**Split The Dependent And Independent Features Into Train Set And Test Set**

**Project Name: statistical machine learning Team ID : PNT2022TMID04163**

**approaches to liver disease prediction**

**In this tutorial, you’ll learn:**

* Why you need to **split your dataset** in supervised machine learning
* Which **subsets** of the dataset you need for an unbiased evaluation of your model
* How to use **train\_test\_split()** to split your data
* How to combine train\_test\_split() with **prediction methods**

## The Importance of Data Splitting

[Supervised machine learning](https://en.wikipedia.org/wiki/Supervised_learning) is about creating models that precisely map the given inputs (independent variables, or **predictors**) to the given outputs (dependent variables, or **responses**).

How you measure the precision of your

## Prerequisites for Using train\_test\_split()

Now that you understand the need to split a dataset in order to perform unbiased model evaluation and identify underfitting or overfitting, you’re ready to learn how to split your own datasets.

You’ll use version 0.23.1 of **scikit-learn**, or **sklearn**. It has many packages for data science and machine learning, but for this tutorial you’ll focus on the **model\_selection** package, specifically on the function **train\_test\_split()**.

You can [install sklearn](https://scikit-learn.org/stable/install.html) with [pip install](https://realpython.com/what-is-pip/):

$ python -m pip install -U "scikit-learn==0.23.1"

If you use [Anaconda](https://www.anaconda.com/), then you probably already have it installed. However, if you want to use a [fresh environment](https://docs.conda.io/projects/conda/en/latest/user-guide/tasks/manage-environments.html), ensure that you have the specified version, or use [Miniconda](https://docs.conda.io/en/latest/miniconda.html), then you can install sklearn from Anaconda Cloud with conda install:

$ conda install -c anaconda scikit-learn=0.23

You’ll also need [NumPy](https://numpy.org/), but you don’t have to install it separately. You should get it along with sklearn if you don’t already have it installed. If you want to refresh your NumPy knowledge, then take a look at the [official documentation](https://numpy.org/doc/stable/user/tutorials_index.html) or check out [Look Ma, No For-Loops: Array Programming With NumPy](https://realpython.com/numpy-array-programming/).

## Application of train\_test\_split()

You need to [import](https://realpython.com/python-import/) train\_test\_split() and NumPy before you can use them, so you can start with the [import](https://realpython.com/courses/python-imports-101/) statements:

>>> import numpy as np

>>> from sklearn.model\_selection import train\_test\_split

With train\_test\_split(), you need to provide the sequences that you want to split as well as any optional arguments. It returns a [list](https://realpython.com/python-lists-tuples/) of [NumPy arrays](https://numpy.org/doc/stable/reference/generated/numpy.ndarray.html), other sequences, or [SciPy sparse matrices](https://docs.scipy.org/doc/scipy/reference/generated/scipy.sparse.csr_matrix.html) if appropriate:

sklearn.model\_selection.train\_test\_split(\*arrays, \*\*options) -> list

**arrays** is the sequence of [lists](https://realpython.com/python-lists-tuples/), [NumPy arrays](https://realpython.com/numpy-array-programming/), [pandas DataFrames](https://realpython.com/pandas-dataframe/), or similar array-like objects that hold the data you want to split. All these objects together make up the dataset and must be of the same length.

**Program:**

>>> x = np.arange(1, 25).reshape(12, 2)

>>> y = np.array([0, 1, 1, 0, 1, 0, 0, 1, 1, 0, 1, 0])

>>> x

array([[ 1, 2],

[ 3, 4],

[ 5, 6],

[ 7, 8],

[ 9, 10],

[11, 12],

[13, 14],

[15, 16],

[17, 18],

[19, 20],

[21, 22],

[23, 24]])

>>> y

array([0, 1, 1, 0, 1, 0, 0, 1, 1, 0, 1, 0])