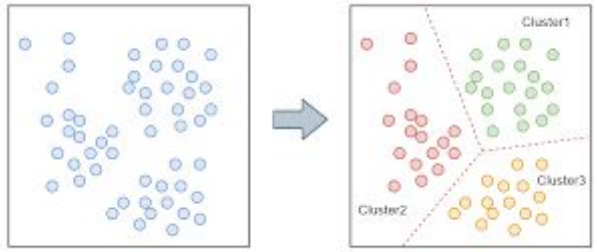
Unsupervised Learning



Data Science & Machine Learning
ITS4 & ISE2

Mously DIAW
ENSAE - 2021

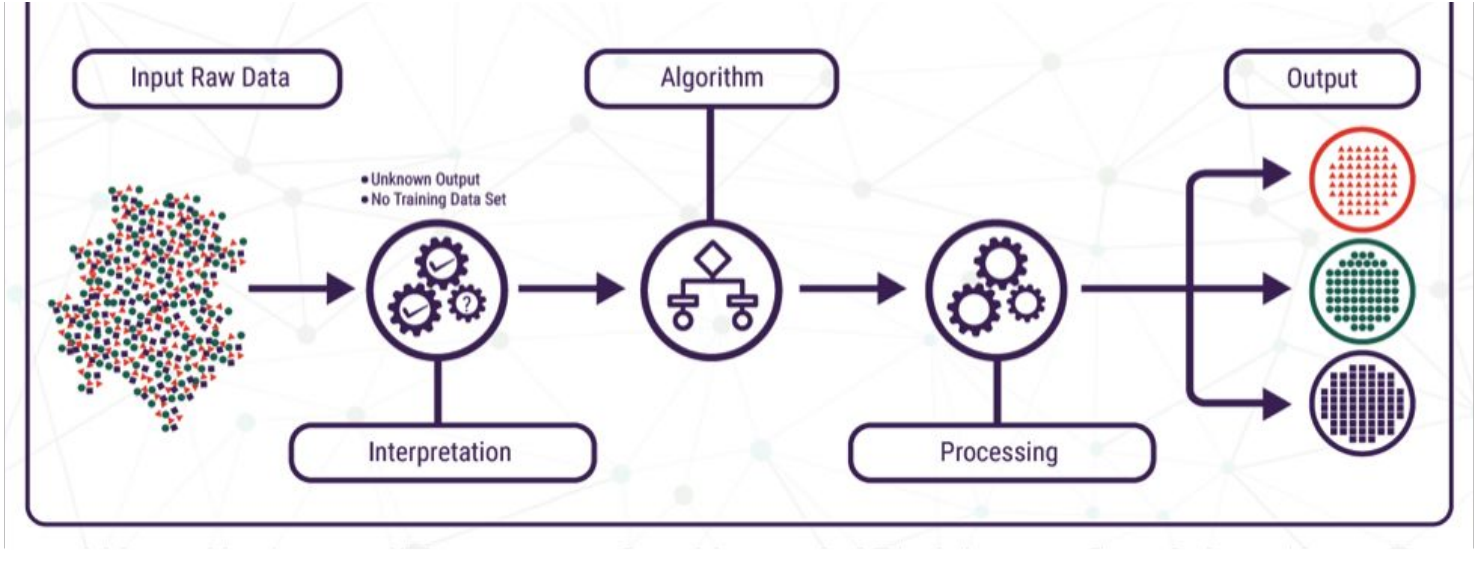
Summary

- What is Artificial Intelligence (AI) ?
- What is Machine Learning (ML) ?
- Deep Learning (DL) overview
- How to learn Data Science ?
- Project description
- Questions

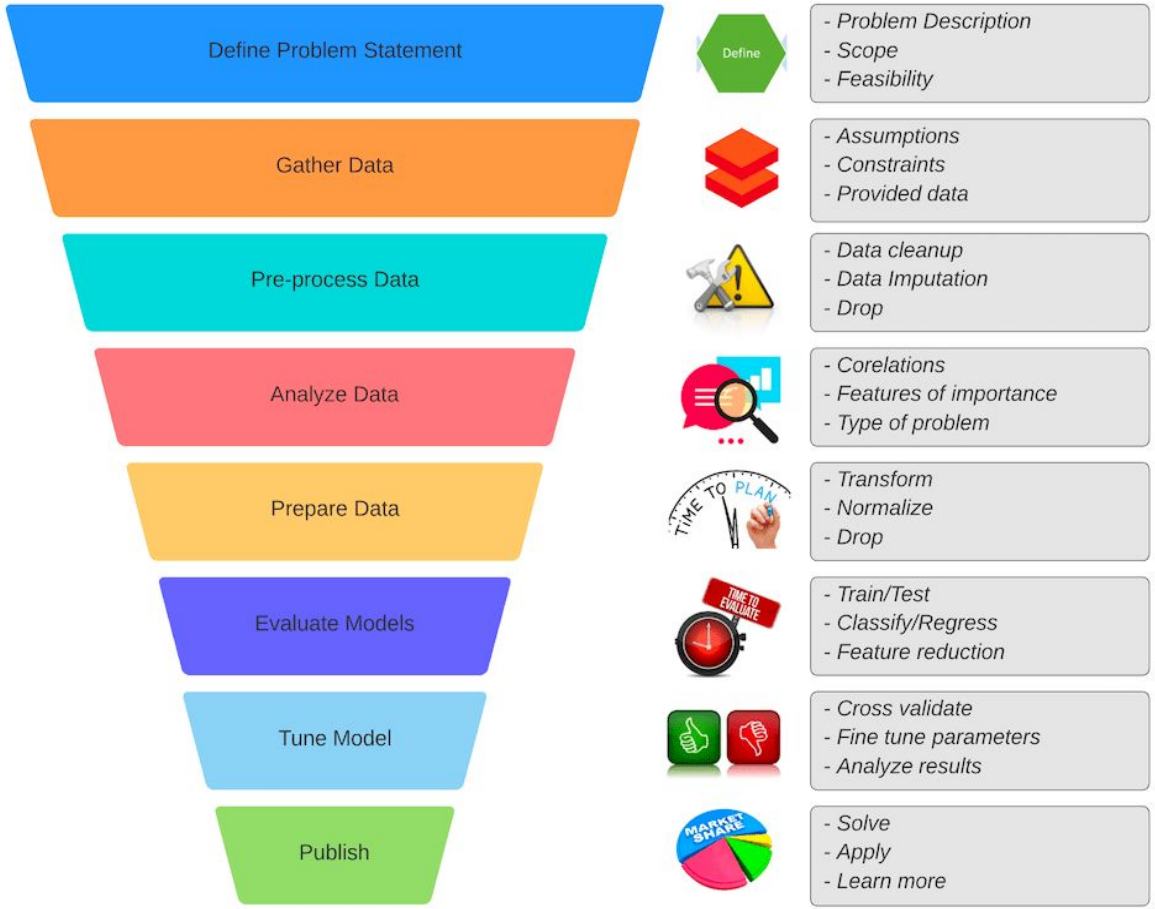
Unsupervised Learning

Unsupervised learning allows the system to identify patterns within data sets.

It is a type of algorithm that learns patterns from untagged data.



Machine learning workflow



Clustering

Overview of clustering methods

Most popular algorithms

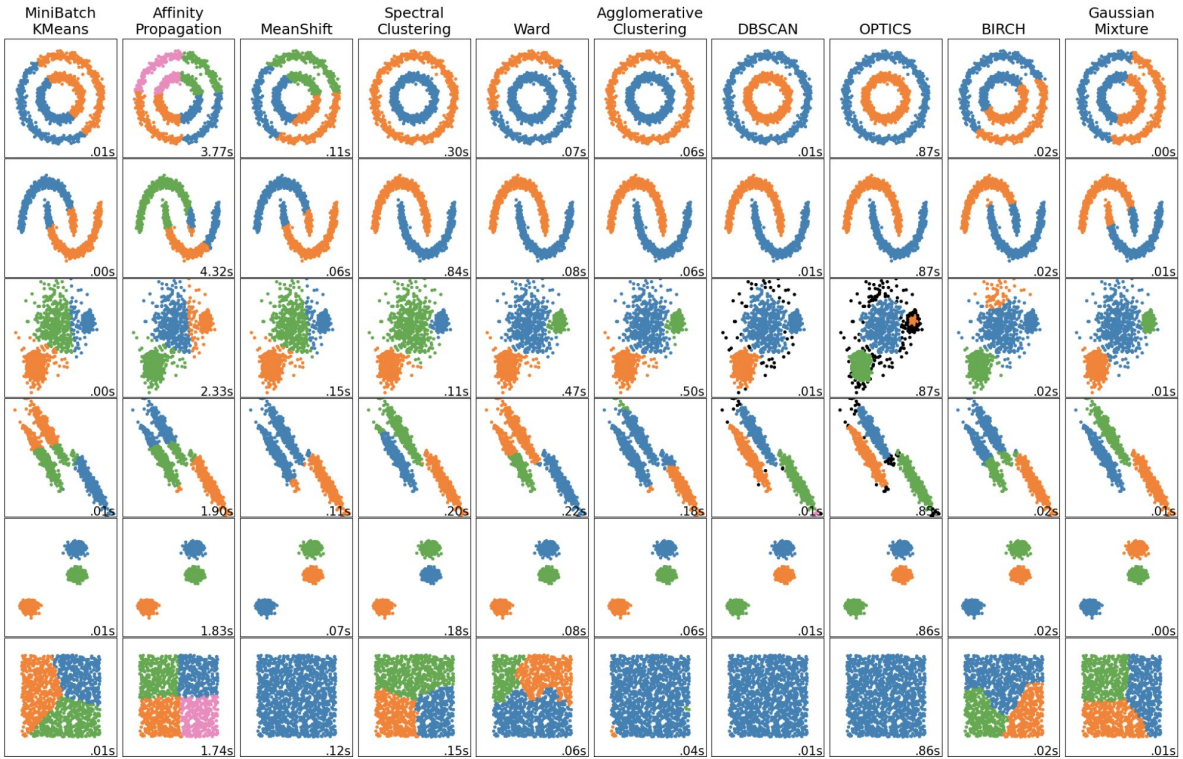
Cluster analysis, or clustering, is an unsupervised machine learning task.

It involves automatically discovering natural grouping in *data. Clustering algorithms only interpret the input data and find natural groups or clusters in feature space.*

Clustering techniques apply when there is no class to be predicted but rather when the instances are to be divided into natural groups.

For example, this involves identifying :

- *customers who have similar behaviors (market segmentation);*
- *users who have similar uses of a tool;*
- *communities in social networks;*
- *recurring patterns in financial transactions.*



K-means

K-Means Clustering may be the **most widely known clustering algorithm** and involves assigning examples to clusters in an effort to **minimize the variance within each cluster**.

K-means aims to **partition *n* observations into *k* clusters** in which each observation belongs to the **cluster with the nearest mean** (cluster centers or cluster centroid), serving as a prototype of the cluster.

The K-means algorithm aims to choose centroids that minimise the **inertia**, or **within-cluster sum-of-squares criterion**:

$$\sum_{i=0}^{n} \min_{\mu_j \in C} (||x_i - \mu_j||^2)$$

Inertia can be recognized as a **measure of how internally coherent clusters are**. It suffers from various drawbacks:

- Inertia makes the assumption that **clusters are convex and isotropic**, which is not always the case. It responds poorly to elongated clusters, or manifolds with **irregular shapes**.
- Inertia is not a **normalized metric**: we just know that **lower values are better and zero is optimal.**

K-means

K-means is often referred to as **Lloyd's algorithm**. In basic terms, the algorithm has **three steps.**

1. The first step chooses the initial centroids, with the most basic method being to choose k samples from the dataset
2. After initialization, K-means consists of looping between the two other steps
    2.1. The first step assigns each sample to its nearest centroid (in the group whose center is closest to it)
    2.2. The second step creates new centroids by taking the mean value of all of the samples assigned to each previous centroid. The difference between the old and the new centroids are computed and the algorithm repeats these last two steps until this value is less than a threshold. In other words, it repeats until the centroids do not move significantly.

Given enough time, **K-means will always converge,** however this may be to a local minimum. This is highly dependent on the initialization of the centroids.

The algorithm can also be understood through the concept of Voronoi diagrams.

The **MiniBatchKMeans** is a variant of the **K-Means** algorithm which uses mini-batches to reduce the computation time

You have to select **how many groups/classes there are**

Hierarchical clustering

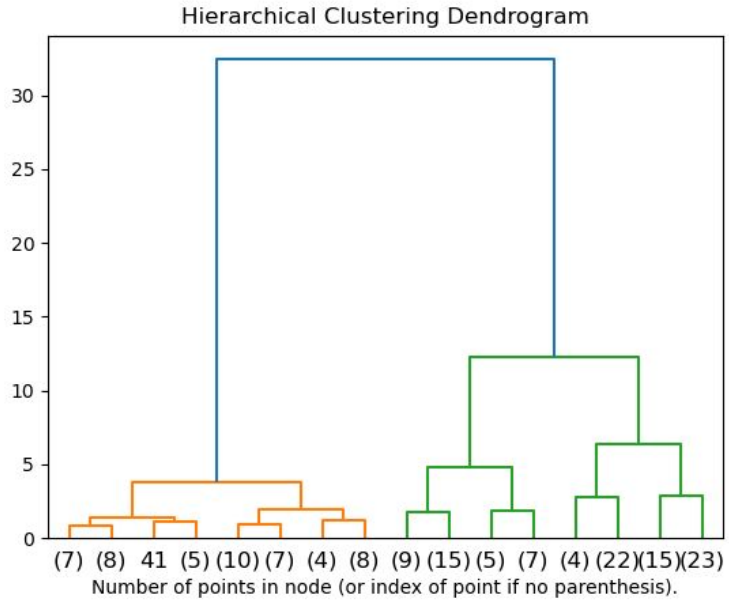
Hierarchical clustering is a general family of clustering algorithms that build nested clusters by merging or splitting them successively.

This hierarchy of clusters is represented as a tree (or dendrogram)

The root of the tree is the unique cluster that gathers all the samples, the leaves being the clusters with only one sample.

Strategies for hierarchical clustering generally fall into two types

- **Agglomerative**: This is a "bottom-up" approach: each observation starts in its own cluster, and pairs of clusters are merged as one moves up the hierarchy.
- **Divisive**: This is a "top-down" approach: all observations start in one cluster, and splits are performed recursively until every object is separate.

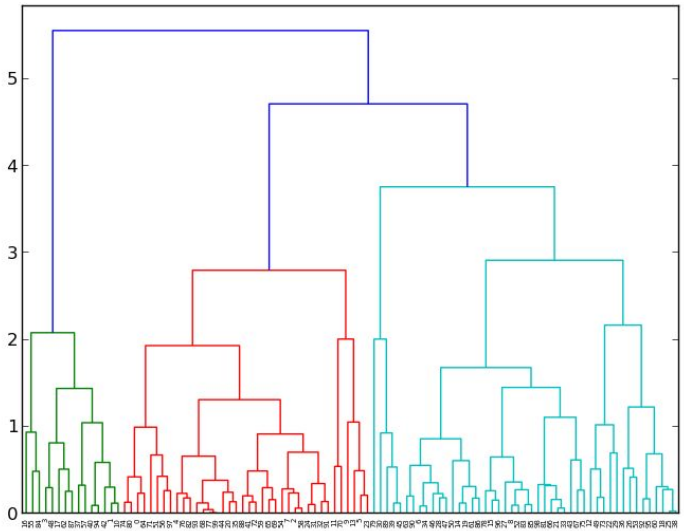


Hierarchical clustering

The linkage criteria determines the metric used for the merge strategy:

- **Ward** minimizes the sum of squared differences within all clusters. It is a variance-minimizing approach and in this sense is similar to the k-means objective function but tackled with an agglomerative hierarchical approach.
- **Maximum** or **complete linkage** minimizes the maximum distance between observations of pairs of clusters.
- **Average linkage** minimizes the average of the distances between all observations of pairs of clusters.
- **Single linkage** minimizes the distance between the closest observations of pairs of clusters.

Hierarchical clustering does not require us to specify the number of clusters



DBSCAN

DBSCAN is short for **Density-Based Spatial Clustering of Applications with Noise**

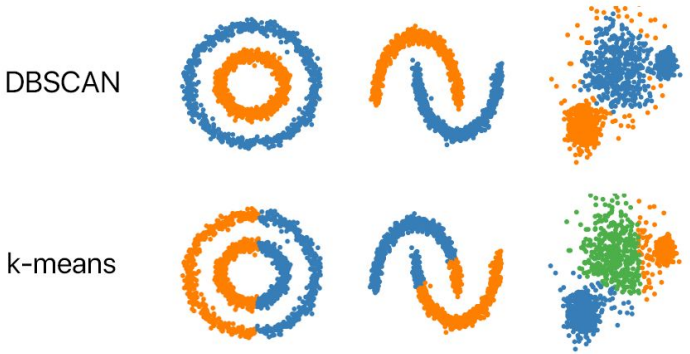
It groups together points that are closely packed together (points with many nearby neighbors), marking as outliers points that lie alone in low-density

DBSCAN can find **non-linearly separable clusters**. It does **not require one to specify the number of clusters** in the data a priori

DBSCAN has a **notion of noise, and is robust to outliers**.

DBSCAN **requires just two parameters (minPoints, ε)** and is mostly insensitive to the ordering of the points in the database. **Higher minPoints or lower ε indicate higher density** necessary to form a cluster.

More formally, we define a core sample as being a sample in the dataset such that there **exist minPoints other samples within a distance of ε**, which are **defined as *neighbors* of the core sample**

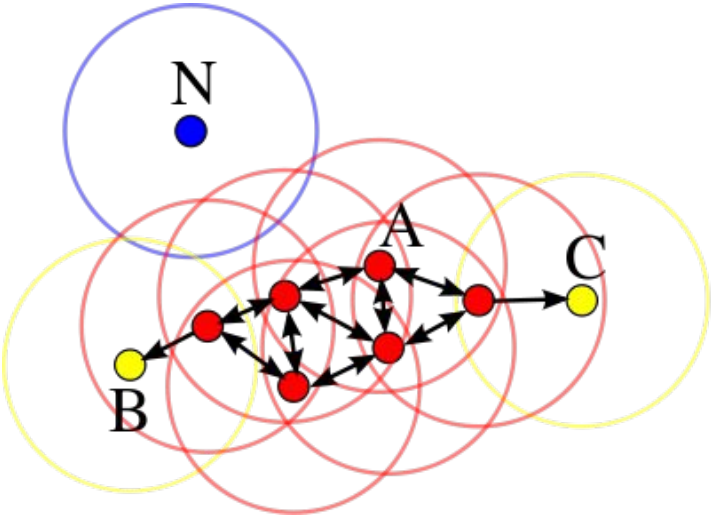


DBSCAN

For the purpose of DBSCAN clustering, the points are classified as *core points*, (*density-*) *reachable points* and *outliers*, as follows:

- A point *p* is a *core point* if at least minPoints points are within distance ε of it (including *p*).
- A point *q* is *directly reachable* from *p* if point *q* is within distance ε from core point *p*. Points are only said to be directly reachable from core points.
- A point *q* is *reachable* from *p* if there is a path $p_1, ..., p_n$ with $p_1 = p$ and $p_n = q$, where each p_{i+1} is directly reachable from p_i. Note that this implies that the initial point and all points on the path must be core points, with the possible exception of *q*.
- All points not reachable from any other point are *outliers* or *noise points*.

➔ Now if p is a core point, then it forms a cluster together with all points (core or non-core) that are reachable from it.



In this diagram, minPoints = 4. Point A and the other red points are core points, because the area surrounding these points in an ε radius contain at least 4 points (including the point itself). Because they are all reachable from one another, they form a single cluster. Points B and C are not core points, but are reachable from A (via other core points) and thus belong to the cluster as well. Point N is a noise point that is neither a core point nor directly-reachable.

Comparison of the clustering algorithms

Method name	Parameters	Usecase	Geometry (metric used)
K-means	number of clusters	General-purpose, even cluster size, flat geometry, not too many clusters, inductive	Distances between points
Agglomerative Clustering	number of clusters or distance threshold, linkage type, distance	Many clusters, possibly connectivity constraints, non Euclidean distances, transductive	Any pairwise distance
DBSCAN	neighborhood size	Non-flat geometry, uneven cluster sizes, outlier removal, transductive	Distances between nearest points

Other methods

Affinity propagation

BIRCH (*balanced iterative reducing and clustering using hierarchies*)

Mini-Batch K-Means

Mean Shift

OPTICS (*ordering points to identify the clustering structure*)

Spectral Clustering

Gaussian Mixture Model

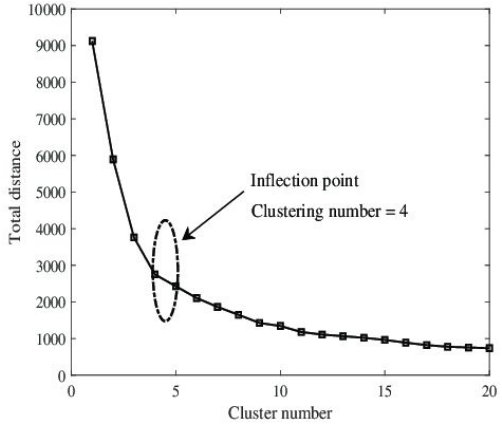
Number of Optimal Clusters, k

There are two major approaches to find optimal number of clusters:

- Domain knowledge
- Data driven approach

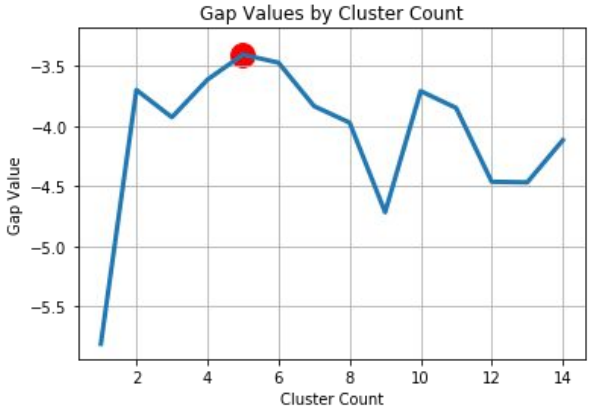
Elbow method

Within-cluster variance is a measure of compactness of the cluster. Lower the value of within cluster variance, higher the compactness of cluster formed.
Sum of within-cluster variance, *W*, is calculated for clustering analyses done with different values of k.



Gap statistic

Similar to Elbow method. It find the ideal k value where ‘deviation’ or ‘Gap’ between two intra-cluster variance is highest



Clustering quality

Before evaluating the clustering performance, making sure that data set we are working has clustering tendency and does not contain uniformly distributed points is very important

To solve this, Hopkins test, a statistical test for spatial randomness of a variable, can be used to measure the probability of data points generated by uniform data distribution.

Evaluating the performance of a clustering algorithm **is not as trivial as counting the number of errors** or the precision and recall of a supervised classification algorithm.

There are majorly two types of measures to assess the clustering performance.

- ***Extrinsic Measures*** which **require ground truth labels**. Examples are Adjusted Rand index (ARI), Fowlkes-Mallows scores, Mutual information based scores, Homogeneity, Completeness and V-measure etc.
- ***Intrinsic Measures*** that **does not require ground truth labels.** Some of the clustering performance measures are Silhouette Coefficient, Calinski-Harabasz Index, Davies-Bouldin Index etc.

Clustering quality - Silhouette Coefficient

If the ground truth labels are not known, evaluation must be performed using the model itself.

The Silhouette Coefficient is defined for each sample and is composed of two scores:

- **a**: The mean distance between a sample and all other points in the same class (mean intra-cluster distance).
- **b**: The mean distance between a sample and all other points in the *next nearest cluster* (mean nearest-cluster distance).

The Silhouette Coefficient *s* for a single sample is then given as:

$$s = \frac{b - a}{max(a, b)}$$

The Silhouette Coefficient for a set of samples is given as the mean of the Silhouette Coefficient for each sample.

The best value is 1 and the **worst value is -1**. Values near **0 indicate overlapping clusters**.

In normal usage, the Silhouette Coefficient is applied to the results of a cluster analysis.

Clustering quality - Calinski-Harabasz Index

Also known as the Variance Ratio Criterion

The index is the ratio of the sum of between-clusters dispersion and of within-cluster dispersion for all clusters (where dispersion is defined as the sum of distances squared):

$$s = \frac{\mathrm{tr}(B_k)}{\mathrm{tr}(W_k)} \times \frac{n_E - k}{k - 1}$$

where

$$W_k = \sum_{q=1}^{k} \sum_{x \in C_q} (x - c_q)(x - c_q)^T$$

$$B_k = \sum_{q=1}^{k} n_q (c_q - c_E)(c_q - c_E)^T$$

A higher Calinski-Harabasz score relates to a model with better defined clusters

The score is higher when clusters are dense and well separated. The score is fast to compute.

Clustering quality - Davies-Bouldin Index

This score signifies the average ‘similarity’ between clusters, where the similarity is a measure that compares the distance between clusters with the size of the clusters themselves.

- s_i: the average distance between each point of cluster *i* and the centroid of that cluster – also know as cluster diameter.
- d_{ij}: the distance between cluster centroids *i* and *j*

$$DB = \frac{1}{k}\sum_{i=1}^{k}\max_{i\neq j} R_{ij} \qquad \text{where} \qquad R_{ij} = \frac{s_i + s_j}{d_{ij}}$$

Zero is the lowest possible score. Values closer to zero indicate a better partition.

Clustering quality - Adjusted Rand index (ARI)

Given the knowledge of the ground truth class assignments, **ARI** is a function that measures **the similarity of the two assignments**, ignoring permutations.

If C is a ground truth class assignment and K the clustering, let us define a and b as:

- a: the number of pairs of elements that are in the same set in C and in the same set in K
- b: the number of pairs of elements that are in different sets in C and in different sets in K

$$\mathrm{ARI} = \frac{\mathrm{RI} - E[\mathrm{RI}]}{\max(\mathrm{RI}) - E[\mathrm{RI}]} \qquad \text{where} \qquad \mathrm{RI} = \frac{a+b}{C_2^{n_{samples}}}$$

$C_2^{n_{samples}}$ is the total number of possible pairs in the dataset

The range is [-1, 1] for the ARI score. **Lower values indicate different labelings**, similar clusterings have a high ARI. **1.0 is the perfect match score** (clusterings are identical)

Homogeneity, completeness and V-measure

The ground truth class assignments of the samples is known.

- **homogeneity**: each cluster contains only members of a single class.
- **completeness**: all members of a given class are assigned to the same cluster.
- **V-measure**: is the **harmonic mean of homogeneity and completeness**

All are bounded below by 0.0 (bad clustering) and above by 1.0 (perfect score, higher is better).

These metrics **require the knowledge of the ground truth classes** while almost never available in practice or requires manual assignment by human annotators (as in the supervised learning setting)

Clustering quality - Fowlkes-Mallows scores

The ground truth class assignments of the samples is known.

The Fowlkes-Mallows score FMI is defined as the geometric mean of the pairwise precision and recall:

$$\mathrm{FMI} = \frac{\mathrm{TP}}{\sqrt{(\mathrm{TP} + \mathrm{FP})(\mathrm{TP} + \mathrm{FN})}}$$

Where:

- **TP** (number of **True Positive**): the number of pair of points that belong to the same clusters in both the true labels and the predicted labels
- **FP** (number of **False Positive):** the number of pair of points that belong to the same clusters in the true labels and not in the predicted labels
- **FN** (number of **False Negative**): the number of pair of points that belongs in the same clusters in the predicted labels and not in the true labels

The score **ranges from 0 to 1**. A high value indicates a good similarity between two clusters.

Dimensionality reduction

Dimensionality reduction - Overview

Dimensionality reduction, or **dimension reduction**, is the transformation of data from a **high-dimensional space into a low-dimensional** space so that the low-dimensional representation **retains some meaningful properties of the original data.**

Methods are commonly divided into **linear** and **nonlinear** approaches.

Dimensionality reduction can be used for **noise reduction, data visualization, cluster analysis**, or as an **intermediate step** to facilitate other analyses.

Principal Component Analysis (PCA)

The main linear technique for dimensionality reduction

PCA performs a linear mapping of the data to a lower-dimensional space in such a way that the variance of the data in the low-dimensional representation is maximized.

PCA is used to **decompose a multivariate dataset** in a set of successive **orthogonal components that explain a maximum amount of the variance**.

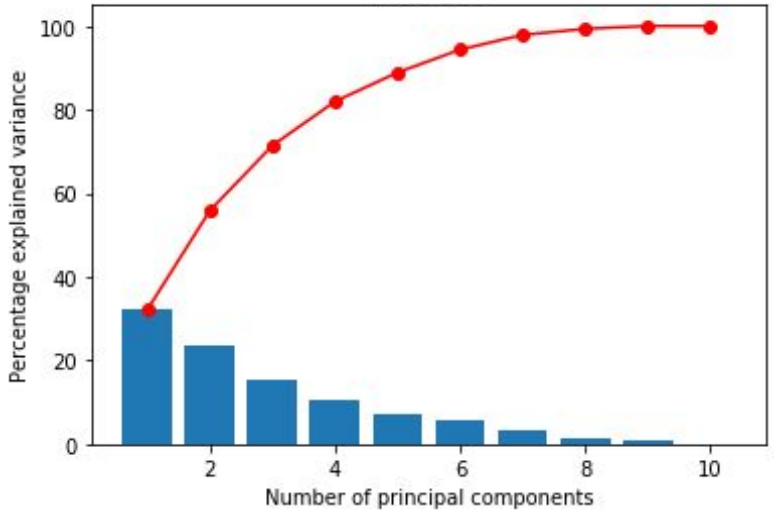
In practice, the covariance (and sometimes the correlation) matrix of the data is constructed and the eigenvectors on this matrix are computed

The eigenvectors that correspond to the largest eigenvalues (the principal components) can now be used to reconstruct a large fraction of the variance of the original data.

PCA - optimal choice of the number of components

Elbow method - explained variance

Percentage of variance explained by each of the selected components.



Kernel PCA (KPCA)

Kernel PCA is an extension of PCA which achieves non-linear dimensionality reduction

Principal component analysis can be employed in a **nonlinear** way by means of the kernel trick.

- Linear kernel: $K(\mathbf{x}, \mathbf{y}) = \mathbf{x}^T \mathbf{y}, \quad \mathbf{x}, \mathbf{y} \in \mathbb{R}^d.$
- Polynomial kernel: $K(\mathbf{x}, \mathbf{y}) = (\mathbf{x}^T \mathbf{y} + r)^n, \quad \mathbf{x}, \mathbf{y} \in \mathbb{R}^d, r \geq 0, n \geq 1.$
- Gaussian kernel (RBF kernel): $K(\mathbf{x}, \mathbf{y}) = e^{-\frac{\|\mathbf{x}-\mathbf{y}\|^2}{2\sigma^2}}, \quad \mathbf{x}, \mathbf{y} \in \mathbb{R}^d, \sigma > 0.$
- Laplacian kernel: $K(\mathbf{x}, \mathbf{y}) = e^{-\alpha\|\mathbf{x}-\mathbf{y}\|}, \quad \mathbf{x}, \mathbf{y} \in \mathbb{R}^d, \alpha > 0.$
- Abel kernel: $K(x, y) = e^{-\alpha|x-y|}, x, y \quad \in \mathbb{R}, \alpha > 0.$

KPCA is capable of constructing nonlinear mappings that maximize the variance in the data.

Non-negative matrix factorization (NMF)

NMF is an alternative approach to decomposition that assumes that the data and the components are non-negative.

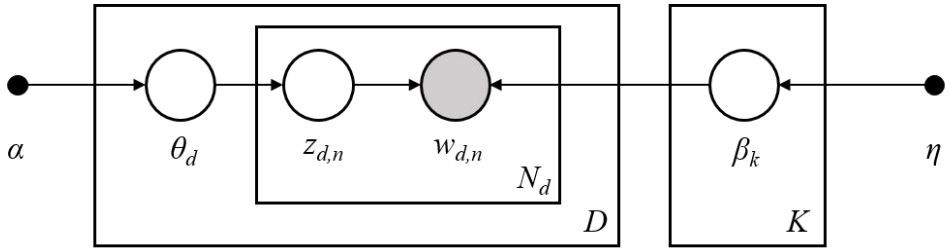
NMF can be plugged in instead of **PCA** or its variants, in the cases where the data matrix does not contain negative values.

Latent Dirichlet Allocation (LDA)

Latent Dirichlet Allocation is a generative probabilistic model for collections of discrete dataset such as text corpora.
The basic idea is that documents are represented as random mixtures over latent topics, where each topic is characterized by a distribution over words.

The goal of LDA is to use the observed words to infer the hidden topic structure

The graphical model of LDA is a three-level generative model:



- The corpus is a collection of D documents.
- A document is a sequence of N words (w).
- There are K topics in the corpus.
- The boxes represent repeated sampling
- Choose a topic zn

Other algorithms

Truncated singular value decomposition and latent semantic analysis
Dictionary Learning
Factor Analysis
Independent component analysis (ICA)
Non-negative matrix factorization (NMF or NNMF)
Latent Dirichlet Allocation (LDA)
T-distributed Stochastic Neighbor Embedding (t-SNE)
Isomap

Canonical Correlation Analysis (CCA)

Questions ???

References

https://towardsdatascience.com/the-5-clustering-algorithms-data-scientists-need-to-know-a36d136ef68
https://scikit-learn.org/stable/modules/clustering.html
https://machinelearningmastery.com/clustering-algorithms-with-python/
https://en.wikipedia.org/wiki/K-means_clustering
https://en.wikipedia.org/wiki/Hierarchical_clustering
https://en.wikipedia.org/wiki/DBSCAN
https://towardsdatascience.com/clustering-evaluation-strategies-98a4006fcfc
https://scikit-learn.org/stable/modules/clustering.html#clustering-evaluation
https://scikit-learn.org/stable/modules/unsupervised_reduction.html
http://pca.narod.ru/scholkopf_kernel.pdf
https://www.jmlr.org/papers/volume3/blei03a/blei03a.pdf