Clustering

Johanni Brea

Einführung Machine Learning

GYMINF 2022

Data Generating Processes Revisited

Recap

It is useful to think of our datasets as samples from **data generating processes** for the input X and the conditional output Y|X.

- **MNIST**
  X: people write digits $\rightarrow$ people take standardized photos thereof.
  Y|X: different people label the same photo X.
- **Weather**
  X: the weather acts on sensors in weather stations.
  Y|X: the weather evolves from X and is measured again 5 hours later.

Using samples from these data generating processes, supervised learning aims at learning something about the conditional processes, i.e how Y depends on X.

Using samples from these data generating processes, **unsupervised learning** aims at learning something about the input generator, i.e how X is generated.

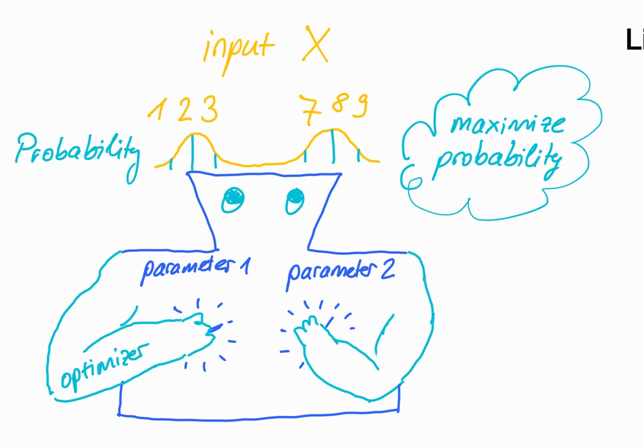
Goals of Unsupervised Learning

- **Exploratory Data Analysis**: Is there an informative way to visualize the data? Can we discover subgroups among the variables or among the observations?
- **Data Processing**: Can we separate signal from noise (denoising)? Can we efficiently compress the data?
- **Uncovering Hidden "Causes" of Observations**: Can we uncover hidden structure in the data? Does the data lie on a low-dimensional manifold?
- **Generating Artificial Data**: Can we generate high-quality novel data samples, e.g. images, text or music?

For the assessment of unsupervised learning there are often no clear objective guidelines.

Table of Contents

How Does Unsupervised Learning Work?

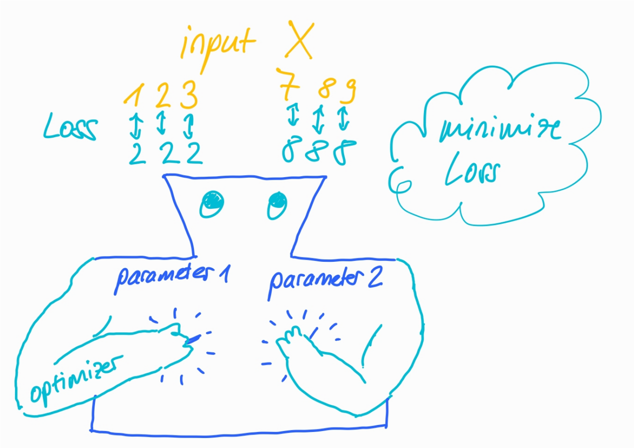


Likelihood Maximizing Machine

- We specify
  1. the training data
  2. the family of probability distributions (model)
  3. the optimizer
- The machine changes the parameters with the help of the optimizer until the likelihood of the parameters is maximal.

E.g.: Gaussian Mixture Model (not further discussed here)

How Does Unsupervised Learning Work?



Loss Minimizing Machine

- We specify
  1. the training data
  2. the function family (model)
  3. the loss function $L(x)$
  4. the optimizer
- The machine changes the parameters with the help of the optimizer until the loss is minimal.

E.g.: K-Means Clustering

Table of Contents

K-Means Clustering

- $C_1, \ldots, C_K$ contain the indices of the observations in each cluster.
- K needs to be chosen.
- Every observation with index $i = 1, \ldots, n$ is in exactly one cluster.
- Goal:

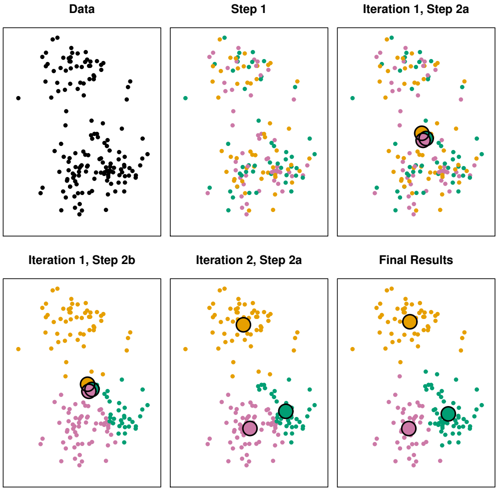
$$\underset{C_1,\ldots,C_K}{\text{minimize}} \sum_{k=1}^{K} W(C_k) \tag{1}$$

where $W(C_k)$ measures the dissimilarity between observations in cluster k, e.g. *squared Euclidean distance*

$$W(C_k) = \frac{1}{|C_k|} \sum_{i,i' \in C_k} \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2 = 2 \sum_{i \in C_k} \sum_{j=1}^{p} (x_{ij} - \bar{x}_{kj})^2$$

with $|C_k|$ the number of observations in cluster k and cluster mean $\bar{x}_{kj} = \frac{1}{|C_k|} \sum_{i \in C_k} x_{ij}$.

K-Means Clustering

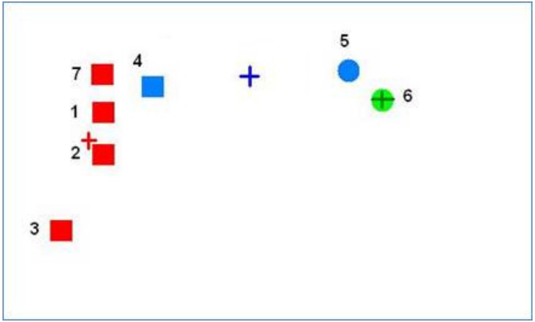


K-Means Clustering Algorithm

1. Randomly assign a number, from 1 to K, to each to the observations.
2. Iterate until the cluster assignments stop changing.
   (a) For each of the K clusters, compute the cluster *centroid* $\bar{x}_{kj} = \frac{1}{|C_k|} \sum_{i \in C_k} x_{ij}$ for $j = 1, \ldots, p$.
   (b) Assign each observation to the cluster whose centroid is closest.

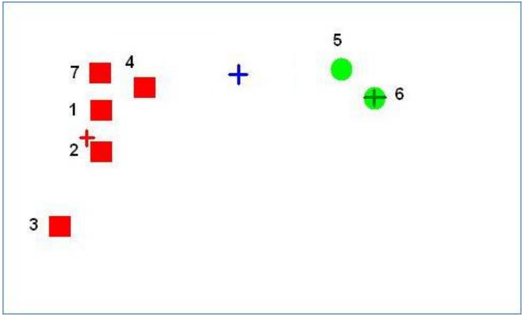
K-Means Empty Cluster Example

Start



Clusters are indicated with colors, centroids with crosses

Next Iteration



Clusters can become empty

Adapted from `http://user.ceng.metu.edu.tr/~tcan/ceng465_f1314/Schedule/KMeansEmpty.html`

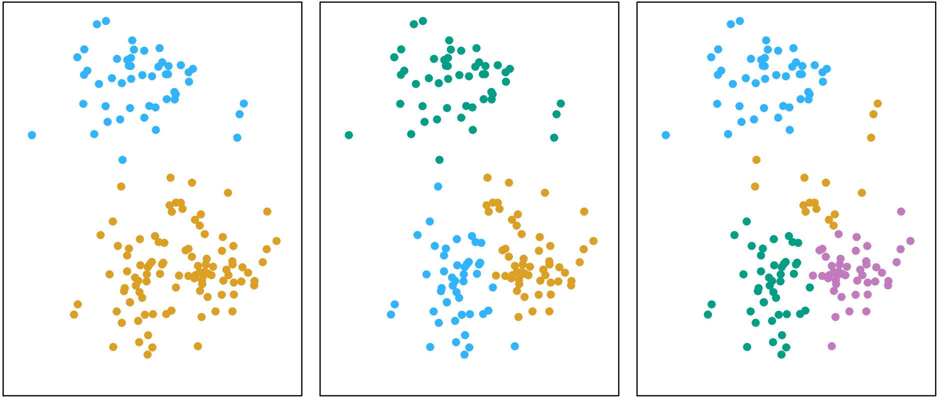
Dependence on the Initial Condition

K-Means Clustering performed six times on the same data set with different random assignments. Above the plot is the value of the loss function (in Equation 1 on slide 8) at convergence.

Three different local optima were obtained. Those labelled in red all achieve the same solution.

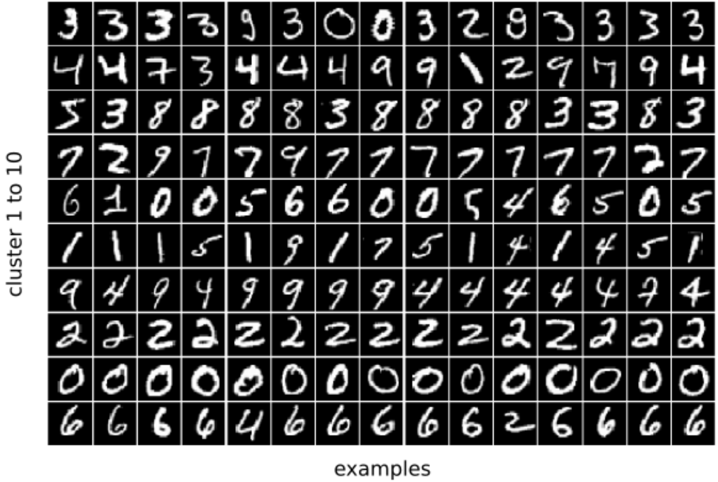
Choosing *k* in K-Means Clustering

K=2 K=3 K=4



Some of the figures in this presentation are taken from "An Introduction to Statistical Learning, with applications in R" (Springer, 2013) with permission from the authors: G. James, D. Witten, T. Hastie and R. Tibshirani

K-Means Clustering of MNIST Images



- All images in the same row are in the same cluster according to one run of K-Means clustering with 10 clusters.
- Some clusters contain images alsmost exclusively from one class; other clusters contain images from a few different classes.

Quiz

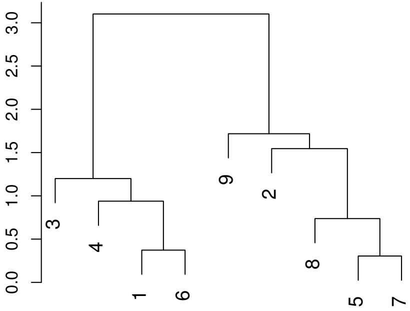
Richtig oder falsch?

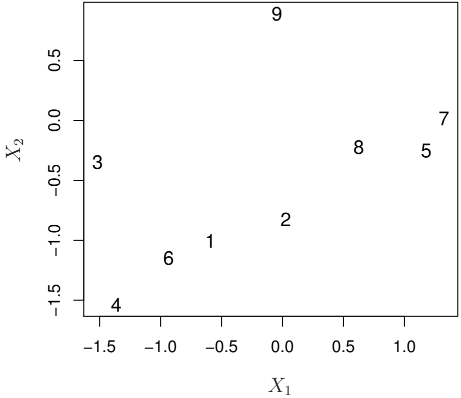
- Am Ende von K-Means Clustering ist jeder Datenpunkt in genau einem Cluster.
- Das Resultat von K-Means Clustering hängt nur von k und dem Verschiedenheitsmass $W(C_k)$ ab.

Table of Contents

Hierarchical Clustering

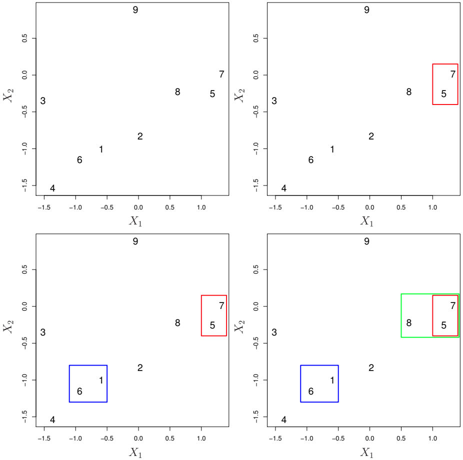
Organize data in a tree called **dendrogram**





The height of the fusion of two branches indicates how different the observations in the two branches are.

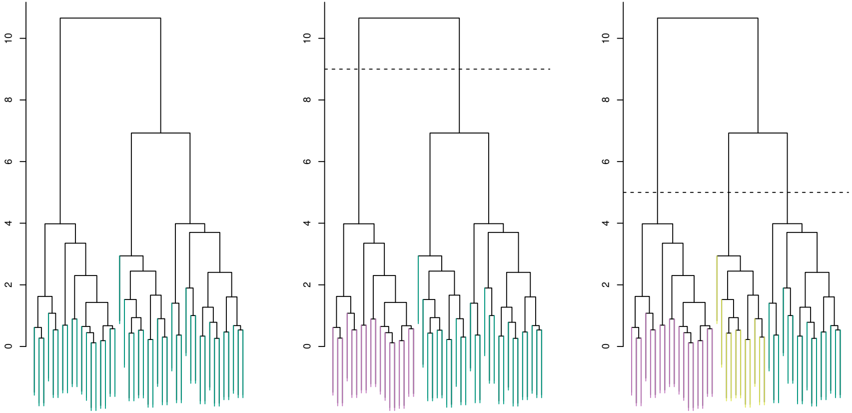
Hierarchical Clustering Algorithm



Euclidean distance, complete linkage

1. Begin with n observations and a measure of all the $\binom{n}{2} = n(n-1)/2$ pairwise dissimilarities. Treat each observation as its own cluster.
2. For $i = n, n-1, \ldots, 2$:
   (a) Examine all pairwise dissimilarities among the i clusters and fuse the most similar pair. The dissimilarity of this pair indicates the height in the dendrogram at which the fusion is placed.
   (b) Compute the new pairwise inter-cluster dissimilarities among the $i-1$ remaining clusters.

Clustering with a Dendrogram



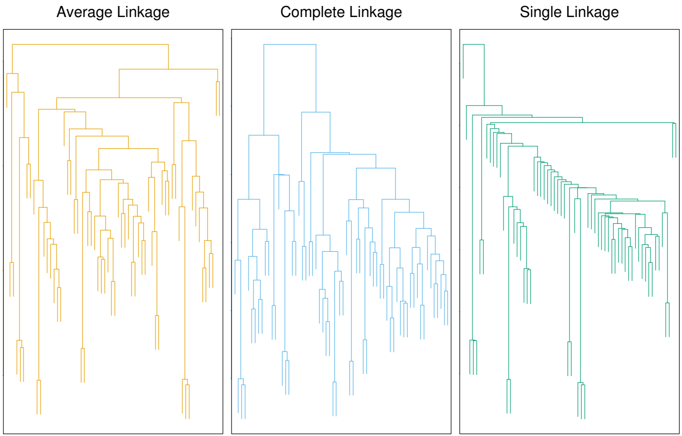
The coloured leaves indicate the class identity. The length of the leaves has no meaning.

Cut the dendrogram at different heights to get different clusterings.

Linkage: Measuring Distances Between Sets

Linkage	Description
Complete	**Maximal intercluster dissimilarity**. Compute all pairwise dissimilarities between the observations in cluster A and the observations in cluster B, and record the largest of these dissimilarities.
Single	**Minimal intercluster dissimilarity**. Compute all pairwise dissimilarities between the observations in cluster A and the observations in cluster B, and record the smallest of these dissimilarities. Single linkage can result in extended, trailing clusters in which single observations are fused one-at-a-time.
Average	**Mean intercluster dissimilarity**. Compute all pairwise dissimilarities between the observations in cluster A and the observations in cluster B, and record the average of these dissimilarities.
Centroid	Dissimilarity between the centroid for cluster A (a mean vector of length p) and the centroid for cluster B. Centroid linkage can result in **undesirable inversions** (i.e. clusters are fused at a height below either of the individual clusters).

The Effect of the Linkage



Average and complete linkage tend to yield more balanced clusters.

Small Decisions with Big Consequences

- What type of dissimilarity measure should be used?
  Euclidean distance is not the most natural for many types of data.
- Should the observations or features be standardized (e.g. variance 1)?
  Scaling can be seen as changing the dissimilarity measure.
- In the case of hierarchical clustering:
  - What type of linkage should be used?
  - Where should we cut the dendrogram?
- In the case of K-means clustering: how should be choose k?

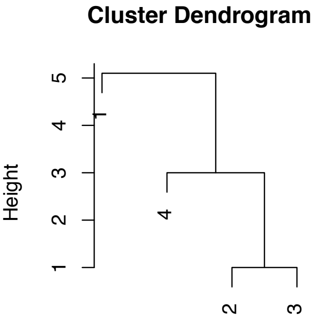
[...] we must be careful about how the results of a clustering analysis are reported. These results should not be taken as the absolute truth about a data set. Rather, they should constitute a starting point for the development of a scientific hypothesis and further study, preferably on an independent data set.

Quiz

Richtig oder falsch?

Stellen Sie sich ein Problem mit 4 Datenpunkten vor:
$x_1 = 1, x_2 = 4, x_3 = 5, x_4 = 7$.

- Nach dem ersten Schritt von hierarchischem Clustering mit euklidschem Verschiedenheitsmass finden wir die 3 Cluster $\{1\}, \{2, 3\}, \{4\}$.
- Mit complete linkage ist der euklidsche Abstand zwischen Cluster $\{1\}$ und $\{2, 3\}$ gleich $\sqrt{(1-5)^2} = 4$.
- Das Dendrogram rechts ist das Resultat von hierarchischem Clustering der 4 Datenpunkte..
- Nachbarn im Dendrogram (beispielsweise 1 und 4) sind Datenpunkte, welche nahe beieinander liegen.



Terminology

- **Supervised Learning**: learn $p(Y|X)$
- **Semi-Supervised Learning**: learn $p(Y|X)$ with typically a small fraction of the data having labels given explicitly by humans and the rest unlabeled, e.g. many images, but only some with labels.
- **Self-Supervised Learning**: learn $p(Y|X)$ where Y is not a label given explicitly by humans (or other supervisors). *Example: auto-regressive models like weather prediction.*
- **Unsupervised Learning**: learn $p(X)$.
  In unsupervised learning one is often more interested in a hidden representation of the data than in plain fitting of $p(X)$, e.g. if the data seems to be clustered, what is the cluster identity of a given point.
  If X is multidimensional one learns sometimes parts of $p(X)$ in a self-supervised manner, e.g. $p(X) = p(X_1)p(X_2|X_1)$.