MACHINE LEARNING FOR COMMUNICATIONS LAB REPORT

MEIC501P

TASK 2

TASK A) PERFORMANCE ANALYSIS OF LINEAR REGRESSION

TASK B) ANALYSIS OF MACHINE LEARNING TOOL – ORANGE SOFTWARE

Name : Durga Sitalakshmi S

Registration Number: 24MEC0024

Phone Number: +91 9731850288

Email ID: durgasitalakshmi.s2024@vitstudent.ac.in

TASK 2A) PERFORMANCE ANALYSIS OF LINEAR REGRESSION

```
#TASK 2(A) : PERFORMANCE ANALYSIS OF LINEAR REGRESSION WITH EEG SIGNAL
# 24MEC0024
#DURGA SITALAKSHMI S
#PHONE NUMBER : 9731850288
#MEIC501P

from glob import glob
import numpy as np
import pandas as pd
import mne# library used for extracting eeg
from matplotlib import pyplot as plt
data_set=glob('Downloads/Dataverse/*.edf')
```

```
def read_data(file_path):
  data=mne.io.read_raw_edf(file_path,preload=True)# defining a function to extract data from our EEG signal
  epochs = mne.make_fixed_length_epochs(data,duration=5,overlap=1)
  array=epochs.get_data() #extracting data inform of array
  return max(array[0][0]*1000000) #taking out the individual maxi value and multiplying with 1Lakh for effective plotting
data_array=[read_data(i) for i in data_set]# taking out the individual maxi value and multiplying with 1L
df= pd.DataFrame()                                          # creating a data frame using pandas
df['EEG_epochs']=data_array#appending our data into a frame
df2=pd.read_csv('Sor.csv')#reading our source file using pandas
print(df2)
df['ANX'] = df2['ANX']
```

```
# Saving the DataFrame to CSV
df.to_csv('epo2.csv', index=False)
# Reading raw data from the first .edf file
raw = mne.io.read_raw_edf(data_set[0])

# Creating epochs of fixed length for visualization
epochs1 = mne.make_fixed_length_epochs(raw, duration=5, overlap=1)

# Getting the data as an array
arr = epochs1.get_data()
print(df)
# Plotting the first epoch of the first channel
pd.Series(arr[0][0]).plot(figsize=(10, 5), lw=1, title='Sample EEG Signal')
plt.xlabel('Samples')
plt.ylabel('Amplitude')
plt.grid()
```

	Unnamed: 0	ANX
0	1	0.3571
1	2	0.7142
2	3	1.0713
3	4	1.4284
4	5	1.7855
5	6	2.1426
6	7	2.4997
7	8	2.8568
8	9	3.2139
9	10	3.5710

Fig 1: List of anxieties of given sample

	EEG_epochs	ANX
0	15.598824	0.3571
1	19.574357	0.7142
2	19.880167	1.0713
3	20.491788	1.4284
4	22.632460	1.7855
5	23.123123	2.1426
6	24.008606	2.4997
7	25.843468	2.8568
8	27.525424	3.2139
9	28.442855	3.5710

Fig 2: EEG_epochs and anxiety of given samples.

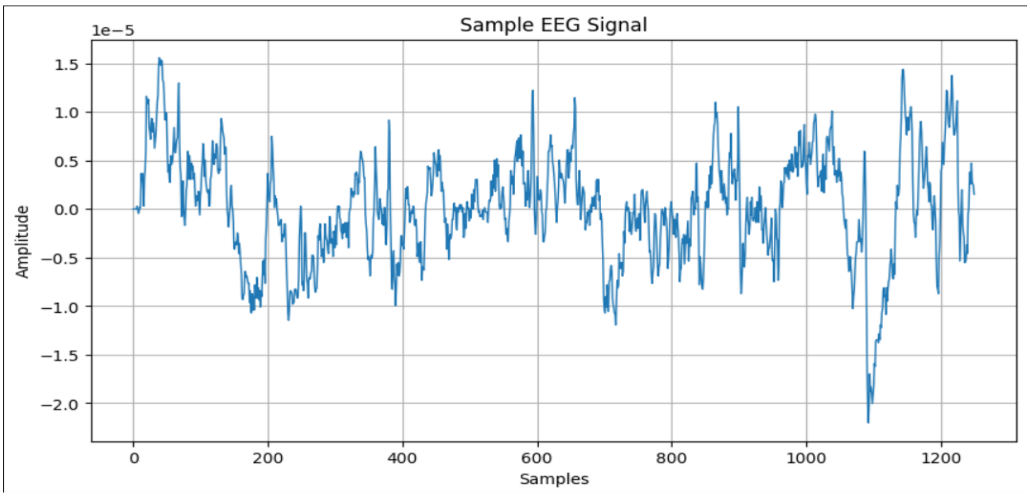


Fig 3: Waveform of sample EEG signal.

```python
#TASK 2(A) : PERFORMANCE ANALYSIS OF EEG SIGNAL W
# 24MEC0024
#DURGA SITALAKSHMI S
#PHONE NUMBER : 9731850288
#MEIC501P
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn import linear_model
import warnings
warnings.filterwarnings("ignore")

# Reading the preprocessed DataFrame
df = pd.read_csv('epo2.csv')

# Creating a linear regression model
reg = linear_model.LinearRegression()

# Fitting the model with ANX as independent variable and EEG_epochs as dependent variable
reg.fit(df[['ANX']], df['EEG_epochs'])

# Predicting EEG epochs for a given ANX value (e.g., ANX = 4.2)
print(reg.predict([[4.2]]))
```

```python
# Plotting the linear regression line
plt.figure(figsize=(10, 6))
plt.scatter(df['ANX'], df['EEG_epochs'], color='blue', label='Data points')
plt.plot(df['ANX'], reg.predict(df[['ANX']]), color='red', linewidth=2, label='Linear Regression')
plt.title('Linear Regression of EEG epochs vs ANX')
plt.xlabel('ANX')
plt.ylabel('EEG Epochs')
plt.legend()
plt.grid(True)
plt.show()
```

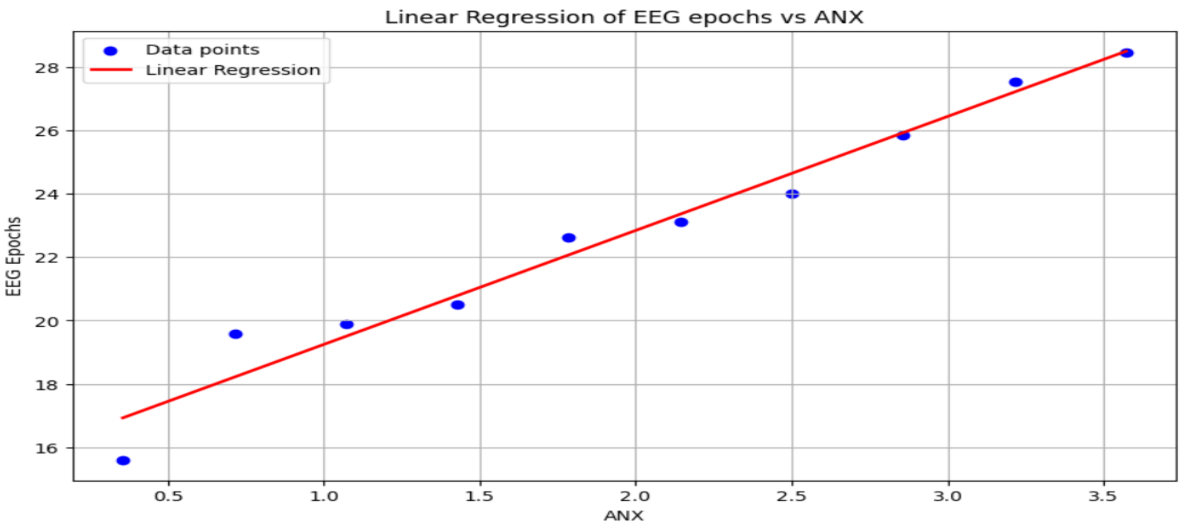


Fig 4:Linear Regression of EEG epochs vs ANX

INFERENCE

1. Objective:

The primary goal of this experiment is to analyze the relationship between EEG signals and anxiety (ANX) levels using a linear regression model. The study focuses on identifying patterns in brain activity that correlate with anxiety, which is known to manifest through changes in EEG signals.

2. Data Acquisition:

EEG data is extracted from `.edf` files using the MNE library, which is commonly used for processing and analyzing electrophysiological data. The data is divided into fixed-length epochs, each representing a segment of the EEG signal.

3. Feature Extraction:

The maximum amplitude of each epoch is computed, and these values are scaled for effective plotting. This transformation aids in visualizing and interpreting the EEG data relative to anxiety levels.

4. Data Integration:

The EEG data is combined with anxiety (ANX) scores, which are obtained from a separate source file (`Sor.csv`). This integration allows for the examination of the relationship between the EEG signals and the anxiety levels.

5. Linear Regression Analysis:

A linear regression model is employed to investigate the relationship between the anxiety scores (independent variable) and the EEG epoch values (dependent variable). The model is trained on the available data to predict EEG epochs based on given anxiety levels.

6. Model Prediction:

The trained model is used to predict EEG epochs for a specific anxiety score (e.g., ANX = 4.2). This prediction showcases the model's capability to estimate EEG signal characteristics based on anxiety levels.

7. Visualization:

The relationship between anxiety and EEG epochs is visualized using a scatter plot with a linear regression line. This visualization highlights the trend in data and the accuracy of the linear model in fitting the observed data points.

CONCLUSION:

The analysis suggests that there is a measurable relationship between EEG signals and anxiety levels. The linear regression model provides a simple yet effective way to quantify this relationship, which can be further refined for more accurate predictions.

TASK B) ANALYSIS OF MACHINE LEARNING TOOL – ORANGE SOFTWARE

Aim:

To thoroughly explore and understand the Orange Data Mining tool, gaining proficiency in its operations. Utilize this low-code/no-code platform to solve machine learning problems with ease and apply it to analyse a specific application.

About Orange:

Link for Orange: orange software

Orange is an open-source toolkit designed for data visualization, machine learning, and data mining. It offers a visual programming interface that facilitates exploratory qualitative data analysis and interactive data visualization.

- Orange is a component-based software package that supports data visualization, machine learning, data mining, and data analysis through a visual programming approach.
- The building blocks of Orange are known as widgets, which encompass a wide range of functionalities, from simple data visualization and subset selection to preprocessing, empirical evaluation of learning algorithms, and predictive modelling.
- The visual programming interface allows users to create workflows by connecting predefined or custom-designed widgets, while advanced users can also leverage Orange as a Python library for data manipulation and widget customization.

About Dataset:

Link for Dataset: Heart Attack Risk Factors Dataset

This dataset is likely used to analyse factors contributing to heart attacks, identify patterns in patient demographics and health metrics, and potentially predict outcomes or treatment responses based on these variables.

1. Gender: Represents the gender of the patient.

2. Age: Indicates the age of the patient in years.

3. Blood Pressure (mmHg): Records the blood pressure of the patient in millimeters of mercury (mmHg).

4. Cholesterol (mg/dL): Represents the cholesterol level of the patient in milligrams per deciliter (mg/dL).

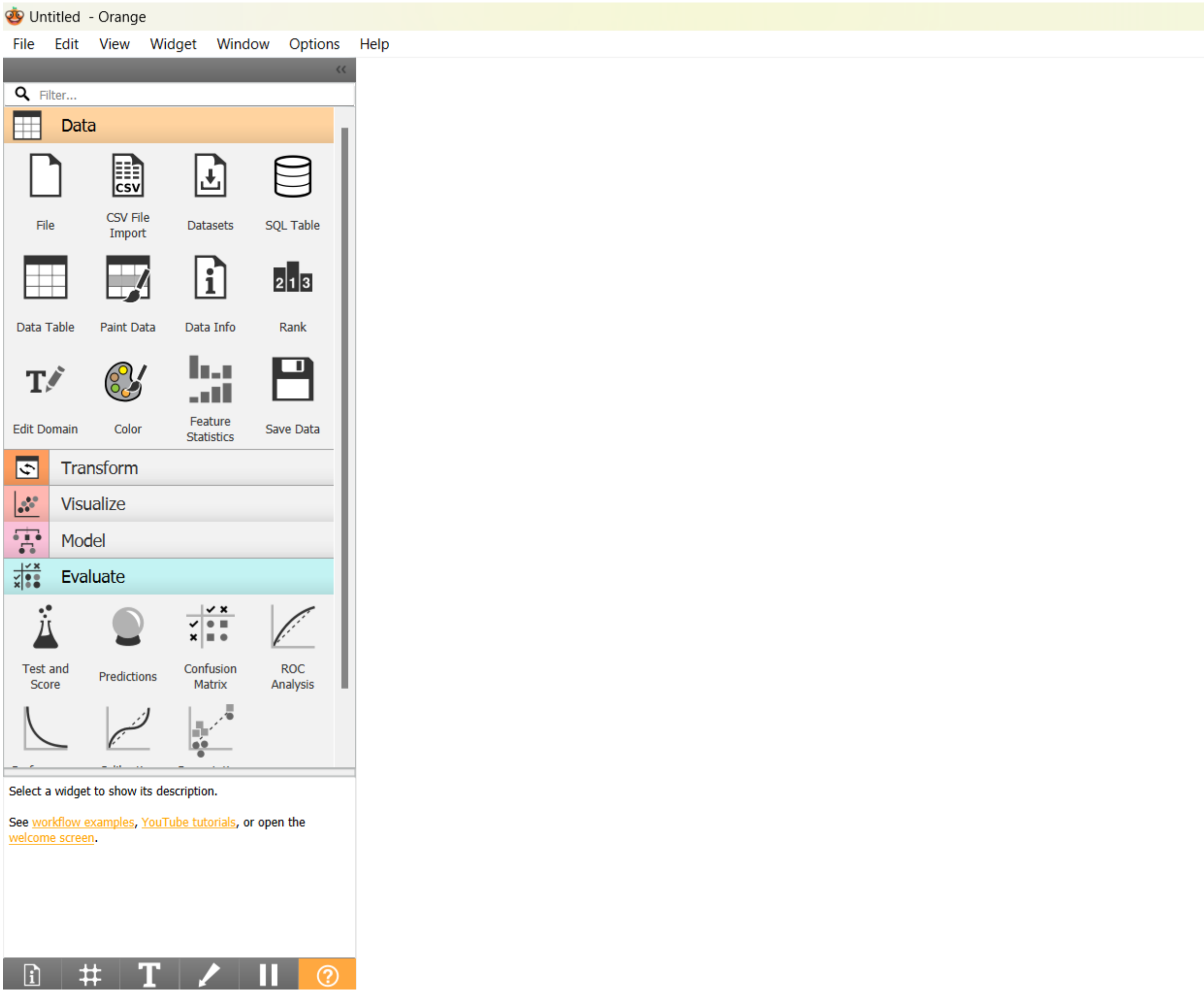
5. Has Diabetes: A binary indicator showing whether the patient has diabetes (typically 1 for yes, 0 for no).

6. Smoking Status: Indicates whether the patient is a smoker.

7. Chest Pain Type: Describes the type of chest pain the patient experiences, which is an important symptom to consider in diagnosing heart conditions.

8. Treatment: Specifies the type of treatment the patient received or is receiving.

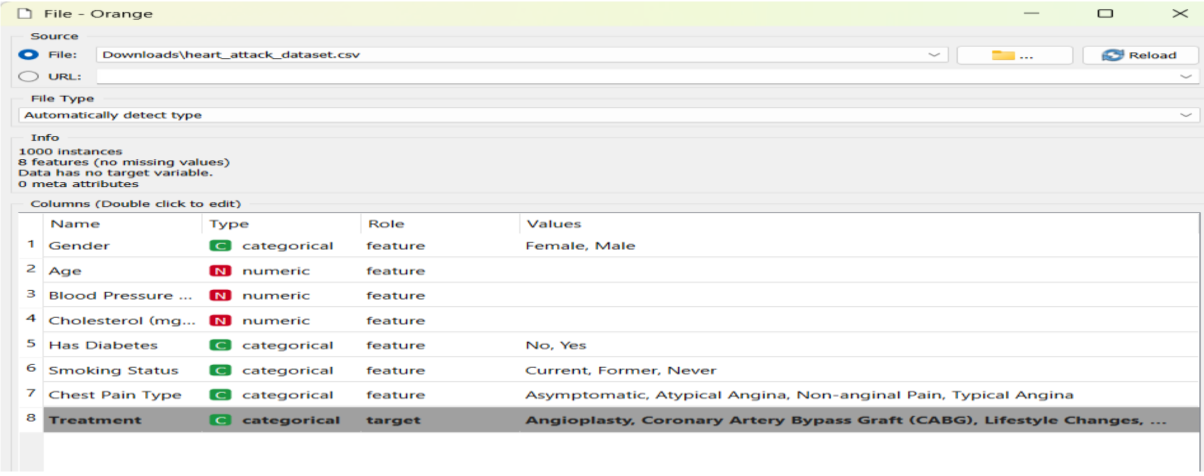
Opening Page :



STEP 1: UPLOAD FILE

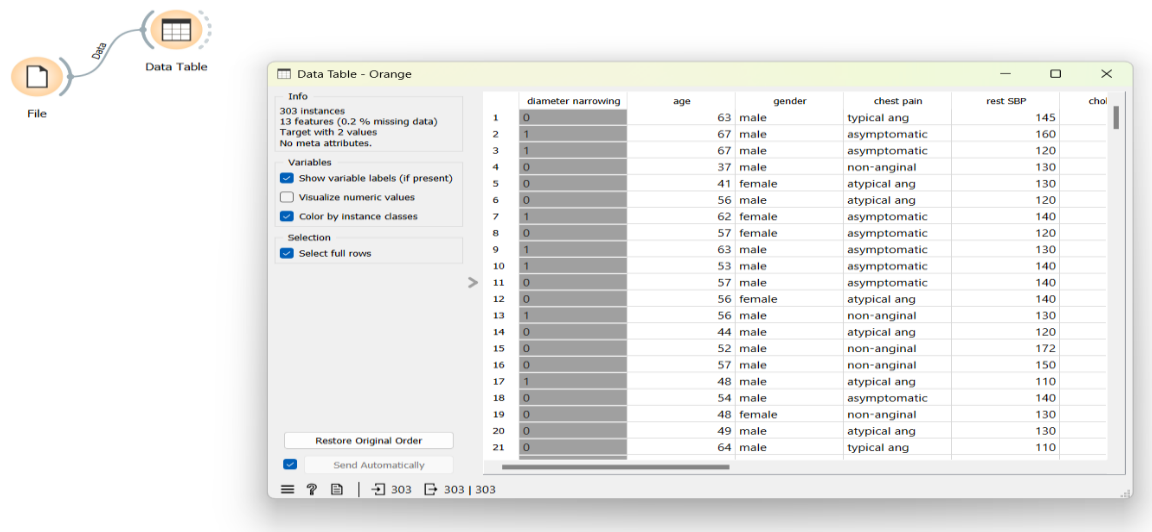
To get the csv file to the orange click on file and bring to the blank page.

After clicking on it, double click on the file to upload the dataset of our choice. Here we are using Heart attack dataset.



Here click on the dataset that we need our model to be trained. In this the treatment is taken as the target. Target is to predict the class based on the other inputs.

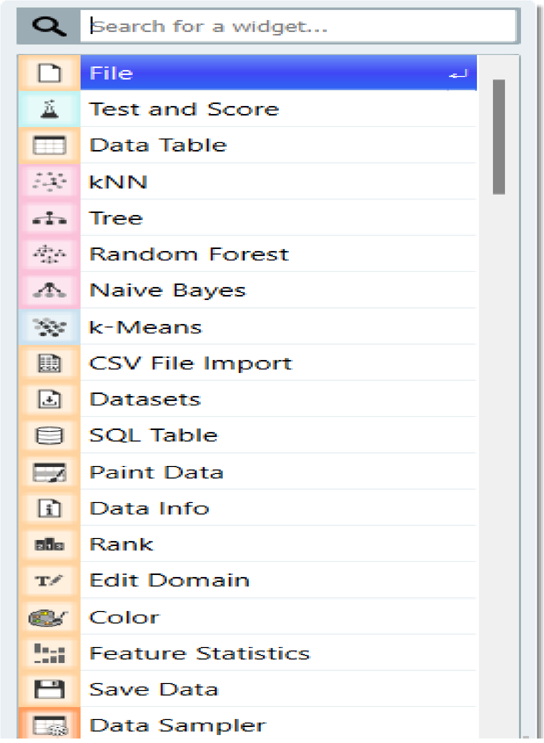
STEP 2: VISUALIZE THE DATA



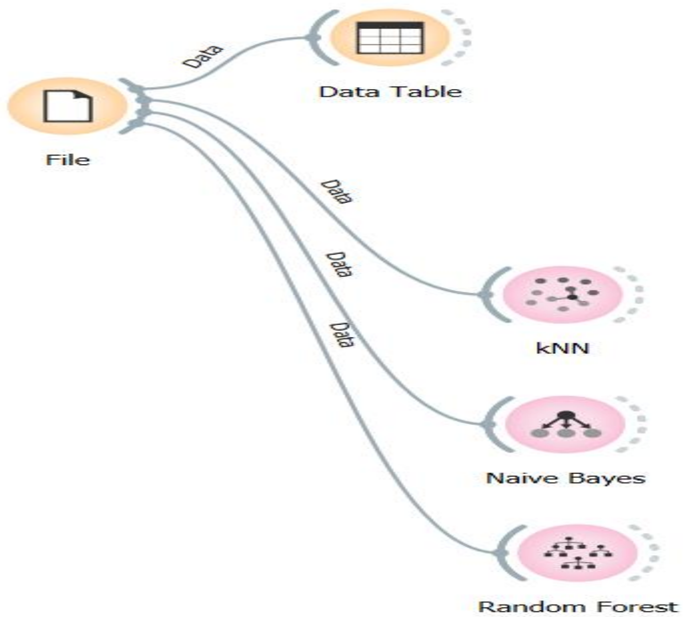
The widgets can be connected to one other. Here the dataset widget is used to visualize the data that was uploaded.

STEP 3 : SELECT MODEL

When we right click on the workspace, a widget list will be opened, we can select the model as per our need and the required dataset.

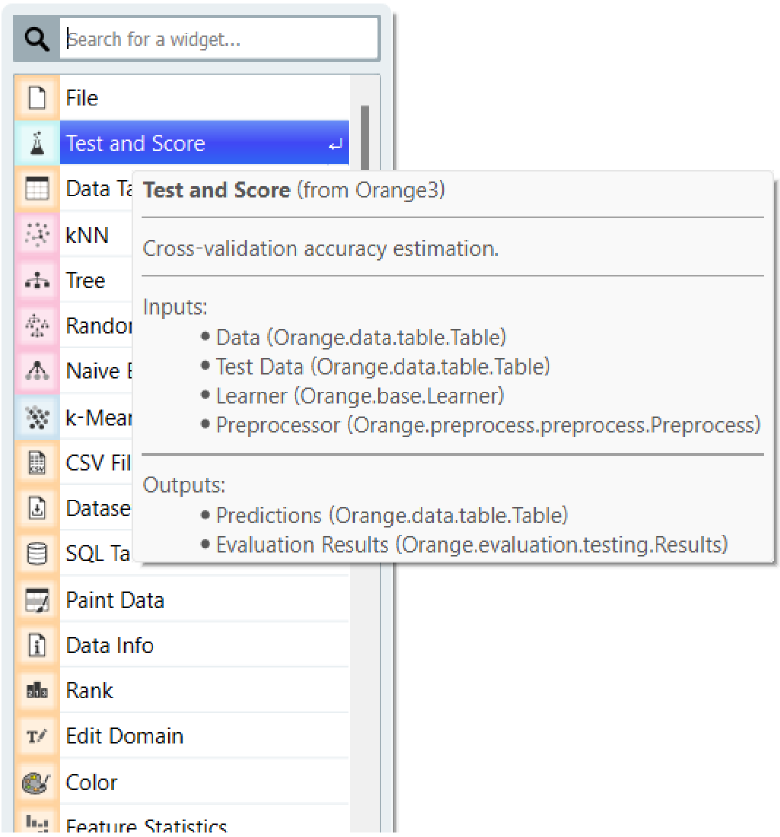


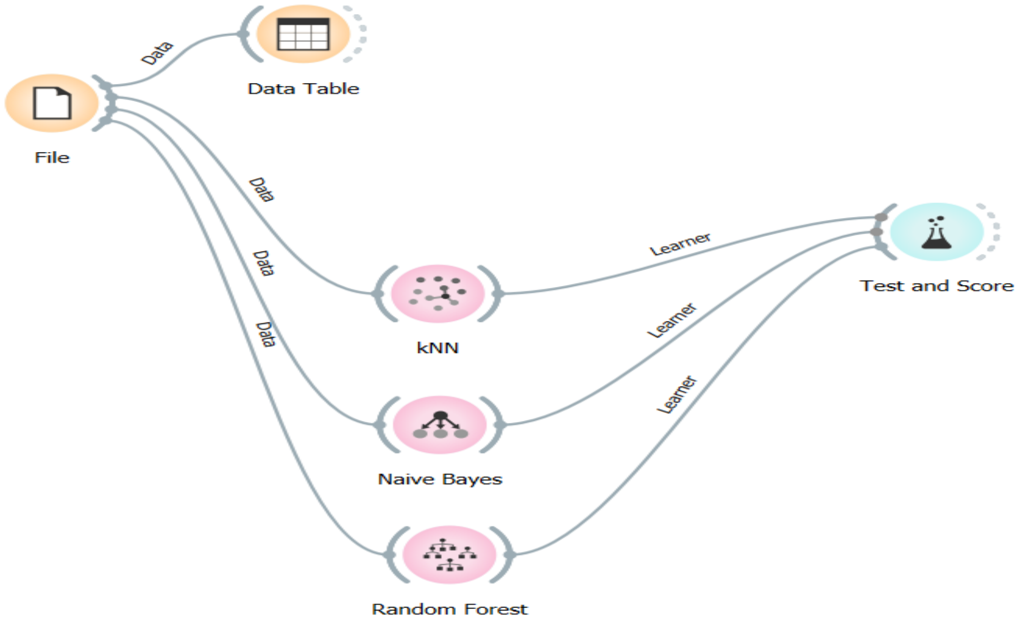
Select the widget or the model of our interest and connect the model and the file.



STEP 4 : TEST AND SCORE FOR PERFORMANCE ANALYSIS

Right click and select test and score.





connect the models to the test and score widget so that the test and score calculates the performance metrics of the models.

STEP 5 : EVALUATION RESULTS AND PREDICTIONS

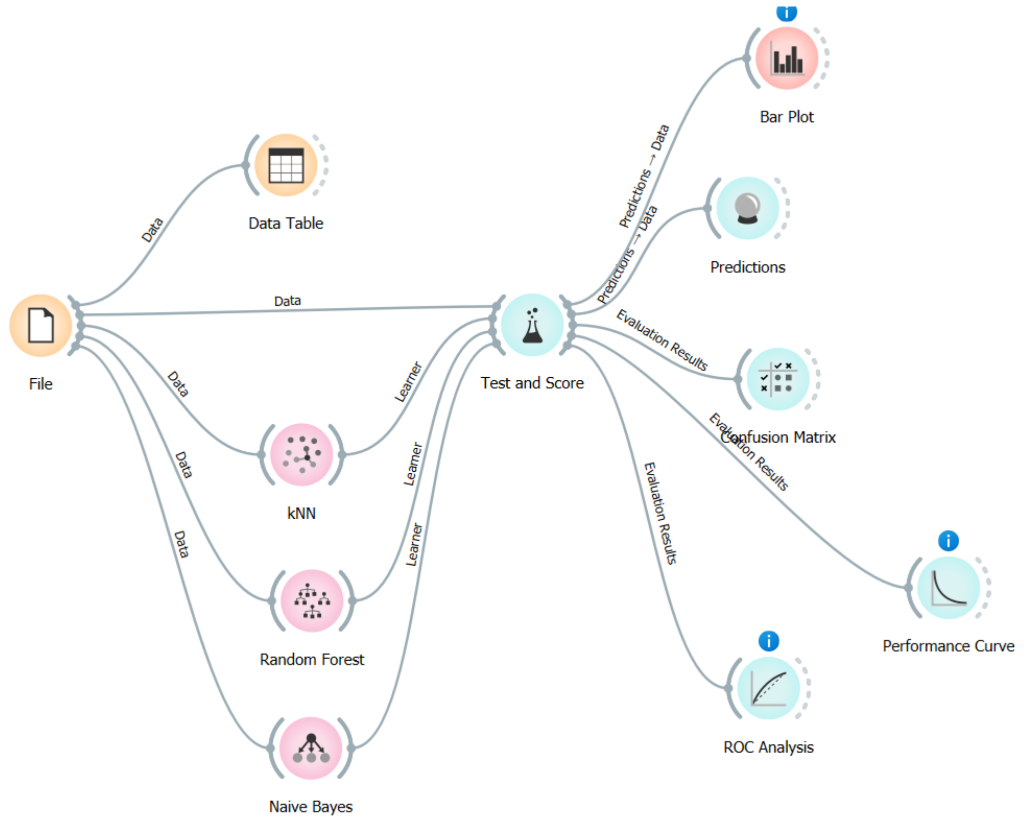
Evaluation results and predictions is to visualize the data in a graphical plot to predict and evaluate the results.

The Evaluation results used here are :

1) ROC Analysis
2) Performance curve
3) Confusion Matrix

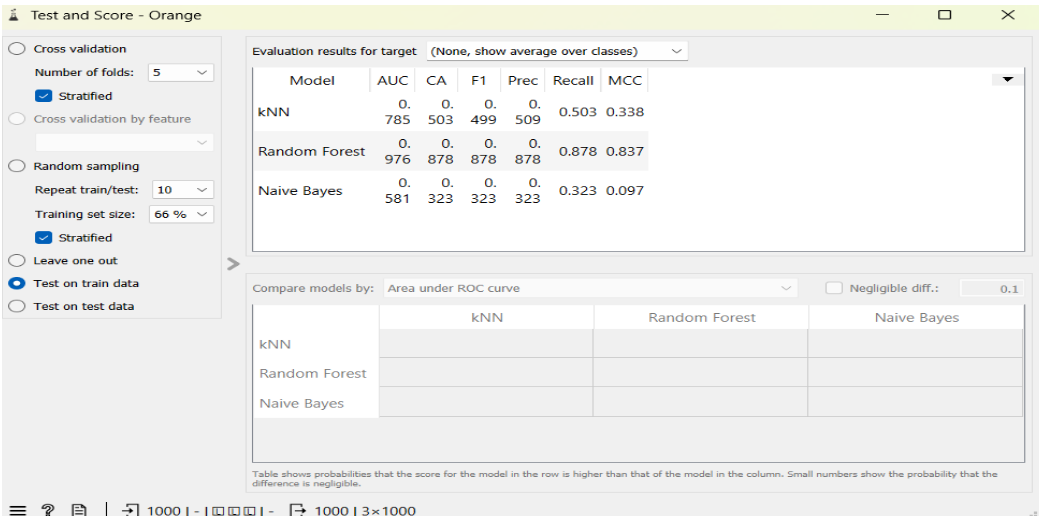
The prediction data used here is :

1) Bar Plot
2) Predictions



STEP 6 : ANALYSING THE TEST SCORE AND THE RESULTS

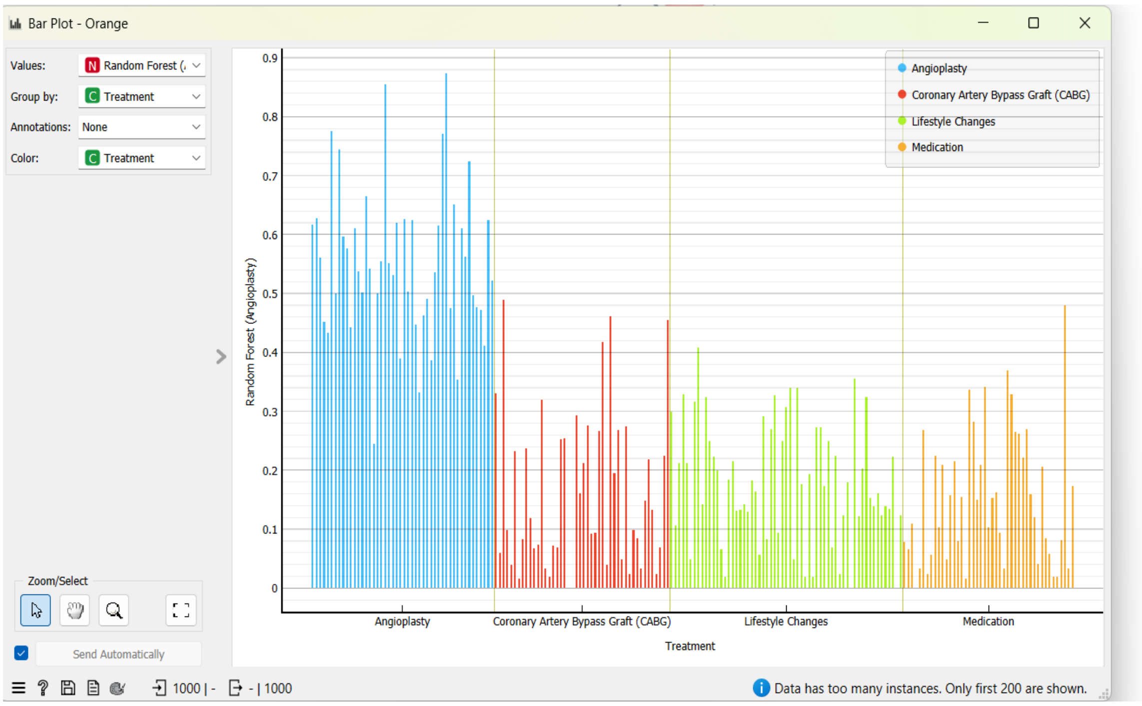
1) TEST AND SCORE



Model	AUC	CA	F1	Prec	Recall	MCC
kNN	0.785	0.503	0.499	0.509	0.503	0.338
Random Forest	0.976	0.878	0.878	0.878	0.878	0.837
Naive Bayes	0.581	0.323	0.323	0.323	0.323	0.097

Here the accuracy for Naïve bayes has highest accuracy, hence we will be considering Naïve Bayes model for our evaluation.

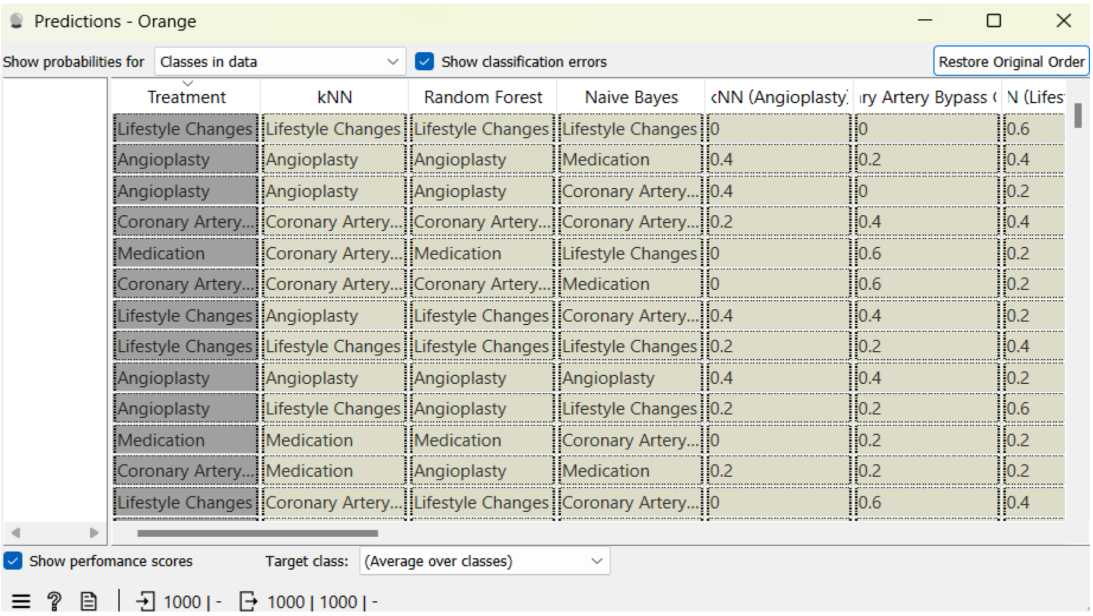
2)BAR GRAPH



We can change the group by the feature of our choice.

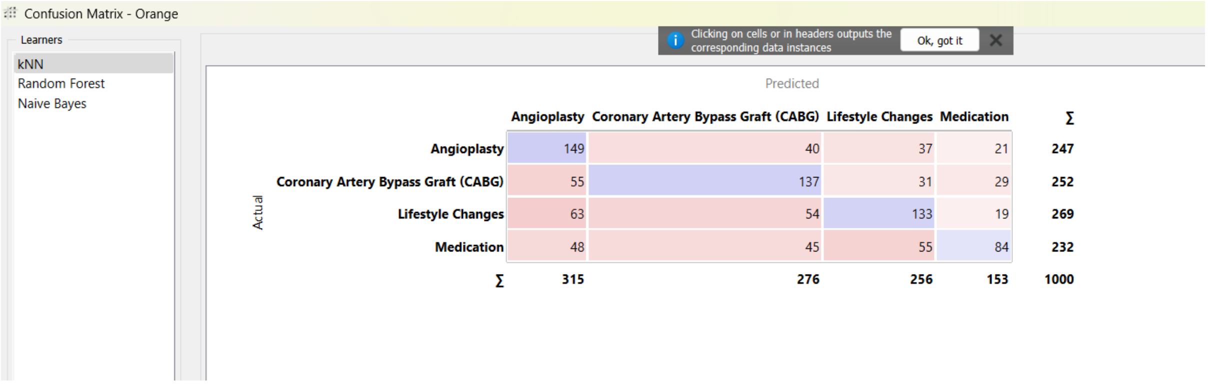
The values of the bar graph ,can be obtained by the model of our choice.

3)PREDICTIONS



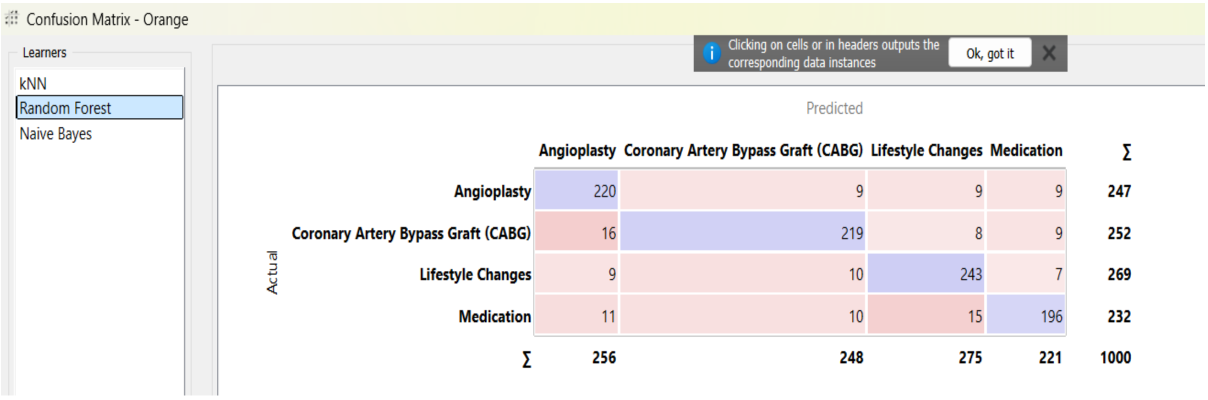
Treatment	kNN	Random Forest	Naive Bayes	‹NN (Angioplasty	ıry Artery Bypass (	N (Lifes
Lifestyle Changes	Lifestyle Changes	Lifestyle Changes	Lifestyle Changes	0	0	0.6
Angioplasty	Angioplasty	Angioplasty	Medication	0.4	0.2	0.4
Angioplasty	Angioplasty	Angioplasty	Coronary Artery...	0.4	0	0.2
Coronary Artery...	Coronary Artery...	Coronary Artery...	Coronary Artery...	0.2	0.4	0.4
Medication	Coronary Artery...	Medication	Lifestyle Changes	0	0.6	0.2
Coronary Artery...	Coronary Artery...	Coronary Artery...	Medication	0	0.6	0.2
Lifestyle Changes	Angioplasty	Lifestyle Changes	Coronary Artery...	0.4	0.4	0.2
Lifestyle Changes	Lifestyle Changes	Lifestyle Changes	Lifestyle Changes	0.2	0.2	0.4
Angioplasty	Angioplasty	Angioplasty	Angioplasty	0.4	0.4	0.2
Angioplasty	Lifestyle Changes	Angioplasty	Lifestyle Changes	0.2	0.2	0.6
Medication	Medication	Medication	Coronary Artery...	0	0.2	0.2
Coronary Artery...	Medication	Angioplasty	Medication	0.2	0.2	0.2
Lifestyle Changes	Coronary Artery...	Lifestyle Changes	Coronary Artery...	0	0.6	0.4

3) CONFUSION MATRIX – KNN



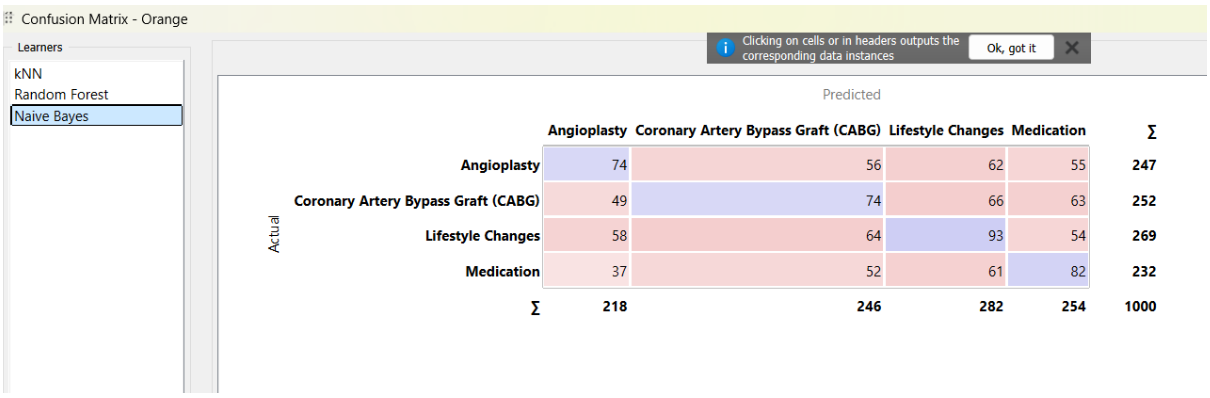
	Angioplasty	Coronary Artery Bypass Graft (CABG)	Lifestyle Changes	Medication	∑
Angioplasty	149	40	37	21	247
Coronary Artery Bypass Graft (CABG)	55	137	31	29	252
Lifestyle Changes	63	54	133	19	269
Medication	48	45	55	84	232
∑	315	276	256	153	1000

4) CONFUSION MATRIX – RANDOM FOREST



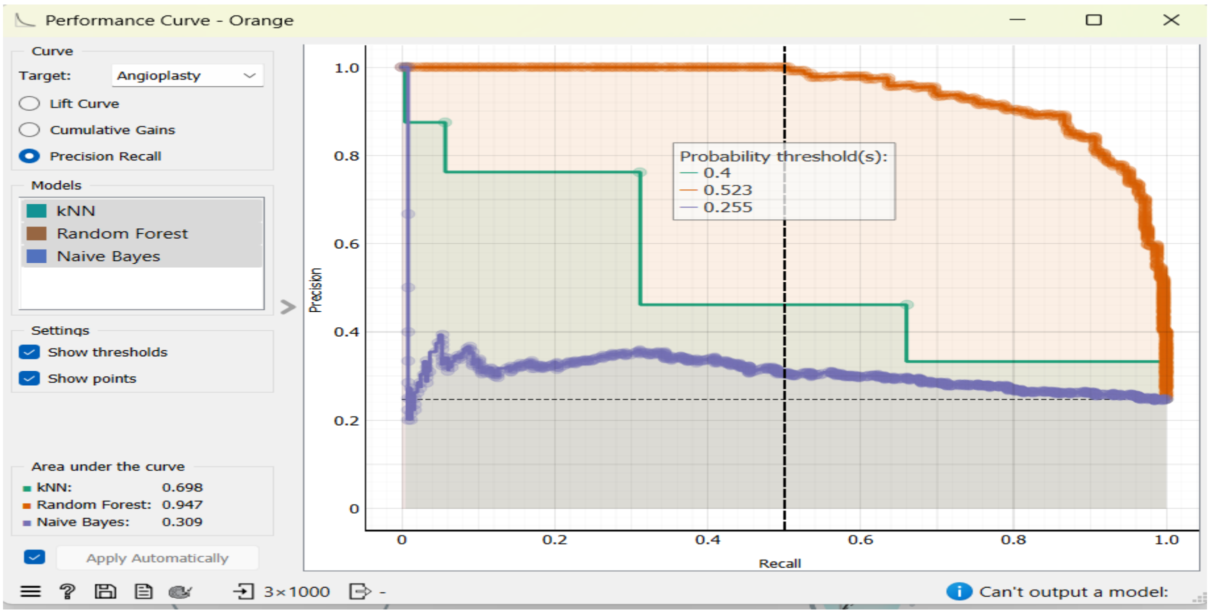
	Angioplasty	Coronary Artery Bypass Graft (CABG)	Lifestyle Changes	Medication	∑
Angioplasty	220	9	9	9	247
Coronary Artery Bypass Graft (CABG)	16	219	8	9	252
Lifestyle Changes	9	10	243	7	269
Medication	11	10	15	196	232
∑	256	248	275	221	1000

5) CONFUSION MATRIX – NAÏVE BAYES

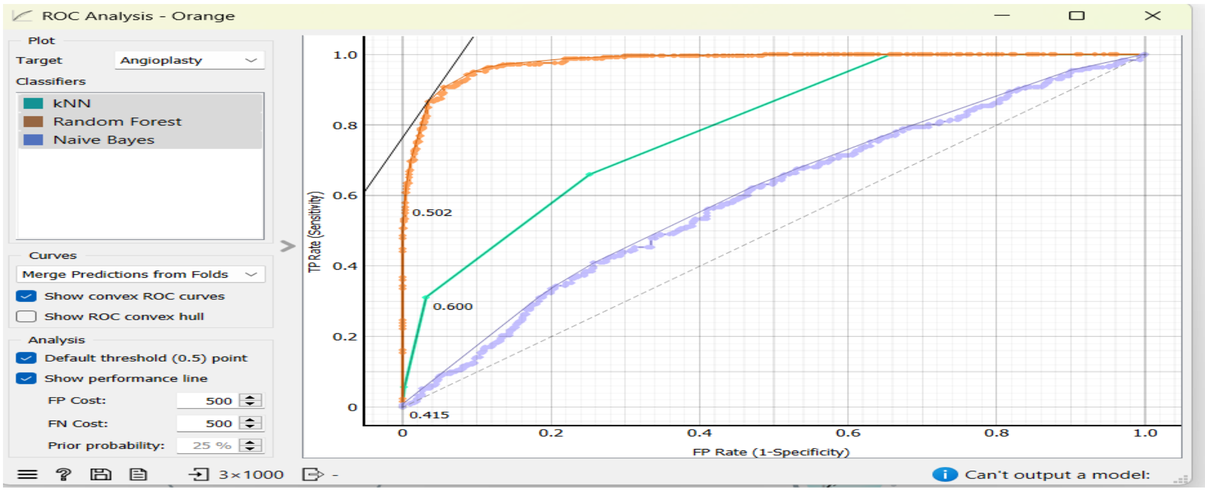


	Angioplasty	Coronary Artery Bypass Graft (CABG)	Lifestyle Changes	Medication	∑
Angioplasty	74	56	62	55	247
Coronary Artery Bypass Graft (CABG)	49	74	66	63	252
Lifestyle Changes	58	64	93	54	269
Medication	37	52	61	82	232
∑	218	246	282	254	1000

6) PERFORMANCE CURVE



7) ROC ANALYSIS



INFERENCE :

1. User-Friendly Design:

- Drag-and-Drop Interface: Users can effortlessly upload datasets, visualize data, and apply machine learning models by linking widgets within a drag-and-drop environment.

- Widgets: The foundational elements of Orange, widgets execute specific tasks such as data preprocessing, model selection, and evaluation, enabling users to build complex data analysis pipelines without the need for coding.

2. **Dataset Integration:**

- Datasets, such as a heart attack risk factors dataset, can be easily uploaded and visualized to immediately understand data distributions and patterns before any machine learning models are applied.

3. **Model Selection:**

- Orange offers a range of machine learning models, including Naïve Bayes, KNN, and Random Forest. These models can be selected and applied to datasets, simplifying the process of building and testing predictive models.

4. **Performance Evaluation:**

- Test and Score: Users can assess the performance of different models by linking them to a test and score widget, which provides accuracy metrics, helping users identify the best model for their task.
- Evaluation Results: Orange enables users to visualize performance metrics, such as ROC curves, performance curves, and confusion matrices, facilitating the interpretation of model outcomes and the comparison of different models.

5. **Prediction and Visualization:**

- Bar Plots and Predictions: The tool supports the visualization of predictions and results through bar plots and other graphical formats, making data insights more accessible and easier to comprehend.

CONCLUSION:

Orange offers a user-friendly approach to data mining and machine learning, featuring a visual programming interface that streamlines the processes of data analysis, model selection, and performance evaluation. It is particularly beneficial for those who prefer not to write code, providing a powerful yet accessible solution for solving machine learning problems.