Python Biella Group

Introduzione al ML

Teoria, setup ambiente e qualche esempio

a cura di: **Federico Angaramo**

Python Biella Group

C’è sempre qualcosa da imparare, insieme!

- **YouTube**: https://www.youtube.com/c/PythonBiellaGroup
- **GitHub**: https://github.com/PythonBiellaGroup
- **Telegram** (PythonBiellaGroup) : https://t.me/joinchat/UZJZzGFKWf9JGGx5
- **Blog e Forum** *:* https://pythonbiella.herokuapp.com/
- **Docs**: https://pythonbiellagroup.it/it/

Tutto questo è stato reso possibile grazie a:

- Tutta la community di P.B.G.
- Maria Teresa Panunzio: https://www.linkedin.com/in/maria-teresa-panunzio-27ba3815/
- Mario Nardi: https://www.linkedin.com/in/mario-nardi-017705100/
- Andrea Guzzo: https://www.linkedin.com/in/andreaguzzo/

Python Biella Group

Agenda

Incontri e serate

01 Introduzione al Machine Learning (teoria)

02 Setup dell’ambiente

03 Introduzione a Scikit Learn

Materiale e codice sorgente

https://github.com/PythonBiellaGroup/

INIZIAMO!

https://github.com/PythonBiellaGroup/MaterialeSerate/

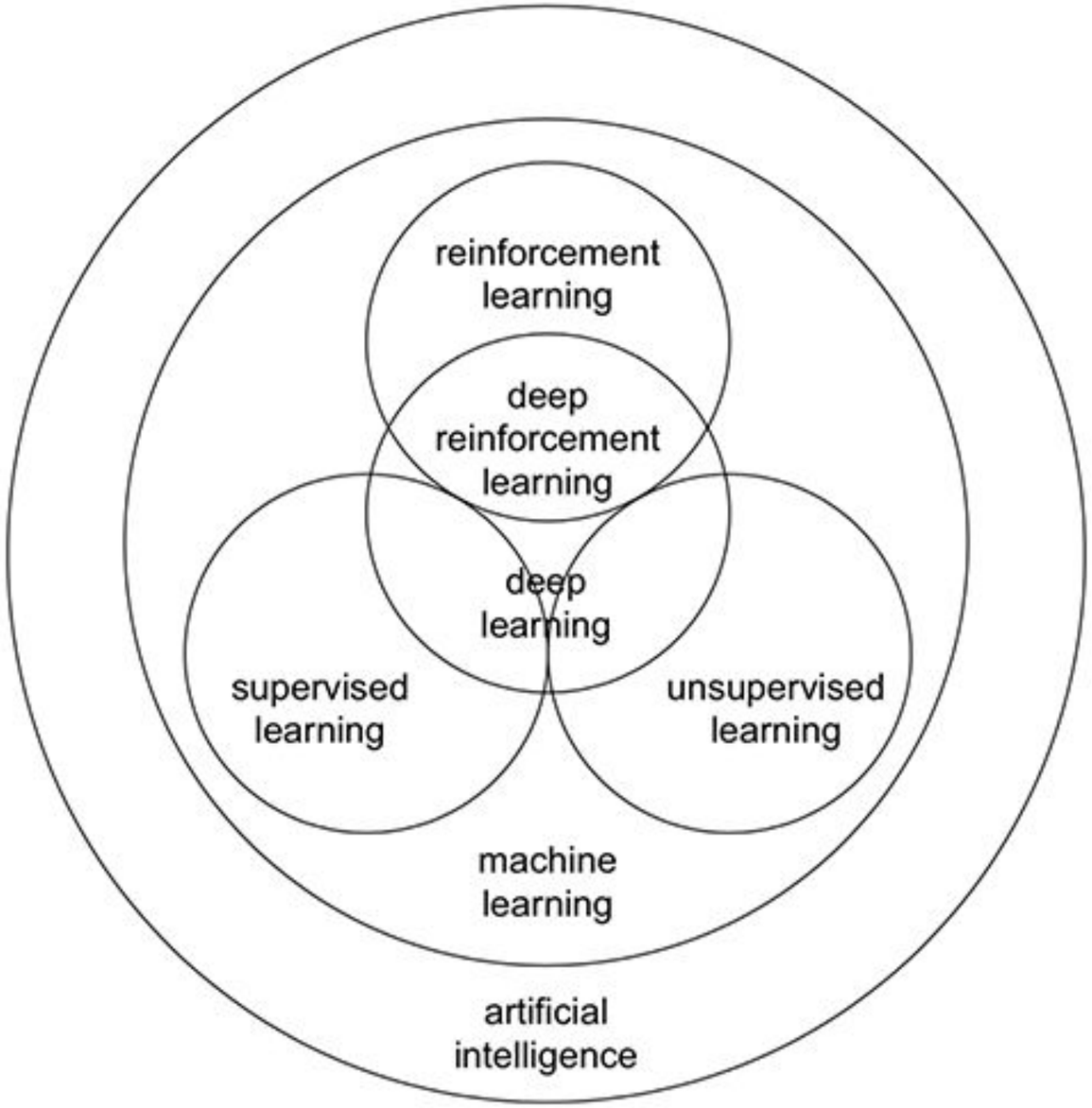
Introduzione al Machine Learning

- Cos’è il ML?

L'apprendimento automatico (anche detto machine learning in inglese) è una branca dell'intelligenza artificiale che raccoglie metodi che utilizzano metodi statistici per migliorare la performance di un algoritmo nell'identificare pattern nei dati.

Nell'ambito dell'informatica, l'apprendimento automatico è una variante alla programmazione tradizionale nella quale in una macchina si predispone l'abilità di apprendere qualcosa dai dati in maniera autonoma, senza istruzioni esplicite.

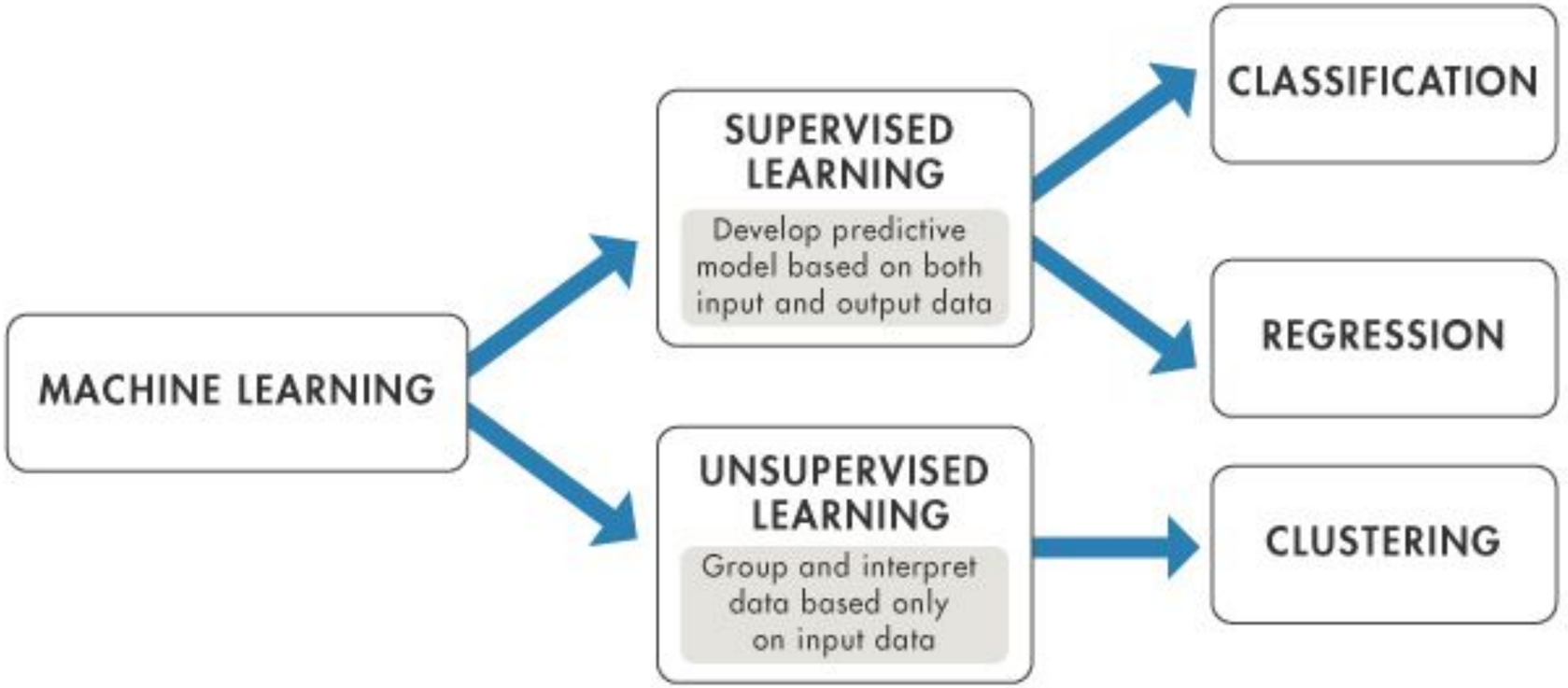
from wikipedia



Yuxi Li, Deep Reinforcement Learning, arXiv, 2018

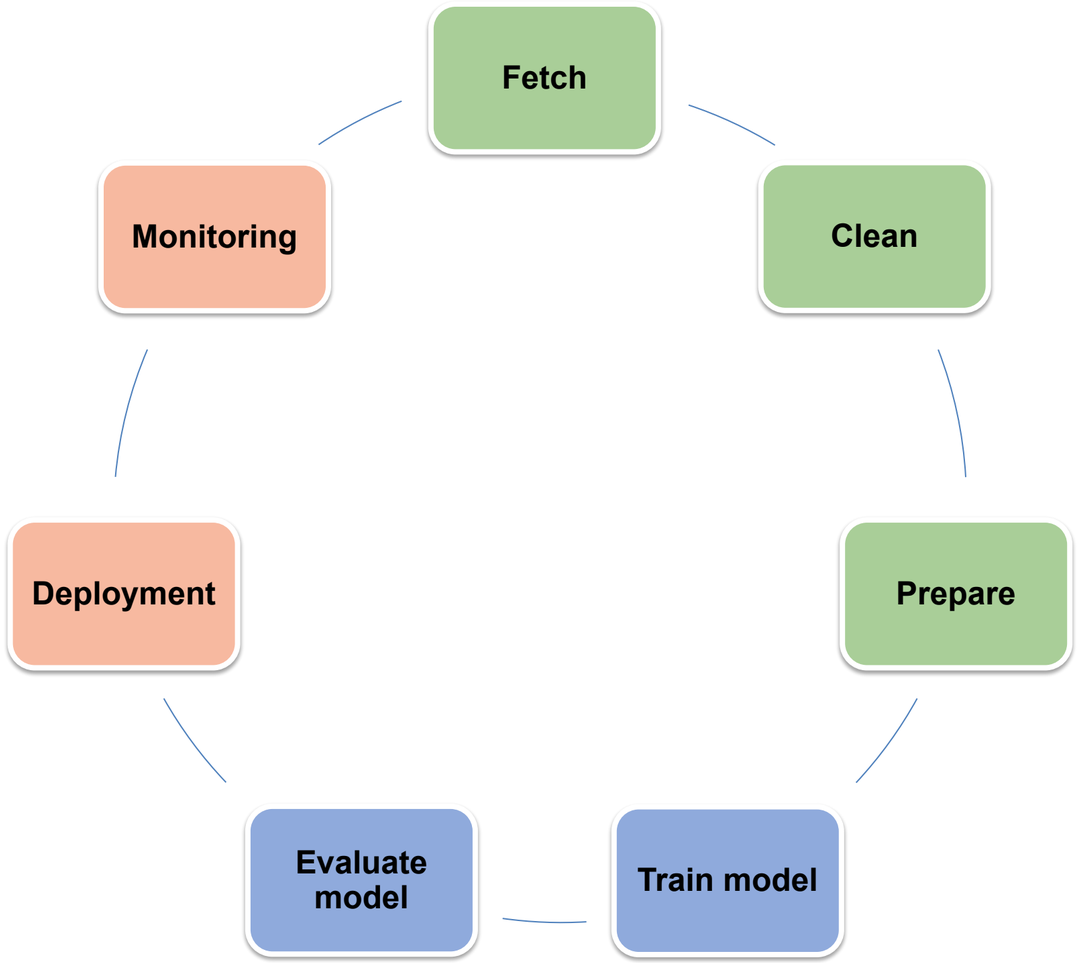
Introduzione al Machine Learning

- Tasks



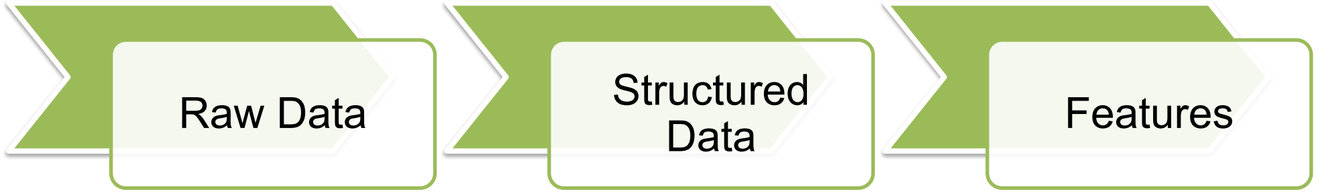
Introduzione al Machine Learning

- Ciclo di sviluppo di un modello
  - Data preparation
  - Model training
  - Model evaluation
  - Deployment + Monitoring

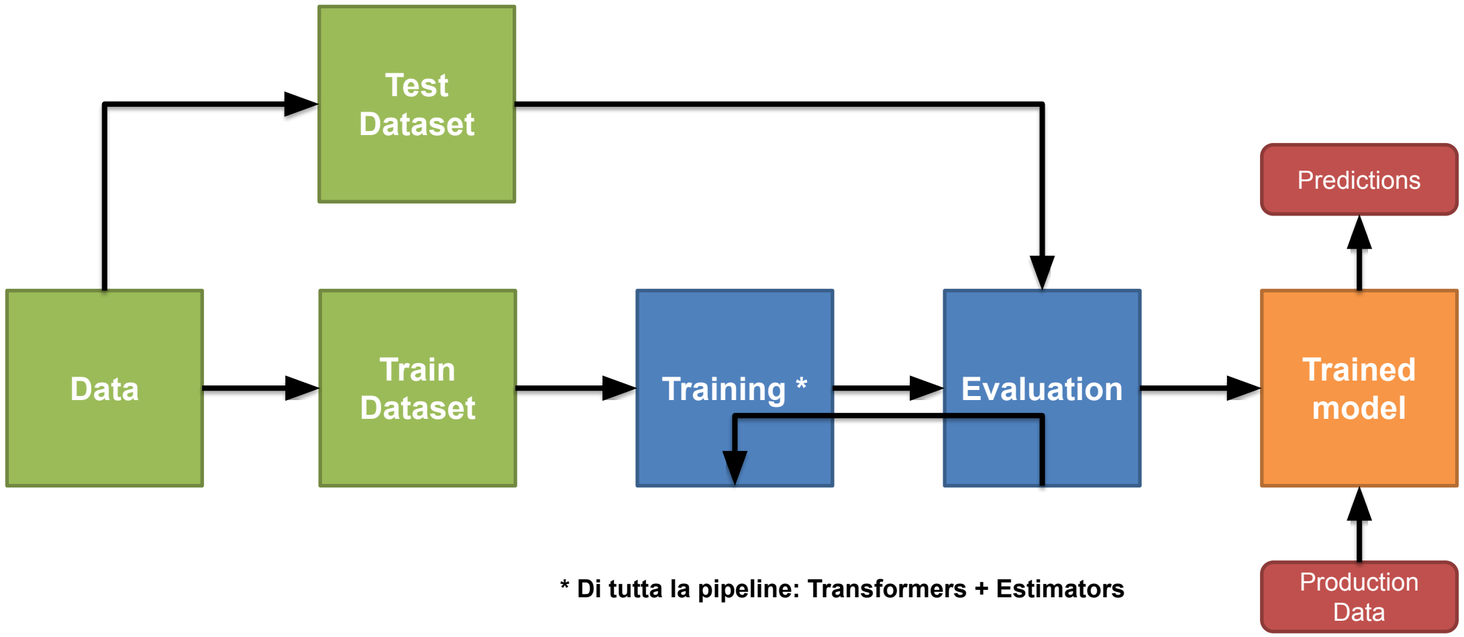


Supervised Learning

- Data Preparation



- Model Training + Model Evaluation



*** Di tutta la pipeline: Transformers + Estimators**

Setup dell'ambiente

Installare Miniconda (https://docs.conda.io/en/latest/miniconda.html#latest-miniconda-installer-links)

git clone https://github.com/PythonBiellaGroup/MaterialeSerate

conda env create -f PercorsoIntroML/envs/jupyterlab_env.yaml -n jupyterlab

conda env create -f PercorsoIntroML/envs/py39_pbg_env.yaml -n py39_pbg

Verificare di avere i nuovi ambienti: *conda env list*

conda activate jupyterlab
jupyter lab

Introduzione a Scikit Learn

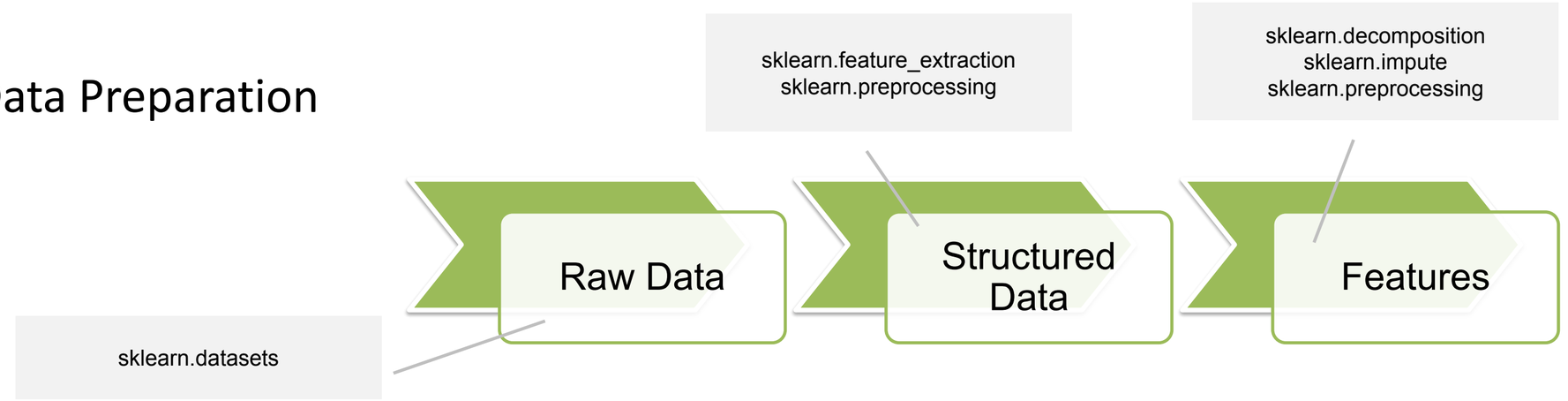
- Simple and efficient tools for predictive data analysis
- Accessible to everybody, and reusable in various contexts
- Built on NumPy, SciPy, and matplotlib
- Open source, commercially usable - BSD license

- Alcuni moduli

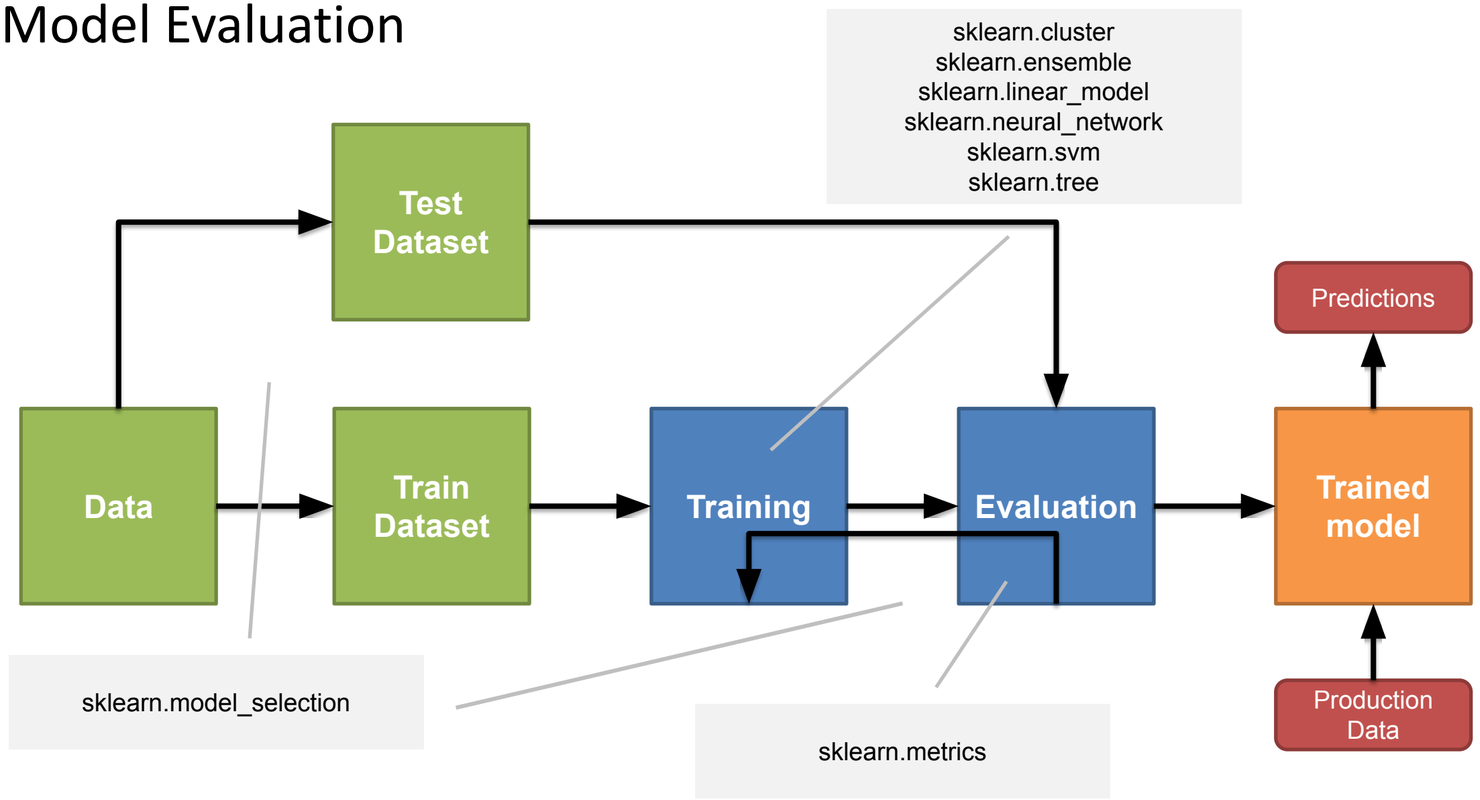
module	short description	description
sklearn.cluster	Clustering	Popular unsupervised clustering algorithms.
sklearn.datasets	Datasets	Utilities to load datasets, including methods to load and fetch popular reference datasets. It also features some artificial data generators.
sklearn.decomposition	Matrix Decomposition	Matrix decomposition algorithms, including among others PCA, NMF or ICA. Most of the algorithms of this module can be regarded as dimensionality reduction techniques.
sklearn.ensemble	Ensemble Methods	Ensemble-based methods for classification, regression and anomaly detection.
sklearn.feature_extraction	Feature Extraction	Feature extraction from raw data. It currently includes methods to extract features from text and images.
sklearn.feature_selection	Feature Selection	Feature selection algorithms. It currently includes univariate filter selection methods and the recursive feature elimination algorithm.
sklearn.impute	Impute	Transformers for missing value imputation
sklearn.inspection	Inspection	Tools for model inspection / explainability
sklearn.linear_model	Linear Models	a variety of linear models (classifiers / regressors)
sklearn.metrics	Metrics	Metrics and scoring to quantify the quality of predictions. It includes score functions, performance metrics and pairwise metrics and distance computations.
sklearn.model_selection	Model Selection	Model selection methods: Cross Validation / hyper-parameters tuning / etc
sklearn.multiclass	Multiclass classification	This module implements multiclass learning algorithms: one-vs-the-rest / one-vs-one / error correcting output codes. All are meta-estimators
sklearn.neighbors	Nearest Neighbors	k-nearest neighbors algorithms
sklearn.neural_network	Neural network models	Simple models based on neural networks.
sklearn.pipeline	Pipeline	Utilities to build a composite estimator, as a chain of transforms and estimators.
sklearn.preprocessing	Preprocessing and Normalization	Includes scaling, centering, normalization, binarization methods.
sklearn.svm	Support Vector Machines	Support Vector Machine algorithms.
sklearn.tree	Decision Trees	Decision Tree-based models for classification and regression.

Introduzione a Scikit Learn

- Data Preparation

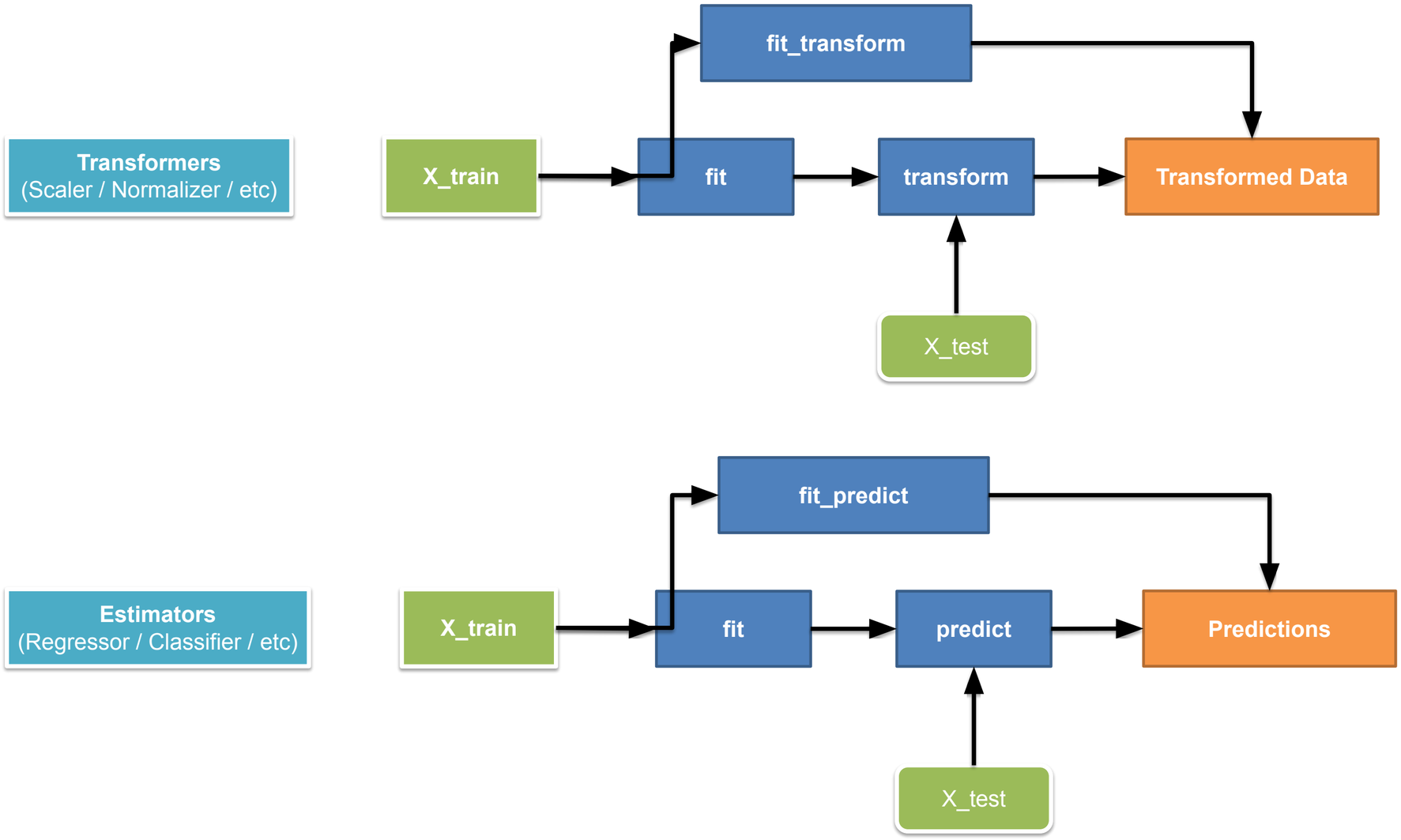


- Model Training + Model Evaluation



Introduzione a Scikit Learn

- Naming convention



scikit-learn algorithm cheat-sheet