DATA PRE-PROCESSING

IMPORTING THE LIBRARIES

DATE	21 NOVEMBER 2022
TEAM ID	PNT2022TMID36295
TEAM MEMBERS	M.MANU VARDHAN, N.BHARGAV, N.BALA CHAITHANYA, K.CHENCHU KISHORE
PROJECT NAME	STATISTICAL MACHINE LEARNING APPROACHES TO LIVER DISEASE PREDICTION

Importing The Libraries

The first step is usually importing the libraries that will be needed in the program.

The required libraries to be imported to Python script are:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import pickle
```

Numpy:

It is an open-source numerical Python library. It contains a multi- dimensional array and matrix data structures. It can be used to perform mathematical operations on arrays such as trigonometric, statistical, and algebraic routines.

Example:

```
import numpy as np
a = np.array([0, 1, 2, 3])       # Create a rank 1 array
print(a)                         #print array a
print(type(a))                   #type of array a
print(a.ndim)                    #dimension of array a
print(a.shape)                   #shape(row,column) of array a
print(len(a))                    #length of array a
```

```
[0 1 2 3]
<class 'numpy.ndarray'>
1
(4,)
4
```

Pandas:

It is a fast, powerful, flexible and easy to use open source data analysis and manipulation tool, built on top of the Python programming language.

Example:

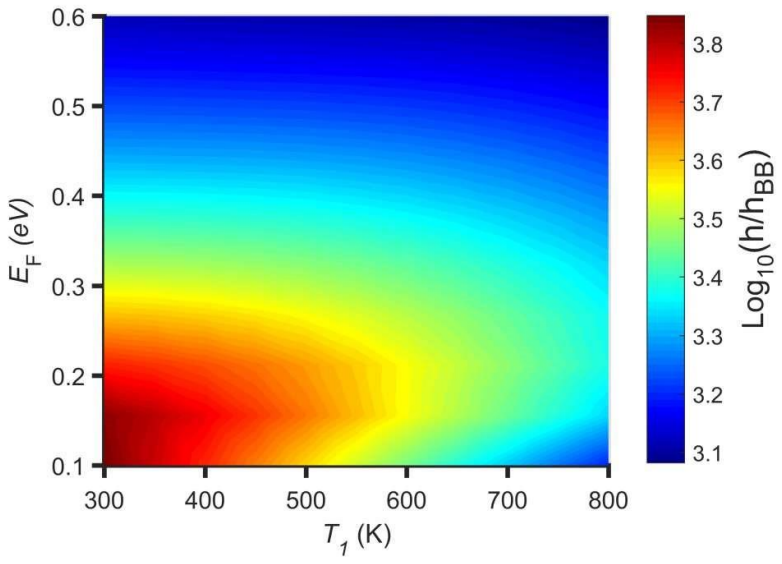
```
label = ['a', 'b', 'c']
my_data = [10, 20, 30]
pd.Series(data = my_data, index = label)
```

```
a    10
b    20
c    30
dtype: int64
```

Matplotlib:

Visualisation with python. It is a comprehensive library for creating static, animated, and interactive visualizations in Python.

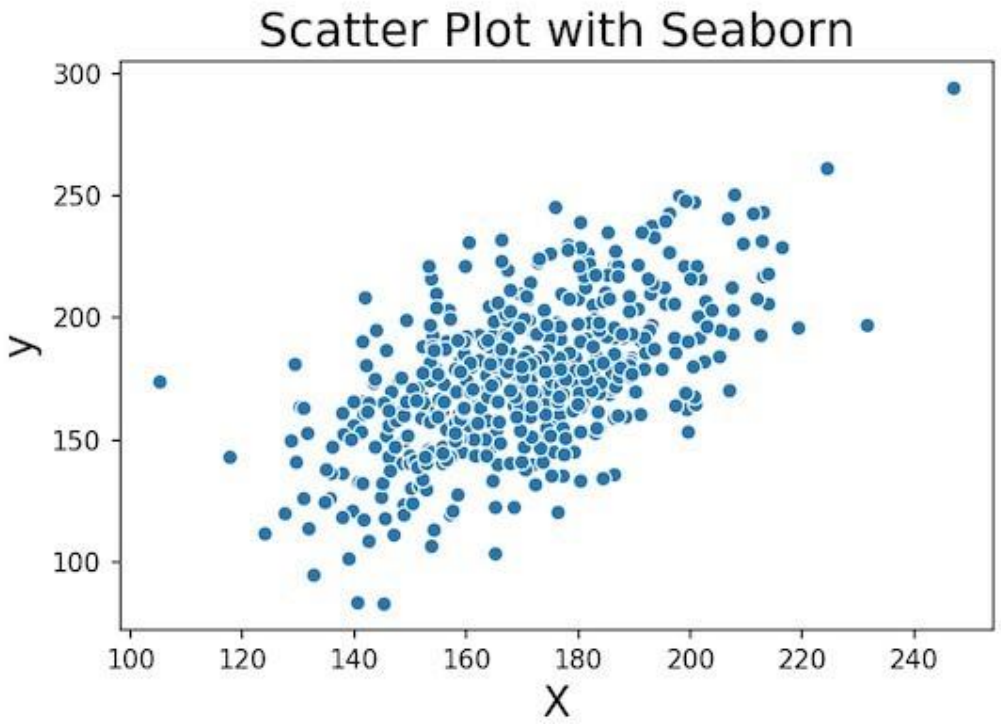
Example:



Seaborn:

Seaborn is a library for making statistical graphics in Python. Seaborn helps you explore and understand your data. Its plotting functions operate on dataFrames and arrays containing whole datasets and internally perform the necessary semantic mapping and statistical aggregation to produce informative plots.

Example:



Pickle:

The pickle module implements serialization protocol, which provides an ability to save and later load Python objects using special binary format.