DATA PRE-PROCESSING

SPLIT THE DEPENDENT AND INDEPENDENT FEATURES INTO TRAIN SET AND TEST SET

DATE	**NOVEMBER 2022**
TEAM ID	**PNT2022TMID43782**
PROJECT NAME	**STATISTICAL MACHINE LEARNING APPROACHES TO LIVER DISEASE PREDICTION**

Split the Dependent and Independent Features into Train set and Test set:

The train-test split procedure is used to estimate the performance of machine learning algorithms when they are used to make predictions on data not used to train the model.

It is a fast and easy procedure to perform, the results of which allow you to compare the performance of machine learning algorithms for your predictive modeling problem. Although simple to use and interpret, there are times when the procedure should not be used, such as when you have a small dataset and situations where additional configuration is required, such as when it is used for classification and the dataset is not balanced.

In this tutorial, you will discover how to evaluate machine learning models using the train-test split.

After completing this tutorial, you will know:

- The train-test split procedure is appropriate when you have a very large dataset, a costly model to train, or require a good estimate of model performance quickly.
- How to use the scikit-learn machine learning library to perform the train-test split procedure.
- How to evaluate machine learning algorithms for classification and regression using the train-test split.

- When you are working on a model and you want to train it, you obviously have a dataset. But after training, we have to test the model on some test dataset. For this, you will a dataset which is different from the training set you used earlier. But it might not always be possible to have so much data during the development phase. In such cases, the solution is to split the dataset into two sets, one for training and the other for testing.
- But the question is, how do you split the data? You can't possibly manually split the dataset into two sets. And you also have to make sure you split the data in a random manner. To help us with this task, the Scikit library provides a tool, called the Model Selection library. There is a class in the library which is,'train_test_split.' Using this we can easily split the dataset into the training and the testing datasets in various proportions.
- The train-test split is a technique for evaluating the performance of a machine learning algorithm.
- Train Dataset: Used to fit the machine learning model.
- Test Dataset: Used to evaluate the fit machine learning model.
- In general you can allocate 80% of the dataset to training set and the remaining 20% to test set.We will create 4 sets— x_train (training

part of the matrix of features), x_test (test part of the matrix of features), y_train (training part of the dependent variables associated with the X train sets, and therefore also the same indices), y_test (test part of the dependent variables associated with the X test sets, and therefore also the same indices.

- There are a few other parameters that we need to understand before we use the class:
- test_size — this parameter decides the size of the data that has to be split as the test dataset. This is given as a fraction. For example, if you pass 0.5 as the value, the dataset will be split 50% as the test dataset
- train_size — you have to specify this parameter only if you're not specifying the test_size. This is the same as test_size, but instead you tell the class what percent of the dataset you want to split as the training set.
- random_state — here you pass an integer, which will act as the seed for the random number generator during the split. Or, you can also pass an instance of the Random_state class, which will become the number generator. If you don't pass anything, the Random_state instance used by np.random will be used instead.
- Now split our dataset into train set and test using train_test_split class from scikit learn library.

Check the shape of both xtrain and xtest.

```
xtrain.shape
```

```
(466, 10)
```

```
xtest.shape
```

```
(117, 10)
```

```
...
# split into train test sets
X_train, X_test, y_train, y_test = train_test_split(X, y, ...)
```