Clustering

Machine Learning, A.Y. 2022/23, Padova

Fabio Aiolli

December 12th 2022

Clustering

- Clustering is the process of grouping a set of objects into groups of similar objects
- The most common form of unsupervised learning
- In contrast to supervised learning, in unsupervised learning we do not have a classification of examples
- Common and very important task that has countless applications
- Not just clustering of examples. For example, clustering of features.

The clustering problem

Given:

- A set of instances $X = \{x_1, \ldots, x_n\}$, $x_i \in \mathcal{X}$
- A measure of similarity
- A desired number of clusters K

Compute:

- An assignment function $\gamma : X \to \{1, \ldots, K\}$ such that no clusters are empty. No relationship of order between the clusters.

The clustering problem is an inherently ill-posed problem. The notion of a group is vague and arbitrary (e.g. fruits can be clustered by shape or color and in both cases clustering would make perfect sense).

Choices to make in clustering

- Representation for clustering
  - Crucial step to get good results from clustering. This choice will be made using any prior knowledge.
  - Data preprocessing. Vector space? Normalization?
  - Need for a notion of similarity/distance (kernels?)
- How many clusters?
  - Fixed a priori? still exploiting any a priori knowledge about the problem
  - Completely data driven?
  - Avoid "trivial" clusters - too big or too small

Objective function

- Often, the goal of a clustering algorithm is to optimize an objective function
- In these cases, clustering is a search (optimization) problem
- $K^n/K!$ different clusterings possible
- Most partitioning algorithms start with an initial (partition) assignment and then refine it
- Having many local minima in the objective function implies that different starting points can lead to very different (and not optimal) final partitions

What is a good clustering? Assessment

- Internal criteria which depend on the notion of similarity and/or on the chosen representation. We evaluate the intra-class similarity (it should be high) and inter-class similarity (it should be low).
- External criteria that, given an external "ground truth", measure its "proximity" to the clustering produced

Internal methods for evaluation

A clustering is a good clustering when it produces clusters in which:

- the intra-class similarity (between examples in the same cluster) is high
- the inter-class similarity (between examples in different clusters) is low
- note that this quality measure strongly depends on the chosen representation and on the similarity measure (distance) used to calculate it

Example:

$$V(X, \gamma) = \sum_{k=1}^{K} \sum_{i:\gamma(x_i)=k} ||\mathbf{x}_i - \mathbf{c}_k||^2$$

where $\mathbf{c}_k$ is the centroid of the k-th cluster (i.e. the mean of the examples assigned to the k-th cluster)

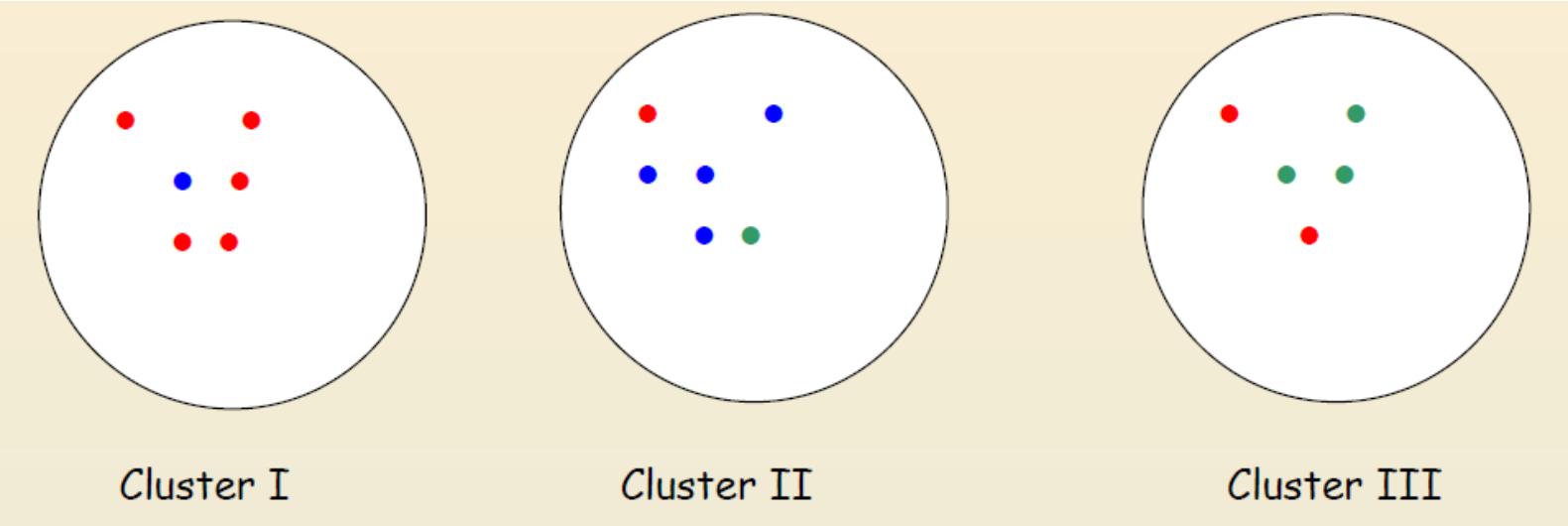
External methods for evaluation

- Quality is measured as the demonstrated ability to recognize some or all hidden patterns and/or latent classes in the data.
- We have a classification (ground truth) for the data we have clustered (this ground truth was NOT used to produce clustering!) and we want to measure how much the clustering produced resembles this ground truth
- Let's assume examples of C different classes clustered on K clusters $\omega_1, \ldots, \omega_K$
- Evaluation methods used for classification are not directly usable. Why?

External methods for evaluation

- Purity, that is the ratio between the number of elements of the dominant class in a cluster and the cardinality of the cluster
- RandIndex, similar to the notion of accuracy used in classification, considering for each pair of examples whether they have been correctly distributed in the clusters (i.e. they are in the same cluster if and only if they are of the same class in the ground truth)
- Other methods such as entropy (or mutual information) among classes of the ground truth and the clusters produced

Purity: an example



- Cluster I : $Purity(1) = \frac{max(5,1,0)}{6} = 5/6$
- Cluster II : $Purity(2) = \frac{max(1,4,1)}{6} = 4/6$
- Cluster III: $Purity(3) = \frac{max(0,2,3)}{5} = 3/5$

The total purity will be the average of the purity in different clusters.

RandIndex

With the RandIndex evaluation method, for each pair of examples, we calculate the following statistic:

- A: number of pairs of the same class assigned to the same cluster (true positives)
- B: number of pairs of different class assigned to the same cluster (false positives)
- C: number of pairs of the same class assigned to different clusters (false negatives)
- D: number of different class pairs assigned to different clusters (true negatives)

Number of points	Same Cluster in clustering	Different Clusters in clustering
Same class in ground truth	A (tp)	C (fn)
Different classes in ground truth	B (fp)	D (tn)

RandIndex: definition

$$RI = \frac{A + D}{A + B + C + D}$$

Number of points	Same Cluster in clustering	Different Clusters in clustering
Same class in ground truth	20	24
Different classes in ground truth	20	72

We can also consider measures corresponding to Precision and Recall, that is:

$$P = \frac{A}{A + B} \qquad R = \frac{A}{A + C}$$

Adjusted RandIndex

Let n_{ij} denote the number of objects in common between X_i and Y_j.
The Adjusted RandIndex (ARI) is:

$$ARI = (RI - Expected\_RI)/(max(RI) - Expected\_RI)$$

$X \backslash Y$	Y_1	Y_2	$\cdots$	Y_s	sums
X_1	n_{11}	n_{12}	$\cdots$	n_{1s}	a_1
X_2	n_{21}	n_{22}	$\cdots$	n_{2s}	a_2
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
X_r	n_{r1}	n_{r2}	$\cdots$	n_{rs}	a_r
sums	b_1	b_2	$\cdots$	b_s	

Then, $RI = \sum_{ij} \binom{n_{ij}}{2}$ and $ARI = \dfrac{\sum_{ij} \binom{n_{ij}}{2} - \frac{\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}}{\binom{n}{2}}}{\frac{1}{2}\left[\sum_i \binom{a_i}{2} + \sum_j \binom{b_j}{2}\right] - \frac{\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}}{\binom{n}{2}}}$

Clustering algorithms

- Partitioning (or flat) algorithms
  - They usually start with a random (partial) partition
  - Refining it iteratively (e.g. K-means clustering and model-based clustering)
- Hierarchical algorithms
  - Bottom-up or agglomerative
  - Top-down or divisive

Partitioning algorithms

They build a partition of n examples in K clusters

- Given a set of examples and a number K
- They find a K clusters partition that optimizes a certain criterion
  - Overall optimal: exhaustively enumerates all possible partitions (definitely inefficient!)
  - Methods based on heuristics: k-means and k-medoids are examples
  - Probabilistic or model-based methods: EM-type approaches (e.g. density mixtures $p(\mathbf{x}) = \sum_{i=1}^{K} p(\mathbf{x}|c_i)p(c_i)$

K-Means Algorithm: idea

- The examples are assumed to be real-valued vectors
- For each cluster we minimize the average of the distance between the examples and the "center" of the cluster (centroid):

$$\mu(c) = \frac{1}{|c|} \sum_{x \in c} x$$

- Instances are assigned to clusters based on the similarity/distance of the examples from the centroids of the current clusters

K-Means Algorithm

1. Generate K points (seeds) in the space. These points represent the initial centroids (e.g. K randomly chosen examples);
2. Assign each example to the cluster whose centroid is the closest according to the considered similarity/distance;
3. After assigning all the examples, it recalculates the position of the K centroids as means of the vectors of the examples belonging to the respective clusters;
4. Repeat steps 2 and 3 until the centroids stabilize.

It can be shown that, at each execution of steps 2 and 3, the value of the objective function is reduced:

$$V(X, \gamma) = \sum_{k=1}^{K} \sum_{i:\gamma(x_i)=k} ||\mathbf{x}_i - c_k||^2$$

Hierarchical Algorithms

The optimal number K of clusters is not usually given. Hierarchical algorithms are a good alternative in these cases.

We build a tree-based taxonomy from a set of examples representing the cluster structure (dendrogram).

A first approach is to consider the recursive application of a partitioning clustering algorithm (divisive or top-down hierarchical algorithm)

A second, more popular approach, is through aggregation of clusters in a bottom-up mode ...

Agglomerative Hierarchical Clustering (HAC)

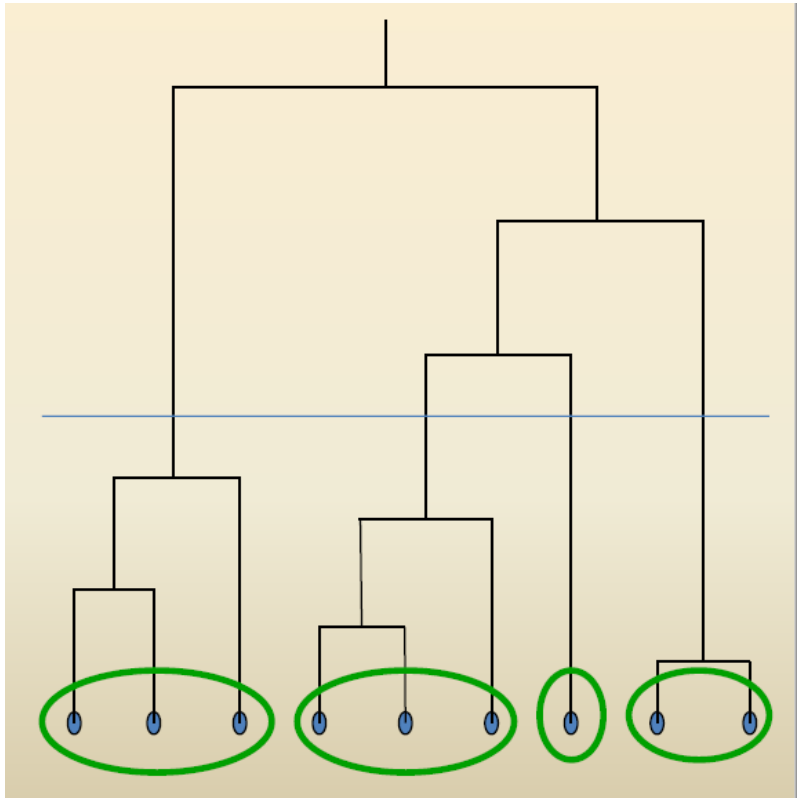
- Let's start with singleton clusters, one for each example. Then, we gradually merge the closest pairs of clusters, until we obtain a single cluster;
- The history of merges forms a binary tree or hierarchy (dendrogram).

What does "closest pairs of clusters" mean? How can we measure the distance between two clusters?

- Single-link: Similarity among the most similar examples in clusters;
- Complete-link: Similarity among the most distant examples in the clusters;
- Centroid: Similarity between the centroids of the clusters;
- Average-link: Mean similarity between pairs of examples of clusters.

Dendrogram for agglomerative algorithms

- The y axis of the dendrogram represents the similarity of the combination, i.e. the similarity between the two clusters being merged;
- The merge operation is assumed to be monotone, i.e. if $s_1, \ldots, s_k$ are successive similarities obtained by the merger, then $s_1 > s_2 > \cdots > s_k$;
- Clustering is achieved by "cutting" the dendrogram to the desired level: each connected component is a cluster.



Summary on agglomerative hierarchical clustering

Single-link	Max sim of any two points	$O(N^2)$	Chaining effect
Complete-link	Min sim of any two points	$O(N^2 \log N)$	Sensitive to outliers
Centroid	Similarity of centroids	$O(N^2 \log N)$	Non monotonic
Group-average	Avg sim of any two points	$O(N^2 \log N)$	OK

Recap

Notions

- Definition of clustering
- Internal clustering evaluation methods
- External clustering evaluation methods (Purity and RandIndex)
- Kmeans (partitioning) and HAC (agglomerative hierarchical) algorithms

Exercizes

- Datasets for clustering (p.e. http://cs.joensuu.fi/sipu/datasets/)
- Try the methods described in sklearn
- Extending k-means with kernels (using the kernel trick)