

PYTHON BASICS

Python for data science

WORKING WITH ARRAYS

Numpy

DATA SCIENCE 2
DATA & A.I. 3

DATA VISUALISATION

Matplotlib

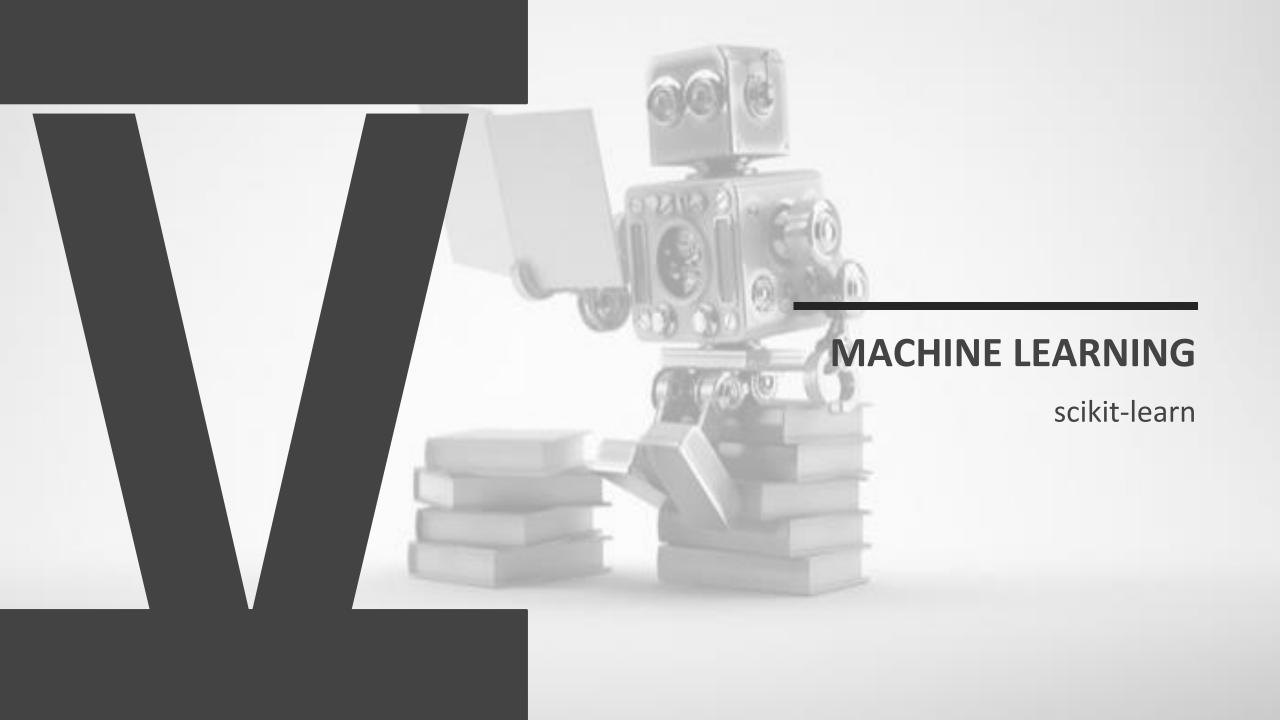
DATA ENGINEERING

pandas

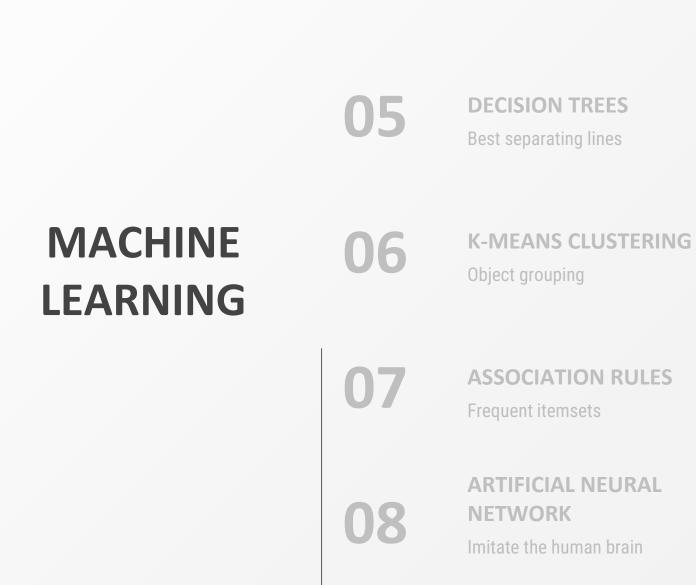
V

MACHINE LEARNING

Automatically find patterns



01	WHAT IS MACHINE LEARNING Automatically find patterns
02	INTRODUCING SCIKIT-LEARN Machine learning with Python
03	HYPERPARAMETERS AND CROSS VALIDATION Holdout samples and cross-validation
04	REGRESSION Best fitting line





SCIKIT-LEARN

PYTHON PACKAGE/LIBRARY FOR MACHINE LEARNING

- General purpose machine learning library (for supervised and unsupervised learning)
- Very commonly used

CONSISTENT INTERFACE TO MANY MACHINE LEARING TECHNIQUES

Common methods to.

- Transform data
- Train model
- Predict data
- Validate model

=> Uniform access to many machine learning techniques

FEATURES

Supervised learning is a kind of dependency model, i.e. a model in which a feature/variable (the dependent variable) is estimated based on a series of other features/variables (independent variables)

In statistics, one talks about dependent and independent variables, in machine learning one talks about target features and predictors or independent features.

- Predictors (independent features/variables): the features that can be used to estimate the target feature
- Target feature (dependent feature/variable): the feature to be estimated

LABELED DATASET

Supervised machine learning is based on training, i.e. deriving a model based on examples, with one example being one set of predictors and the known outcome for that set of predictors. Hence, for supervised machine learning, one needs a labeled dataset, a dataset with example with for every example the known outcome (the known outcome of an example is also called a label)

So a labeled dataset contains a set of example features (dependent features or predictors) and the known outcome or label for each of those examples.

SCIKIT-LEARN: FEATURE MATRIX AND TARGET ARRAY

- X = feature matrix: the feature matrix is commonly rerred to as X (capital X because it is a matrix)
- y = target array : the target array is commonly referred to as y (small y because it is mostly a vector,)
 (capital Y if it is a matrix for multivariate scikit)

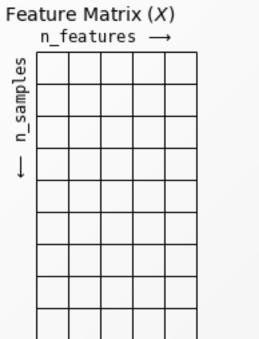
Mind that feature matrix X must be a 2-dimensional structure or matrix in scikit-lean, even it there is only one single feature or predictor. So in case of a single feature in a Pandas series or a one-dimensional Numpy array, that Pandas series needs to be converted to a Pandas dataframe explicitly, or the one-dimensional Numpy array must be converted to a two-dimensional Numpy array explicitly.

SCIKIT-LEARN: FEATURE MATRIX AND TARGET ARRAY

So the starting point of supervised machine learning is a labeled dataset. In scikit-learn, this labeled dataset is split into two arrays or matrices:

- Feature matrix: the table with the values of the independent features or predictors. It is a table structure because it consists of rows (examples) and columns (independent features or predictors. As it is a table, it can be represented in Python by a pandas dataframe, or by a 2-dimensional Numpy array (matrix)
- Target array: the vector with the labels, the values of the dependent features or target feature. It is mostly a vector because it contains the labels for a single target feature (one label for every example). As it is a vector, it can be represented in Python by a pandas series, or by a 1-dimensional Numpy array (vector). However, some scikit-learn methods can handle multile target features at once, and in that case the targe array also becomes a table that can be represented by a pandas dataframe or a 2-dimensional Numpy array (matrix)

Mind that in many cases the independent features ore predictors and the target feature are interchangeable, i.e. any feature can play the role of predictor or target.



Target Vector (y)

n_samples

BASIC DATA ANALYTICS PIPELINE

DATA PREPARATION

- Load (labeled) source data
- Compile feature matrix
- Compile target array (for supervised methods)

MODEL SELECTION AND HYPERPARAMETER SELECTION (MODEL SPECIFIC))

- Decide on the method to use (linear regression, decision tree, K-means clustering, ...)
- Decide on the hyperparameter to use (degree of polynomial, tree depth, number of clusters, ...)
 - Hyperparameters are parameters that the algorithm uses to derive a model
 - Hyperparameters depend on the method (every methods has it's on kind of hyperparameters)

DERIVE MODEL (TRAIN MODEL/FIT MODEL)

Apply the method/algorithm on the data to derive the model

DISPLAY MODEL (MODEL SPECIFIC)

Display the resulting model

The resulting model depends on the method used (equation of regression line with intercept and slope, decision tree with nodes and split conditions, groups of observations with centroid, ...)

APPLY MODEL ON NEW DATA

Apply the model on new data

In case of supervised machine learning methods, this will predict the target feature for new data

In case of unsupervised machine learning methods, this will restructure the feature data (clusters, association rules, new feature matrix with reduced dimensions)

BASIC PIPELINE WITH SCIKIT-LEARN

```
# DATA PREPARATION
import pandas as pd
pd.options.display.max rows = None
import seaborn as sns
iris = sns.load dataset('iris')
X = iris[['sepal_width', 'sepal_length', 'petal_width']] # Predictors
y = iris['petal_length'] # Target feature to predict
# MODEL SELECTION AND HYPERPARAMETER SELECTION (MODEL SPECIFIC)
from sklearn.linear model import LinearRegression
model = LinearRegression(fit intercept=True)
print(model)
# List all selected hyperparameters
print(model.get params(deep=True))
# DERIVE MODEL (TRAIN MODEL/FIT MODEL)
model.fit(X,y)
```

BASIC PIPELINE WITH SCIKIT-LEARN

```
# DISPLAY MODEL (MODEL SPECIFIC)
print(model.intercept_, model.coef_)

# APPLY MODEL ON NEW DATA

X_pred = .... (new feature data to predict the target feature for)
y_pred = model.predict(X_pred)
```