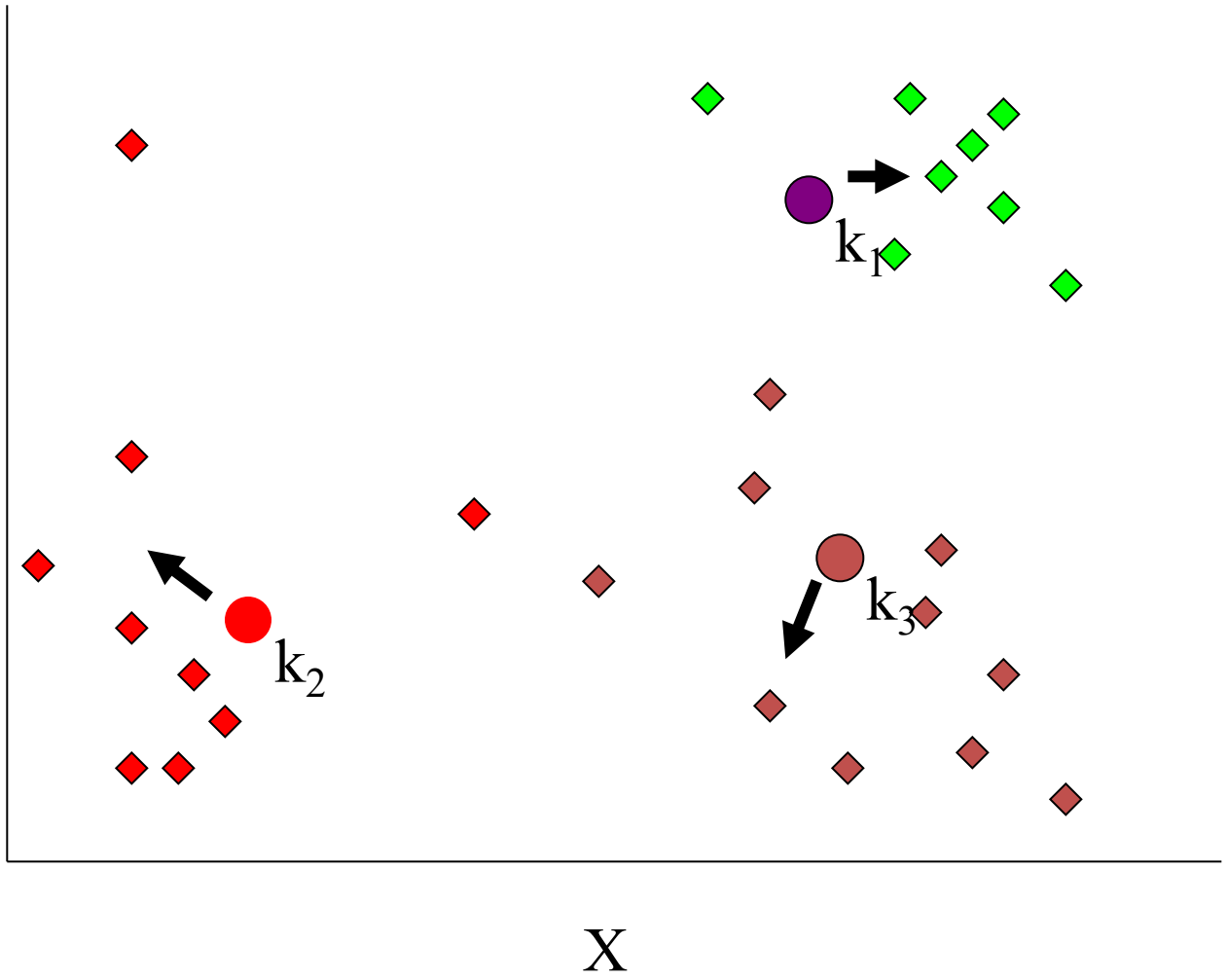


K-means example, step 4 …

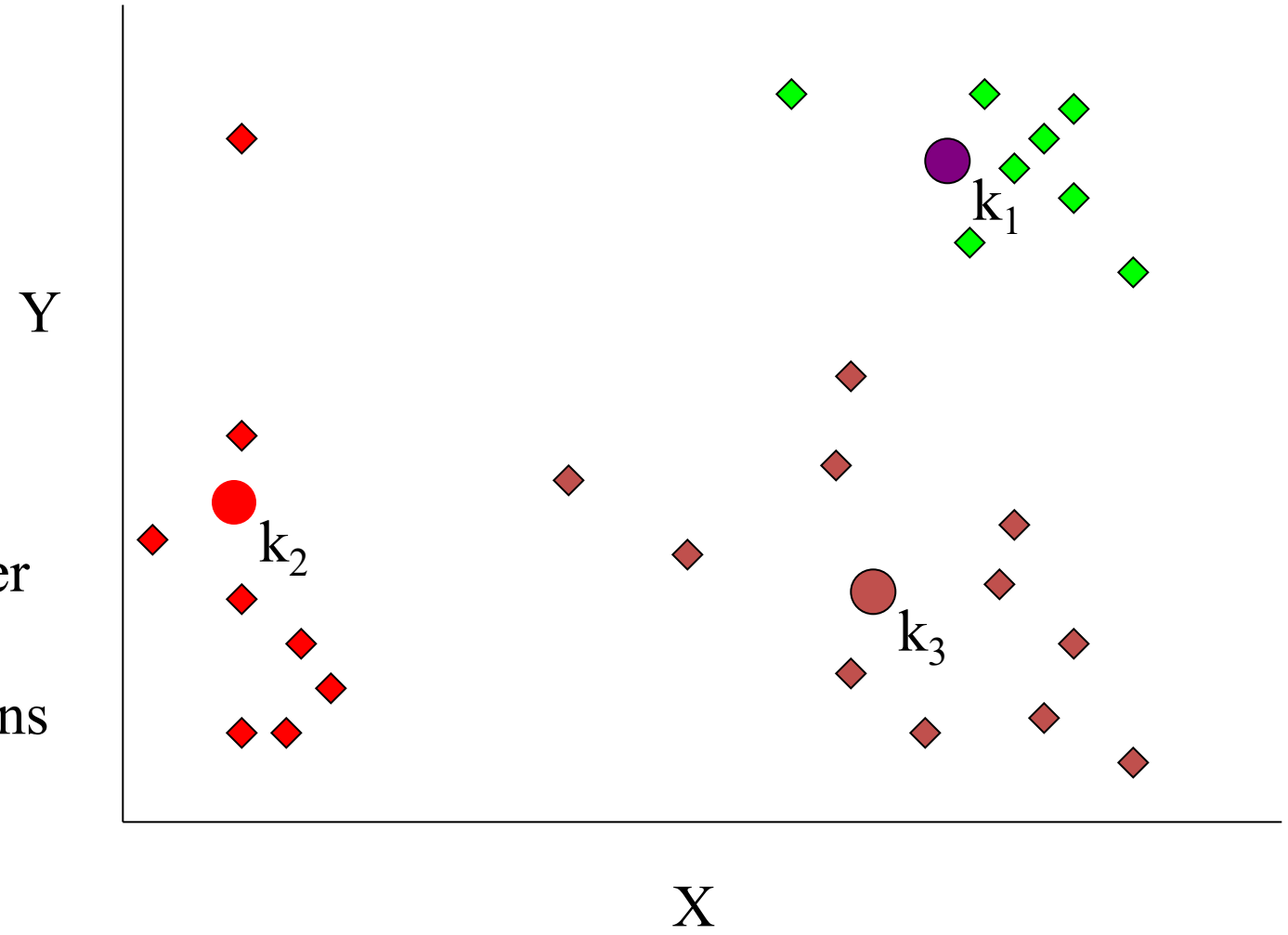


K-means example, step 4b

re-compute cluster means

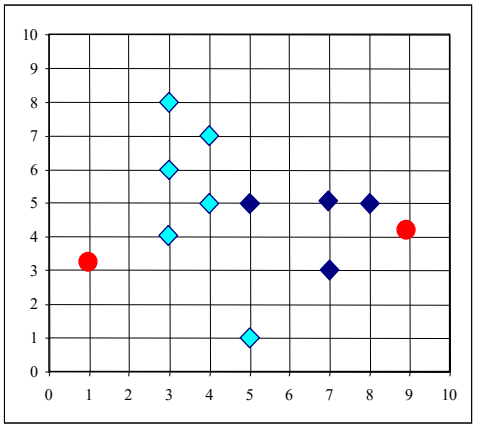


K-means example, step 5



move cluster centers to cluster means

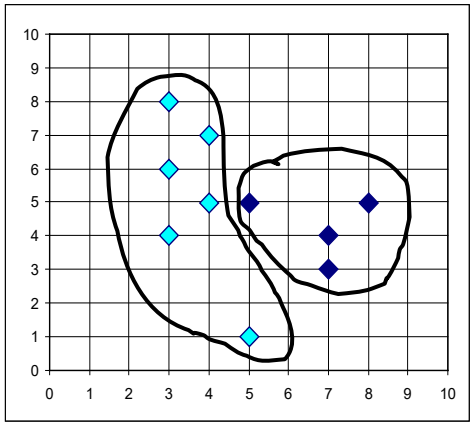
Another k-means example



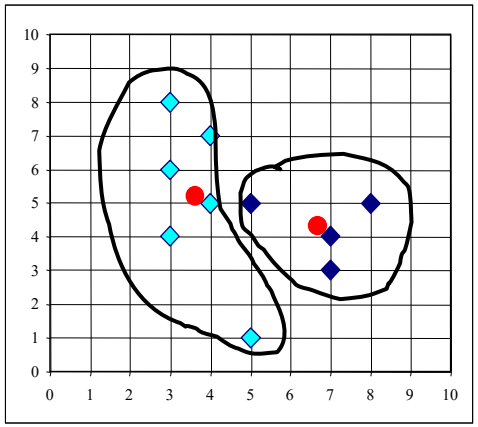
K=2

Arbitrarily choose K objects as initial cluster centers

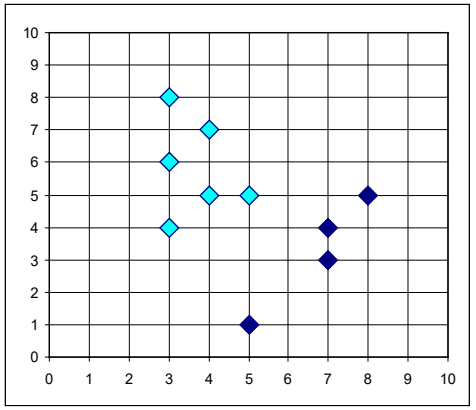
Assign each object to most similar centroid



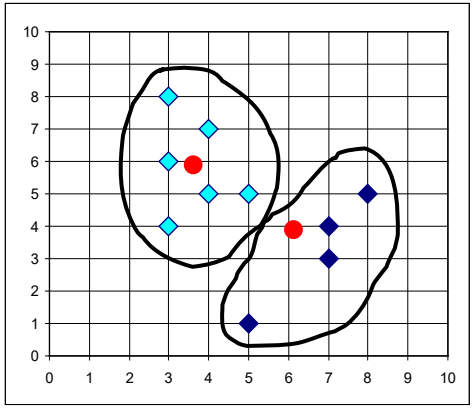
Update the cluster means



reassign



Update the cluster means



reassign

K-Means: Remarks

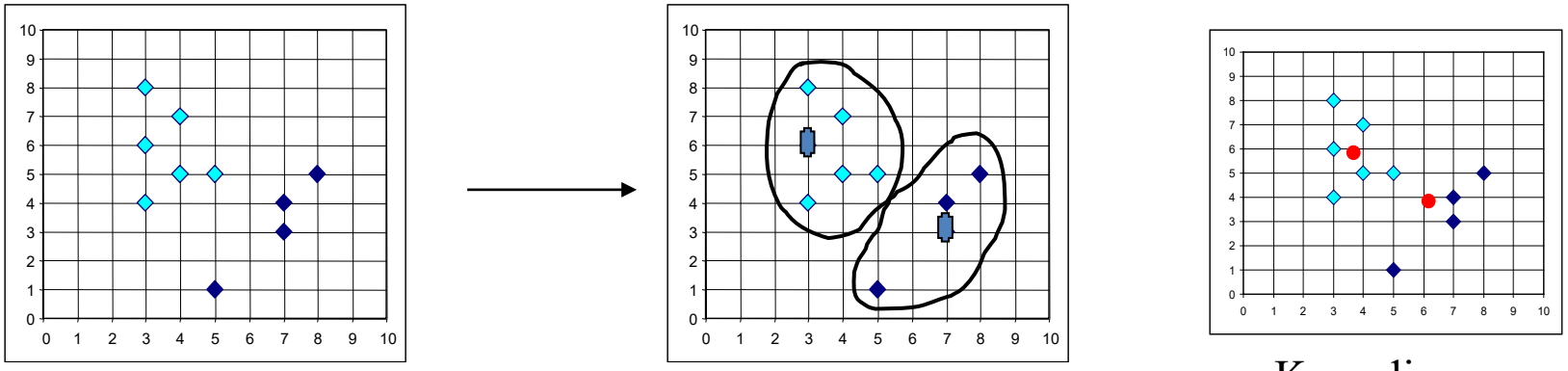
- Strength
  - *Efficient*: $O(tkn)$, where n is # objects, k is # clusters, and t is # iterations. Normally, $k, t << n$.
  - Comment: Often terminates at a *local optimal*
- Weakness
  - Applicable only to objects in a continuous n-dimensional space
  - Sensitive to noisy data and *outliers*
  - Need to specify *k,* the *number* of clusters, in advance (there are ways to automatically determine the best k)

Variations of *K-Means*

- Handling categorical data: *k-modes*
  - Replacing means of clusters with modes: most common value for each attribute
  - Using new dissimilarity measures to deal with categorical objects
- Handling mixed data: *k-prototype*
  - a simple combination of K-Means and K-Modes
- Most of the variants of *k-means* differ in
  - Selection of the initial *k* centroids (eg. Start with random instances rather than centroids)
  - Strategies to calculate cluster means
  - Distance measures (e.g. *k-median*)

K-median algorithm

- The k-means algorithm is sensitive to outliers
  - Since an object with an extremely large value may substantially distort the distribution of the data (e.g., 1, 3, 5, 7, 1009)
- K-Medoids: Instead of taking the **mean** value of the object in a cluster as a reference point, **medoids** can be used, which is the **most centrally located** object in a cluster

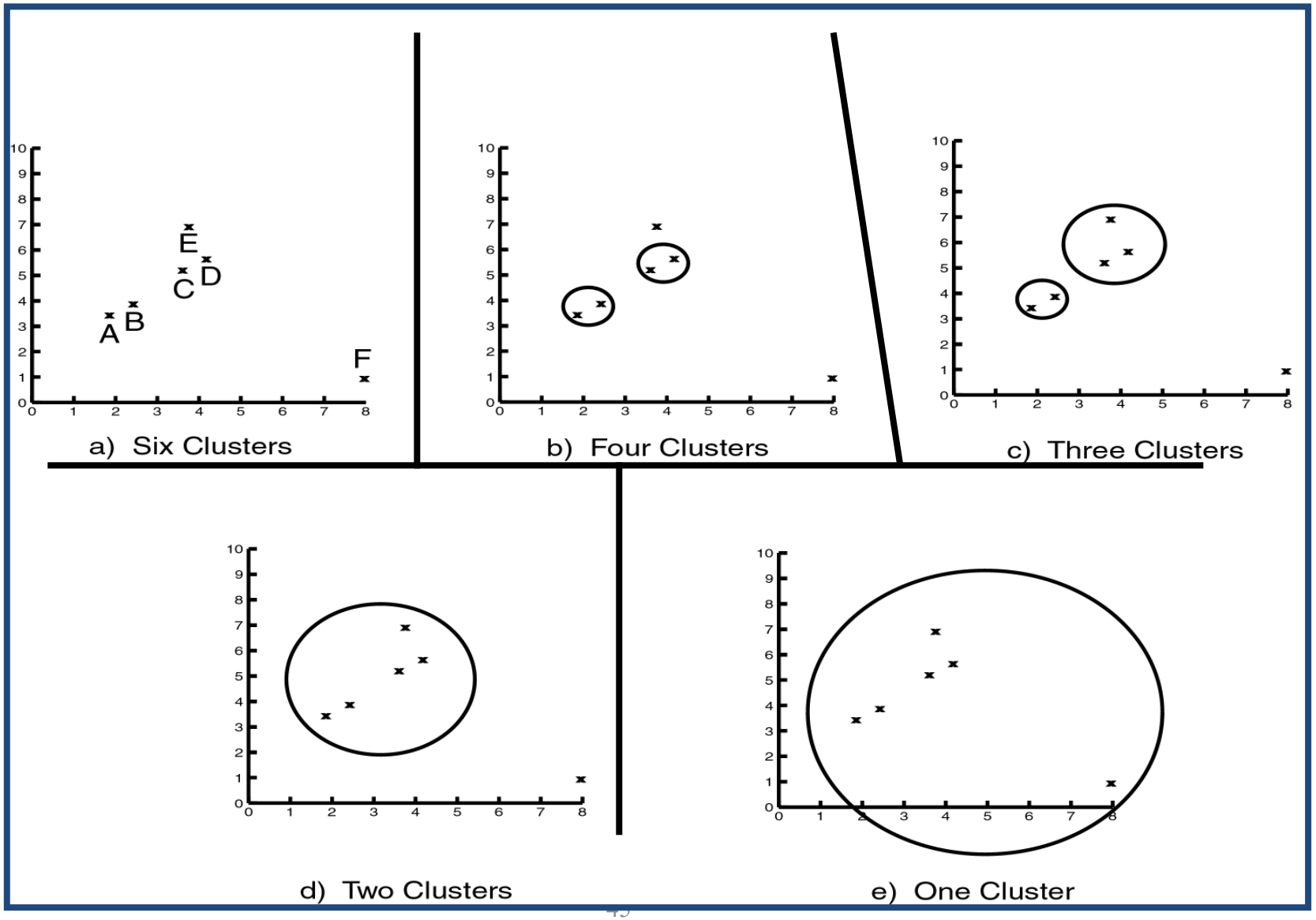


Distance Based Clustering Methods: Finding Cluster Numbers in K-means family of algorithms

Finding Number of Clusters (k)

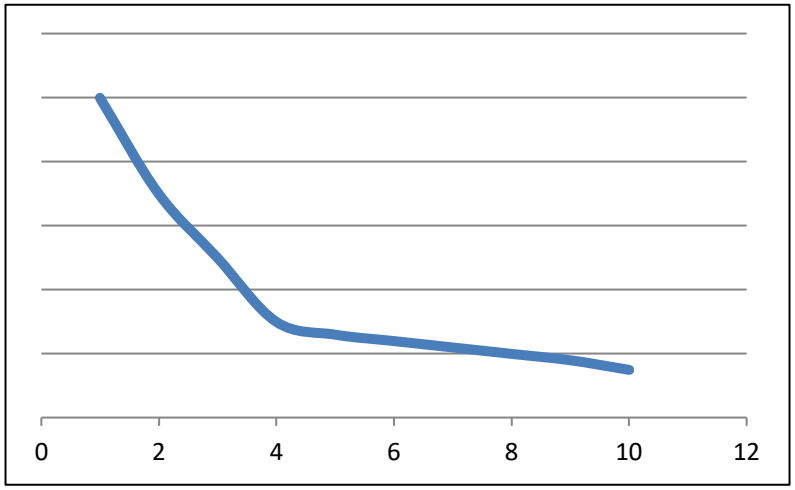
- What is the optimal number ? An open issue
  - Not always easy to determine how many clusters to expect from a data set
  - Maybe based on some domain knowledge
- A different number of clusters can give different results
  - One solution may be better than other.
- Many clustering techniques require the number of clusters to be generated as an input and/or use a default number

Finding Numbers of Clusters



Determine the Number of Clusters

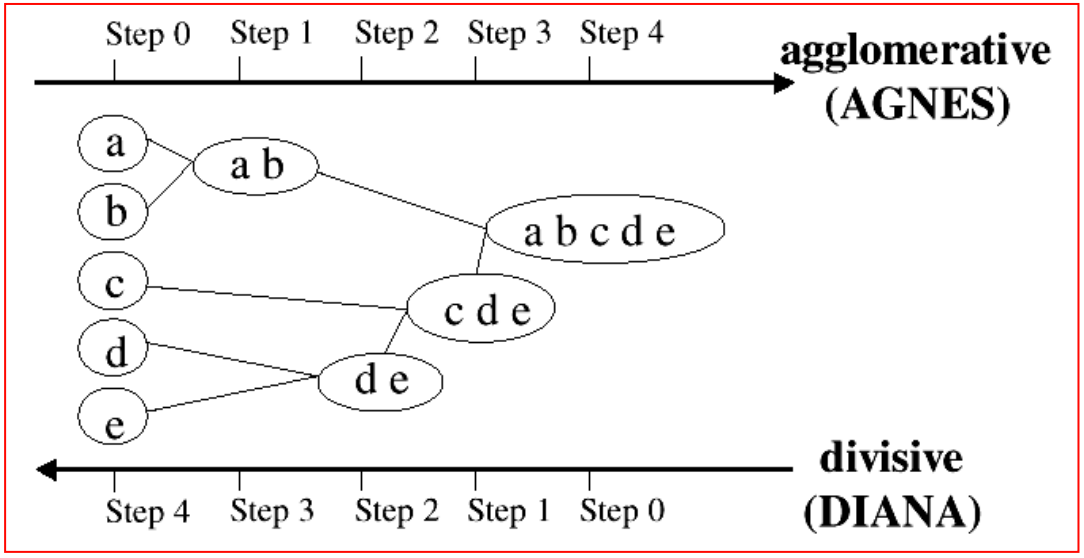
- Empirical method
  - # of clusters ≈√n/2 for a dataset of n objects
- Silhouette coefficient or Silhouette score
  - Calculates the goodness of a clustering solution using intra-cluster and inter-cluster distances.
  - Ranges from -1 to 1. 1: Means clusters are well apart from each other and clearly distinguished. 0 means overlapping clusters.
- Elbow method
  - Use the turning point in the curve of sum of within cluster variance w.r.t the # of clusters



Distance Based Clustering Methods: Hierarchical Methods

Hierarchical clustering

- Organizes instances in a tree of clusters.
- Popular because of the flexibility of the number of clusters (and multi-level visualization and access)
- Use the distance matrix as clustering criteria
  - does not require the number of clusters *k* as an input, but needs a termination condition



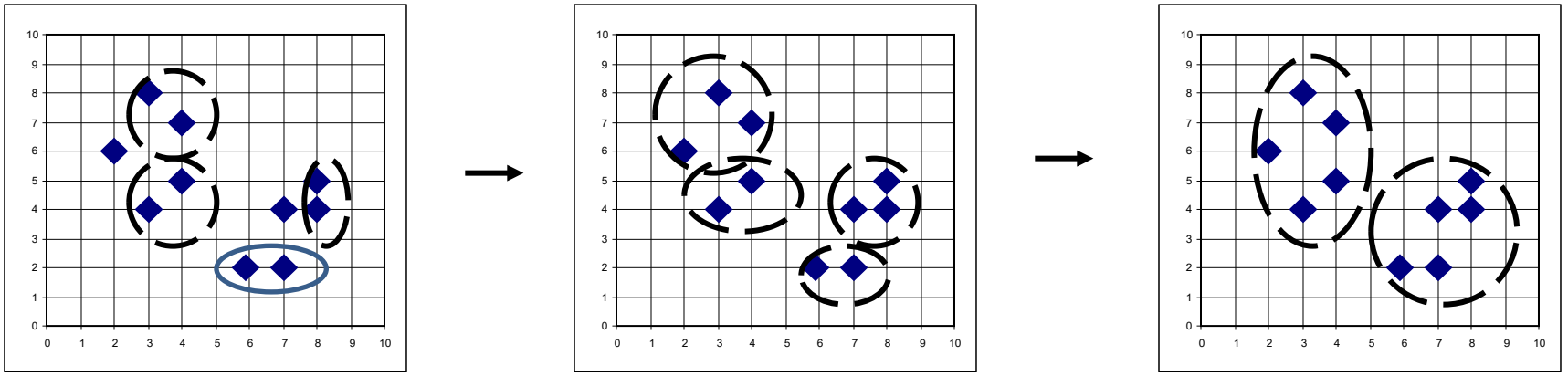
Agglomerative Nesting: Agnes

General Idea:

Step 1 : Find the 2 objects most similar to each other and connect them

Step 2: Replace the 2 objects by a tuple of their "centre"
(centroid, medoid or mode)

Repeat Steps 1 and 2 until
the distance between the most similar objects is above a threshold

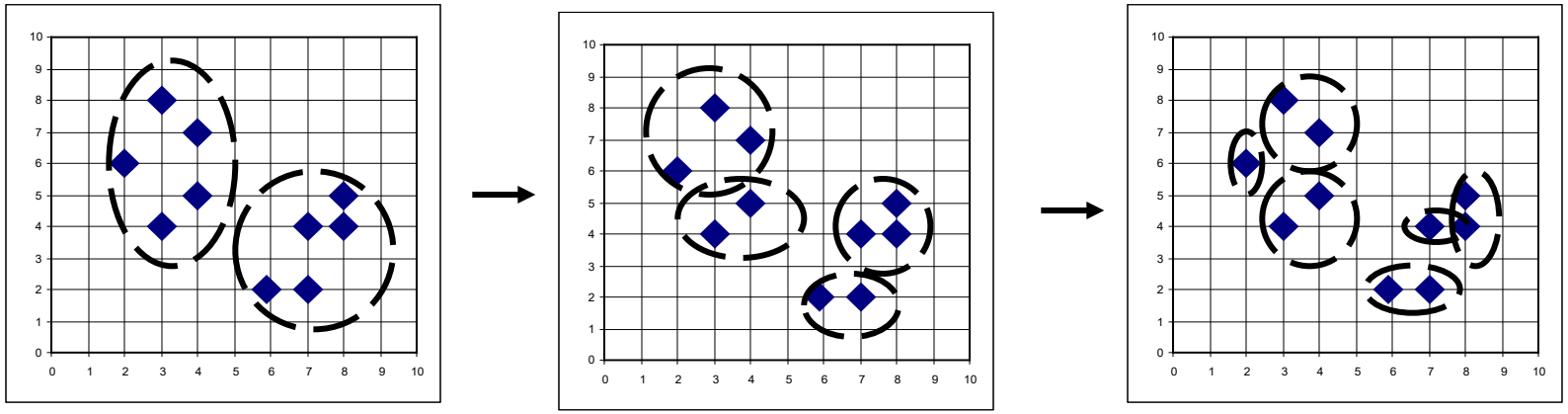


Divisive Analysis: DIANA

General Idea (Inverse order of AGNES):

Step 1 : All objects form a cluster
Step 2: Find the objects most similar to each other and split them into another cluster
Repeat Step 2 until
the distance between the most similar objects is above a threshold

- Eventually each node in the hierarchy forms a cluster on its own



Hierarchical Clustering

- The output can be a number of clusters or the cluster tree.

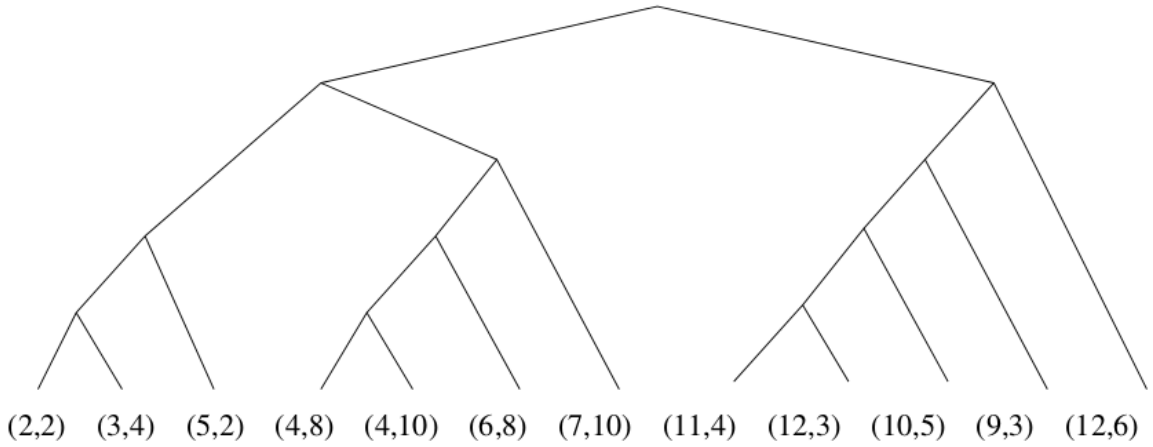


Figure 7.6: Tree showing the complete grouping of the points of Fig. 7.2

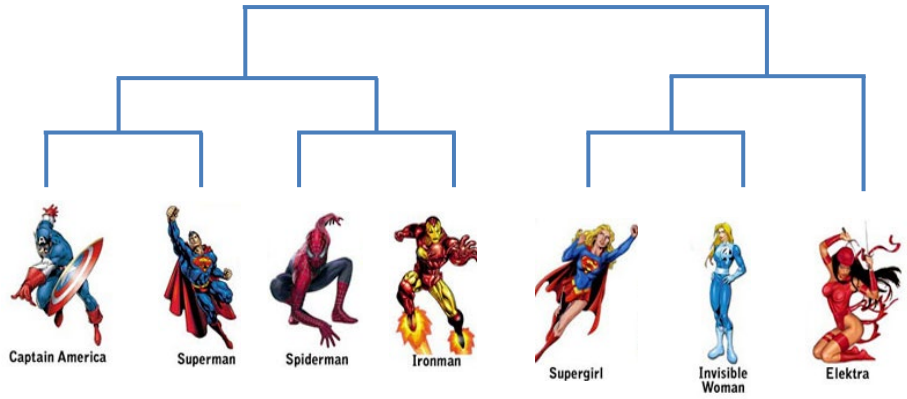
Hierarchical Clustering

- Strength
  - Does not require a pre-defined number of clusters
- Major weakness of agglomerative clustering methods
  - Can never undo what was done previously
  - Do not scale well: time complexity of at least $O(n^2)$, where n is the number of total objects

Partitional Vs Hierarchical

- Partitional: Flat structure
  - Need to specify the number of classes in advance

- Hierarchical: Rich internal structure
  - No need to specify the number of clusters



Partitional Vs Hierarchical

- Hierarchical clustering
  - Efficiency: $O(n^3)$, slow
- Assumptions
  - No assumption
  - Only need distance metric
- Output
  - Create a hierarchical decomposition of objects

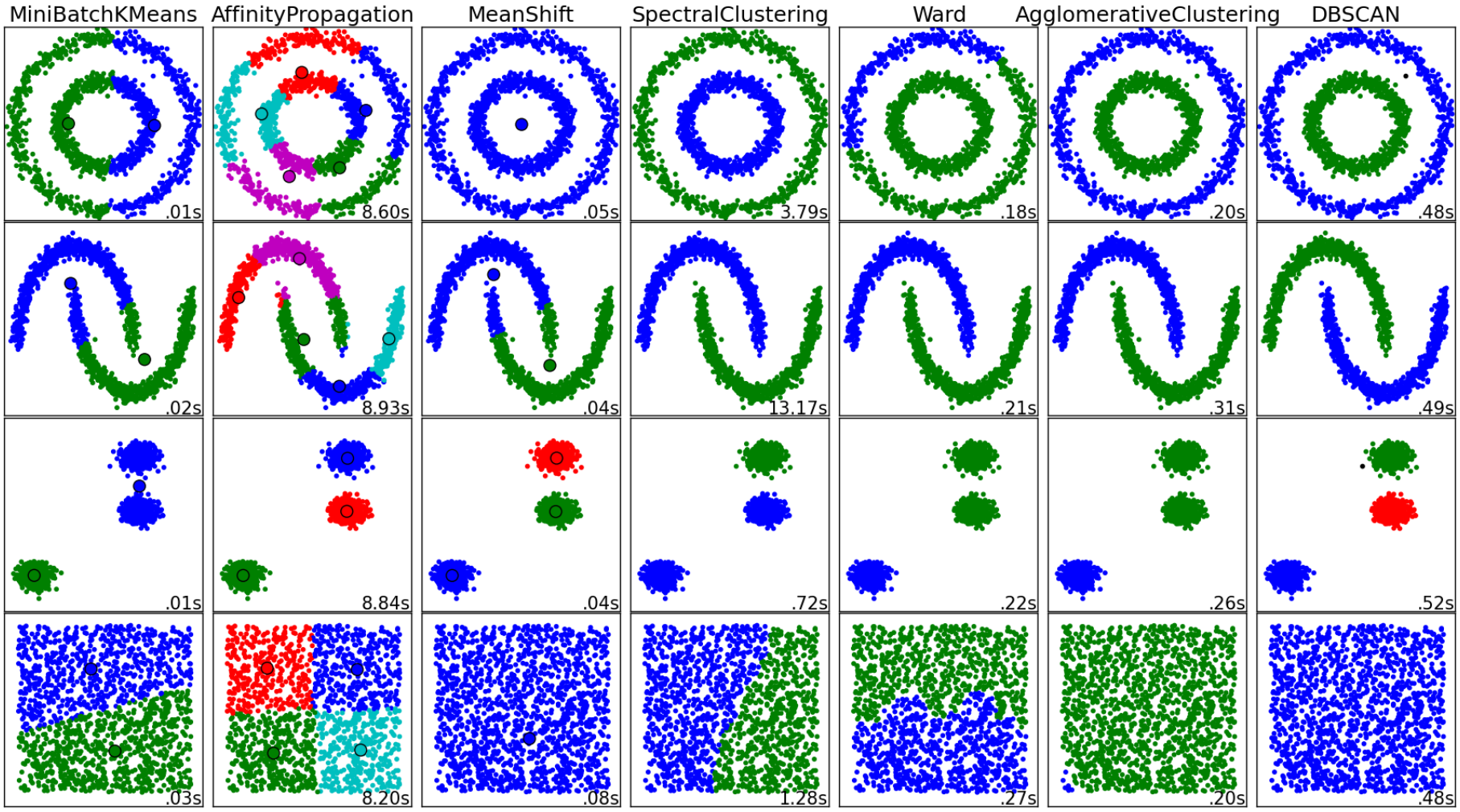
- k-means clustering
  - Efficiency: $O(knl)$, fast
- Assumptions
  - Strong assumption – centroid, latent cluster membership
  - Need to specify k
- Output
  - Flat structure
  - k clusters

Other Clustering Methods
A brief List

Other Clustering Approaches

- Density-based approach (e.g., DBSACN, OPTICS, DenClue)
  - Based on connectivity and density functions
- Model-based (e.g., EM, SOM, COBWEB)
  - A model is hypothesized for each of the clusters and tries to find the best fit of that model to each cluster
- Dimensionality reduction-based approach
  - Subspace (CLIQUE, ProClus): find clusters in all the possible subspaces
  - Matrix factorization (NMF): find clusters in the projected lower-dimensional space
- Frequent pattern-based (pCluster)
  - Based on the analysis of frequent patterns
- User-guided or constraint-based
  - Clustering by considering user-specified or application-specific constraints

Performance with different clustering algorithms



@Source: http://scikit-learn.org/stable/modules/clustering.html

Clustering: Evaluation

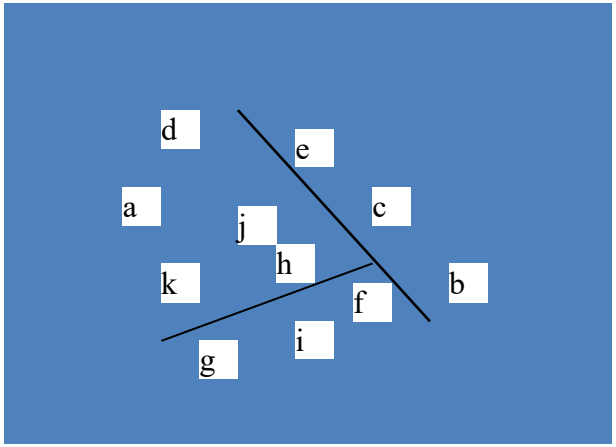
What Is Good Clustering?

- A good clustering method will produce high quality clusters with
  - high intra-class similarity
  - low inter-class similarity
  - Reasonable number of clusters
    - Automatic discovery of natural classes in data
    - Discovery of clusters with arbitrary shape
  - Complete and disjoint Coverage
- The quality of a clustering result depends on both the similarity measure used by the method and its implementation.
- + good visualisation support for “playing” with clusters, to see if they make sense and are useful in a business context

Clusters:
exclusive (hard) vs. overlapping (soft or fuzzy)

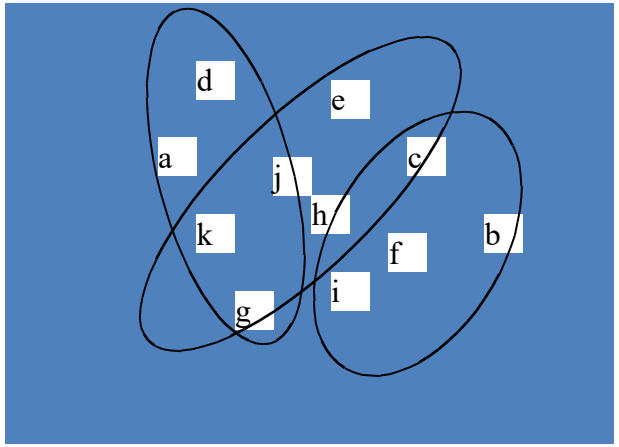
Simple 2-D representation

Hard Clustering



Venn diagram

Soft Clustering



Measuring Clustering Quality

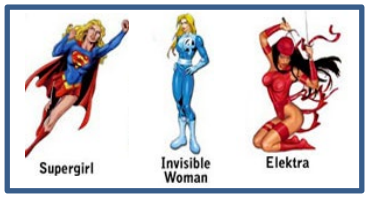
- Two methods: extrinsic vs. intrinsic
- Extrinsic: the ground truth is available only for testing
  - Compare a clustering solution against the ground truth using certain clustering quality measures
  - Ex. Purity, entropy, precision and recall metrics
- Intrinsic: the ground truth is unavailable
  - Evaluate the goodness of a clustering solution by considering how well the clusters are separated, and how compact the clusters are
  - Ex. Silhouette coefficient
    - ranges from -1 to 1, where a high value indicates that the data point is well matched to its cluster and poorly matched to neighboring cluster.

What is the “natural grouping”?

Clustering is very subjective!

Distance metric is important!

Group by gender



Group by source of ability

Group by costume

Clustering Summary

- Unsupervised Learning
- Cluster analysis groups objects based on their similarity and has wide applications
- Measure of similarity can be computed for various types of data
- Many approaches
  - K-means – simple and useful
    - K-medoids is less sensitive to outliers
  - Hierarchical clustering – works better for symbolic attributes
  - Density-based methods, Model-based methods, dimensionality reduction-based methods
- Evaluation is a problem.

References

- Data Mining techniques and concepts by Han J et al, 2011.
- A. K. Jain and R. C. Dubes. Algorithms for Clustering Data. Printice Hall, 1988.
- L. Kaufman and P. J. Rousseeuw. Finding Groups in Data: an Introduction to Cluster Analysis. John Wiley & Sons, 1990.