Multi-dimensional index structures

- … cluster objects when the individual dimensions (features) are known…
- …what if we do not have explicit features??

Multi-dimensional index structures

- … cluster objects when the individual dimensions (features) are known…
- …what if we do not have explicit features??
  - …we may have a "black-box program" which can compare two objects

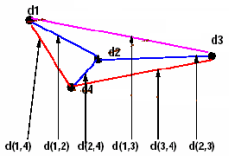
Multi-dimensional index structures

- … cluster objects when the individual dimensions (features) are known…
- …what if we do not have explicit features??
  - …we may have a "black-box program" which can compare two objects
  - …we may have "users" evaluating the similarity of objects

Multi-dimensional index structures

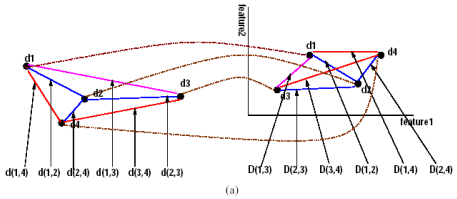
- … cluster objects when the individual dimensions (features) are known…
- …what if we do not have explicit features??
  - …we may have a "black-box program" which can compare two objects
  - …we may have "users" evaluating the similarity of objects
- we need a clustering scheme that does not need explicit features!!!!

What if we do not have features??



We know the distances, but
- we do not have explicit features
- distances are not metric….

Multi Dimensional Scaling



(a)

$d(1,2) \approx D(1,2)$ $d(1,3) \approx D(1,3)$ $d(1,4) \approx D(1,4)$
$d(2,3) \approx D(2,3)$ $d(2,4) \approx D(2,4)$ $d(3,4) \approx D(3,4)$

MDS

- The criterion for the mapping is to minimize *stress*

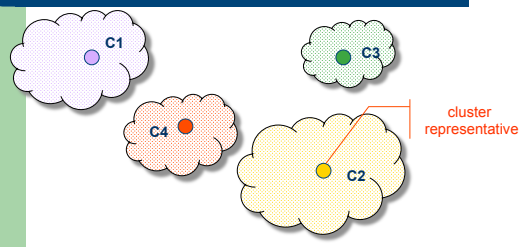
$$stress = \sqrt{\frac{\sum_{i,j}(d'_{ij} - d_{ij})^2}{\sum_{i,j} d_{ij}^2}}$$

- Start with a (random) configuration of points with low dimensions
- Apply some form of steepest descent iteratively to minimize the stress.
  - move objects
  - add dimensions

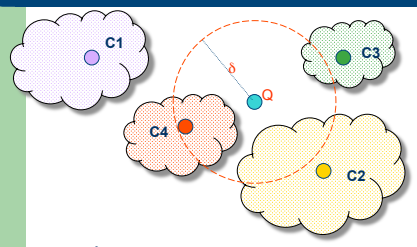
MDS

- How to map the query?

Use of clusters (prune search space)

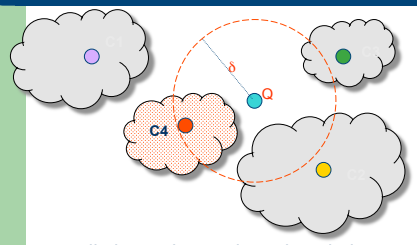


Use of clusters (prune search space)



- …given a query

Use of clusters (prune search space)



- …eliminate clusters based on their representatives

Clustering methods

- Sound methods:
  - need a fixed document-to-document similarity matrix
- Iterative methods:
  - use document vectors iteratively

Outline of sound methods

- Find the similarity of each object pair

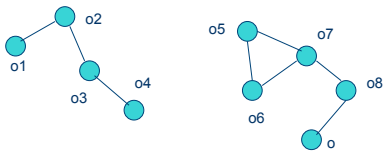
Outline of sound methods

- Find the similarity of each object pair
- Setup a threshold
  - sim(o1,o2) < T (objects are very different!)
  - sim(o1,o2) >= T (objects are comparable)

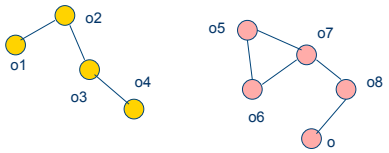
Outline of sound methods

- Find the similarity of each object pair
- Setup a threshold
  - sim(o1,o2) < T (objects are very different!)
  - sim(o1,o2) >= T (objects are comparable)
- Create a graph which represents object similarities
  - Each pair of objects that are comparable is connected with an edge

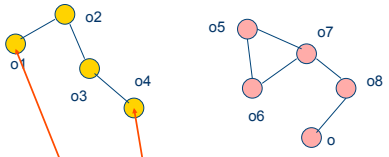
Example collection



Connected components

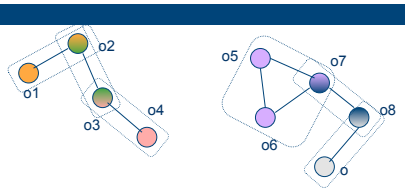


Connected components

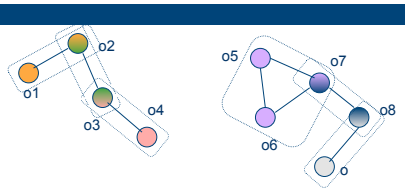


- …are o1 and o4 really similar????

Clique…



Clique…



- Clusters are overlapping!
- Costlier to compute (NP-complete)

…or single-pass iterative method

- choose an object, and make it a cluster

…or single-pass iterative method

- choose an object, and make it a cluster
- choose another object, o
  - find the closest cluster, c
  - if dist(o,c) < T add o to c
  - else o is a new cluster

…or single-pass iterative method

- choose an object, and make it a cluster
- choose another object, o
  - find the closest cluster, c
  - if dist(o,c) < T add o to c
  - else o is a new cluster
- repeat until all objects are processed

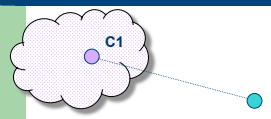
…or single-pass iterative method

- choose an object, and make it a cluster
- choose another object, o
  - find the closest cluster, c
  - if dist(o,c) < T add o to c
  - else o is a new cluster
- repeat until all objects are processed

also called the “leader” algorithm

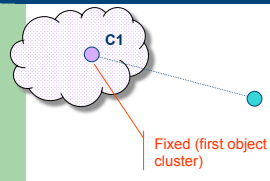
….how do we compute distance?

….how do we compute distance?



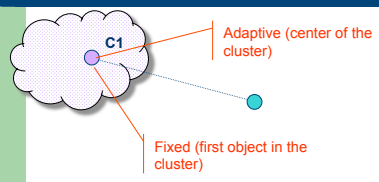
- use a cluster representative…

….how do we compute distance?



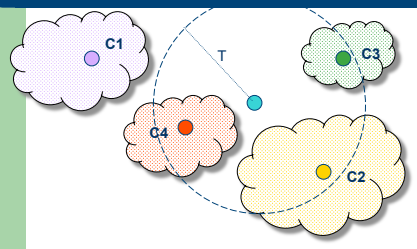
- use a cluster representative…

….how do we compute distance?

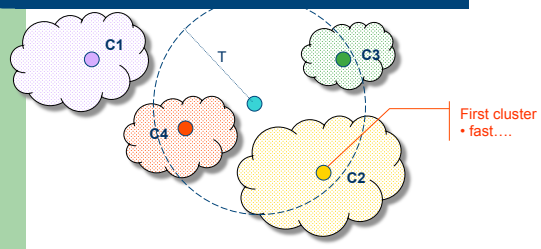


- use a cluster representative…

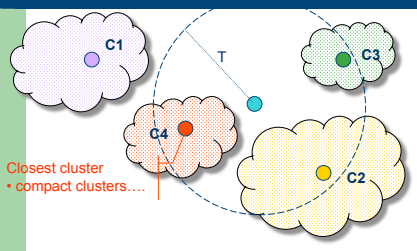
….alternatives…



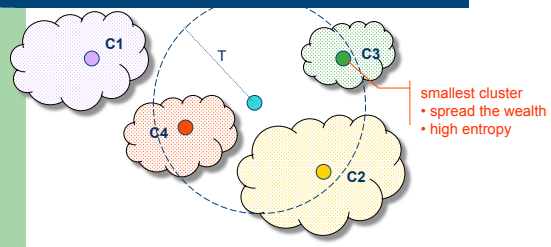
….alternatives…



….alternatives…



….alternatives…



What if we do not have a threshold??

- Find a minimum spanning tree of the input graph
  - $O(N^2)$ edges to $O(N)$ edges

What if we do not have a threshold??

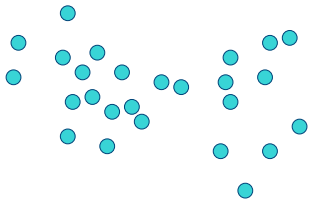
- Find a minimum spanning tree of the input graph
  - $O(N^2)$ edges to O(N) edges
- Remove all edges longer than the average of their neighbors
  - threshold is determined based on the neighborhood

What if we do not have a threshold??

- Find a minimum spanning tree of the input graph
  - $O(N^2)$ edges to O(N) edges
- Remove all edges longer than the average of its neighbors
  - threshold is determined based on the neighborhood
- Apply connected-components or clique..

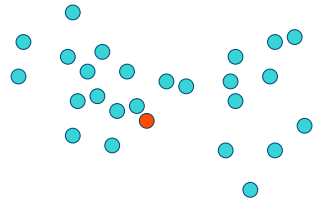
Max-a-min

- The number of clusters is known, r (say 4)



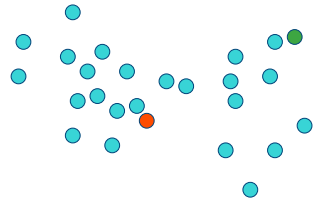
Max-a-min

- Choose a random leader



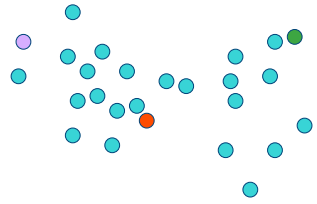
Max-a-min

- Find the furthest point to the leader



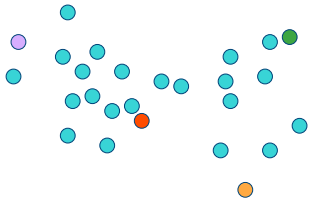
Max-a-min

- Find the furthest point to the 2 leaders



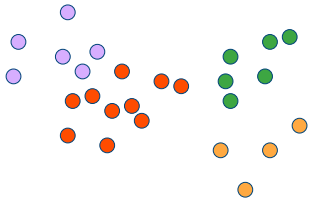
Max-a-min

- Find the furthest point to the 3 leaders



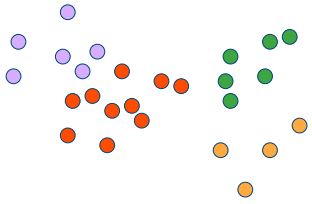
Max-a-min

- Assign points to closest leader..



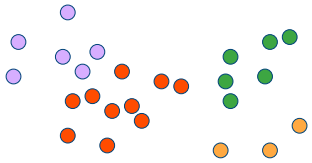
K-means (iterative improvement)

- Minimize a “global cost function”



K-means (iterative improvement)

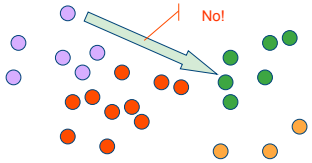
- Minimize a "global cost function"



- …each item is checked whether moving to another cluster would reduce global cost

K-means (iterative improvement)

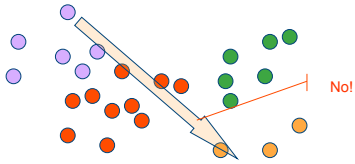
- Minimize a "global cost function"



- …each item is checked whether moving to another cluster would reduce global cost

K-means (iterative improvement)

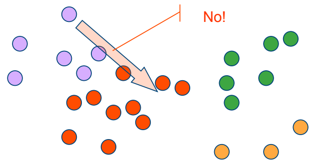
- Minimize a "global cost function"



- …each item is checked whether moving to another cluster would reduce global cost

K-means (iterative improvement)

- Minimize a “global cost function”

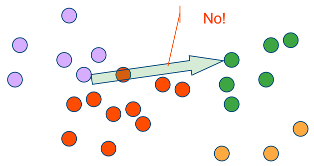


- …each item is checked whether moving to another cluster would reduce global cost

Maria Luisa Sapino (BDM 2018)

K-means (iterative improvement)

- Minimize a “global cost function”

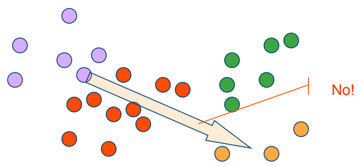


- …each item is checked whether moving to another cluster would reduce global cost

Maria Luisa Sapino (BDM 2018)

K-means (iterative improvement)

- Minimize a “global cost function”

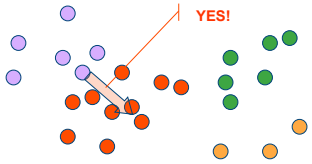


- …each item is checked whether moving to another cluster would reduce global cost

Maria Luisa Sapino (BDM 2018)

K-means (iterative improvement)

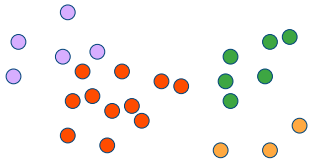
- Minimize a "global cost function"



- …each item is checked whether moving to another cluster would reduce global cost

K-means (iterative improvement)

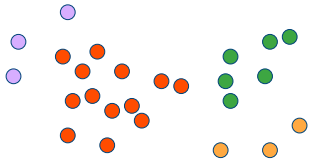
- Minimize a "global cost function"



- …each item is checked whether moving to another cluster would reduce global cost

K-means (iterative improvement)

- Minimize a "global cost function"



- …each item is checked whether moving to another cluster would reduce global cost

What are the possible criteria??

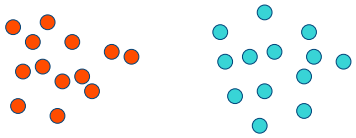
- Compactness (minimize root-mean-square)

$$RMSE_m = \sqrt{\frac{1}{N}\sum_{i=0}^{N}\left[\hat{F}_m(x_i) - F_m(x_i)\right]^2}$$

$$RMSE = \frac{1}{M}\sum_{m=1}^{M} RMSE_m$$

What are the possible criteria??

- Evenly sized clusters (maximize entropy)



$$H(X) \equiv \sum_{x \in \mathcal{A}_X} P(x) \log \frac{1}{P(x)}$$

What if we do not have distances???

- …we need to learn from
  - user feedback or
  - user access patterns!!

Confidence clustering…

- Assumption:
  - each cluster can have maximum r objects
- …keeps adapting the clusters to user access pattern

Confidence clustering…

- Assign each object o_i to cluster C_j with random confidence ($0 <= conf(i,j) <= 10$)

Confidence clustering…

- Assign each object o_i to cluster C_j with random confidence ($0 <= conf(i,j) <= 10$)
- If user accesses o_a and o_b, then

Confidence clustering…

- Assign each object o_i to cluster C_j with random confidence (0 <= conf(i,j)<= 10)
- If user accesses o_a and o_b, then
  - if o_a and o_b are in the same cluster C_j then
    - conf(a,j)++
    - conf(b,j)++

Confidence clustering…

- Assign each object o_i to cluster C_j with random confidence (0 <= conf(i,j)<= 10)
- If user accesses o_a and o_b, then
  - if o_a is in cluster C_i and o_b is in cluster C_j then

Confidence clustering…

- Assign each object o_i to cluster C_j with random confidence (0 <= conf(i,j)<= 10)
- If user accesses o_a and o_b, then
  - if o_a is in cluster C_i and o_b is in cluster C_j then
    - if conf(a,i) > 1 and conf(b,j)>1 then
      - conf(a,i)--
      - conf(b,j)--

Confidence clustering…

- Assign each object o_i to cluster C_j with random confidence (0 <= conf(i,j)<= 10)
- If user accesses o_a and o_b, then
  - if o_a is in cluster C_i and o_b is in cluster C_j then
    - if conf(a,i) == 1 and conf(b,j)==1 then
      - conf(b,i) = 1 (move o_b from C_j to C_i)
      - conf(c,j) = 1 (move some o_c from C_i to C_j)

Maria Luisa Sapino (BDM 2018)

Confidence clustering…

- Assign each object o_i to cluster C_j with random confidence (0 <= conf(i,j)<= 10)
- If user accesses o_a and o_b, then
  - if o_a is in cluster C_i and o_b is in cluster C_j then
    - if conf(a,i) >1 and conf(b,j)==1 then
      - there exists conf(c,i) == 1
        - conf(b,i) = 1 (move o_b from C_j to C_i)
        - conf(c,j) = 1 (move o_c from C_i to C_j)

Maria Luisa Sapino (BDM 2018)

Confidence clustering…

- Assign each object o_i to cluster C_j with random confidence (0 <= conf(i,j)<= 10)
- If user accesses o_a and o_b, then
  - if o_a is in cluster C_i and o_b is in cluster C_j then
    - if conf(a,i) >1 and conf(b,j)==1 then
      - there does not exists conf(c,i) == 1
        - conf(a,i)--

Maria Luisa Sapino (BDM 2018)

Adaptive clustering…

- What if we do not know the number of clusters????
- …keeps adapting the clusters to user access pattern

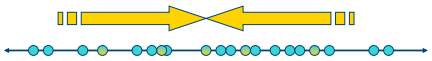
Adaptive clustering…

- start with a random assignment of objects to a line

Adaptive clustering…

- If a set of objects are accessed together…

Adaptive clustering…



- ..pull the objects closer to their average point..

Adaptive clustering…

- ..choose another random set of objects..

Adaptive clustering…

- ..push these away from their average point..

Adaptive clustering…

- …over time…
  - similar objects will come closer…
  - different objects will get apart…

Maria Luisa Sapino (BDM 2018)