sklearn.pipeline.make_pipeline

sklearn.pipeline.make_pipeline(*steps, memory=None, verbose=False) [source]

Construct a Pipeline from the given estimators.

This is a shorthand for the Pipeline constructor; it does not require, and does not permit, naming the estimators. Instead, their names will be set to the lowercase of their types automatically.

Parameters:

steps : *list of Estimator objects

List of the scikit-learn estimators that are chained together.

memory : *str or object with the joblib.Memory interface, default=None*

Used to cache the fitted transformers of the pipeline. By default, no caching is performed. If a string is given, it is the path to the caching directory. Enabling caching triggers a clone of the transformers before fitting. Therefore, the transformer instance given to the pipeline cannot be inspected directly. Use the attribute `named_steps` or `steps` to inspect estimators within the pipeline. Caching the transformers is advantageous when fitting is time consuming.

verbose : *bool, default=False*

If True, the time elapsed while fitting each step will be printed as it is completed.

Returns:

p : *Pipeline*

Returns a scikit-learn Pipeline object.

See also:

Pipeline

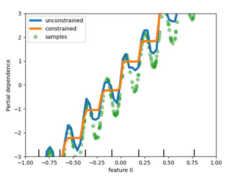
Class for creating a pipeline of transforms with a final estimator.

Examples

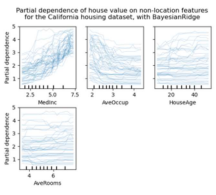
```
>>> from sklearn.naive_bayes import GaussianNB
>>> from sklearn.preprocessing import StandardScaler
>>> from sklearn.pipeline import make_pipeline
>>> make_pipeline(StandardScaler(), GaussianNB(priors=None))
Pipeline(steps=[('standardscaler', StandardScaler()),
                ('gaussiannb', GaussianNB())])
```

Examples using `sklearn.pipeline.make_pipeline`

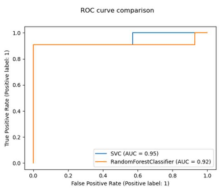
Release Highlights for scikit-learn 1.0



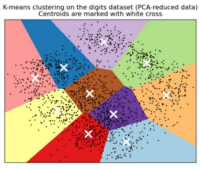
Release Highlights for scikit-learn 0.23



Release Highlights for scikit-learn 0.24



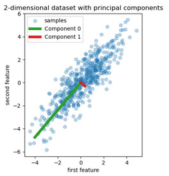
Release Highlights for scikit-learn 0.22



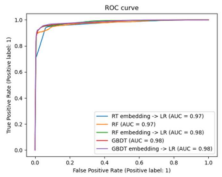
A demo of K-Means clustering on the handwritten digits

Toggle Menu

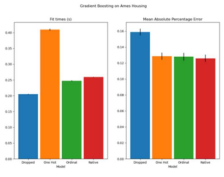
data



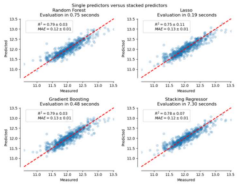
Principal Component Regression vs Partial Least Squares Regression



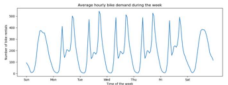
Feature transformations with ensembles of trees



Categorical Feature Support in Gradient Boosting

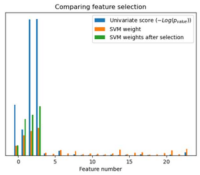


Combine predictors using stacking

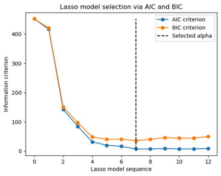


Time-related feature engineering

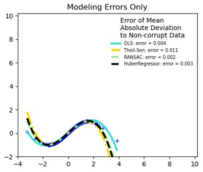
Pipeline ANOVA SVM



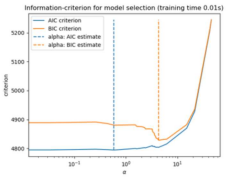
Univariate Feature Selection



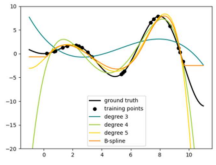
Lasso model selection via information criteria



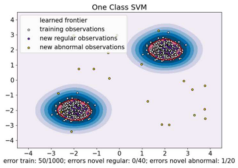
Robust linear estimator fitting



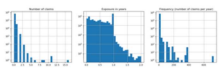
Lasso model selection: AIC-BIC / cross-validation



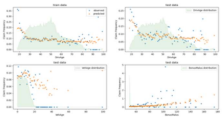
Polynomial and Spline interpolation



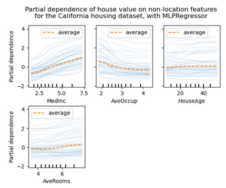
One-Class SVM versus One-Class SVM using Stochastic Gradient Descent



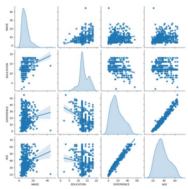
Poisson regression and non-normal loss



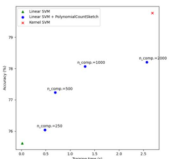
Tweedie regression on insurance claims



Partial Dependence and Individual Conditional Expectation Plots

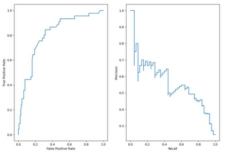


Common pitfalls in the interpretation of coefficients of linear models

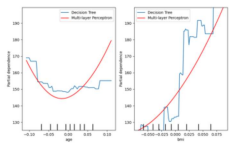


Scalable learning with polynomial kernel approximation

Manifold learning on handwritten digits: Locally Linear Embedding, Isomap...

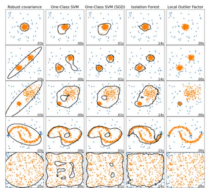


Visualizations with Display Objects

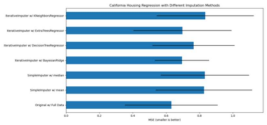


Advanced Plotting With Partial Dependence

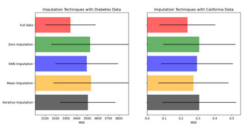
Displaying Pipelines



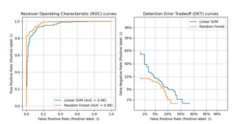
Comparing anomaly detection algorithms for outlier detection



Imputing missing values with variants of IterativeImputer

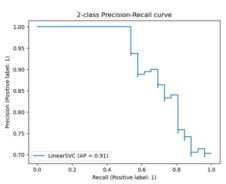


Imputing missing values before building an estimator

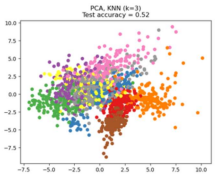


Detection error tradeoff (DET) curve

on toy datasets

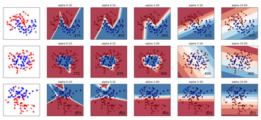


Precision-Recall

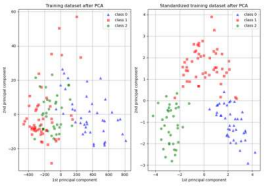


Dimensionality Reduction with Neighborhood Components Analysis

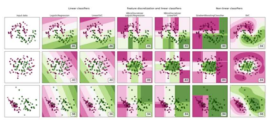
Approximate nearest neighbors in TSNE



Varying regularization in Multi-layer Perceptron



Importance of Feature Scaling



Feature discretization

Clustering text documents using k-means