Machine Learning

Clustering II

Prof. Matthias Hein

Machine Learning Group
Department of Mathematics and Computer Science
Saarland University, Saarbrücken, Germany

Lecture 25, 31.01.2014

Roadmap

Clustering

- Goal of clustering,
- k-means clustering (prototype-based clustering)
- Spectral clustering (graph-based clustering),
- Agglomerative and hierarchical clustering,
- Density based clustering.

Clustering is one instance of unsupervised learning

What is clustering ?

Clustering:
Construction of a grouping of the points into sets of *similar* points, the so called *clusters*.

- no generally accepted objective for clustering $\Longrightarrow$ without specifying a suitable objective clustering is **ill-defined**,
- clustering objective depends largely on application,
- in clustering the modelling aspect is even more important than in supervised learning $\Longrightarrow$ do not use a clustering method if you have not understood what the objective implies !

Hierarchical clustering

Hierarchical clustering
generates a hierarchical representation of the *n* data points.

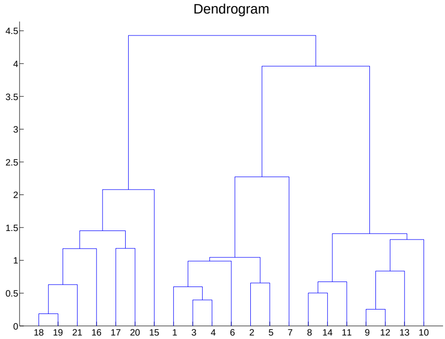
- **agglomerative:** start with all *n* points as individual clusters and consecutively join cluster which are *most similar*,
- **divisive:** start with one cluster containing all *n* points and consecutively divide the clusters so that they are *most dissimilar*.

$\Longrightarrow$ generates a tree structure on the data - the **dendrogram**.

Hierarchical clustering II

Definition

A **dendrogram** is a binary tree with a distinguished root, that has the data points as its leaves. The height where two clusters are merged is equal to their dissimilarity.



Agglomerative hierarchical clustering

Agglomerative hierarchical clustering

Requirement: a distance measure between point sets.

Definition

A **dissimilarity measure D between finite subsets** of $\mathcal{X}$ is defined as $D : 2^{\mathcal{X}} \times 2^{\mathcal{X}} \to \mathbb{R}$ with

- $D(A, B) \geq 0$ for all $A, B \subseteq \mathcal{X}$,
- $D(A, B) = 0$ if and only if $A = B$,
- $D(A, B) = D(B, A)$.

Note: triangle inequality not required - not necessarily a metric.

Agglomerative hierarchical clustering II

Algorithm:

- given: set of n points in $\mathcal{X}$, dissimilarity D between subsets of $\mathcal{X}$.
- initialize: we have n clusters at level n, $C_1^{(n)}, \ldots, C_n^{(n)}$ with $C_i^{(n)} = \{x_i\}$.
- **do**
  1. compute for all l clusters in $C_1^{(l)}, \ldots, C_l^{(l)}$ their dissimilarity $d_{ij} = D(C_i^{(l)}, C_j^{(l)})$
  2. merge the least dissimilar clusters, with indices $(r, s) = \underset{1 \le i,j \le l,\, i \ne j}{\arg\min}\ d_{ij}$.
  3. for $i \neq r$ and $i \neq s$, $C_i^{(l-1)} = C_i^{(l)}$ and $C_r^{(l-1)} = C_r^{(l)} \cup C_s^{(l)}$.
  4. height in the dendrogram of the merger between $C_r^{(l)}$ and $C^{(l)}$ is
  $$\alpha^{(l)} = d_{rs} = \min_{i,j} d_{ij}.$$
  5. relabel the clusters of level $l-1$ from 1 to $l-1$,
- **while** $l > 1$
- output: the sets of clusters $C^{(l)}$ for each level $l = 1, \ldots, n$.

Hierarchical clustering III

Agglomerative clustering:
consecutively join clusters which are *most similar*.

How to measure dissimilarity of clusters C_1 and C_2 ?

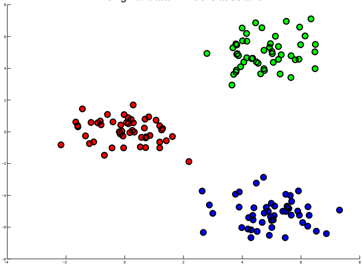
- **Single-linkage:** $d_{\min}(C_1, C_2) = \min_{i \in C_1, j \in C_2} d(x_i, x_j)$,
- **Average-linkage:** $d_{\mathrm{avg}}(C_1, C_2) = \frac{1}{|C_1| |C_2|} \sum_{i \in C_1, j \in C_2} d(x_i, x_j)$,
- **Complete-linkage:** $d_{\max}(C_1, C_2) = \max_{i \in C_1, j \in C_2} d(x_i, x_j)$,

Two clusters are similar:

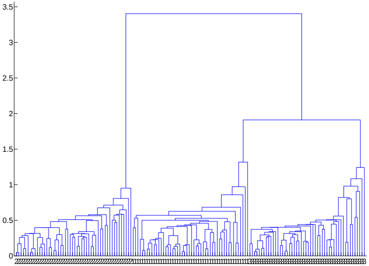
- single linkage: if for all points in each cluster there exists a path so that all points in the path are similar,
- complete-linkage: if all points for both clusters are similar,
- average-linkage: if on average the points of both clusters are similar.

Compact, spherical clusters

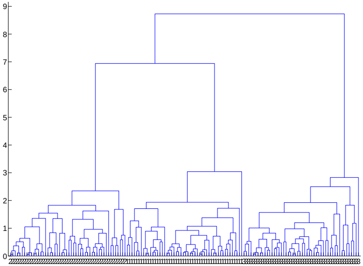
Original data: Three Gaussians



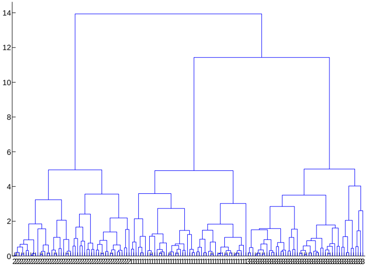
Single linkage clustering



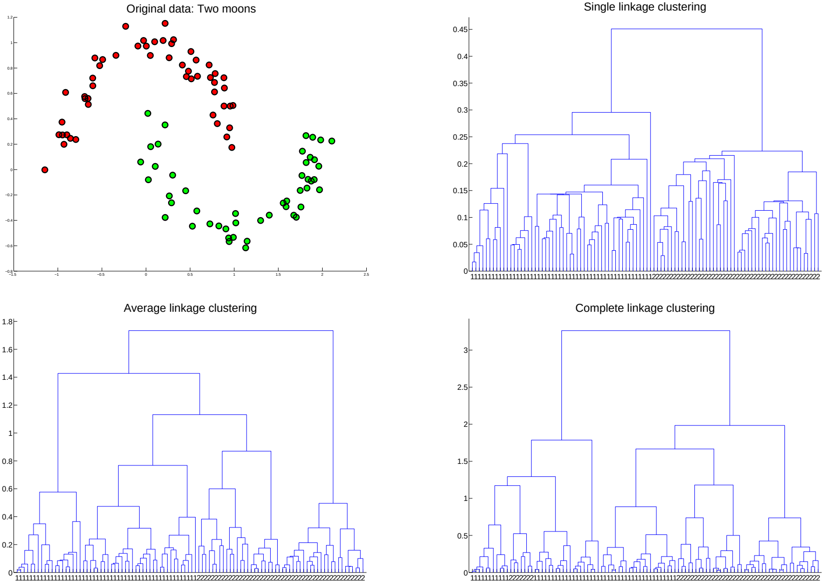
Average linkage clustering



Complete linkage clustering



Non-compact clusters



Original data: Two moons

Single linkage clustering

Average linkage clustering

Complete linkage clustering

Hierarchical clustering

Problems of dendrograms

- **instability** small changes in the data can lead to huge changes in the dendrogram,
- **hierarchy:** multi-scale partitioning but different distance measures are hard to interpret.
- **dissimilarity:** the dissimilarity of clusters at which one joins clusters encodes their dissimilarity - this is a quite strange distance measure $\Longrightarrow$ comparing data using this distance is highly non-intuitive.

Ultrametric

Definition

An **ultra-metric** d on $\mathcal{X}$ is a metric d which satisfies for all $x, y, z \in \mathcal{X}$,

$$d(x, y) \leq \max\{d(x, z),\, d(y, z)\}$$

This inequality is called **strong triangle or ultrametric inequality**.

The ultrametric inequality is stronger than the triangle inequality since

$$\begin{aligned}\max\{d(x, z),\, d(y, z)\} &\leq \max\{d(x, z),\, d(y, z)\} + \min\{d(x, z),\, d(y, z)\}\\ &= d(x, z) + d(y, z).\end{aligned}$$

$\Rightarrow$ very strange effects for this metric !

Ultrametric and hierarchical clustering

Theorem

Let D be a dissimilarity measure for sets in $\mathcal{X}$ and let $C^{(l)}$ be the induced hierarchical clustering on the set $T = \{x_1, \ldots, x_n\}$. If the dissimilarity of consecutively merged clusters is monotonically increasing, that is $\alpha^{(l)} \leq \alpha^{(m)}$ for $l > m$, then, $d' : T \times T \to \mathbb{R}$, defined as

$$\begin{aligned} d'(i,j) &= \max_{l \textit{ such that } x_i \in C_r^{(l)} \textit{ and } x_j \in C_s^{(l)} \textit{ with } r \neq s} D(C_r^{(l)}, C_s^{(l)}) \\ &= \max_{l \textit{ such that } x_i \in C_r^{(l)} \textit{ and } x_j \in C_s^{(l)} \textit{ with } r \neq s} \alpha^{(l)}, \end{aligned}$$

is an ultrametric.

$\Longrightarrow$ distance measure d' integrates the hierarchical structure.

$\Longrightarrow$ need not be much related to original distances on the data.

Ultrametric and hierarchical clustering

Proof: All properties except the triangle inequality follow from D.
Let x, y, z be three points in T. We denote by l_1 the level at which x and z are merged and by l_2 the level at which y and z are merged. Thus,

$$d'(x,z) = \alpha^{(l_1)}, \text{ and } d'(y,z) = \alpha^{(l_2)}.$$

Since the clusters are hierarchical, we have that x, y, z are in the same cluster for the level $\min\{l_1, l_2\} \Longrightarrow$ the level l_3 where the points x and y are merged is larger than or equal to $\min\{l_1, l_2\}$.
Using that $\alpha^{(l)}$ is monotonically decreasing in l, we have that $\alpha^{(l_3)} \leq \max\{\alpha^{(l_1)}, \alpha^{(l_2)}\}$ which yields,

$$d'(x,y) = \alpha^{(l_3)} \leq \max\{\alpha^{(l_1)}, \alpha^{(l_2)}\} = \max\{d'(x,z), d'(z,y)\}.$$

Single-linkage clustering and MST

Single-linkage and minimal spanning trees:

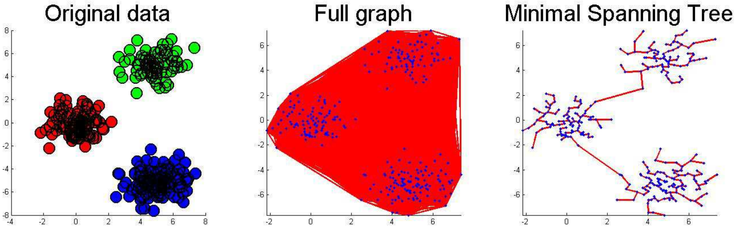
In single-linkage clustering the merging of two clusters can be interpreted as placing an edge into the graph which has as its vertex set all the data points.

- single linkage constructs a spanning tree,
- It is a Euclidean minimal spanning tree if we use the Euclidean distance for the weights.

$\Longrightarrow$ divisive clustering method by deleting the edge with the largest weight (largest distance) in the MST.

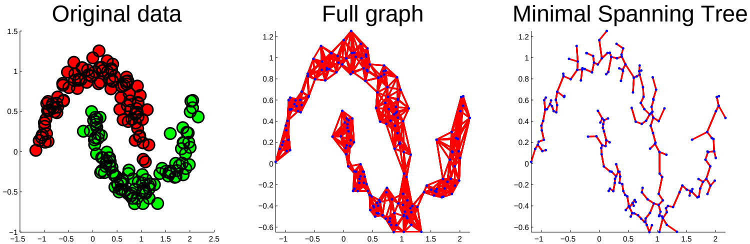
Single linkage and minimal spanning tree

The minimal spanning tree of a complete graph



Clustering using minimal spanning trees

Transfer the method to arbitrary graphs:



Original data

Full graph

Minimal Spanning Tree

Divisive clustering:

- construct hierarchical partioning of the graph by consecutively eliminating the edge with the smallest/largest edge weight.

Theoretical results

Consistency of single-linkage clustering:

Hartigan proves one of the first theoretical results for clustering (1981).

Clustering model:

- Statistical setting: data in $\mathbb{R}^d$ is drawn from some probability measure,
- The clusters are the connected components of the level set L_t

$$L_t = \{x \in \mathbb{R}^d \,|\, p(x) \geq t\},$$

  of the density to the level t.
- **Theorem:** Given that the connected components of L_t have a sufficiently large distance, there exists a threshold for single linkage such that the found clusters contain a large fraction of the corresponding points in the level set L_t.

Summary

Pro:

- nice hierarchical representation of the data,
- single-linkage has a nice theoretical foundation,
- computationally relatively cheap.

Contra:

- single-linkage and complete very sensitive to data fluctuations,
- complete linkage has problems with non-spherical clusters,
- interpretation of the data requires profound understanding of the cluster similarity measures.

Density-based clustering

Statistical setting:

- sample $\{X_i\}_{i=1}^n$ is drawn i.i.d. from probability measure in $\mathbb{R}^d$,
- the probability measure has a density in $\mathbb{R}^d$,

Clustering model: The clusters of the density p are the connected components of the level set L_t,

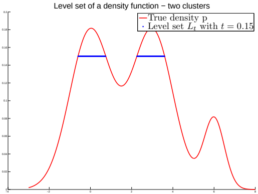
$$L_t = \{x \in \mathbb{R}^d \,|\, p(x) \geq t\},$$

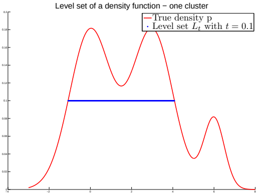
of the density to the level t.
$\Longrightarrow$ the only general model for clustering.

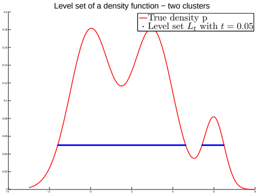
Main difference to approaches up to now

- we have clusters **and** "background noise" $\Rightarrow$ the clusters define **not** a partitioning of the space !

Level set of a density function







- Level set of a mixture of three Gaussians at three different level $t = 0.05, 0.1, 0.15$,
- different level-sets lead to multi-scale cluster analysis.

Density-based clustering II

Naive approach:

- estimate density $\hat{p}(x)$ at each point using a density estimator,
- we define the estimated level-set $\hat{L}_t$ as $\hat{L}_t = \{x \in \mathbb{R}^d \,|\, \hat{p}(x) \geq t\}$,
- compute connected components of $\hat{L}_t$.

Main ingredients:

- how to compute a density based on the sample $\{X_i\}_{i=1}^n$,
- how to compute the connected components of $\hat{L}_t$.

$\Rightarrow$ density based clustering is interesting for outlier-detection.

Density estimation

Kernel density estimation:
We need a kernel function $k : \mathbb{R} \to \mathbb{R}$ and a bandwidth h, then

$$\hat{p}_h(x) = \frac{1}{n\, h^d} \sum_{i=1}^{n} k(\|x - X_i\| / h).$$

e.g. $k(\|x - X_i\| / h) = \frac{1}{(2\pi)^{\frac{d}{2}}} \exp\left(- \frac{\|x-X_i\|^2}{2h^2}\right).$

With this choice, we have

$$\int_{\mathbb{R}^d} \hat{p}_h(x) = 1.$$

$\Rightarrow$ $\hat{p}_h$ is a true density function.
$\Rightarrow$ bandwidth parameter can be adjusted using cross-validation.

Density estimation II

Theoretical background for density estimation:

The expected value of the kernel density estimate is given as

$$\mathbb{E}[\hat{p}_h(x)] = \int_{\mathbb{R}^d} \frac{1}{h^d} k(\|x-y\|/h)\, p(y)\, dy.$$

Given $p \in C^3(\mathbb{R}^d)$, can prove using Taylor's theorem that,

$$\int_{\mathbb{R}^d} \frac{1}{h^d} k(\|x-y\|/h)\, p(y)\, dy = p(x) + O(h^2).$$

Using Bernstein's inequality one can show, for some constant $C > 0$

$$\mathrm{P}\Big(\big|\hat{p}_h(x) - \mathbb{E}[\hat{p}_h(x)]\big| > \varepsilon\Big) \leq 2e^{-C\, n\, h^d \varepsilon^2}.$$

$\Rightarrow$ thus $\hat{p}_h(x)$ converges (pointwise) towards the true density at x if $nh^d \to \infty$ as $n \to \infty$ and $h \to 0$.

Density-based clustering

Connected components of the level set:

- generate graph for all points with $\hat{p}_h(X_i) \geq t$,
- weights are generated using k-NN graph,
- compute connected components of this graph,
- generate partition of whole space by nearest neighbor partitioning.

$\Longrightarrow$ consistency of method including third step can be shown.

Other ones:

- DBSCAN,
- one-class SVM.