Train In Data

Open-source Python packages for Feature Selection

Open-source for Feature engineering

scikit learn

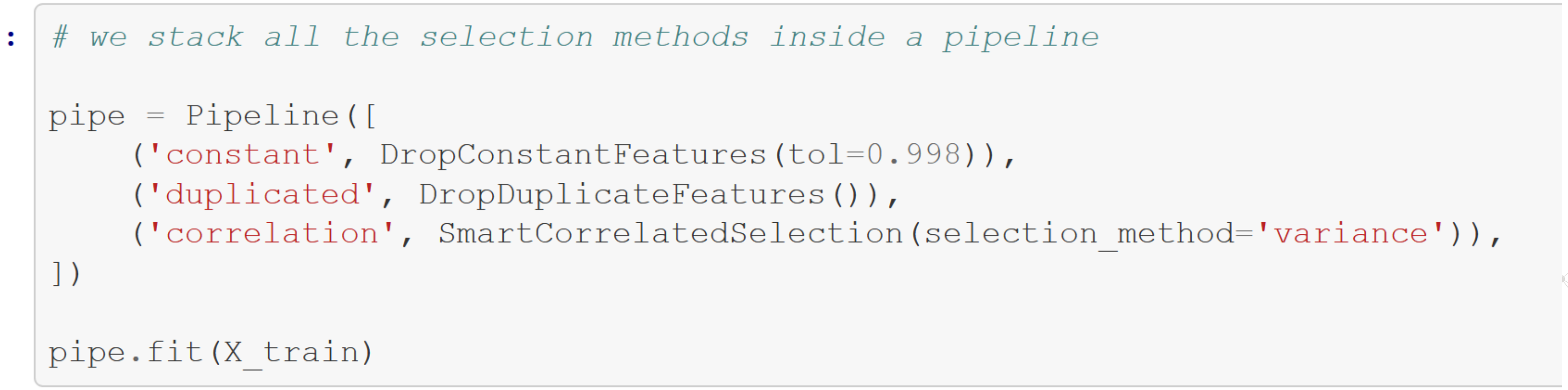
MLXTEND

Feature-engine

Fit – transform functionality

- fit() ➔ finds important features
- transform() ➔ transforms data
  - Removes unwanted features

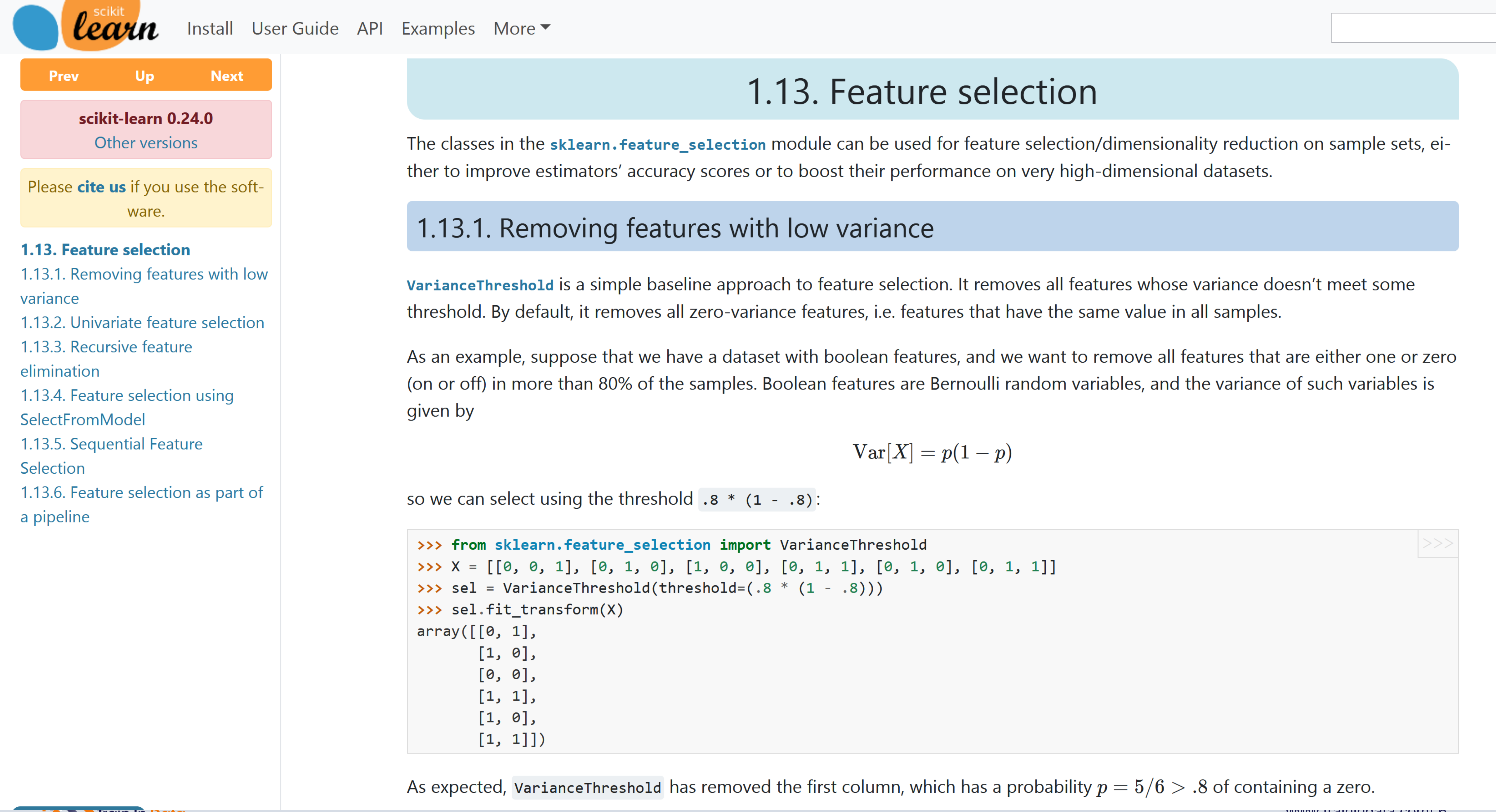
Pipeline



```
# we stack all the selection methods inside a pipeline

pipe = Pipeline([
    ('constant', DropConstantFeatures(tol=0.998)),
    ('duplicated', DropDuplicateFeatures()),
    ('correlation', SmartCorrelatedSelection(selection_method='variance')),
])

pipe.fit(X_train)
```

Pipeline

```
# train pipeline
price_pipe.fit(X_train, y_train)

# transform data
price_pipe.transform(X_train)
price_pipe.transform(X_test)
```



scikit-learn 0.24.0
Other versions

Please **cite us** if you use the software.

1.13. Feature selection

The classes in the `sklearn.feature_selection` module can be used for feature selection/dimensionality reduction on sample sets, either to improve estimators' accuracy scores or to boost their performance on very high-dimensional datasets.

1.13.1. Removing features with low variance

`VarianceThreshold` is a simple baseline approach to feature selection. It removes all features whose variance doesn't meet some threshold. By default, it removes all zero-variance features, i.e. features that have the same value in all samples.

As an example, suppose that we have a dataset with boolean features, and we want to remove all features that are either one or zero (on or off) in more than 80% of the samples. Boolean features are Bernoulli random variables, and the variance of such variables is given by

$$\mathrm{Var}[X] = p(1 - p)$$

so we can select using the threshold `.8 * (1 - .8)`:

```
>>> from sklearn.feature_selection import VarianceThreshold
>>> X = [[0, 0, 1], [0, 1, 0], [1, 0, 0], [0, 1, 1], [0, 1, 0], [0, 1, 1]]
>>> sel = VarianceThreshold(threshold=(.8 * (1 - .8)))
>>> sel.fit_transform(X)
array([[0, 1],
       [1, 0],
       [0, 0],
       [1, 1],
       [1, 0],
       [1, 1]])
```

As expected, `VarianceThreshold` has removed the first column, which has a probability $p = 5/6 > .8$ of containing a zero.

MLXTEND

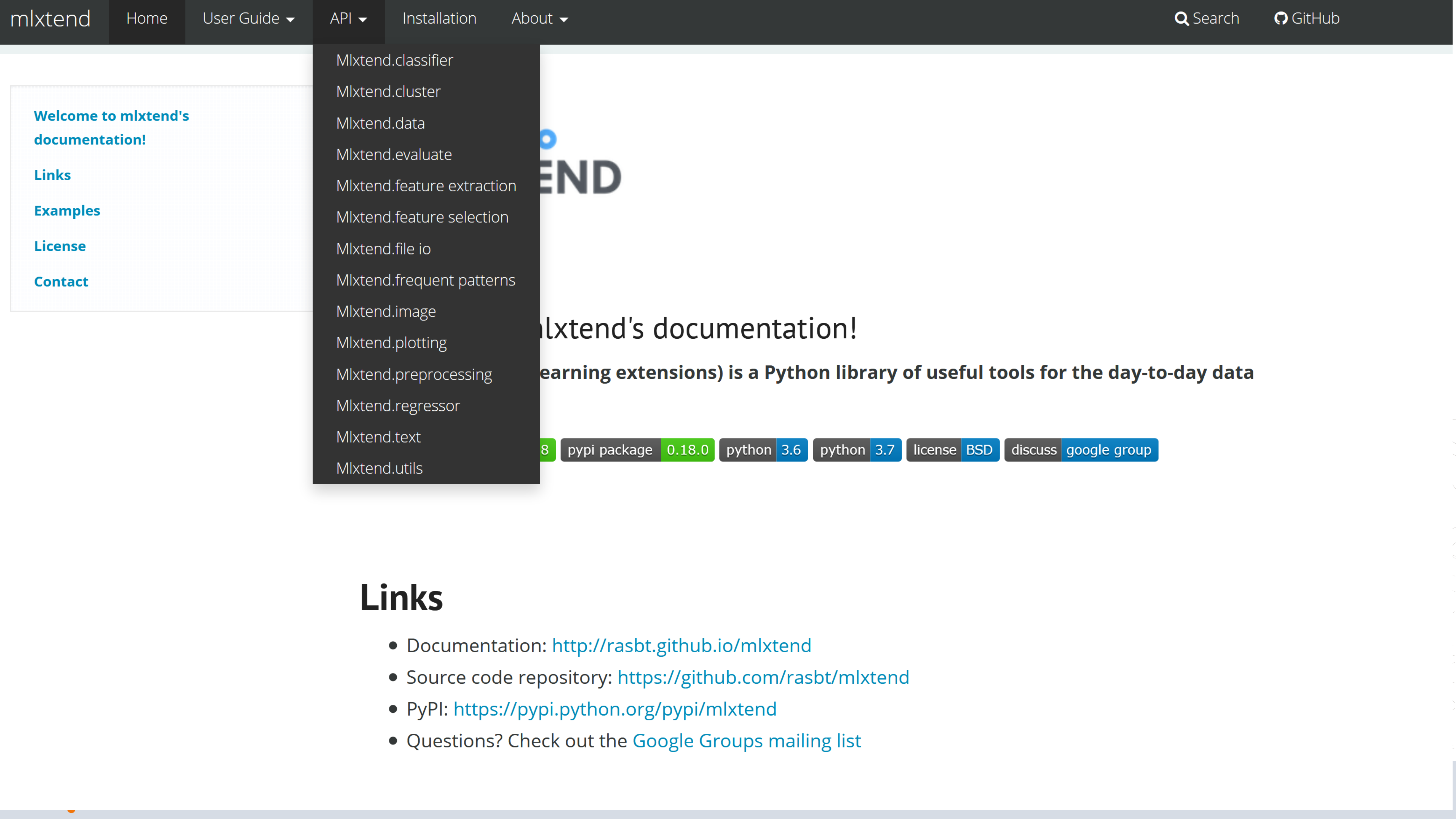
Welcome to mlxtend's documentation!

Mlxtend (machine learning extensions) is a Python library of useful tools for the day-to-day data science tasks.

JOSS 10.21105/joss.00638 pypi package 0.18.0 python 3.6 python 3.7 license BSD discuss google group

Links

- Documentation: http://rasbt.github.io/mlxtend
- Source code repository: https://github.com/rasbt/mlxtend
- PyPI: https://pypi.python.org/pypi/mlxtend
- Questions? Check out the Google Groups mailing list



lxtend's documentation!

earning extensions) is a Python library of useful tools for the day-to-day data

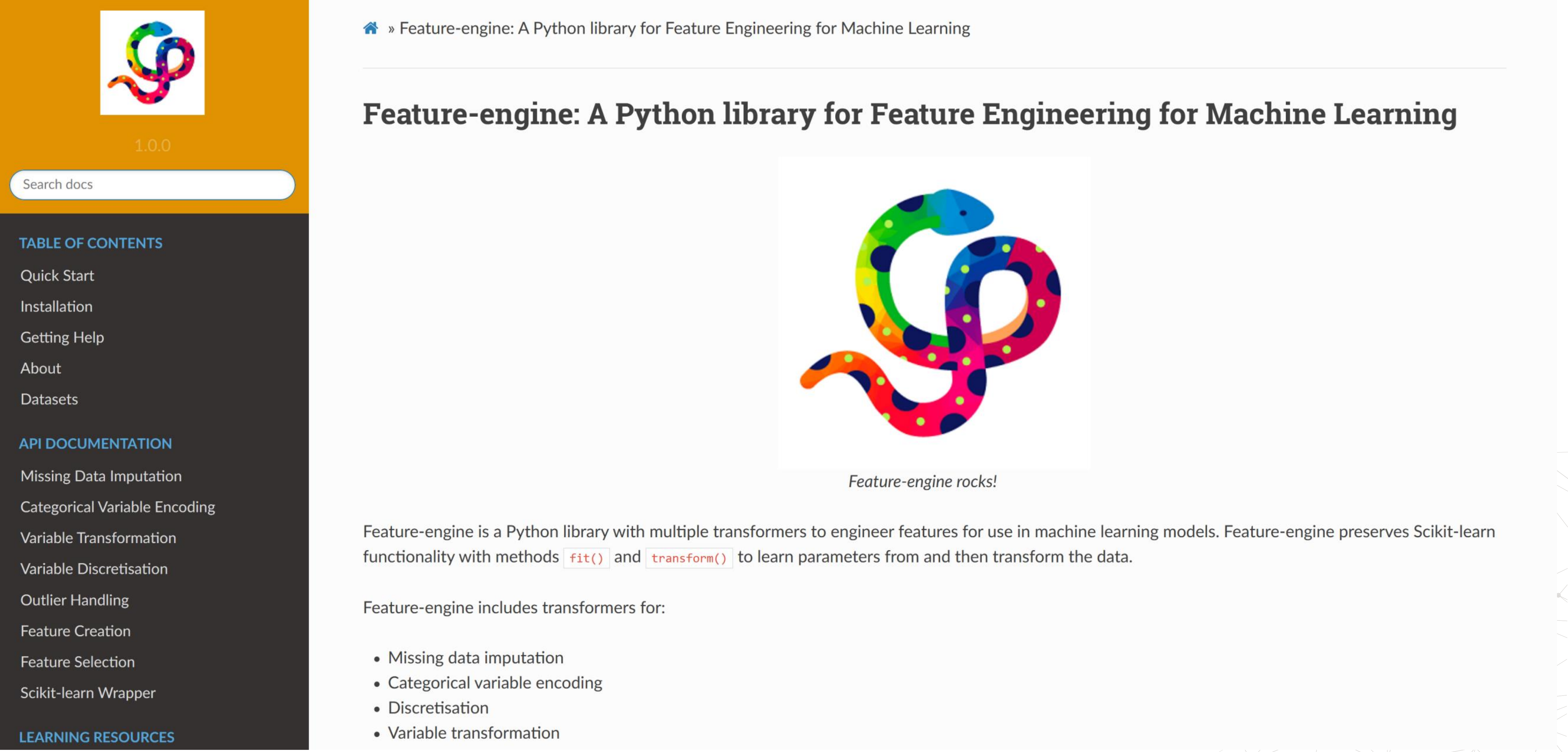
Links

- Documentation: http://rasbt.github.io/mlxtend
- Source code repository: https://github.com/rasbt/mlxtend
- PyPI: https://pypi.python.org/pypi/mlxtend
- Questions? Check out the Google Groups mailing list

Feature-engine

- https://www.trainindata.com/feature-engine
- https://feature-engine.readthedocs.io/en/latest/
- https://github.com/solegalli/feature_engine

```
pip install feature-engine

conda install –c conda-forge feature_engine
```



1.0.0

Search docs

» Feature-engine: A Python library for Feature Engineering for Machine Learning

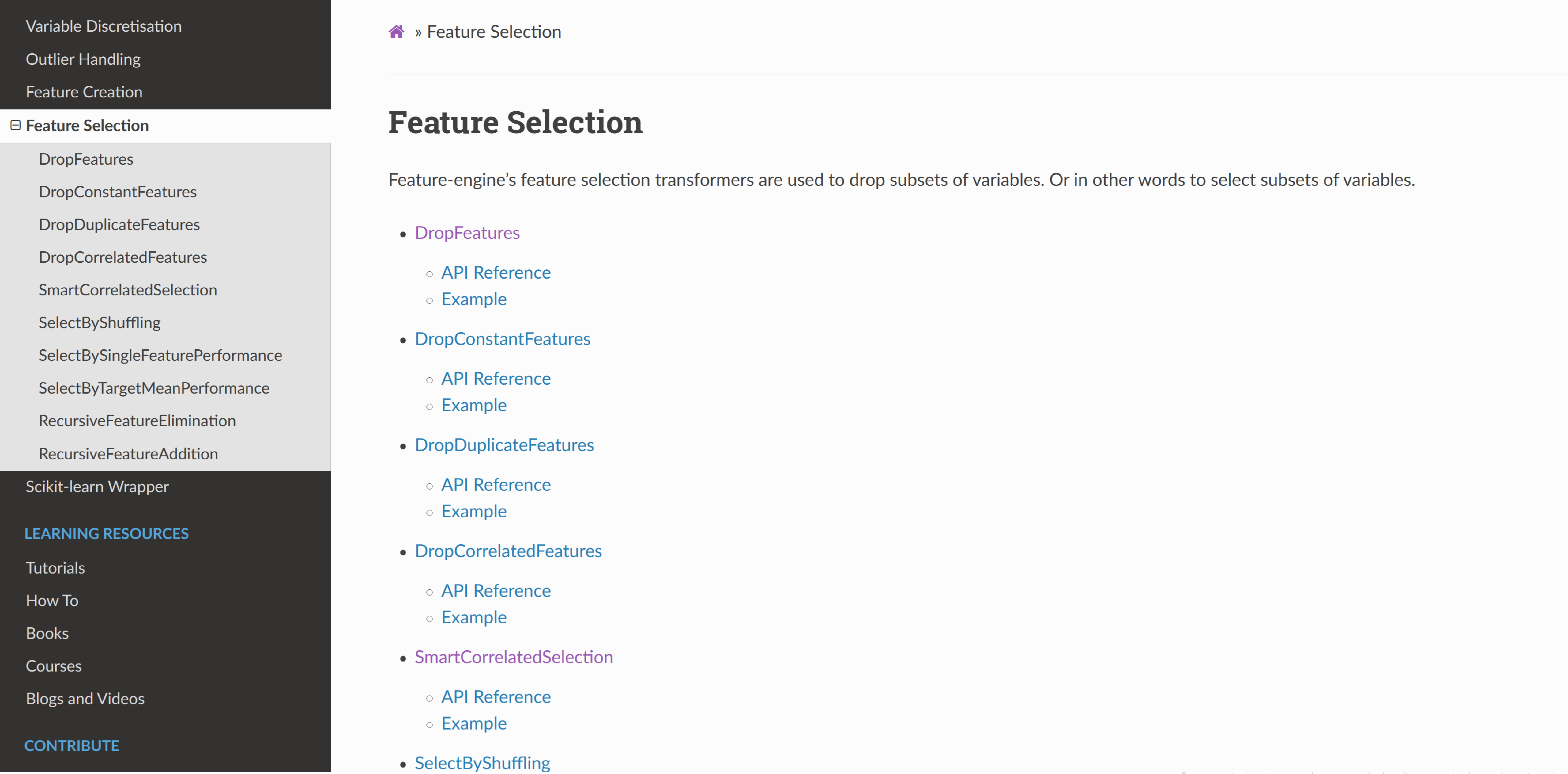
Feature-engine: A Python library for Feature Engineering for Machine Learning

Feature-engine rocks!

Feature-engine is a Python library with multiple transformers to engineer features for use in machine learning models. Feature-engine preserves Scikit-learn functionality with methods `fit()` and `transform()` to learn parameters from and then transform the data.

Feature-engine includes transformers for:

- Missing data imputation
- Categorical variable encoding
- Discretisation
- Variable transformation



Variable Discretisation
Outlier Handling
Feature Creation
⊟ Feature Selection
DropFeatures
DropConstantFeatures
DropDuplicateFeatures
DropCorrelatedFeatures
SmartCorrelatedSelection
SelectByShuffling
SelectBySingleFeaturePerformance
SelectByTargetMeanPerformance
RecursiveFeatureElimination
RecursiveFeatureAddition
Scikit-learn Wrapper

LEARNING RESOURCES
Tutorials
How To
Books
Courses
Blogs and Videos

CONTRIBUTE

» Feature Selection

Feature Selection

Feature-engine's feature selection transformers are used to drop subsets of variables. Or in other words to select subsets of variables.

- DropFeatures
  - API Reference
  - Example
- DropConstantFeatures
  - API Reference
  - Example
- DropDuplicateFeatures
  - API Reference
  - Example
- DropCorrelatedFeatures
  - API Reference
  - Example
- SmartCorrelatedSelection
  - API Reference
  - Example
- SelectByShuffling

Train In Data

Thank you

www.trainindata.com