



scikit-learn

Machine Learning in Python

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

Classification

Identifying to which category an object belongs to.

Applications: Spam detection, Image recognition.

Algorithms: SVM, nearest neighbors, random forest, ... — Examples

Regression

Predicting a continuous-valued attribute associated with an object.

Applications: Drug response, Stock prices.

Algorithms: SVR, ridge regression, Lasso, ... — Examples

Clustering

Automatic grouping of similar objects in

Applications: Customer segmentation, Grouping experiment outcomes

Algorithms: k-Means, spectral clustering, mean-shift, ... — Examples

Dimensionality reduction

Reducing the number of random variables to consider.

Applications: Visualization, Increased efficiency

Algorithms: PCA, feature selection, non-negative matrix factorization. — Examples

Model selection

Comparing, validating and choosing parameters and models.

Goal: Improved accuracy via parameter tuning

Modules: grid search, cross validation, metrics. — Examples

Preprocessing

Feature extraction and normalization.

Application: Transforming input data to text for use with machine learning algorithms

Modules: preprocessing, feature extraction. — Examples

Classification

- ▶ **Description:** Identifying to which category an object belongs to.
- ▶ **Applications:** Spam detection, Image recognition.
- ▶ **Algorithms:** SVM, nearest neighbors, random forest,

Regression

- ▶ **Description:** Predicting a continuous-valued attribute associated with an object.
- ▶ **Applications:** Drug response, Stock prices.
- ▶ **Algorithms:** SVR, ridge regression, Lasso,

Clustering

Automatic grouping of similar objects into sets. Applications: Customer segmentation, Grouping experiment outcomes Algorithms: k-Means, spectral clustering, mean-shift, ...

Dimensionality Reduction

- ▶ **Description:** Reducing the number of random variables to consider.
- ▶ **Applications:** Visualization, Increased efficiency
- ▶ **Algorithms:** PCA, feature selection, non-negative matrix factorization.

Model selection

- ▶ **Description:** Comparing, validating and choosing parameters and models.
- ▶ **Goal:** Improved accuracy via parameter tuning
- ▶ **Modules:** grid search, cross validation, metrics

Preprocessing

- ▶ **Description:** Feature extraction and normalization.
- ▶ **Application:** Transforming input data such as text for use with machine learning algorithms.
- ▶ **Modules:** preprocessing, feature extraction.