PROBLEM DESCRIPTION:

Cumulative Grade Point Average (CGPA) refers to the overall Grade Point Average (GPA), obtained by dividing the total Grade Points (GPs) earned in all courses attempted by the total degree-credit hours in all attempted courses. You are required to develop a machine learning system to predict final CGPA of a student at the end of fourth year given GPs of the courses obtained in initial years (up to first, second or third year).

- ✓ The dataset to be used is attached with this file with name The_Grades_Dataset.csv.
- ✓ Model 1: predict final CGPA based on GPs of first year only.
- ✓ Model 2: predict final CGPA based on GPs of first two years.
- ✓ Model 3: predict final CGPA based on GPs of first three years.

DATA EXPLORATION:

First of all we import all the useful libraries and load our dataset named as The_Grades_Dataset.csv. Then we explore it by head() function, find out the insights by info() function and check out for the missing values in the dataset by isnull().sum() function.

DATA PREPROCESSING:

- ✓ First of all we check for the missing values in our dataset by plotting it using a heatmap.
- ✓ As NaN values are one of the major problems in Data Analysis and it is one of the common ways to represent the missing values in the data so it is very essential to deal with NaN in order to get the desired result. So for this we replace all the NaN values with zeros in Pandas Dataframe.
- ✓ Furthermore we drop all the 4^{th} year courses along with roll nos. and CGPA and pass the remaining attributes as our input to the model.
- ✓ Declare CGPA as our target variable.
- ✓ Then we find out the unique values of our attributes and replace those by numeric values and stored it in a dictionary named as grades_enc.
- ✓ Now we have replaced all the string values of our dataset with the numeric values stored in grades_enc dictionary. In this way our dataset converts into numeric values hence which are easy to deal with.
- ✓ Then we concatenate the inputs and targets for data insights.
- ✓ In order to visualize the data now, we called the hist() function. A histogram is a representation of the distribution of data. This function calls matplotlib.pyplot.hist(), on each series in the DataFrame, resulting in one histogram per column. (Since we have 34 columns altogether including target variable)
- ✓ Then we show the existence of correlation by plotting heatmap.

- ✓ Furthermore we calculate some statistical data such as count, mean, std, min, etc using describe() method.
- ✓ In the end we convert DataFrame into CSV data and passed a file object to write the CSV data into a file. Hence the data has been cleaned and ready for model implementations.
- ✓ In the last step we split the entire file into different columns based on our model prediction requirements.

MODELS USED IN THE SYSTEM:

- ✓ MODEL 1: This model predicts the final CGPA based on the GP'S of First year only.
- ✓ MODEL 2: This model predicts the final CGPA based on the GP'S of First two years.
- ✓ MODEL 3: This model predicts the final CGPA based on the GP'S of First three years.

ALGORIHMS IMPLEMENTED:

- ✓ **LINEAR REGRESSION:** This algorithm is used as it finds the best fit linear line and finds the relation between dependent variable and independent variable such that the error is minimized. Linear regression is used in our models as they are well understood and can be trained easily.

- ✓ **RANDOM FOREST ALGORITHM:** This algorithm is implemented as it produces good predictions and can handle large datasets easily. Random forest algorithm provides high level of accuracy in predicting the outcomes of our models.

PERFORMANCE OF OUR SYSTEM:

There are no such issues like overfitting or underfitting in our models. Underfitting happens when the model does not fits the data well enough or it maps poorly to the trend of data. There is no underfitting in our case as our models fits the data very well as majority of the points lies near the regression line and there are some points that lies out of this region as illustrated in the graphs. As far as overfitting is concerned it happens when the model fits too well on train data and fails to generalize on test data. In our models we don't have this issue as there are some points that lie outside the region and our model fits well on training data and also generalizes on test data. As a result of which the accuracy and efficiency of our model is good.

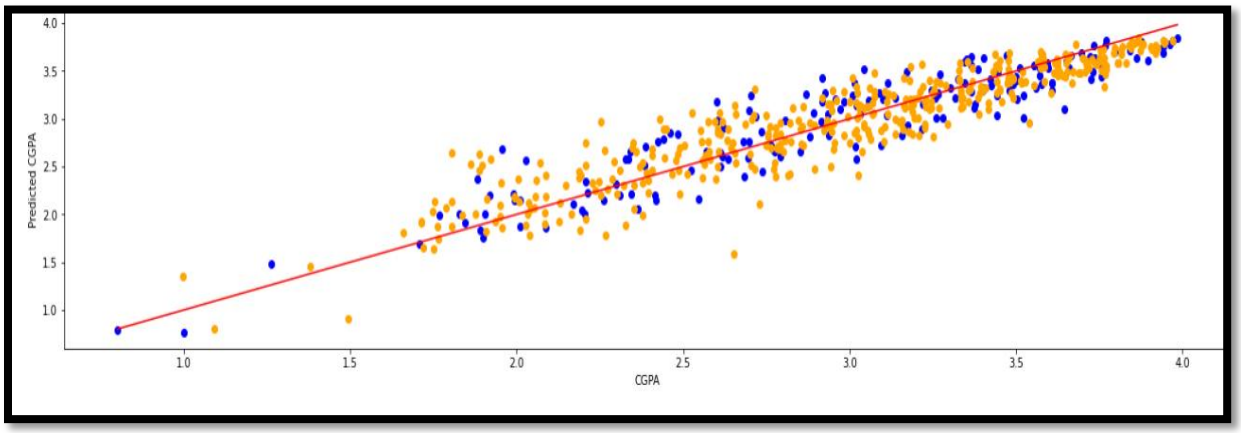
DISTINGUISHING FEATURES:

- ✓ The algorithms which we used for our model implementations i.e. Linear Regression and Random Forest Algorithm are designed in such a way that it provides greater weightage to CS courses as compared to others.
- ✓ We have made used of histograms and heat maps to plot different graphs in order to visualize and explore data in a more appropriate way.

GRAPHICAL COMPARISON OF MODELS:

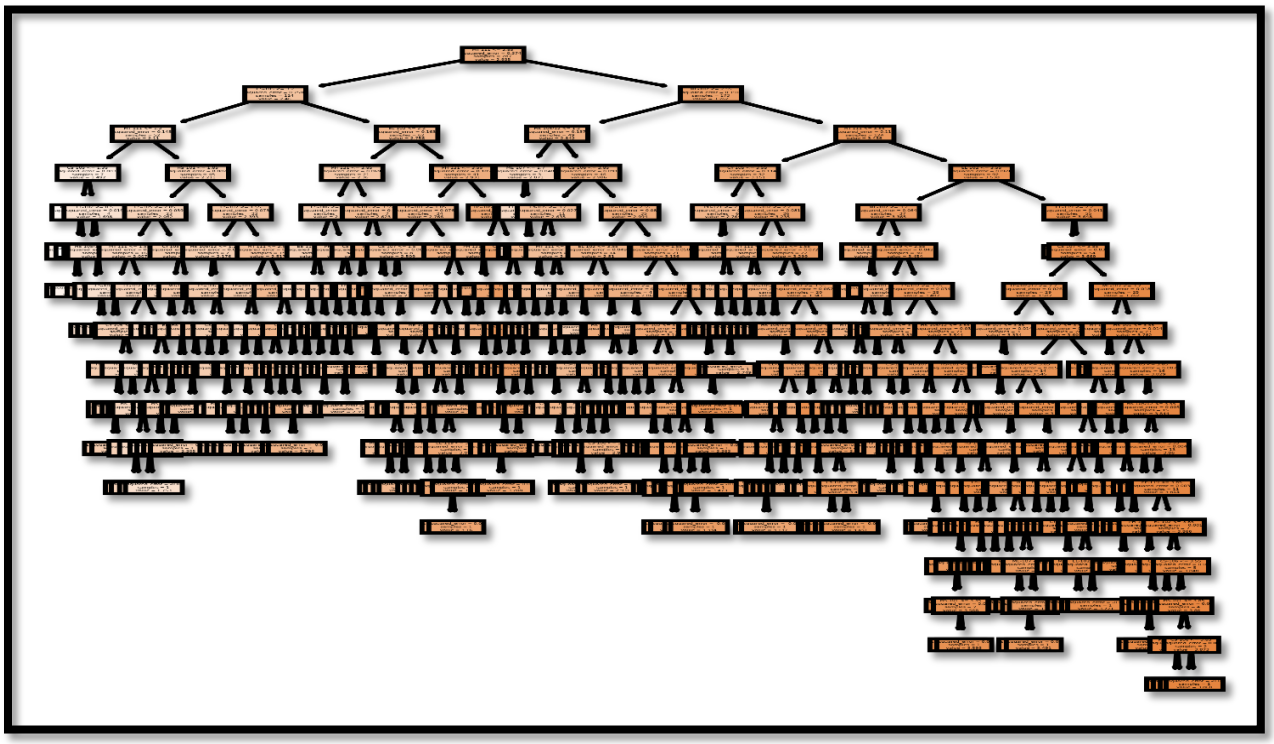
- **VISUALIZATION OF LINEAR REGRESSION ON MODEL 1:**

The graph is plotted between actual CGPA and predicted CGPA with actual CGPA on X axis and predicted CGPA on Y axis. The test score of the model when linear regression is implemented is 86%.



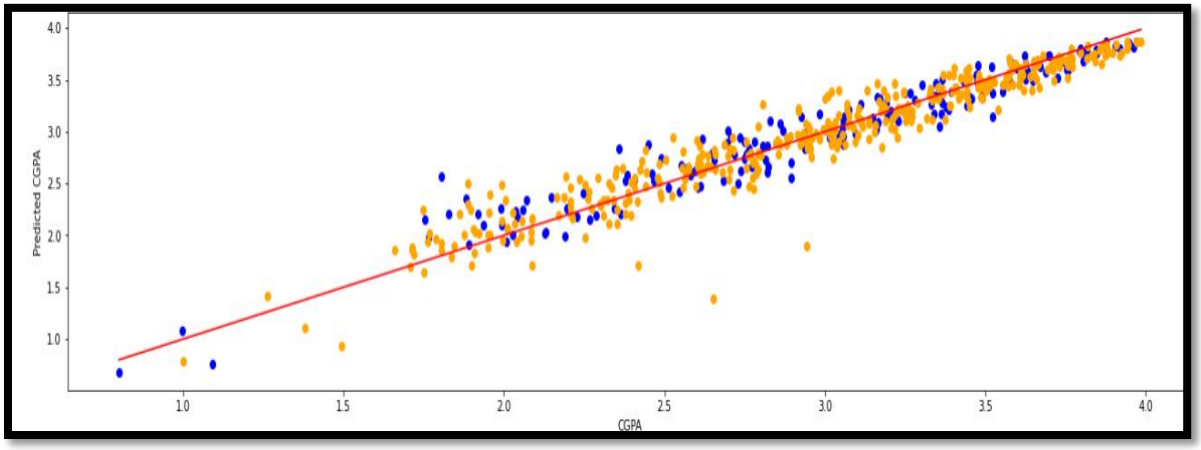
- **VISUALIZATION OF RANDOM FOREST ON MODEL 1:**

The test score of the model when random forest algorithm is implemented is 82%.



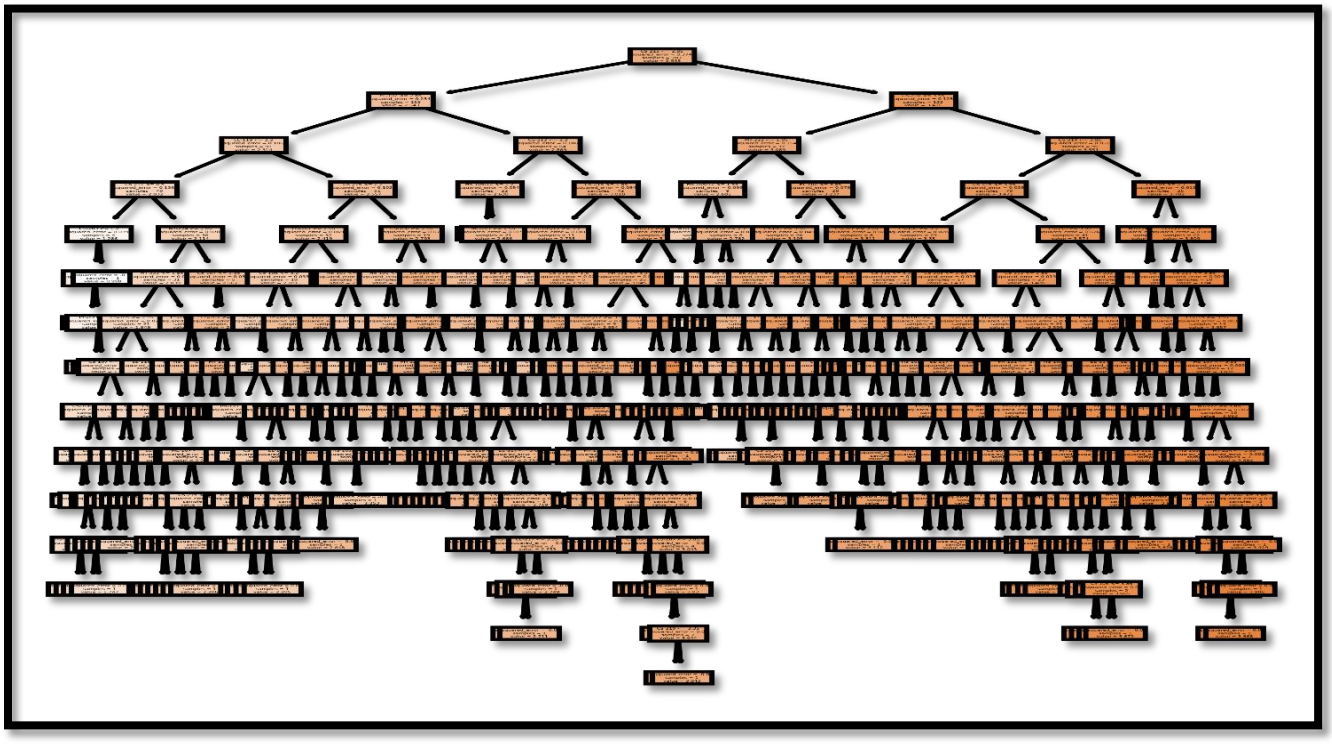
- **VISUALIZATION OF LINEAR REGRESSION ON MODEL 2:**

The test score of the model when linear regression is implemented is 93%.



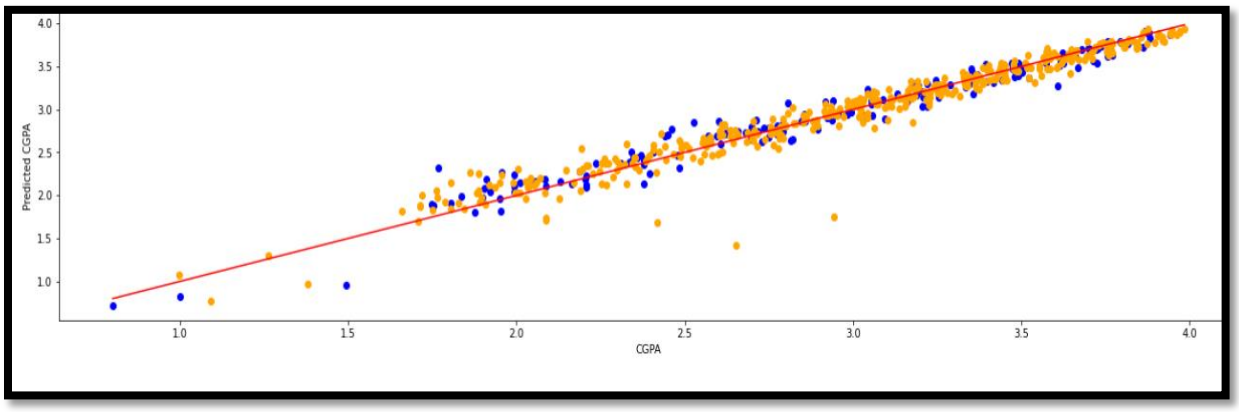
- **VISUALIZATION OF RANDOM FOREST ON MODEL 2:**

The test score of the model when random forest algorithm is implemented is 92%.



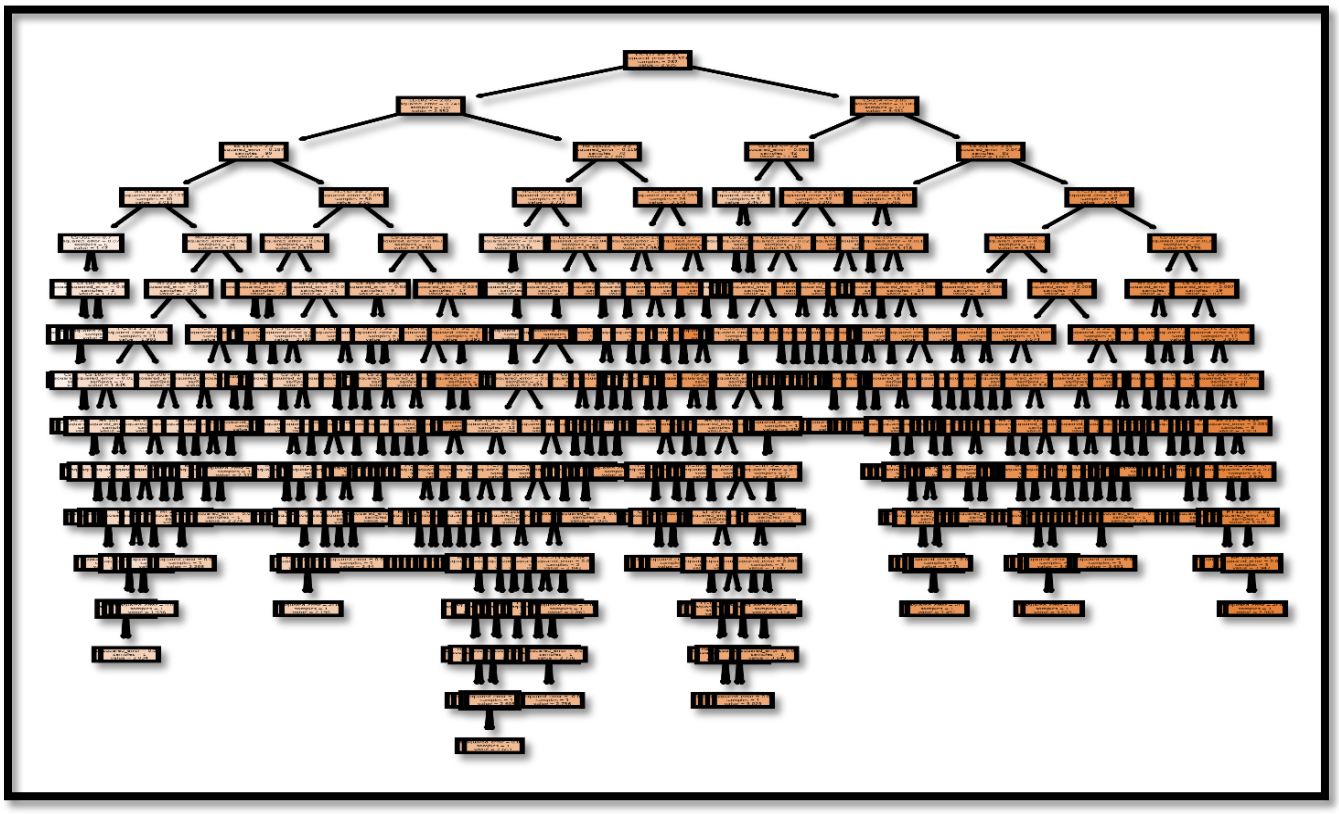
- **VISUALIZATION OF LINEAR REGRESSION ON MODEL 3:**

The test score of the model when linear regression is implemented is 96%.



- **VISUALIZATION OF RANDOM FOREST ON MODEL 3:**

The test score of the model when random forest algorithm is implemented is 95%.



GUI IMPLEMENTATION:

```
------------------------------------------------------------
-------------Hello from GPA Predictor----------------------
------------------------------------------------------------
What is your name? Bushra
From which model would you like to predict your GPA ? 2
Enter your PH-121 GPA : A+
Enter your HS-101 GPA : A
Enter your CY-105 GPA : B
Enter your HS-105/12 GPA : B-
Enter your MT-111 GPA : C
Enter your CS-105 GPA : D
Enter your CS-106 GPA : A
Enter your EL-102 GPA : A
Enter your EE-119 GPA : A
Enter your ME-107 GPA : W
Enter your CS-107 GPA : A
Enter your HS-205/20 GPA : B
Enter your MT-222 GPA : F
Enter your EE-222 GPA : D
Enter your MT-224 GPA : A
Enter your CS-210 GPA : C
Enter your CS-211 GPA : A
Enter your CS-203 GPA : C-
Enter your CS-214 GPA : A
Enter your EE-217 GPA : A
Enter your CS-212 GPA : A
Enter your CS-215 GPA : A
Bushra, your predictive CGPA according to Linear Regression Algorithm is :[3.07284546]
Bushra, Your predictive CGPA according to Random Forest Algorithm is :[2.95525]
Have a nice day !!
```

Snapshot of GUI implementation is attached above. First of all a user will come and enter its name as input and then select a model by which he wants to get it's GPA predicted.

- ✓ If a user select model 1 then all first year courses will appear and user is asked to enter its GPA individually.
- ✓ If a user selects model 2 then all first year and second year courses will appear and user is asked to enter its GPA individually.
- ✓ If a user selects model 3 then all the three year courses will appear and user is asked to enter its GPA individually.

Now according to the selected model our two implemented algorithms will predict the GPA of the particular student. This is the overall GUI interface which we have implemented for our given problem.