Machine Learning

Clustering - KMeans

Claudio Sartori

DISI
Department of Computer Science and Engineering – University of Bologna, Italy
claudio.sartori@unibo.it

The problem of clustering

For a comprehensive review see [Jain et al.(1999)Jain, Murty, and Flynn]

- Given – a set of N objects x_i, each described by D values x_{id}
- Task – find a natural partitioning in K clusters and, possibly, a number of noise objects
- Result – a clustering scheme, i.e. a function mapping each data object to the sequence $[1 \dots K]$ (or to noise)

Desired property of clusters

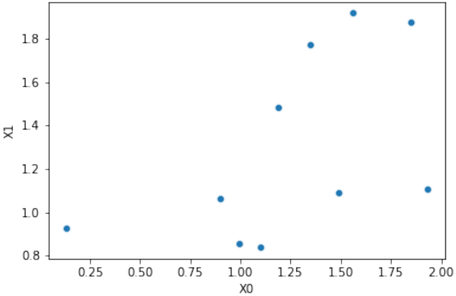
- objects in the same cluster are similar
    - look for a clustering scheme which maximizes intra–cluster similarity

A little bit of formality

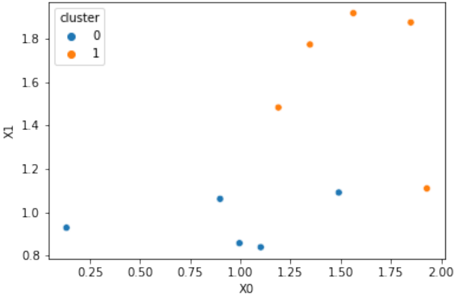
- find a function $clust()$ from $\mathcal{X}$ to $1..K$ such that:
  - $\forall x_1, x_2 \in \mathcal{X}$, $clust(x_1) = clust(x_2)$ when x_1 and x_2 are similar
  - $\forall x_1, x_2 \in \mathcal{X}$, $clust(x_1) \neq clust(x_2)$ when x_1 and x_2 are not similar

Clustering

2-d Data



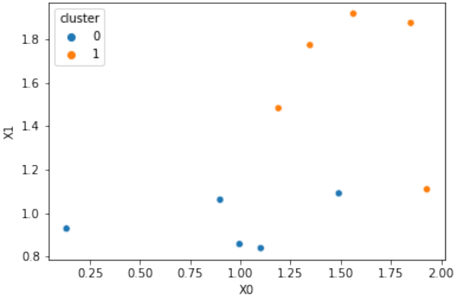
2-d Data clustered



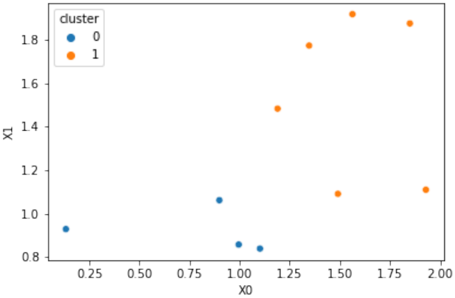
The clustering function maps points to clusters

Clustering - two different clustering functions

clust^a



clust^b



Which of the two is better, i.e. maximises intra–cluster similarity?

A measure is needed

Ideas for a measure?

- the sum of the distances between a point and the others in the same cluster?
    -
- the average of the distances between a point and the others in the same cluster?
    -
- the sum of the squared distances?
    -
- we could choose a point which, in some sense, represents the points of the cluster
    -
- . . .

Ideas for a measure? Discussion

- the sum of the distances between a point and the others in the same cluster?
  - bigger for bigger clusters
- the average of the distances between a point and the others in the same cluster?
  - not influenced by cluster size
- the sum of the squared distances?
  - more penalisation for sparsity
- we could choose a point which, in some sense, represents the points of the cluster
  - what about the average of the coordinates?
- ...

Centroid

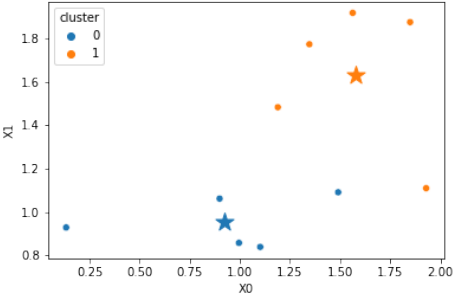
- a point with coordinates computed as the average of the coordinates of all the points in the cluster
- in physics it is the center of gravity of a set of points of equal mass
- for each cluster k and dimension d, the d coordinate of the centroid is

$$centroid_d^k = \frac{1}{|x_i : clust(x_i) = k|} \sum_{x_i : clust(x_i) = k} x_{id}$$

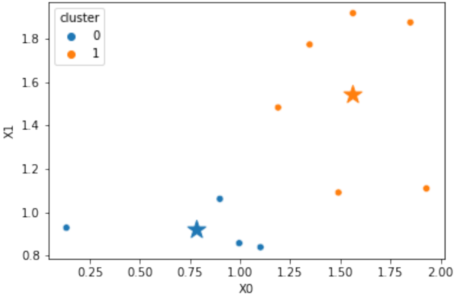
Clustering - two different clustering functions

The stars represent the centroids

$clust^a$



$clust^b$



Which of the two is better, i.e. maximises intra–cluster similarity?

A measure is needed

Taxonomy of the clustering methods

- Partitioning
  - K-means (MacQueen 67), expectation maximization (Lauritzen 95), CLARANS (Ng and Han 94)
- Hierarchic
  - agglomerative/divisive, BIRCH (Zhang et al 96), CURE (Guha et al 98)
- Based on linkage
- Based on density
  - DBSCAN (Ester et al 96), DENCLUE (Hinnenburg and Keim 98)
- Statistics
  - IBM-IM demographic clustering, COBWEB (Fisher 87), Autoclass (Cheeseman 96)

1 Introduction to clustering 2

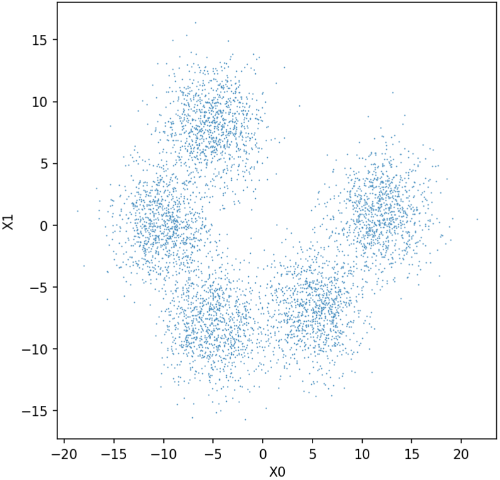
2 K-means 12

- Minimize distortion 24
- Issues about K–means 30

3 Evaluation of a clustering scheme 42

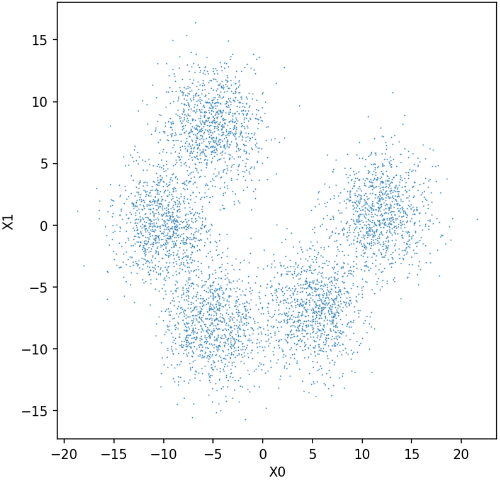
Some data

- Could be modeled as a five component gaussian mixture
  - . . . statistician voice. . .
- How do we guess the number five in a D–dimensional space (with $D \geqslant 2$)?



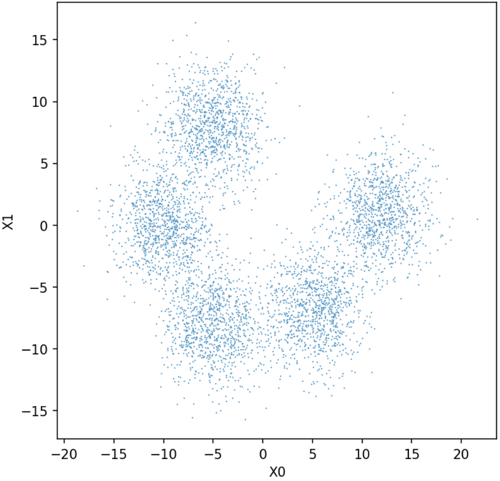
Transmission

- Transmit the coordinates of points
- Allow only two bits per point
  - the transmission will be lossy
- Need a coding/decoding mechanism



How much loss?

- Sum of the squared errors between the real points and their encoding/decoding
- Which encoding/decoding minimizes the loss?

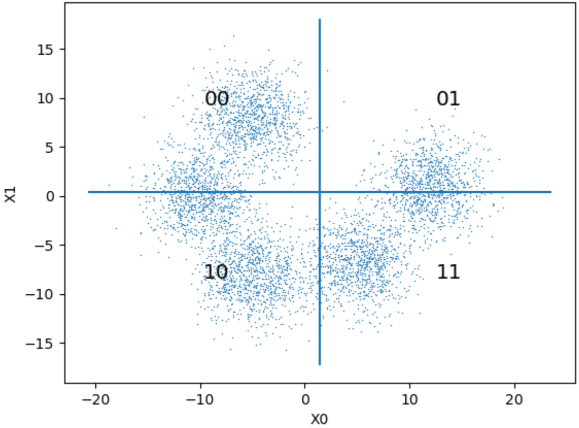


How much loss?

- Sum of the squared errors between the real points and their encoding/decoding

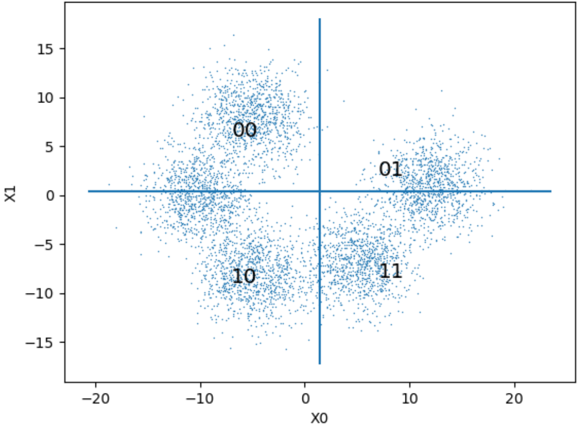
First idea

- partition the space into a grid of cells
- decode each pair of bits with the center of the grid cell



Improvement

- partition the space into a grid of cells
- decode each pair of bits with the centeroid of the points in the grid cell



Let’s be more practical

1. Try a different number of clusters

   in this case we can just look at the number of “clouds”

Let’s be more practical

1. Try a different number of clusters

   in this case we can just look at the number of “clouds”

2. Let’s emerge the splitting of the space from the data

Let’s be more practical

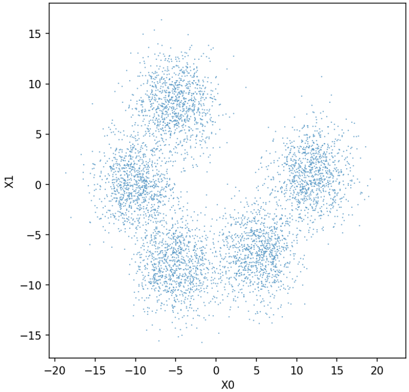
1. Try a different number of clusters

   in this case we can just look at the number of “clouds”

2. Let’s emerge the splitting of the space from the data
3. Let’s try a random start

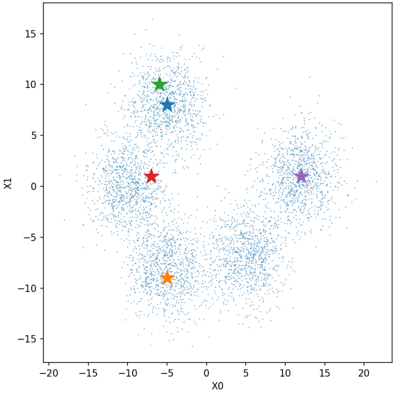
kMeans at work

1. Ask the user the number of clusters $K \Rightarrow 5$



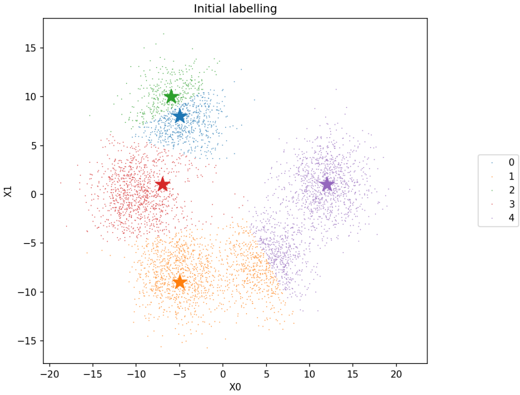
kMeans at work

1. Ask the user the number of clusters $K \Rightarrow 5$
2. Random choice of K points as temporary centers



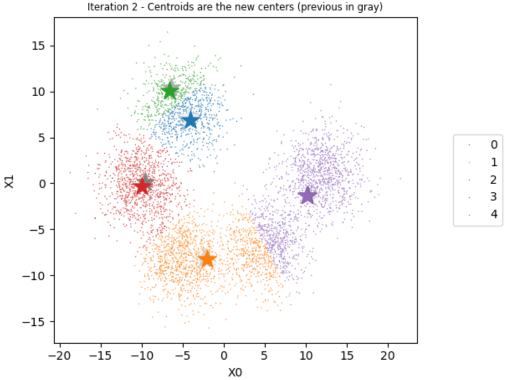
kMeans at work

1. Ask the user the number of clusters $K \Rightarrow 5$
2. Random choice of K points as temporary centers
3. For each point find his nearest center and label it



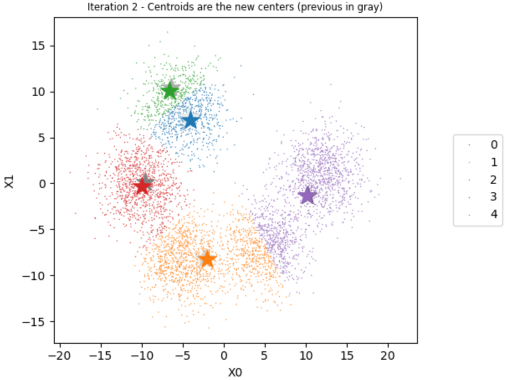
kMeans at work

1. Ask the user the number of clusters $K \Rightarrow 5$
2. Random choice of K points as temporary centers
3. For each point find his nearest center and label it
4. For each center find the centroid of its points and move there the center

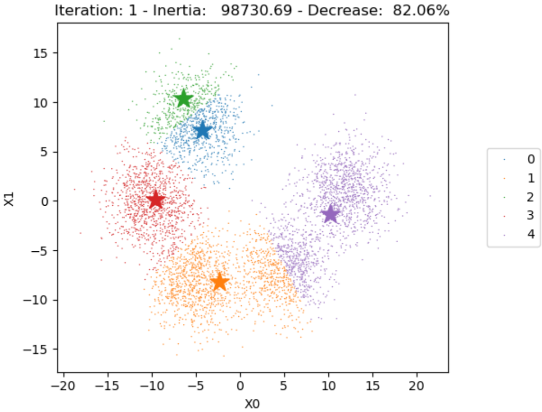


kMeans at work

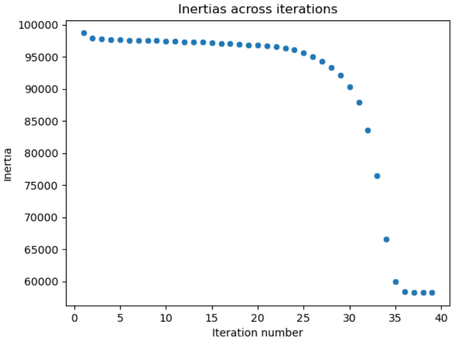
1. Ask the user the number of clusters $K \Rightarrow 5$
2. Random choice of K points as temporary centers
3. For each point find his nearest center and label it
4. For each center find the centroid of its points and move there the center
5. . . . repeat from 3 till end!

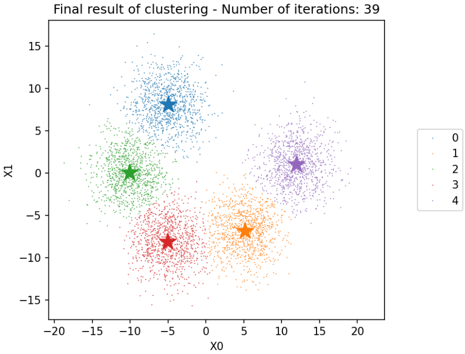


kMeans at work



K–means ends


Inertias across iterations
Inertia
100000
95000
90000
85000
80000
75000
70000
65000
60000
0
5
10
15
20
25
30
35
40
Iteration number


Final result of clustering - Number of iterations: 39
X1
15
10
5
0
−5
−10
−15
−20
−15
−10
−5
0
5
10
15
20
X0
0
1
2
3
4

Questions

1. What are we trying to optimize?
2. Is termination guaranteed?
3. Are we sure that the best clustering scheme is found?
   3.1 Which is the definition of best clustering scheme?
4. How should we start?
5. How can we find the number of clusters?

Question 1: Distortion

Frequetly called in the literature **Inertia**

Given:

a dataset	$\{x_i, i = 1 \ldots\ N\}$
a coding function	$\text{Encode} : \{R\}\hat{}D \rightarrow [1..K]$
a decoding function	$\text{Decode} : [1..K] \rightarrow \mathbb{R}^D$
define	$\text{Distortion} = \sum_{i=1}^{N} (x_i - \text{Decode}(\text{Encode}(x_i)))^2$
shortcut	$\text{Decode}(k) = \mathbf{c}_k$

then

$$\text{Distortion} = \sum_{i=1}^{N} (x_i - \mathbf{c}_{\text{Encode}(x_i)})^2$$

Minimal distortion I

$$\text{Distortion} = \sum_{i=1}^{N} (x_i - \mathbf{c}_{\text{Encode}(x_i)})^2$$

Which properties are requested to $\mathbf{c}_1, \ldots, \mathbf{c}_K$ for the minimal distortion?

1. x_i must be encoded with the nearest center

Why?
Because otherwise the distortion could be reduced by substituting Encode(x_i) with the nearest center

$$\mathbf{c}_{\text{Encode}(x_i)} = \underset{\mathbf{c}_j \in \{\mathbf{c}_1, \ldots, \mathbf{c}_K\}}{\text{argmin}} (x_i - \mathbf{c}_j)^2$$

Minimal distortion II

$$\text{Distortion} = \sum_{i=1}^{N} \left(x_i - \mathbf{c}_{\text{Encode}(x_i)}\right)^2$$

Which properties are requested to $\mathbf{c}_1, \ldots, \mathbf{c}_K$ for the minimal distortion?
2. The partial derivative of distortion w.r.t. the position of each center must be zero

Why?
Because in that case the function has either a maximum or a minimum

Step 2. The partial derivative of distortion w.r.t. the position of each center must be zero

Optional

$$
\begin{aligned}
\text{Distortion} &= \sum_{i=1}^{N} \left(x_i - \mathbf{c}_{\text{Encode}(x_i)}\right)^2 \\
&= \sum_{j=1}^{K} \sum_{i \in OwnedBy(\mathbf{c}_j)} (x_i - \mathbf{c}_j)^2 \\
\frac{\partial\, \text{Distortion}}{\partial \mathbf{c}_j} &= \frac{\partial}{\partial \mathbf{c}_j} \sum_{i \in OwnedBy(\mathbf{c}_j)} (x_i - \mathbf{c}_j)^2 \\
&= -2 \sum_{i \in OwnedBy(\mathbf{c}_j)} (x_i - \mathbf{c}_j)
\end{aligned}
$$

2. The partial derivative of distortion w.r.t. the position of each center must be zero

When distortion is minimal

$$\mathbf{c}_j = \frac{1}{|OwnedBy(\mathbf{c}_j)|} \sum_{i \in OwnedBy(\mathbf{c}_j)} x_i$$

Minimal distortion III

$$\text{Distortion} = \sum_{i=1}^{N} \left(x_i - \mathbf{c}_{\text{Encode}(x_i)}\right)^2$$

Which properties are requested to $\mathbf{c}_1, \ldots, \mathbf{c}_K$ for the minimal distortion?

1. x_i must be encoded with the nearest center
2. each center must be the centroid of the points it owns

Algorithm: Improving a sub–optimal solution

$$\text{Distortion} = \sum_{i=1}^{N} \left(x_i - \mathbf{c}_{\text{Encode}(x_i)}\right)^2$$

Which properties are requested to $\mathbf{c}_1, \ldots, \mathbf{c}_K$ for the minimal distortion?

1. x_i must be encoded with the nearest center
2. each center must be the centroid of the points it owns

$\Rightarrow$ Alternately perform steps 1 and 2

It can be proven that after a finite number of steps the system reaches a state where neither of the two operations changes the state

Why?

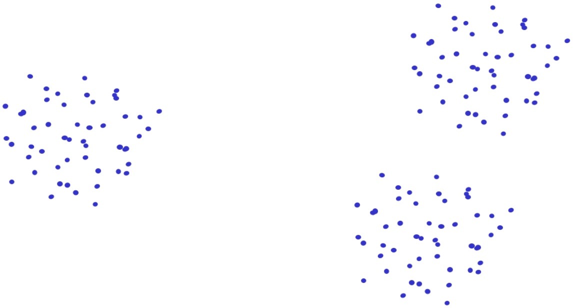
Question 2: Algorithm termination

- There is only a finite number of ways to partition N objects into K groups
- The state of the algorithm is given by the two encode/decode functions
- The number of configurations where all the centers are the centroids of the points they own is finite
- If after one iteration the state changes, the distortion is reduced
- Therefore each change of state bring to a state which was never visited before
- In summary, sooner or later the algorithm will stop because there are no new states reachable

Question 3: Local or global minimum?

Is the ending state the best possible?

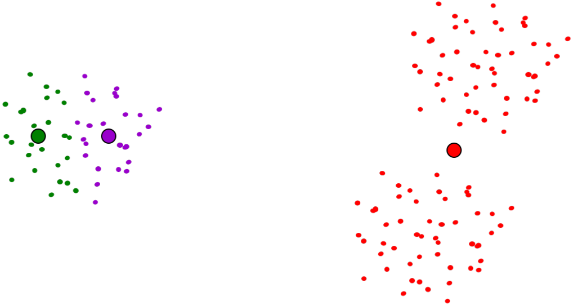
- Not necessarily
- An example



Question 3: Local or global minimum?

Is the ending state the best possible?

- Not necessarily
- An example



Question 4: Looking for a good ending state

- The starting point is important
  - choose randomly the first starting point
  - choose in sequence the 2..Kstarting points as far as possibile from the preceding ones
- Re-run the algorithm with different starting points

Question 5: Choose the number of clusters

not so easy...

- try various values
- use a quantitative evaluation of the quality of the clustering scheme to decide among the different values
- the best value finds the optimal compromise between the minimization of intra-cluster distances and the maximization of the inter cluster distances

The proximity function

- The most obvious solution, used in the previous formulas is the euclidean distance
  - good choice, in general, for vector spaces
- Several alternative solutions for specific data types and data sets
  - see the "Data" module for additional discussions

Sum of Squared Errors I

The official name of the distortion

$$
\begin{aligned}
\mathrm{SSE} &= \sum_{j=1}^{K} \sum_{i \in OwnedBy(\mathbf{c}_j)} (x_i - \mathbf{c}_j)^2 \\
&= \sum_{j=1}^{K} \mathrm{SSE}_j
\end{aligned}
$$

Sum of Squared Errors II

- A cluster j with high SSE_j has low quality
- $SSE_j = 0$ if and only if all the points are coincident with the centroid
- SSE decreases for increasing K, is zero when $K = N$
- $\Rightarrow$ minimizing SSE is not a viable solution to choose the best K
  - more discussions on this in the section on “Evaluation of the quality of a clustering scheme”

Outliers

- are points with high distance from their centroid
    - high contribution to *SSE*
- have a bad influence on the clustering results
    - sometimes it is a good idea to remove them
    - the choice is related to the application domain

Common uses of K–means

- It can be easily used in the beginning, for the esploration of data
- In a one-dimension space it is a good way to discretize the values of a domain in non-uniform buckets
- It is the basis for vector quantization, a classical technique for signal processing and compression
- Used for choosing the color palettes
  - gif compressed images: color quantization

Complexity

Given:

- T number of iterations
- K number of clusters
- N number of data points
- D number of dimensions

the time complexity is

$$\mathcal{O}(TKND)$$

Pros and cons of K–means

- Strong points
  - fairly efficient, nearly linear in the number of data points
    - in general $T, K, D << N$
- Weak points
  - in essence it is defined for spaces where the centroid can be computed
    - e.g. when the Euclidean distance is available, also other distance functions work well
    - cannot work with nominal data
  - requires the K parameter
    - nevertheless the best K can be found with iterations
  - it is very sensitive to outliers
  - does not deal with noise
  - does not deal properly with non convex clusters

Evaluation of a clustering scheme

- It is related only to the result, not to the clustering technique
- Clustering is a non supervised method
    - the evaluation is critical, because whenever there is very little apriori information, such as class labels
    - we need one or more score function to measure various properties of the clusters and of the clustering scheme as a whole
        - in the literature the words score and index are considered synonyms in this context
    - if some supervised data are available, they can be used to evaluate the clustering scheme
- In 2D the clusters can be examined visually
- In higher order spaces the 2D projections can help, but in general it is better to use more formal methods

Issues on the evaluation of clustering

- Distinguish patterns from random apparent regularities
- Find the best number of clusters
- Non supervised evaluation
- Supervised evaluation
- Relative comparison of clustering schemes

Proximity and others

- Similarity – Proximity
  - a two variable function measuring how much two objects are similar, according to the values of their properties
- Dissimilarity
  - a two variable function measuring how much two objects are different, according to the values of their properties
  - e.g. the Euclidean distance

Global separation of a clustering scheme

SSB – Sum of Squares Between clusters

$$\mathbf{c} = \text{global centroid of the dataset}$$

$$\text{SSB} = \sum_{i=1}^{K} N_i \, \text{Dist}(\mathbf{c}_i, \mathbf{c})^2$$

Link between cohesion and separation – I

- TSS = Total Sum of Squares
    - sum of squared distances of the points from the global centroid
- TSS = SSE + SSB
- the total sum of squares is a global property of the dataset, independent from the clustering scheme
- for a given dataset, minimise SSE $\Leftrightarrow$ maximise SSB

Link between cohesion and separation – II

Optional

$$\begin{aligned}
\mathrm{TSS} &= \sum_{i=1}^{K} \sum_{x \in k_i} (x - \mathbf{c})^2 = \sum_{i=1}^{K} \sum_{x \in k_i} ((x - \mathbf{c}_i) - (\mathbf{c} - \mathbf{c}_i))^2 \\
&= \sum_{i=1}^{K} \sum_{x \in k_i} (x - \mathbf{c}_i)^2 - 2 \sum_{i=1}^{K} \sum_{x \in k_i} (x - \mathbf{c}_i)(\mathbf{c} - \mathbf{c}_i) + \sum_{i=1}^{K} \sum_{x \in k_i} (\mathbf{c} - \mathbf{c}_i)^2 \\
&= \sum_{i=1}^{K} \sum_{x \in k_i} (x - \mathbf{c}_i)^2 + \sum_{i=1}^{K} \sum_{x \in k_i} (\mathbf{c} - \mathbf{c}_i)^2 \\
&= \sum_{i=1}^{K} \sum_{x \in k_i} (x - \mathbf{c}_i)^2 + \sum_{i=1}^{K} |k_i| (\mathbf{c} - \mathbf{c}_i)^2 = \mathrm{SSE} + \mathrm{SSB}
\end{aligned}$$

since $\sum_{x \in k_i} (x - \mathbf{c}_i) = 0$ by definition of $\mathbf{c}_i$

Evaluation of specific clusters and objects

- Each cluster can have its own evaluation
  - the worst clusters can be considered for additional split
- A weakly separated pair of clusters could be considered for merging
- Single objects can give negative contribution to the cohesion of a cluster or to the separation between two clusters
  - border objects

Silhouette score of a cluster – I

Requirements for a clustering quality score

- values are in a standard range, e.g. $-1, 1$
- increases with the separation between clusters
- decreases for clusters with low cohesion

$$
\begin{aligned}
\text{SSE} &= \sum_{j=1}^{K} \sum_{i \in OwnedBy(\mathbf{c}_j)} (x_i - \mathbf{c}_j)^2 \\
&= \sum_{j=1}^{K} \text{SSE}_j
\end{aligned}
$$

Silhouette score of a cluster – I

Requirements for a clustering quality score

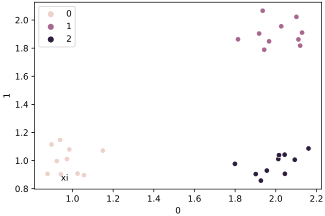
- values are in a standard range, e.g. $-1, 1$
- increases with the separation between clusters
- decreases for clusters with low cohesion, or, in other words, with high sparsity

Silhouette score of a cluster – II

Consider the individual contribution of each object, say x_i

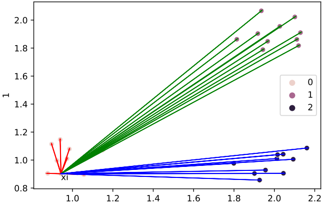
Contribution to cluster sparsity: the average of red distances

$$a_i = \underset{j, y(x_j)=y(x_i)}{\text{average}} \; dist(x_i, x_j)$$



Contribution to separation from other clusters: the minimum of the two averages of green and blue distances

$$b_i = \min_{k \in \mathcal{Y}, k \neq y(x_i)} \left(\underset{j, y(x_j)=k}{\text{average}} \, dist(x_i, x_j) \right)$$



Silhouette score of a cluster – III

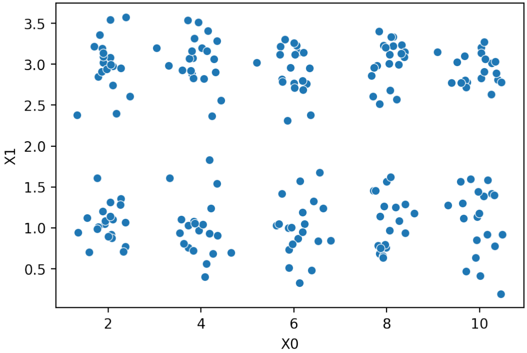
- Silhouette score of x_i

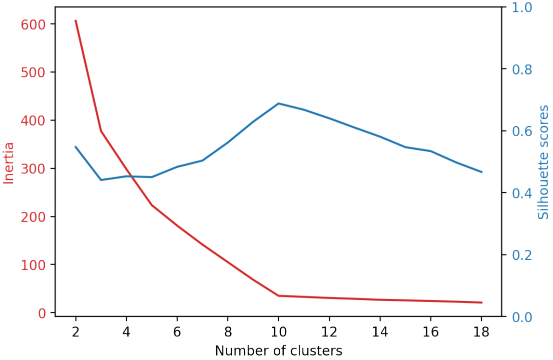
$$s_i = \frac{b_i - a_i}{\max(a_i, b_i)} \in [-1, 1]$$

- For the global score of a cluster/clustering scheme compute the average score over the cluster/dataset
- Intuition
  - when the score is less than zero for an object it means that there is a dominance of objects in other clusters at a distance smaller than objects of the same cluster

Example: Inertia and silhouette scores

Testing K-means with different numbers of clusters





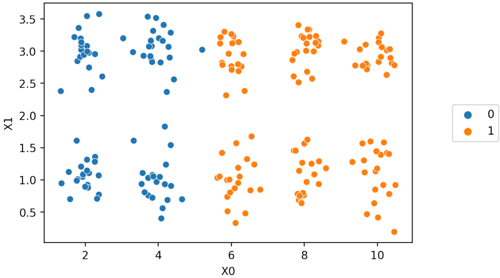
Looking for the best number of clusters – I

- Some algorithms, such as K-means, require the number of clusters as a parameter
- Measures, such as SSE and Silhouette, are obviously influenced by the number of clusters
    - they can be used to optimize K
- Computation of Silhouette score is expensive
- SSE decreases monotonically for increasing K
    - is equal to TSS for $K = 1$
    - goes to zero when $K = N$

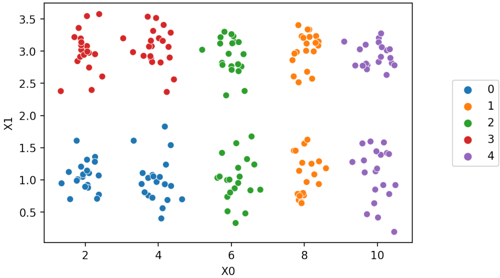
Looking for the best number of clusters – II

- The inertia varying K has frequently one or more points where the slope decreases: one of this points is frequently a plausible value for K
    - this is called elbow method
- The silhouette score varying K has frequently a maximum, in this case it indicates the best value for K

K–means results on the dataset of page 54



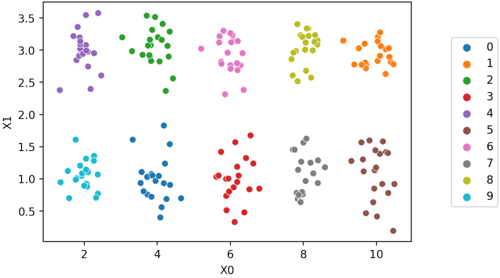
$K = 2$

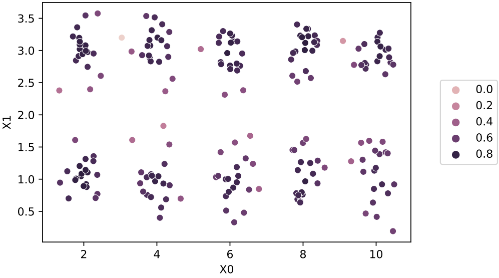


$K = 5$

K–means results on the dataset of page 54

Best silhouette for $K = 10$





Supervised measures: Gold Standard I

- Let be available a partition of a dataset similar to the data to be clustered, which we call gold standard, and defined by a labelling scheme $y_g(.)$
    - it is the same as the labels attached to supervised data for training a classifier
- Consider a clustering scheme $y_k(.)$
    - the cardinalities of the sets of distinct labels generated by the two schemes $\mathcal{V}_g$ and $\mathcal{V}_k$ can be different, and also in case of identity of the two grouping schemes, a permutation of labels could be necessary to make them equal

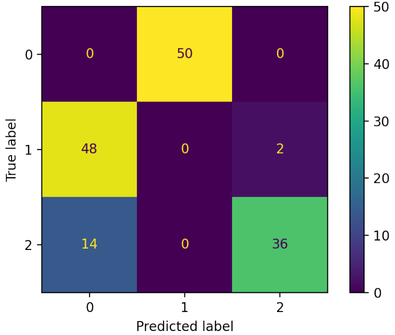
Why should we compare it with the gold standard?

- validate a clustering technique which can be applied later to new, unlabelled data
- the purpose is quite similar to testing a classifier
- the difference is that in this case we are more interested in grouping new data than in labelling them following the Gold Standard scheme

Classification–oriented measures

- Measure how the gold standard classes are distributed among the clusters
    - confusion matrix, precision, recall, f-measure
- On the right the confusion matrix for the Iris dataset
    - best match with permutation[1] of the predicted labels $0 \rightarrow 1, 1 \rightarrow 0, 2 \rightarrow 2$

```
X,y = load_iris(return_X_y=True)
estimator = KMeans(n_clusters=3
                   , random_state=363)
y_km = estimator.fit_predict(X)
disp = ConfusionMatrixDisplay(confusion_matrix(y,y_km))
disp.plot()
```



Similarity oriented measures - I

- Analogous to compare binary data
- Consider a clustering scheme $y_k(.)$ and compare it with the Gold Standard $y_g(.)$
- Any pair of objects can be labelled as
  - *SGSK* if they belong to the same set in $y_g(.)$ and $y_k(.)$
  - *SGDK* if they belong to the same set in $y_g(.)$ and not in $y_k(.)$
  - *DGSK* if they belong to the same set in $y_k(.)$ but not in $y_g(.)$
  - *DGDK* if they belong to different sets both in $y_g(.)$ and $y_k(.)$

Similarity oriented measures - II

Results given by `pair_confusion_matrix(y_g,y_k)`

	SK	DK
SG	13512	1488
DG	1200	6150

Rand Score $\frac{SGSK+DGDK}{SGSK+DGDK+SGDK+DGSK} = 0.88$

Adjusted Rand Score Excludes the count of matches expected by chance[1] $= 0.73$

Jaccard Coefficient for label c $\frac{SG_cSK_c}{SG_cSK_c+SG_cDK_c+DG_cSK_c} = (1, 0.75, 0.69)$

- it requires remapping of $y_g(.)$ to obtain the best match

[1] See the Wikipedia page for a reference

Bibliography I

- A. K. Jain, M. N. Murty, and P. J. Flynn.
  Data clustering: A review.
  ACM Comput. Surv., 31(3):264–323, September 1999.
  ISSN 0360-0300.
  doi: 10.1145/331499.331504.
  URL http://doi.acm.org/10.1145/331499.331504.