Team ID: PNT2022TMID11450

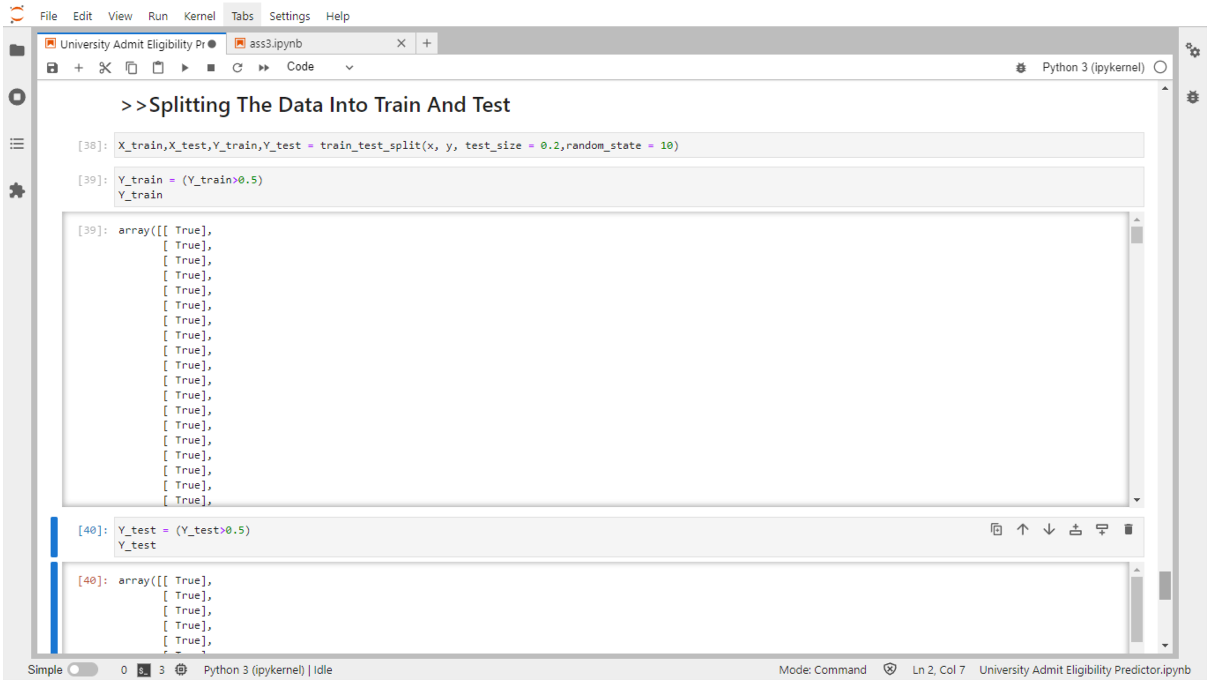
DATA COLLECTION

Splitting The Data Into Train And Test

- **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, we can allocate **80%** of the dataset to the **training set** and the remaining **20%** to the **test set** and create 4 sets are
    - **X_train**
    - **X_test**
    - **Y_train**
    - **Y_test**
- There are a few other parameters that need to understand before using this 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 and remaining a train dataset.
    - **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.

Step-1:

- Firstly, we need to split the data into test and train set.
- In Scikit library, **train_test_split () function** is used to split data by train set 80% and test set 20% present in the dataset.
- Then, we are assigning variables such as **X_train, X_test, Y_train, Y_test** by using parameter like test_size and random_state.


File Edit View Run Kernel Tabs Settings Help
University Admit Eligibility Pr
ass3.ipynb
Code
Python 3 (ipykernel)
Simple 0 3 Python 3 (ipykernel) | Idle Mode: Command Ln 2, Col 7 University Admit Eligibility Predictor.ipynb

>>Splitting The Data Into Train And Test

```
[38]: X_train,X_test,Y_train,Y_test = train_test_split(x, y, test_size = 0.2,random_state = 10)
```

```
[39]: Y_train = (Y_train>0.5)
      Y_train
```

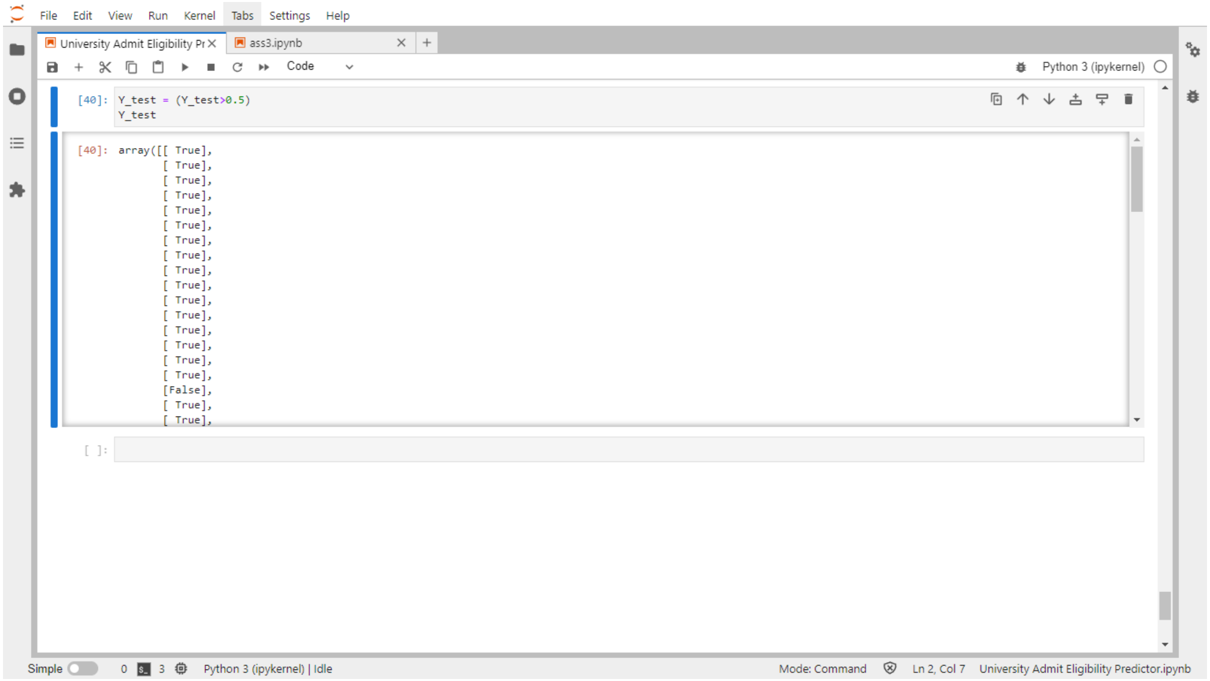
```
[39]: array([[ True],
             [ True],
             [ True],
             [ True],
             [ True],
             [ True],
             [ True],
             [ True],
             [ True],
             [ True],
             [ True],
             [ True],
             [ True],
             [ True],
             [ True],
             [ True],
             [ True],
             [ True],
             [ True],
```

```
[40]: Y_test = (Y_test>0.5)
      Y_test
```

```
[40]: array([[ True],
             [ True],
             [ True],
             [ True],
             [ True],
```

Step-2:

- Using **Y_train** variable will produce **80%** of train set and **Y_test** variable will produce **20**% of test set for admitting the college students.


File Edit View Run Kernel Tabs Settings Help
University Admit Eligibility Pr
ass3.ipynb
Code
Python 3 (ipykernel)
Simple 0 3 Python 3 (ipykernel) | Idle Mode: Command Ln 2, Col 7 University Admit Eligibility Predictor.ipynb

```
[40]: Y_test = (Y_test>0.5)
      Y_test
```

```
[40]: array([[ True],
             [ True],
             [ True],
             [ True],
             [ True],
             [ True],
             [ True],
             [ True],
             [ True],
             [ True],
             [ True],
             [ True],
             [ True],
             [ True],
             [ True],
             [ True],
             [False],
             [ True],
             [ True],
```

```
[ ]:
```