Clustering

Victor Kitov

v.v.kitov@yandex.ru

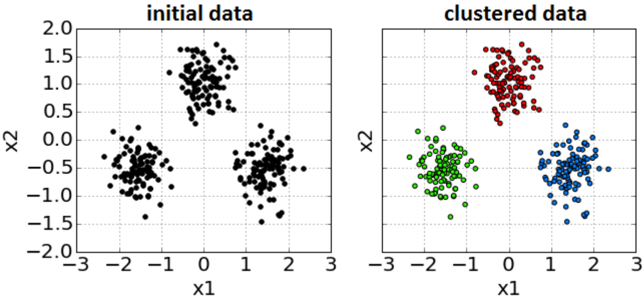
Yandex
School of Data Analysis.

Table of Contents

Aim of clustering

- Clustering is partitioning of objects into groups so that:
  - inside groups objects are very similar
  - objects from different groups are dissimilar
- Unsupervised learning
- No definition of “similar”
  - different algorithms use different formalizations of similarity

Clustering demo



Applications of clustering

- data summarization
    - feature vector is replaced by cluster number
- feature extraction
    - cluster number, cluster average target, distance to native cluster center / other clusters
- customer segmentation
    - e.g. for recommender service
- community detection in networks
    - nodes - people, similarity - number of connections
- outlier detection
    - outliers do not belong any cluster

Clustering algorithms comparison

We can compare clustering algorithms in terms of:

- computational complexity
- do they build flat or hierarchical clustering?
- can the shape of clustering be arbitrary?
  - if not is it symmetrical, can clusters be of different size?
- can clusters vary in density of contained objects?
- robustness to outliers

Table of Contents

K-means algorithm

- Suppose we want to cluster our data into K clusters.
- Cluster i has a center μ_i, i=1,2,...K.
- Consider the task of minimizing

$$\sum_{n=1}^{N} \|x_n - \mu_{z_n}\|_2^2 \to \min_{z_1,...z_N,\mu_1,...\mu_K} \tag{1}$$

  where $z_i \in \{1, 2, ...K\}$ is cluster assignment for x_i and $\mu_1, ...\mu_K$ are cluster centers.
- Direct optimization requires full search and is impractical.
- K-means is a suboptimal algorithm for optimizing (1).

K-means algorithm

```
Initialize μ_j, j = 1,2,...K.

WHILE not converged:

    FOR i = 1,2,...N:
        find cluster number of x_i:
        z_i = arg min_{j∈{1,2,...K}} ||x_i − μ_j||_2^2

    FOR j = 1,2,...K:
        μ_j = 1/(∑_{n=1}^N I[z_n=j]) ∑_{n=1}^N I[z_n = j] x_i
```

Initialize μ_j, $j = 1, 2, ...K$.

WHILE not converged:

FOR $i = 1, 2, ...N$:

find cluster number of x_i:

$z_i = \arg\min_{j \in \{1,2,...K\}} ||x_i - \mu_j||_2^2$

FOR $j = 1, 2, ...K$:

$\mu_j = \frac{1}{\sum_{n=1}^N \mathbb{I}[z_n=j]} \sum_{n=1}^N \mathbb{I}[z_n = j] x_i$

K-means properties

Convergence conditions:

- maximum number of iterations reached
- cluster assignments $z_1, ...z_N$ stop to change (exact)
- $\{\mu_i\}_{i=1}^K$ stop changing significantly (approximate)

Initialization:

- typically $\{\mu_i\}_{i=1}^K$ are initialized to randomly chosen training objects

K-means properties

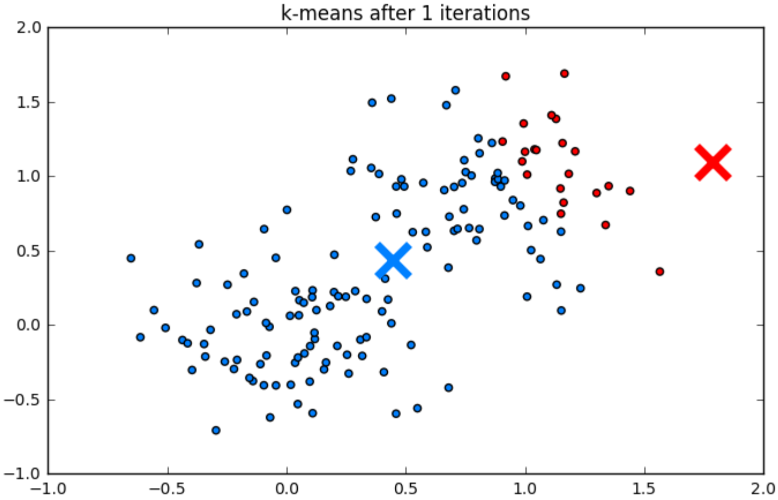
Optimality:

- criteria is non-convex
- solution depends on starting conditions
- may restart several times from different initializations and select solution giving minimal value of (1).

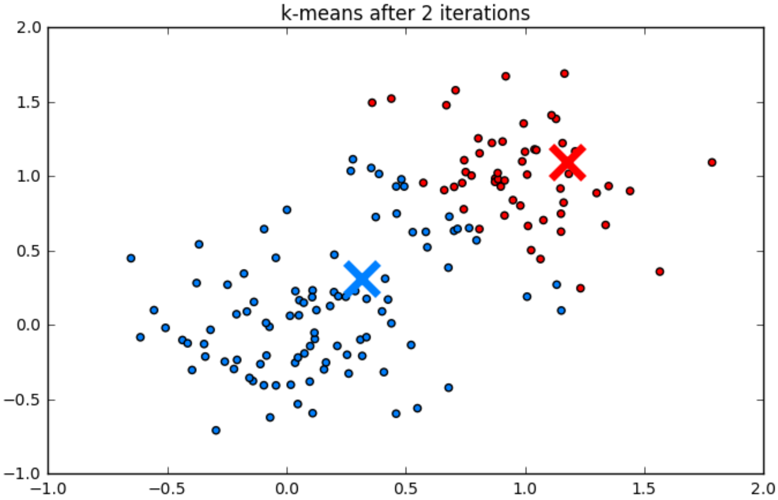
Complexity: $O(NDKI)$

- K is the number of clusters
- I is the number of iterations.
    - usually few iterations are enough for convergence.

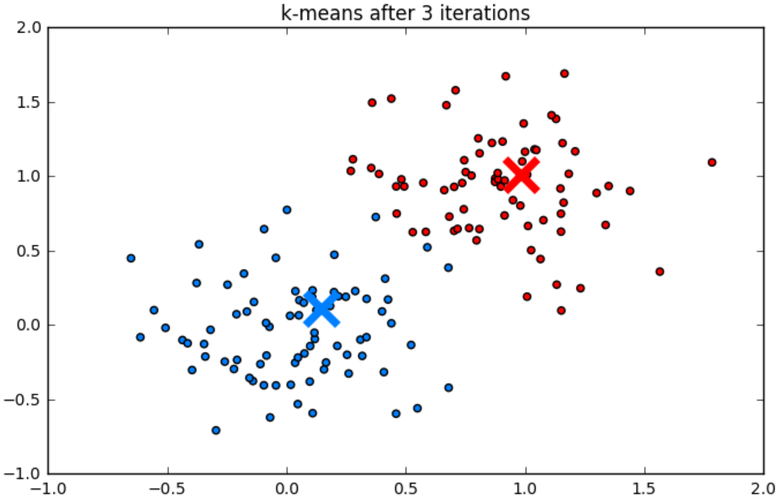
Example of K-means



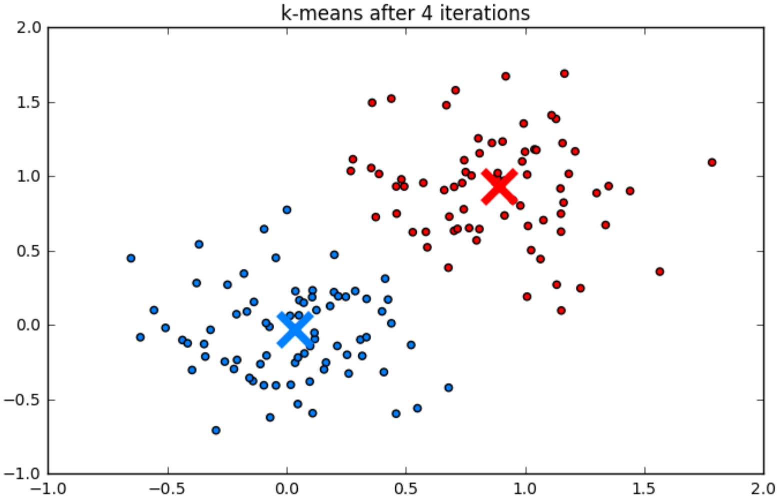
Example of K-means



Example of K-means

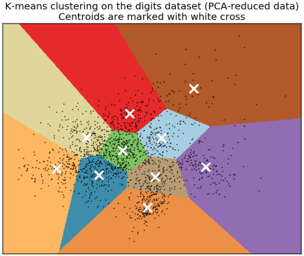


Example of K-means



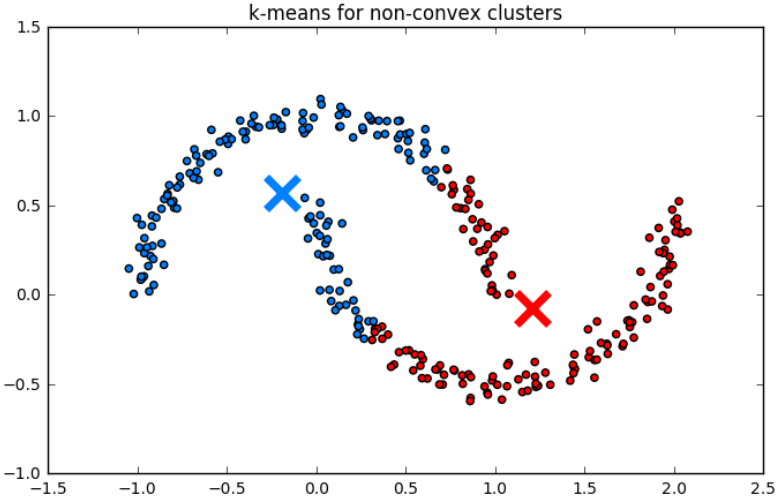
Gotchas

- K-means assumes that clusters are convex:



- It always finds clusters even if none actually exist
  - need to control cluster quality metrics

K-means for non-convex clusters



K-means for data without clusters

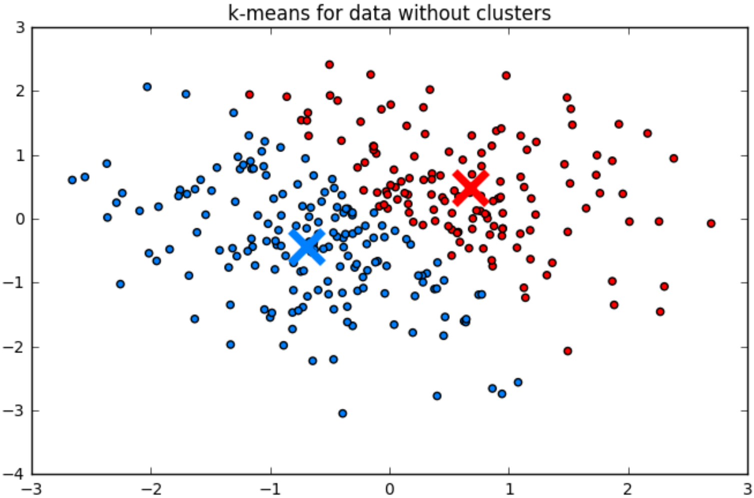


Table of Contents

Motivation

- Number of clusters K not known a priory.
- Clustering is usually not flat, but hierarchical with different levels of granularity:
    - sites in the Internet
    - books in library
    - animals in nature

Hierarchical clustering

Hierarchical clustering may be:

- top-down
    - hierarchical K-means
- bottom-up
    - agglomerative clustering

3. Hierarchical clustering
    - Top-down hierarchical clustering
    - Bottom-up hierarchical clustering

Algorithm

INPUT:
data D, flat clustering algorithm A
leaf selection criterion, termination criterion

Initialize tree T to root, containing all data

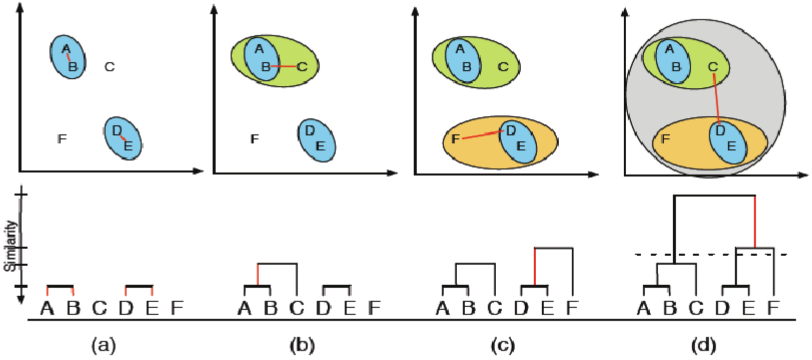
REPEAT
   based on selection criterion, select leaf L
   using algorithm A split L into children $L_1, ...L_K$
   add $L_1, ...L_K$ as child nodes to tree T
UNTIL termination criterion

Comments

- Leaf selection criterion:
    - split leaf most close to the root
        - result: balanced tree by height
    - split leaf with maximum elements
        - result: balanced tree by cluster size
- Building hierarchy top-down is more natural for a human

3. Hierarchical clustering
    - Top-down hierarchical clustering
    - Bottom-up hierarchical clustering

Bottom-up clustering demo



Algorithm

initialize distance matrix $M \in \mathbb{R}^{N \times N}$ between
singleton clusters $\{x_1\}, ...\{x_N\}$

REPEAT:
1) pick closest pair of clusters i and j
2) merge clusters i and j
3) delete rows/columns i, j from M and add new row/column for merged cluster

UNTIL 1 cluster is left

RETURN hiearchical clustering of objects

- Early stopping is possible when:
  - K clusters are left
  - distance between most close clusters $\geq$threshold

Agglomerative clustering - distances

- Consider clusters $A = \{x_{i_1}, x_{i_2}, ...\}$ and $B = \{x_{j_1}, x_{j_2}, ...\}$.
- We can define the following natural distances
  - single link (nearest neighbour)

$$\rho(A, B) = \min_{a \in A, b \in B} \rho(a, b)$$

  - complete-link (furthest neighbour)

$$\rho(A, B) = \max_{a \in A, b \in B} \rho(a, b)$$

  - group average link

$$\rho(A, B) = \text{mean}_{a \in A, b \in B} \rho(a, b)$$

  - closest centroid

$$\rho(A, B) = \rho(\mu_A, \mu_B)$$

where $\mu_U = \frac{1}{|U|} \sum_{x \in U} x$ or $m_U = median_{x \in U}\{x\}$