



[← Go Back to Practical Data Science](#)

[☰ Course Content](#)

## New Package Introduction - Decision Trees

Now, let's go through some of the common functions used in this LVC from the Scikit-learn library

### Decision Tree

A decision tree is a non-parametric supervised learning approach that can be used for classification as well as regression problems. It has a tree structure that is hierarchical and consists of a root node, branches, internal nodes, and leaf nodes.

### Decision Tree Regression

```
sklearn.tree.DecisionTreeRegressor(*, criterion='squared_error', splitter='best', max_depth=None, min_samples_split=2, min_samples_leaf=1, min_weight_fraction=0.0)
```

#### Example

```
from sklearn.tree import DecisionTreeRegressor
dtr = DecisionTreeRegressor()
dtr.fit(x_train, y_train)
```

You can refer to the decision tree regressor sklearn documentation for a better understanding of the parameters and attributes [here](#).

Similarly, we can implement the decision tree classifier through the sklearn library.

### Decision Tree Classification

```
sklearn.tree.DecisionTreeClassifier(*, criterion='gini', splitter='best', max_depth=None, min_samples_split=2, min_samples_leaf=1, min_weight_fraction=0.0)
```

#### Example

```
from sklearn.tree import DecisionTreeClassifier
dtr = DecisionTreeClassifier()
dtr.fit(x_train, y_train)
```

You can refer to the decision tree regressor sklearn documentation for a better understanding of the parameters and attributes [here](#).

### Label Encoding

Sklearn provides a very efficient tool for encoding the levels of categorical features into numeric values. LabelEncoder encodes labels with a value between 0 and n\_classes-1, where n is the number of distinct labels. If a label repeats, it assigns the same value as assigned earlier.

#### Example

```
from sklearn.preprocessing import LabelEncoder
le = LabelEncoder()
le.fit_transform(data['column']) # column should be any categorical variable
```

You can refer to the label encoder sklearn documentation for a better understanding of the parameters and attributes [here](#).

Happy Learning!

Proprietary content.©Great Learning. All Rights Reserved. Unauthorized use or distribution prohibited.