

Scikit-Learn Important Modules and Functions

1. `sklearn.datasets`:

- **Functions:** Load and generate datasets for machine learning tasks.
 - `load_*`: Load pre-existing datasets like Iris, Boston, etc.
 - `fetch_*`: Download datasets from online repositories.
 - `make_*`: Generate synthetic datasets with specific properties.

2. `sklearn.preprocessing`:

- **Classes:** Preprocessing and feature scaling.
 - `StandardScaler`: Standardize features by removing the mean and scaling to unit variance.
 - `MinMaxScaler`: Scale features to a specified range (e.g., [0, 1]).
 - `RobustScaler`: Scale features using statistics that are robust to outliers.
 - `LabelEncoder`: Encode target labels with values between 0 and `n_classes-1`.
 - `OneHotEncoder`: Encode categorical features as a one-hot numeric array.
 - `PolynomialFeatures`: Generate polynomial and interaction features from input data.

3. `sklearn.model_selection`:

- **Functions:** Model selection and evaluation.
 - `train_test_split`: Split arrays or matrices into random train and test subsets.
 - `cross_val_score`: Evaluate a score by cross-validation.
 - `GridSearchCV`: Exhaustively search for the best hyperparameters within a specified parameter grid.
 - `StratifiedKFold`: Provides stratified k-fold cross-validation.

4. `sklearn.feature_extraction`:

- **Classes:** Feature extraction from raw data.
 - `CountVectorizer`: Convert a collection of text documents to a matrix of token counts.
 - `TfidfVectorizer`: Convert a collection of raw documents to a matrix of TF-IDF features.

5. `sklearn.feature_selection`:

- **Classes:** Feature selection and dimensionality reduction.
 - `SelectKBest`: Select features based on the k highest scores.
 - `RFE`: Perform recursive feature elimination with cross-validation.

6. `sklearn.decomposition`:

- **Classes:** Matrix factorization-based dimensionality reduction.
 - `PCA`: Principal Component Analysis.
 - `NMF`: Non-Negative Matrix Factorization.

7. `sklearn.linear_model`:

- **Classes:** Linear models for regression and classification.
 - `LinearRegression`: Ordinary least squares Linear Regression.
 - `Ridge`: Linear regression with L2 regularization.
 - `Lasso`: Linear regression with L1 regularization.
 - `LogisticRegression`: Logistic Regression.
 - `SGDClassifier`: Linear classifiers with Stochastic Gradient Descent (SGD) training.

8. `sklearn.tree`:

- **Classes:** Decision tree-based models.
 - `DecisionTreeClassifier`: Decision Tree Classifier.
 - `DecisionTreeRegressor`: Decision Tree Regressor.

9. `sklearn.ensemble`:

- **Classes:** Ensemble methods for classification and regression.
 - `RandomForestClassifier`: Random Forest Classifier.
 - `RandomForestRegressor`: Random Forest Regressor.
 - `GradientBoostingClassifier`: Gradient Boosting Classifier.
 - `AdaBoostClassifier`: AdaBoost Classifier.

10. `sklearn.cluster`:

- **Classes:** Clustering algorithms.
 - `KMeans`: K-means clustering algorithm.
 - `DBSCAN`: Density-Based Spatial Clustering of Applications with Noise.
 - `AgglomerativeClustering`: Agglomerative hierarchical clustering.
 - `MeanShift`: Mean shift clustering algorithm.

11. `sklearn.metrics`:

- **Functions:** Metrics for evaluating model performance.
 - `accuracy_score`: Accuracy classification score.
 - `precision_score`: Compute the precision score.
 - `recall_score`: Compute the recall score.
 - `f1_score`: Compute the F1 score.
 - `mean_squared_error`: Mean squared error regression loss.

12. `sklearn.pipeline`:

- **Classes:** Pipeline of transforms with a final estimator.
 - `Pipeline`: Pipeline construction and execution.

13. `sklearn.externals.joblib`:

- **Functions:** Saving and loading scikit-learn models.
 - `dump`: Save an estimator, transformer, or pipeline to disk.
 - `load`: Load an estimator, transformer, or pipeline from disk.

14. `sklearn.compose`:

- **Classes:** Combine multiple transformers or estimators into a single transformer or estimator.
 - `ColumnTransformer`: Applies different transformers to different columns of an array or dataframe.

- **TransformerMixin:** Base class for transformers.
- **FunctionTransformer:** Applies a user-defined function to each input.

15. `sklearn.impute:`

- **Classes:** Imputation of missing values.
 - **SimpleImputer:** Imputation transformer for completing missing values.
 - **KNNImputer:** Imputation transformer based on k-nearest neighbors.

16. `sklearn.metrics.cluster:`

- **Functions:** Clustering performance evaluation metrics.
 - **adjusted_rand_score:** Rand index adjusted for chance.
 - **silhouette_score:** Compute the mean Silhouette Coefficient of all samples.
 - **calinski_harabasz_score:** Calinski-Harabasz index.

17. `sklearn.metrics.pairwise:`

- **Functions:** Pairwise distances and kernels.
 - **cosine_similarity:** Compute cosine similarity between samples.
 - **euclidean_distances:** Compute pairwise Euclidean distances between samples.

18. `sklearn.mixture:`

- **Classes:** Gaussian Mixture Models (GMM) for clustering and density estimation.
 - **GaussianMixture:** Gaussian Mixture Model.

19. `sklearn.multioutput:`

- **Classes:** Multi-output regression and classification.
 - **MultiOutputRegressor:** Multi-output regression.
 - **MultiOutputClassifier:** Multi-output classification.

20. `sklearn.naive_bayes:`

- **Classes:** Naive Bayes classifiers.
 - **GaussianNB:** Gaussian Naive Bayes.
 - **MultinomialNB:** Multinomial Naive Bayes.
 - **ComplementNB:** Complement Naive Bayes.

21. `sklearn.neighbors:`

- **Classes:** Nearest neighbors-based methods.
 - **KNeighborsClassifier:** k-nearest neighbors classifier.
 - **KNeighborsRegressor:** k-nearest neighbors regressor.
 - **RadiusNeighborsClassifier:** radius-based neighbors classifier.
 - **RadiusNeighborsRegressor:** radius-based neighbors regressor.

22. `sklearn.neural_network:`

- **Classes:** Feedforward neural networks.
 - **MLPClassifier:** Multi-layer Perceptron classifier.
 - **MLPRegressor:** Multi-layer Perceptron regressor.

23. `sklearn.pipeline:`

- **Classes:** Pipeline of transforms with a final estimator.
 - **FeatureUnion:** Combine several transformer objects into a single transformer.

24. `sklearn.preprocessing:`

- **Classes:** Encoding and scaling.

- `LabelBinarizer`: Binarize labels in a one-vs-all fashion.
- `LabelEncoder`: Encode target labels with values between 0 and `n_classes-1`.
- `OrdinalEncoder`: Encode categorical features as an integer array.
- `PowerTransformer`: Apply a power transformation to make data more Gaussian-like.

25. `sklearn.svm`:

- **Classes**: Support Vector Machines for classification and regression.
 - `SVC`: Support Vector Classification.
 - `SVR`: Support Vector Regression.

26. `sklearn.tree`:

- **Classes**: Decision tree-based models.
 - `ExtraTreeClassifier`: Extremely Randomized Trees classifier.
 - `ExtraTreeRegressor`: Extremely Randomized Trees regressor.

27. `sklearn.utils`:

- **Functions**: Utility functions for scikit-learn.
 - `shuffle`: Randomly shuffle data samples and labels.

28. `sklearn.calibration`:

- **Classes**: Probability calibration of classifiers.
 - `CalibratedClassifierCV`: Probabilistic classifier with probability calibration.

29. `sklearn.compose`:

- **Classes**: Combine multiple transformers or estimators into a single transformer or estimator.
 - `ColumnTransformer`: Applies different transformers to different columns of an array or dataframe.
 - `TransformedTargetRegressor`: Apply a transformer to the target variable before fitting a regressor.

30. `sklearn.covariance`:

- **Classes**: Covariance estimation algorithms.
 - `EmpiricalCovariance`: Maximum likelihood covariance estimator.
 - `GraphicalLasso`: Sparse inverse covariance estimation using Graphical Lasso.

31. `sklearn.discriminant_analysis`:

- **Classes**: Linear and Quadratic Discriminant Analysis.
 - `LinearDiscriminantAnalysis`: Linear Discriminant Analysis.
 - `QuadraticDiscriminantAnalysis`: Quadratic Discriminant Analysis.

32. `sklearn.ensemble`:

- **Classes**: Ensemble methods for classification and regression.
 - `VotingClassifier`: Combine multiple classifiers by majority voting.
 - `VotingRegressor`: Combine multiple regressors by averaging predictions.

33. `sklearn.exceptions`:

- **Exceptions**: Custom exceptions raised by scikit-learn.
 - `NotFittedError`: Raised when an unfitted estimator is used.

34. `sklearn.isotonic`:

- **Classes**: Isotonic regression.

- `IsotonicRegression`: Isotonic regression model.

35. `sklearn.kernel_approximation`:

- **Classes:** Kernel approximation.
 - `RBFSampler`: Approximate feature map of an RBF kernel by Monte Carlo approximation.

36. `sklearn.kernel_ridge`:

- **Classes:** Kernel ridge regression.
 - `KernelRidge`: Kernel ridge regression.

37. `sklearn.manifold`:

- **Classes:** Manifold learning and t-SNE.
 - `TSNE`: t-distributed Stochastic Neighbor Embedding.
 - `LocallyLinearEmbedding`: Locally Linear Embedding.

38. `sklearn.mixture`:

- **Classes:** Gaussian Mixture Models (GMM) for clustering and density estimation.
 - `BayesianGaussianMixture`: Bayesian Gaussian Mixture Model.

39. `sklearn.model_selection`:

- **Classes:** Model selection.
 - `TimeSeriesSplit`: Time Series cross-validator.

40. `sklearn.multiclass`:

- **Classes:** Strategies for multiclass classification.
 - `OneVsRestClassifier`: One-vs-the-rest multiclass strategy.
 - `OneVsOneClassifier`: One-vs-one multiclass strategy.

41. `sklearn.multioutput`:

- **Classes:** Multi-output regression and classification.
 - `RegressorChain`: Chains regressors in sequence to handle multi-output problems.

42. `sklearn.naive_bayes`:

- **Classes:** Naive Bayes classifiers.
 - `BernoulliNB`: Bernoulli Naive Bayes.

43. `sklearn.neighbors`:

- **Classes:** Nearest neighbors-based methods.
 - `RadiusNeighborsRegressor`: Radius-based neighbors regressor.

44. `sklearn.neural_network`:

- **Classes:** Feedforward neural networks.
 - `BernoulliRBM`: Bernoulli Restricted Boltzmann Machine.

45. `sklearn.pipeline`:

- **Classes:** Pipeline of transforms with a final estimator.
 - `make_pipeline`: Construct a pipeline from a list of estimators.

46. `sklearn.preprocessing`:

- **Classes:** Data normalization and encoding.
 - `QuantileTransformer`: Transform features to follow a uniform or normal distribution.
 - `FunctionTransformer`: Apply a user-defined function to each input.

47. `sklearn.random_projection`:

- **Classes:** Random projection for dimensionality reduction.
 - `SparseRandomProjection`: Random projection using sparse random matrices.

48. `sklearn.semi_supervised`:

- **Classes:** Semi-supervised learning.
 - `LabelPropagation`: Label propagation and label spreading.

49. `sklearn.svm`:

- **Classes:** Support Vector Machines for classification and regression.
 - `NuSVC`: Nu-Support Vector Classification.
 - `NuSVR`: Nu-Support Vector Regression.

50. `sklearn.utils`:

- **Functions:** Utility functions for scikit-learn.
 - `check_X_y`: Check that X and y have correct shape and type.
 - `check_array`: Check that array-like input is a 2D array.

51. `sklearn.calibration`:

- **Classes:** Probability calibration of classifiers.
 - `CalibrationDisplay`: Visualize probability calibration.

52. `sklearn.compose`:

- **Classes:** Combine multiple transformers or estimators into a single transformer or estimator.
 - `make_column_transformer`: Construct a `ColumnTransformer` from a list of transformers.
 - `make_column_selector`: Construct a column selector function based on column names or data types.

53. `sklearn.covariance`:

- **Classes:** Covariance estimation algorithms.
 - `LedoitWolf`: Covariance estimator using the Ledoit-Wolf shrinkage method.
 - `OAS`: Covariance estimator using the Oracle Approximating Shrinkage method.

54. `sklearn.discriminant_analysis`:

- **Classes:** Linear and Quadratic Discriminant Analysis.
 - `QuadraticDiscriminantAnalysis`: Quadratic Discriminant Analysis with shrinkage.

55. `sklearn.ensemble`:

- **Classes:** Ensemble methods for classification and regression.
 - `HistGradientBoostingClassifier`: Histogram-based Gradient Boosting Classification Tree.
 - `HistGradientBoostingRegressor`: Histogram-based Gradient Boosting Regression Tree.
 - `StackingClassifier`: Stacked generalization for classification.
 - `StackingRegressor`: Stacked generalization for regression.

56. `sklearn.exceptions`:

- **Exceptions:** Custom exceptions raised by scikit-learn.

- `ConvergenceWarning`: Warning raised when an iterative fitting procedure does not converge.

57. `sklearn.feature_extraction`:

- **Classes**: Text feature extraction.
 - `HashingVectorizer`: Convert a collection of text documents to a matrix of token occurrences.

58. `sklearn.inspection`:

- **Functions**: Model inspection and interpretation.
 - `permutation_importance`: Compute feature importance scores via permutation.

59. `sklearn.isotonic`:

- **Classes**: Isotonic regression.
 - `IsotonicCalibration`: Isotonic regression for probability calibration.

60. `sklearn.kernel_approximation`:

- **Classes**: Kernel approximation.
 - `Nystroem`: Approximate feature map of a kernel using a subset of the training data.

61. `sklearn.manifold`:

- **Classes**: Manifold learning and t-SNE.
 - `MDS`: Multi-Dimensional Scaling.
 - `SpectralEmbedding`: Spectral embedding for non-linear dimensionality reduction.

62. `sklearn.mixture`:

- **Classes**: Gaussian Mixture Models (GMM) for clustering and density estimation.
 - `GaussianMixture`: Gaussian Mixture Model with Expectation-Maximization (EM).

63. `sklearn.model_selection`:

- **Classes**: Model selection.
 - `TimeSeriesSplit`: Time Series cross-validator with fixed training and test sets.

64. `sklearn.multiclass`:

- **Classes**: Strategies for multiclass classification.
 - `OutputCodeClassifier`: Multi-class classification using binary classifiers.

65. `sklearn.multioutput`:

- **Classes**: Multi-output regression and classification.
 - `MultiOutputRegressor`: Multi-output regression with different regressors.

66. `sklearn.naive_bayes`:

- **Classes**: Naive Bayes classifiers.
 - `CategoricalNB`: Naive Bayes classifier for categorical features.

67. `sklearn.neighbors`:

- **Classes**: Nearest neighbors-based methods.
 - `NearestCentroid`: Nearest centroid classifier.

68. `sklearn.neural_network`:

- **Classes**: Feedforward neural networks.
 - `MLPRegressor`: Multi-layer Perceptron regressor with different activation functions.

69. `sklearn.pipeline`:

- **Classes**: Pipeline of transforms with a final estimator.

- `FeatureUnion`: Combine several transformer objects into a single transformer.

70. `sklearn.preprocessing`:

- **Classes**: Data normalization and encoding.
 - `LabelPropagation`: Semi-supervised learning with label propagation.

71. `sklearn.random_projection`:

- **Classes**: Random projection for dimensionality reduction.
 - `GaussianRandomProjection`: Random projection using Gaussian random matrices.

72. `sklearn.semi_supervised`:

- **Classes**: Semi-supervised learning.
 - `LabelSpreading`: Label propagation and label spreading.

73. `sklearn.svm`:

- **Classes**: Support Vector Machines for classification and regression.
 - `LinearSVC`: Linear Support Vector Classification.
 - `LinearSVR`: Linear Support Vector Regression.

74. `sklearn.tree`:

- **Classes**: Decision tree-based models.
 - `DecisionTreeClassifier`: Decision Tree Classifier.
 - `DecisionTreeRegressor`: Decision Tree Regressor.

75. `sklearn.utils`:

- **Functions**: Utility functions for scikit-learn.
 - `check_random_state`: Generate pseudo-random numbers.

76. `sklearn.decomposition`:

- **Classes**: Dimensionality reduction and matrix factorization.
 - `DictionaryLearning`: Dictionary Learning.
 - `FactorAnalysis`: Factor Analysis.
 - `IncrementalPCA`: Incremental Principal Component Analysis.
 - `MiniBatchDictionaryLearning`: Mini-Batch Dictionary Learning.

77. `sklearn.ensemble`:

- **Classes**: Ensemble methods for classification and regression.
 - `AdaBoostClassifier`: AdaBoost Classifier.
 - `AdaBoostRegressor`: AdaBoost Regressor.
 - `GradientBoostingClassifier`: Gradient Boosting Classifier.
 - `GradientBoostingRegressor`: Gradient Boosting Regressor.
 - `RandomForestClassifier`: Random Forest Classifier.
 - `RandomForestRegressor`: Random Forest Regressor.

78. `sklearn.exceptions`:

- **Exceptions**: Custom exceptions raised by scikit-learn.
 - `SkipTest`: Exception indicating that a test should be skipped.

79. `sklearn.feature_extraction`:

- **Classes**: Feature extraction from text and images.
 - `CountVectorizer`: Convert a collection of text documents to a matrix of token counts.

- `TfidfVectorizer`: Convert a collection of raw documents to a matrix of TF-IDF features.
- `ImageFeatureExtractor`: Extract features from images.

80. `sklearn.inspection`:

- **Functions: Model inspection and interpretation.**
 - `plot_partial_dependence`: Plot partial dependence of features.

81. `sklearn.isotonic`:

- **Classes: Isotonic regression.**
 - `IsotonicRegressionCV`: Cross-validated isotonic regression.

82. `sklearn.kernel_approximation`:

- **Classes: Kernel approximation.**
 - `RBFSampler`: Approximate feature map of an RBF kernel by random Fourier features.

83. `sklearn.manifold`:

- **Classes: Manifold learning and t-SNE.**
 - `TSNE`: t-distributed Stochastic Neighbor Embedding.

84. `sklearn.mixture`:

- **Classes: Gaussian Mixture Models (GMM) for clustering and density estimation.**
 - `BayesianGaussianMixture`: Bayesian Gaussian Mixture Model with variational inference.

85. `sklearn.model_selection`:

- **Classes: Model selection and evaluation.**
 - `GroupKFold`: K-fold iterator variant with non-overlapping groups.
 - `GridSearchCV`: Grid Search with Cross-Validation.
 - `RandomizedSearchCV`: Randomized Search with Cross-Validation.

86. `sklearn.multioutput`:

- **Classes: Multi-output regression and classification.**
 - `MultiOutputClassifier`: Multi-output classification with different classifiers.

87. `sklearn.naive_bayes`:

- **Classes: Naive Bayes classifiers.**
 - `GaussianProcessClassifier`: Gaussian process classification.

88. `sklearn.neighbors`:

- **Classes: Nearest neighbors-based methods.**
 - `NearestNeighbors`: Unsupervised Nearest Neighbors.

89. `sklearn.neural_network`:

- **Classes: Feedforward neural networks.**
 - `MLPClassifier`: Multi-layer Perceptron classifier with advanced features.

90. `sklearn.pipeline`:

- **Classes: Pipeline of transforms with a final estimator.**
 - `make_union`: Construct a `FeatureUnion` from a list of transformers.

91. `sklearn.preprocessing`:

- **Classes: Data preprocessing and scaling.**
 - `MaxAbsScaler`: Scale features to $[-1,1]$ by dividing by the maximum absolute value.

- `RobustScaler`: Scale features using statistics that are robust to outliers.

92. `sklearn.random_projection`:

- **Classes**: Random projection for dimensionality reduction.
 - `SparseRandomProjection`: Random projection using sparse random matrices.

93. `sklearn.semi_supervised`:

- **Classes**: Semi-supervised learning.
 - `LabelPropagation`: Label Propagation.

94. `sklearn.svm`:

- **Classes**: Support Vector Machines for classification and regression.
 - `SVC`: C-Support Vector Classification.
 - `SVR`: Epsilon-Support Vector Regression.

95. `sklearn.tree`:

- **Classes**: Decision tree-based models.
 - `ExtraTreeClassifier`: Extremely Randomized Trees Classifier.

96. `sklearn.calibration`:

- **Classes**: Probability calibration of classifiers.
 - `CalibratedClassifierCV`: Probability calibration with cross-validation.

97. `sklearn.compose`:

- **Classes**: Combine multiple transformers or estimators into a single transformer or estimator.
 - `ColumnTransformer`: Applies transformers to columns of an array or pandas DataFrame.
 - `TransformedTargetRegressor`: Applies a transformer to the target variable during fitting.

98. `sklearn.covariance`:

- **Classes**: Covariance estimation algorithms.
 - `GraphicalLasso`: Covariance estimator using Graphical Lasso.

99. `sklearn.ensemble`:

- **Classes**: Ensemble methods for classification and regression.
 - `VotingClassifier`: Combine multiple classifiers by majority voting.

100. `sklearn.exceptions`:

- **Exceptions**: Custom exceptions raised by scikit-learn.
 - `NotFittedError`: Exception indicating that a method was called on an estimator that was not fitted.