

Data Preprocessing

```
from sklearn.preprocessing import <classname>
StandardScaler, MinMaxScaler, RobustScaler
QuantileTransformer, PowerTransformer,
FunctionTransformer
KBinsDiscretizer, PolynomialFeatures,
Normalizer
scaler = StandardScaler()
# Apply a user-defined function to the data
transformer = FunctionTransformer(n_jobs=-1)
# Discretize features into k bins
discretizer = KBinsDiscretizer(n_bins=3,
                                encode='ordinal', strategy='uniform')
poly_features = PolynomialFeatures(degree=2)
X_scaled = <object>.fit_transform(X)
```

Encoding Categorical Data

```
from sklearn.preprocessing import <classname>
LabelEncoder, OneHotEncoder, OrdinalEncoder,
LabelBinarizer
tb = OneHotEncoder()
le = LabelEncoder()
lb = LabelBinarizer()
y = le.fit_transform(['Yes', 'No', 'Yes'])
y = lb.fit_transform(['Yes', 'No', 'Yes'])
X_encoded = tb.fit_transform(X)
```

Handling missing values

```
from sklearn.impute import SimpleImputer,
KNNImputer, IterativeImputer, MissingIndicator
from sklearn.experimental import enable_iterative_imputer
imputer = SimpleImputer(strategy='mean')
imputer = KNNImputer(n_neighbors=2)
imputer = IterativeImputer(random_state=0)
indicator = MissingIndicator()
X_imputed = imputer.fit_transform(X)
```

Feature Selection:

```
from sklearn.feature_selection import
SelectKBest, SelectPercentile, SelectFromModel,
VarianceThreshold, RFE, RFECV,
SequentialFeatureSelector
```

Dimensionality Reduction

```
from sklearn.decomposition import
PCA, IncrementalPCA, TruncatedSVD, KernelPCA,
NMF, FastICA, LatentDirichletAllocation
pca = PCA(n_components=2)
kpca = KernelPCA(n_components=2, kernel='rbf')
tsne = TSNE(n_components=2)
X_new = any.fit_transform(X)
```

Pipelines:

```
from sklearn.pipeline import
Pipeline
FeatureUnion
ColumnTransformer
```



Supervised Learning Models:

Linear Models:

LinearRegression, Ridge, Lasso, ElasticNet, LogisticRegression, SGDClassifier, SGDRegressor, Perceptron

Naive Bayes:

GaussianNB, BernoulliNB, MultinomialNB,

Tree-Based Models:

DecisionTreeClassifier, DecisionTreeRegressor,

Support Vector Machines (SVM):

SVC, SVR, LinearSVC, LinearSVR, NuSVC, NuSVR, OneClassSVM

Nearest Neighbors:

KNeighborsClassifier, KNeighborsRegressor, RadiusNeighborsClassifier, RadiusNeighborsRegressor

Neural Networks:

MLPClassifier, MLPRegressor

Ensemble

RandomForestClassifier, RandomForestRegressor, GradientBoostingClassifier, GradientBoostingRegressor, ExtraTreesClassifier, ExtraTreesRegressor, AdaBoostClassifier, AdaBoostRegressor

xgboost

XGBClassifier, XGBRegressor

lightgbm

LGBMClassifier, LGBMRegressor

catboost

CatBoostClassifier, CatBoostRegressor,

```
from sklearn.linear_model import LogisticRegression, SGDClassifier, SGDRegressor, Perceptron
from sklearn.naive_bayes import GaussianNB, BernoulliNB, MultinomialNB
from sklearn.tree import DecisionTreeClassifier, DecisionTreeRegressor
from sklearn.ensemble import RandomForestClassifier, RandomForestRegressor, GradientBoostingClassifier, GradientBoostingRegressor, ExtraTreesClassifier, ExtraTreesRegressor, AdaBoostClassifier, AdaBoostRegressor
from xgboost import XGBClassifier, XGBRegressor
from lightgbm import LGBMClassifier, LGBMRegressor
from catboost import CatBoostClassifier, CatBoostRegressor
from sklearn.svm import SVC, SVR, LinearSVC, LinearSVR, NuSVC, NuSVR, OneClassSVM
from sklearn.neighbors import KNeighborsClassifier, KNeighborsRegressor, RadiusNeighborsClassifier, RadiusNeighborsRegressor
from sklearn.neural_network import MLPClassifier, MLPRegressor
```

Semi-Supervised Learning:

LabelPropagation

LabelSpreading

Unsupervised Learning Models

Clustering:

KMeans, AgglomerativeClustering, DBSCAN, Birch, SpectralClustering

Dimensionality Reduction:

PCA, IncrementalPCA, TruncatedSVD, KernelPCA, NMF, FastICA, LatentDirichletAllocation

Clustering

KMeans

AgglomerativeClustering

DBSCAN

Birch

SpectralClustering

Model Evaluation Metrics

Regression Metrics:

mean_squared_error, r2_score, mean_absolute_error, explained_variance_score, median_absolute_error, mean_squared_log_error

Classification Metrics:

accuracy_score, precision_score, recall_score, f1_score, roc_auc_score, average_precision_score, log_loss, confusion_matrix, classification_report

```
from sklearn.metrics import
```



By **Daryabi**

cheatography.com/daryabi/

Not published yet.

Last updated 21st March, 2023.

Page 2 of 2.

Sponsored by **Readable.com**

Measure your website readability!

<https://readable.com>