# Clustering

## **Clustering task overview**

A small introduction video for clustering task:

* [Clustering: K-means and Hierarchical](https://www.youtube.com/watch?v=QXOkPvFM6NU)
* List of clustering algorithms in scikit-learn:  
  <https://scikit-learn.org/stable/modules/classes.html#module-sklearn.cluster>

## **KMeans**

* A description of the algorithm in scikit-learn:  
  <https://scikit-learn.org/stable/modules/clustering.html#k-means>
* An explanatory video for the algorithm:  
  [StatQuest: K-means clustering](https://www.youtube.com/watch?v=4b5d3muPQmA)
* Examples of code usage:  
  <https://towardsdatascience.com/understanding-k-means-clustering-in-machine-learning-6a6e67336aa1>

## **DBSCAN**

* A description of the algorithm in scikit-learn:  
  <https://scikit-learn.org/stable/modules/clustering.html#dbscan>  
  [DBSCAN: Part 1](https://www.youtube.com/watch?v=sKRUfsc8zp4)  
  [DBSCAN: Part 2](https://www.youtube.com/watch?v=6jl9KkmgDIw)
* Examples of code usage:  
  <https://towardsdatascience.com/dbscan-clustering-explained-97556a2ad556#:~:text=DBSCAN%20stands%20for%20density%2Dbased,many%20points%20from%20that%20cluster.>
* (Additionally) HDBSCAN library:  
  <https://hdbscan.readthedocs.io/en/latest/how_hdbscan_works.html>

## **Agglomerative (hierarchical) clustering**

* A description of the algorithm in Scikit-learn:  
  <https://scikit-learn.org/stable/modules/clustering.html#hierarchical-clustering>
* [StatQuest: Hierarchical Clustering](https://www.youtube.com/watch?v=7xHsRkOdVwo&t=94s)
* Examples of code usage:  
  <https://towardsdatascience.com/hierarchical-clustering-explained-e58d2f936323>

## **Clustering performance metrics**

* A description of metrics in scikit-learn:  
  <https://scikit-learn.org/stable/modules/clustering.html#clustering-performance-evaluation>
* How to select the amount of clusters with the silhouette coefficient:  
  <https://scikit-learn.org/stable/auto_examples/cluster/plot_kmeans_silhouette_analysis.html#sphx-glr-auto-examples-cluster-plot-kmeans-silhouette-analysis-py>
* How to select the amount of clusters with inertia (KMeans)  
  <https://blog.cambridgespark.com/how-to-determine-the-optimal-number-of-clusters-for-k-means-clustering-14f27070048f>

# Outlier detection

## **Outlier detection (OD) task overview**

* An overview from Scikit-learn:  
  <https://scikit-learn.org/stable/modules/outlier_detection.html>
* A set of small lectures with overview of the OD task and some methods:  
  [Outlier Analysis/Detection with Univariate Methods Using Tukey boxplots in Python - Tutorial 20](https://www.youtube.com/watch?v=nbNiD76yE8o&list=PL8Bgdi87Y1lWJtBDuStNuEGoXKVFmMrF3&index=2&t=0s)
* Andrew NG's lectures about outliers:  
  [Lecture 15.1 — Anomaly Detection Problem | Motivation — [ Machine Learning | Andrew Ng ]](https://www.youtube.com/watch?v=086OcT-5DYI&list=PLwgXNx7TiGV6UH3aEzmdZwzFRwvEnRb0N&index=2&t=0s)

## **Mahalanobis rule**

* An explanation of intuition behind the rule:  
  [Mahalanobis Distance - intuitive understanding through graphs and tables](https://www.youtube.com/watch?v=3IdvoI8O9hU)
* An implementation of the distance from SciPy:  
  <https://docs.scipy.org/doc/scipy/reference/generated/scipy.spatial.distance.mahalanobis.html>
* Pairwise distances between observations in n-dimensional space (With Mahalanobis distance option):
* https://docs.scipy.org/doc/scipy/reference/generated/scipy.spatial.distance.pdist.html
* An example of Mahalanobis distance + PCA:  
  <https://nirpyresearch.com/detecting-outliers-using-mahalanobis-distance-pca-python/>

## **Local Outlier Factor (LOF)**

* An explanatory video:  
  [Tutorial | Anomaly Detection | Local Outlier Factor | LOF Algorithm](https://www.youtube.com/watch?v=CePgbdVdLvg)
* Scikit-learn docs:
  + <https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.LocalOutlierFactor.html>
  + <https://scikit-learn.org/stable/auto_examples/neighbors/plot_lof_outlier_detection.html#:~:text=The%20Local%20Outlier%20Factor%20(LOF,lower%20density%20than%20their%20neighbors.>
* An explanation of LOF:  
  <https://towardsdatascience.com/local-outlier-factor-for-anomaly-detection-cc0c770d2ebe>
* Examples of the code with Scikit-learn:  
  <https://www.datatechnotes.com/2020/04/anomaly-detection-with-local-outlier-factor-in-python.html>

## **Isolation Forest**

* Scikit-learn docs:
  + <https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.IsolationForest.html?highlight=isolation%20forest>
  + <https://scikit-learn.org/stable/modules/outlier_detection.html#isolation-forest>
* An explanation of Isolation Forest from PyData:  
  [Unsupervised Anomaly Detection with Isolation Forest - Elena Sharova](https://www.youtube.com/watch?v=5p8B2Ikcw-k)
* Examples of the code with Scikit-learn:  
  <https://blog.paperspace.com/anomaly-detection-isolation-forest/>
* An example of Isolation Forest usage on time series data:  
  <https://towardsdatascience.com/anomaly-detection-with-isolation-forest-visualization-23cd75c281e2>

## **One-Class SVM**

* Scikit-learn docs:
  + <https://scikit-learn.org/stable/modules/generated/sklearn.svm.OneClassSVM.html?highlight=one%20class%20svm#sklearn.svm.OneClassSVM>
  + <https://scikit-learn.org/stable/modules/outlier_detection.html#novelty-detection>
* An explanation page:  
  <https://towardsdatascience.com/outlier-detection-with-one-class-svms-5403a1a1878c>