Clustering

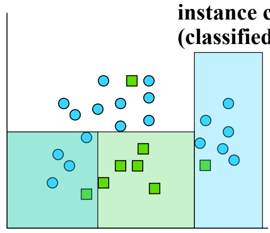
Outline

- Introduction
- K-means clustering
- Hierarchical clustering: COBWEB

Classification vs. Clustering

Classification: Supervised learning:

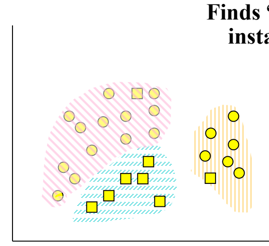
Learns a method for predicting the instance class from pre-labeled (classified) instances



Clustering

Unsupervised learning:

Finds "natural" grouping of instances given un-labeled data



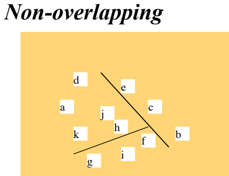
Clustering Methods

- Many different method and algorithms:
  - For numeric and/or symbolic data
  - Deterministic vs. probabilistic
  - Exclusive vs. overlapping
  - Hierarchical vs. flat
  - Top-down vs. bottom-up

Clusters: exclusive vs. overlapping

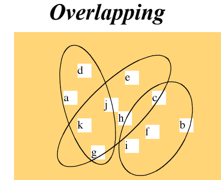
Simple 2-D representation

Non-overlapping



Venn diagram

Overlapping



Clustering Evaluation

- Manual inspection
- Benchmarking on existing labels
- Cluster quality measures
  - distance measures
  - high similarity within a cluster, low across clusters

The distance function

- Simplest case: one numeric attribute A
  - Distance(X,Y) = A(X) – A(Y)
- Several numeric attributes:
  - Distance(X,Y) = Euclidean distance between X,Y
- Nominal attributes: distance is set to 1 if values are different, 0 if they are equal
- Are all attributes equally important?
  - Weighting the attributes might be necessary

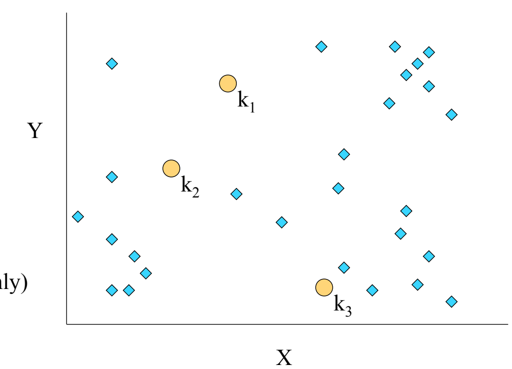
Simple Clustering: K-means

Works with numeric data only

1) Pick a number (K) of cluster centers (at random)
2) Assign every item to its nearest cluster center (e.g. using Euclidean distance)
3) Move each cluster center to the mean of its assigned items
4) Repeat steps 2,3 until convergence (change in cluster assignments less than a threshold)

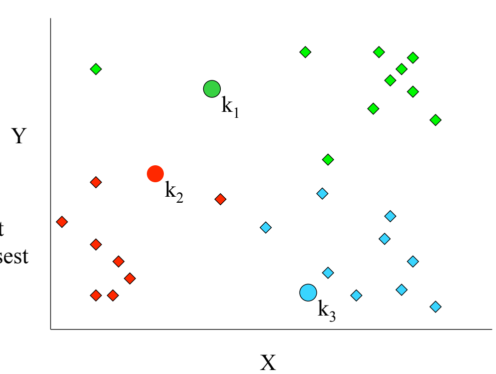
K-means example, step 1

Pick 3 initial cluster centers (randomly)



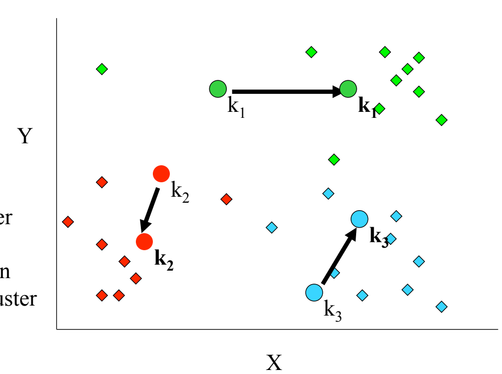
K-means example, step 2

Assign each point to the closest cluster center



K-means example, step 3

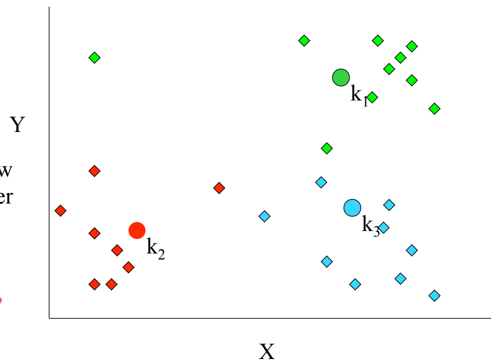
Move each cluster center to the mean of each cluster



K-means example, step 4

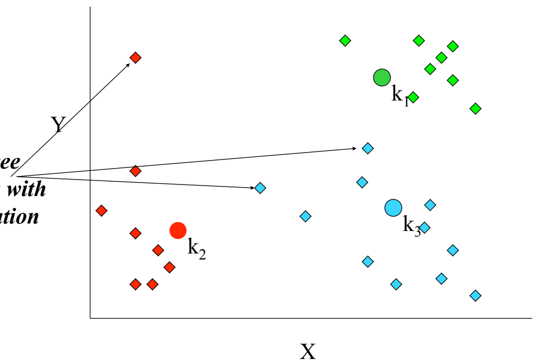
Reassign points closest to a different new cluster center

Q: Which points are reassigned?



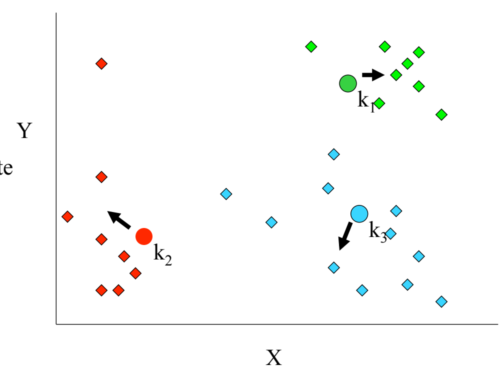
K-means example, step 4 …

A: three points with animation



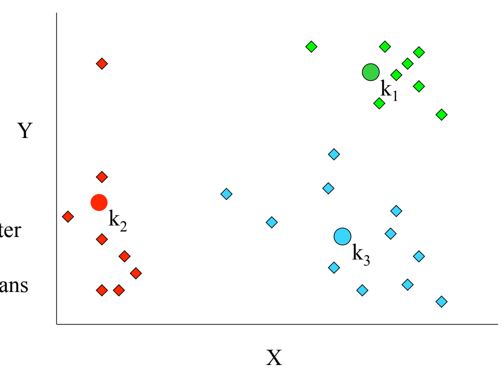
K-means example, step 4b

re-compute cluster means



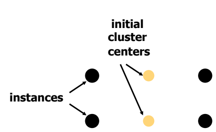
K-means example, step 5

move cluster centers to cluster means



Discussion

- Result can vary significantly depending on initial choice of seeds
- Can get trapped in local minimum
  - Example:



- To increase chance of finding global optimum: restart with different random seeds

K-means clustering summary

Advantages

- Simple, understandable
- items automatically assigned to clusters

Disadvantages

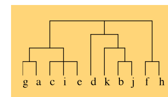
- Must pick number of clusters before hand
- All items forced into a cluster
- Too sensitive to outliers

K-means variations

- **K-medoids** – instead of mean, use medians of each cluster
  - Mean of 1, 3, 5, 7, 9 is **5**
  - Mean of 1, 3, 5, 7, 1009 is **205**
  - Median of 1, 3, 5, 7, 1009 is **5**
  - Median advantage: not affected by extreme values
- For large databases, use sampling

*Hierarchical clustering

- Bottom up
  - Start with single-instance clusters
  - At each step, join the two closest clusters
  - Design decision: distance between clusters
    - E.g. two closest instances in clusters vs. distance between means
- Top down
  - Start with one universal cluster
  - Find two clusters
  - Proceed recursively on each subset
  - Can be very fast
- Both methods produce a *dendrogram*

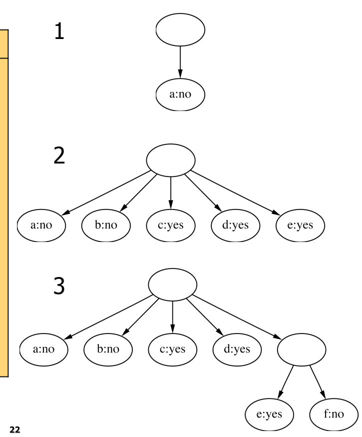


*Incremental clustering

- Heuristic approach (COBWEB/CLASSIT)
- Form a hierarchy of clusters incrementally
- Start:
  - tree consists of empty root node
- Then:
  - add instances one by one
  - update tree appropriately at each stage
  - to update, find the right leaf for an instance
  - May involve restructuring the tree
- Base update decisions on *category utility*

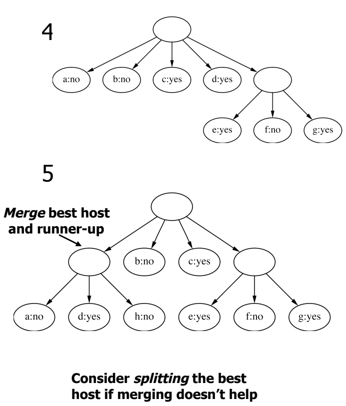
*Clustering weather data

ID	Outlook	Temp.	Humidity	Windy
A	Sunny	Hot	High	False
B	Sunny	Hot	High	True
C	Overcast	Hot	High	False
D	Rainy	Mild	High	False
E	Rainy	Cool	Normal	False
F	Rainy	Cool	Normal	True
G	Overcast	Cool	Normal	True
H	Sunny	Mild	High	False
I	Sunny	Cool	Normal	False
J	Rainy	Mild	Normal	False
K	Sunny	Mild	Normal	True
L	Overcast	Mild	High	True
M	Overcast	Hot	Normal	False
N	Rainy	Mild	High	True



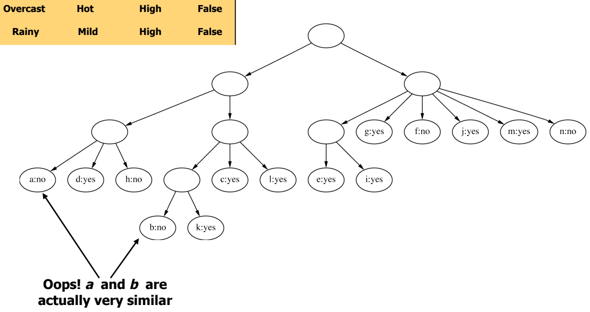
*Clustering weather data

ID	Outlook	Temp.	Humidity	Windy
A	Sunny	Hot	High	False
B	Sunny	Hot	High	True
C	Overcast	Hot	High	False
D	Rainy	Mild	High	False
E	Rainy	Cool	Normal	False
F	Rainy	Cool	Normal	True
G	Overcast	Cool	Normal	True
H	Sunny	Mild	High	False
I	Sunny	Cool	Normal	False
J	Rainy	Mild	Normal	False
K	Sunny	Mild	Normal	True
L	Overcast	Mild	High	True
M	Overcast	Hot	Normal	False
N	Rainy	Mild	High	True

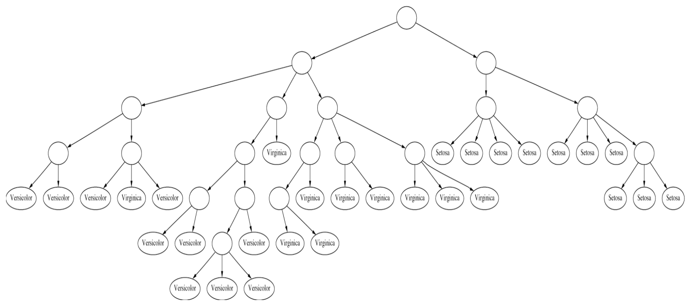


*Final hierarchy

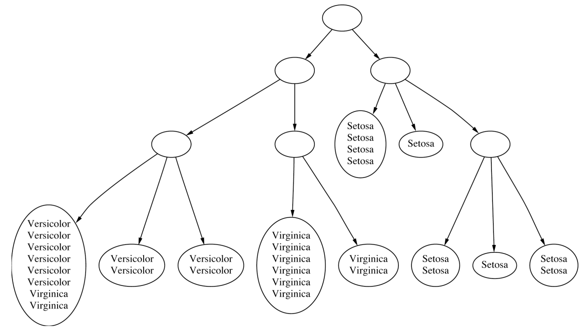
ID	Outlook	Temp.	Humidity	Windy
A	Sunny	Hot	High	False
B	Sunny	Hot	High	True
C	Overcast	Hot	High	False
D	Rainy	Mild	High	False



*Example: the iris data (subset)



*Clustering with cutoff



*Category utility

- Category utility: quadratic loss function defined on conditional probabilities:

$$CU(C_1, C_2, \ldots, C_k) = \frac{\sum_l \Pr[C_l] \sum_i \sum_j (\Pr[a_i = v_{ij} \mid C_l]^2 - \Pr[a_i = v_{ij}]^2)}{k}$$

- Every instance in different category ⇒ numerator becomes

$$m - \Pr[a_i = v_{ij}]^2 \longleftarrow \textbf{\textit{maximum}}$$

number of attributes

*Overfitting-avoidance heuristic

- If every instance gets put into a different category the numerator becomes (maximal):

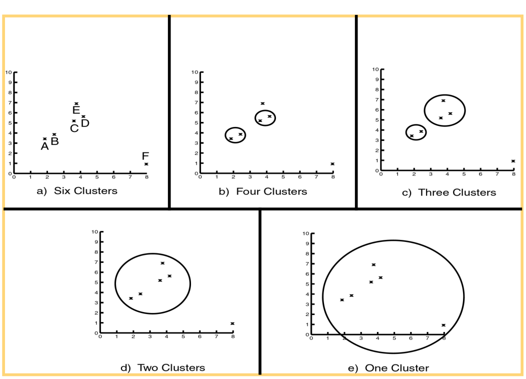
$$n - \sum_i \sum_j \Pr[a_i = v_{ij}]^2 \longleftarrow$$

Maximum value of CU

Where *n* is number of all possible attribute values.

- So without *k* in the denominator of the CU-formula, every cluster would consist of one instance!

Levels of Clustering

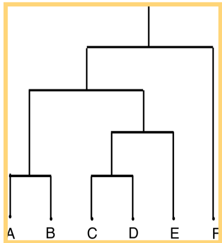


Hierarchical Clustering

- Clusters are created in levels actually creating sets of clusters at each level.
- ***Agglomerative***
  - Initially each item in its own cluster
  - Iteratively clusters are merged together
  - Bottom Up
- ***Divisive***
  - Initially all items in one cluster
  - Large clusters are successively divided
  - Top Down

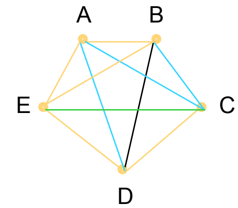
Dendrogram

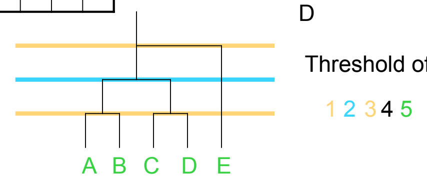
- *Dendrogram:* a tree data structure which illustrates hierarchical clustering techniques.
- Each level shows clusters for that level.
  - Leaf – individual clusters
  - Root – one cluster
- A cluster at level i is the union of its children clusters at level i+1.



Agglomerative Example

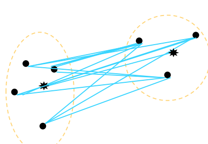
	A	B	C	D	E
A	0	1	2	2	3
B	1	0	2	4	3
C	2	2	0	1	5
D	2	4	1	0	3
E	3	3	5	3	0



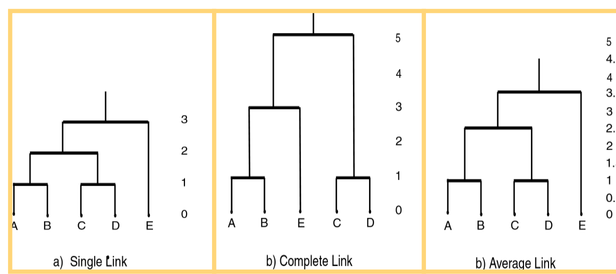


Distance Between Clusters

- ***Single Link***: smallest distance between points
- ***Complete Link:*** largest distance between points
- ***Average Link:*** average distance between points
- ***Centroid:*** distance between centroids



Single Link Clustering



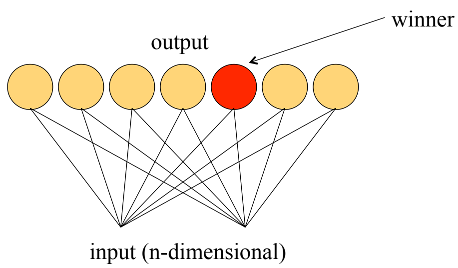
Other Clustering Approaches

- EM – probability based clustering
- Bayesian clustering
- SOM – self-organizing maps
- …

Self-Organizing Map

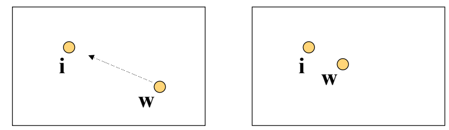
Self Organizing Map

- Unsupervised learning
- Competitive learning



Self Organizing Map

- Determine the winner (the neuron of which the weight vector has the smallest distance to the input vector)
- Move the weight vector **w** of the winning neuron towards the input **i**

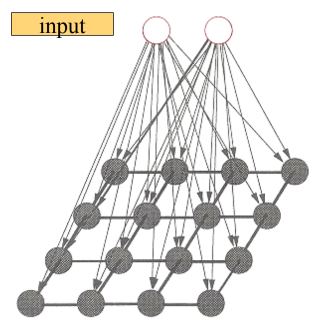


Before learning *After learning*

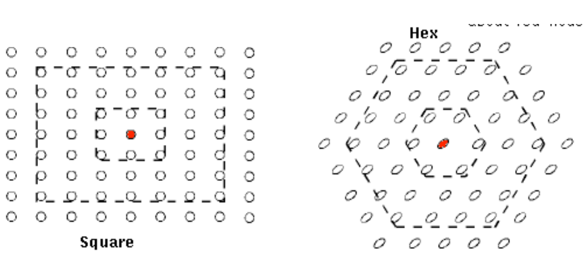
Self Organizing Map

- Impose a topological order onto the competitive neurons (e.g., rectangular map)
- Let neighbors of the winner share the "prize" (The "postcode lottery" principle)
- After learning, neurons with similar weights tend to cluster on the map

Self Organizing Map



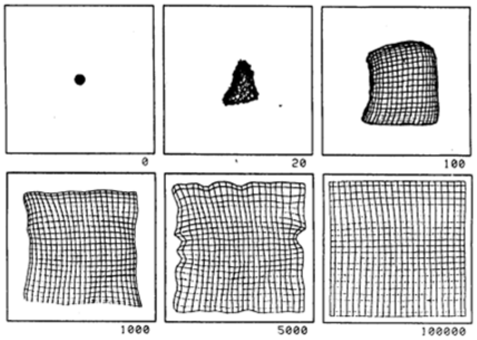
Self Organizing Map



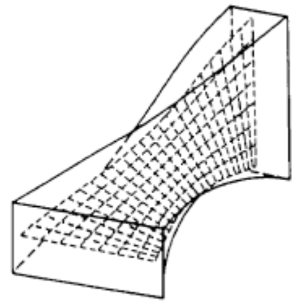
Self Organizing Map

- Input: uniformly randomly distributed points
- Output: Map of 20^2 neurons
- Training
  - Starting with a large learning rate and neighborhood size, both are gradually decreased to facilitate convergence

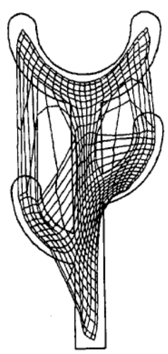
Self Organizing Map

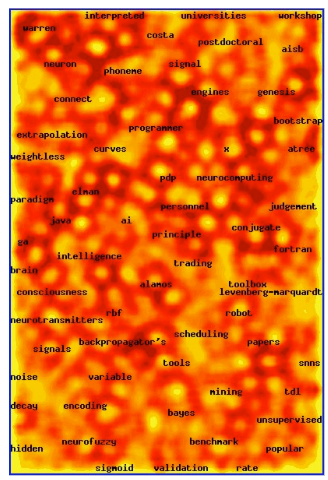


Self Organizing Map

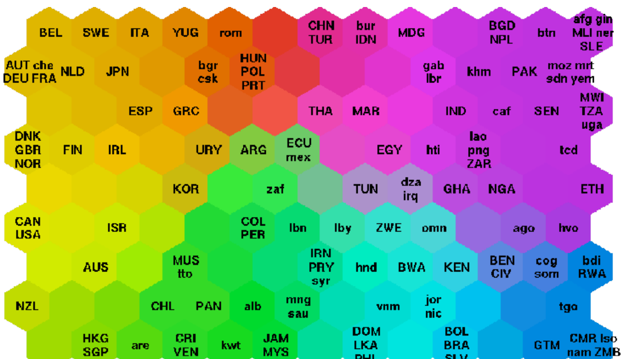


Self Organizing Map

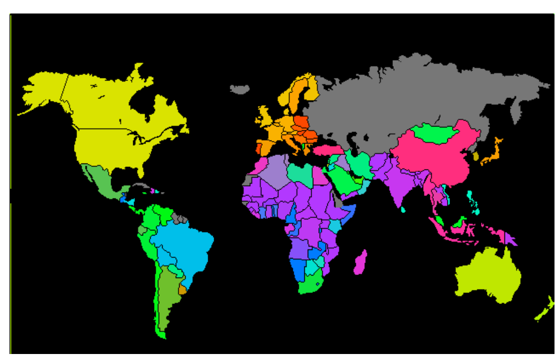




Self Organizing Map



Self Organizing Map



Discussion

- Can interpret clusters by using supervised learning
  - learn a classifier based on clusters
- Decrease dependence between attributes?
  - pre-processing step
  - E.g. use *principal component analysis*
- Can be used to fill in missing values
- Key advantage of probabilistic clustering:
  - Can estimate likelihood of data
  - Use it to compare different models objectively

Examples of Clustering Applications

- **Marketing:** discover customer groups and use them for targeted marketing and re-organization
- **Astronomy:** find groups of similar stars and galaxies
- **Earth-quake studies:** Observed earth quake epicenters should be clustered along continent faults
- **Genomics:** finding groups of gene with similar expressions
- …

Clustering Summary

- unsupervised
- many approaches
  - K-means – simple, sometimes useful
    - K-medoids is less sensitive to outliers
  - Hierarchical clustering – works for symbolic attributes
- Evaluation is a problem