STUDENT PERFORMANCE ANALYSIS

(Group members: Jaishankar Harshit Geddam, Avani Jain)

1. Abstract:

Use of machine learning and data science makes life easier in every aspect, using machine learning and predicting the outcomes before the exam will allow the students as well as their parents and teachers to analyze the improvement area. we adopt numerous machine learning concepts and algorithms in order to predict the grades of the student before an exam. And machine learning is booming, and machine learning is firmly identified with (and frequently covers with) computational insights, which also focuses on prediction making through the use of technology. It has solid connections to numerical improvement, which conveys strategies, hypothesis and application areas to the field. Machine learning is some of the time conflated with data mining where the latter subfield concentrates more on exploratory information analysis and is known as supervised learning.

The main idea of this project is:

- In this project our aim is to analyze all the factors affecting the grades of a student and then predicting the grades of the student based upon those factors.
- We are implementing different supervised regression models for prediction of the grades.
- Using R-Squared, MAE, MSE, RMSE to evaluate the best model among all the implemented supervised regression models.
- The outcome of the algorithms predicts the number of students who are likely to pass, fail or promoted to next year. The results provide steps to improve the performance of the students who were predicted to fail or promoted.

2. Background:

Data sources and data preparation:

- We are using the Student Performance Data set available on the UCI Machine Learning Repository.
- There is a total of 33 attributes in a data with 649 data instances, so the dataset is enough for our analysis and generating predictions.
- All the datasets used were in csv format, so we didn't have perform any additional data preparation tasks in order to use the data.

Data exploration, visualization, cleansing and transformation

The data set available in UCI Machine Learning Repository, which is been used in this project is already preprocessed and clean. The data set has 33 attributes and none of the attributes has null values. So, we did not do any cleansing to our data set. The data set includes variable such as sex, address, family size, mother's education, father's education, mothers' job, fathers' job and many other parameters, which contributes towards the prediction. The main task here is to predict the G3 grades.

After some initial exploration of the datasets, we came up with the following dataset with which we conducted our analysis on to achieve our goals.

	school	sex	age	address	famsize	Pstatus	Medu	Fedu	Mjob	Fjob	...	famrel	freetime	goout	Dalc	Walc	health	absences	G1	G2	G3
0	GP	F	18	U	GT3	A	4	4	at_home	teacher	...	4	3	4	1	1	3	6	5	6	6
1	GP	F	17	U	GT3	T	1	1	at_home	other	...	5	3	3	1	1	3	4	5	5	6
2	GP	F	15	U	LE3	T	1	1	at_home	other	...	4	3	2	2	3	3	10	7	8	10
3	GP	F	15	U	GT3	T	4	2	health	services	...	3	2	2	1	1	5	2	15	14	15
4	GP	F	16	U	GT3	T	3	3	other	other	...	4	3	2	1	2	5	4	6	10	10

Checking the null values:

```
#Check the Null values
df.isnull().any()
```

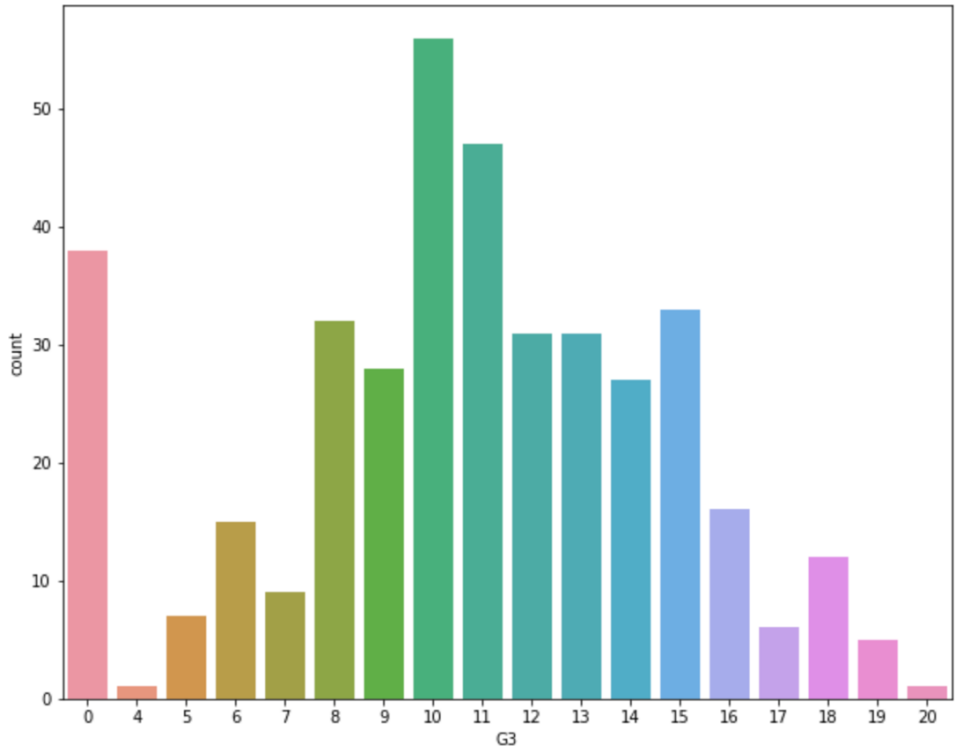
```
school        False
sex           False
age           False
address       False
famsize       False
Pstatus       False
Medu          False
Fedu          False
Mjob          False
Fjob          False
reason        False
guardian      False
traveltime    False
studytime     False
failures      False
schoolsup     False
famsup        False
paid          False
activities    False
nursery       False
higher        False
internet      False
romantic      False
famrel        False
freetime      False
goout         False
Dalc          False
Walc          False
health        False
absences      False
G1            False
G2            False
G3            False
dtype: bool
```

The major problem we faced with our data set was the data available to us was mostly categorical data, and in order to use the data for modelling we had to convert all the categorical variables to numerical variables. When all the categorical variables were converted to the numerical variables, the data set was good to go for modelling and also good for prediction.

After converting the categorical variables to numerical variables:

	age	Medu	Fedu	traveltime	studytime	failures	famrel	freetime	goout	Dalc	...	activities_no	activities_yes	nursery_no	nursery_yes	higher_no	high
0	18	4	4	2	2	0	4	3	4	1	...	1	0	0	1	0	
1	17	1	1	1	2	0	5	3	3	1	...	1	0	1	0	0	
2	15	1	1	1	2	3	4	3	2	2	...	1	0	0	1	0	
3	15	4	2	1	3	0	3	2	2	1	...	0	1	0	1	0	
4	16	3	3	1	2	0	4	3	2	1	...	1	0	0	1	0	
5	16	4	3	1	2	0	5	4	2	1	...	0	1	0	1	0	
6	16	2	2	1	2	0	4	4	4	1	...	1	0	0	1	0	
7	17	4	4	2	2	0	4	1	4	1	...	1	0	0	1	0	
8	15	3	2	1	2	0	4	2	2	1	...	1	0	0	1	0	
9	15	3	4	1	2	0	5	5	1	1	...	0	1	0	1	0	
10	15	4	4	1	2	0	3	3	3	1	...	1	0	0	1	0	

We have visualized the count plot for the predicted variable:



According to the count plot more students are getting the passing marks that is 10. And also, we can see the students getting full marks are very few. Some students are also getting 0 marks and the count of students getting 0 marks is 38 approx.

For the Classification modeling we have to create our dataset according to our problem, our main aim to use classification modelling is to predict weather the student will be passed or failed, for that we have to pre-process the data accordingly.

So, in this pre-processing part we have considered the 'G3' column and then we have classified as if the marks are greater than the half of the total marks if so, then the student is pass or else the student has failed.

3. Related Work and Methodology:

The main task of this project after loading the data is to eliminate the variables which don't have impact towards the prediction. We are achieving this by following the backward elimination process. We are using the Ordinary least square method to find the P values of each attribute. We have set significance level value to be 0.05. The variables having P value greater than 0.05 don't contribute towards the prediction and are to be ignored.

Backward elimination process: This process is carried out to remove the variables which don't contribute towards the prediction. The first task is to throw all the independent variables to the equation, we set a significance level, the algorithm checks for the highest P-value from all the independent variables, if the P value is greater than the significance level, that independent variable is been removed from the dataset as it doesn't contribute toward the prediction, and the process is continued till the algorithm removes all the independent variable with greater P value than the significance level.

We used various supervised learning algorithms to model our data. Models we used in our project are Multiple linear regression, Support vector regression, Random forest regression. Then we will be evaluating each model to check which model is best for our prediction using R-Squared, MAE, MSE, RMSE.

We used Classification modelling in order to predict weather the student would be passed or failed in the next exam, and we have implemented 2 classification models, logistic regression and XG Boost and we have also done the evaluation part by plotting the confusion matrix to get the accuracy, recall, F1 scores for both of the algorithms. The ROC and the AUC curve also we plot in order to see which model works better in our data.

Models:

Multiple Linear Regression: Multiple linear regression (MLR), also known simply as multiple regression, is a statistical technique that uses several explanatory variables to predict the outcome of a response variable. The goal of multiple linear regression (MLR) is to model the linear relationship between the explanatory (independent) variables and response (dependent) variable.

Support Vector Machine Regression: A support vector machine Regression is a supervised machine learning algorithm which is used in regression environment. In this calculation, we plot every data thing as a point in n- dimensional space (where n is number of highlights you have) with the estimation of each element being the estimation of a specific facilitate. At that point, we perform grouping by finding the hyper-plane that separate the two classes and form the best fit line and give up the predictions.

Random Forest Regression: A Random Forest is an ensemble technique capable of performing both regression and classification tasks with the use of multiple decision trees

and a technique called Bootstrap Aggregation, commonly known as bagging. The basic idea behind this is to combine multiple decision trees in determining the final output rather than relying on individual decision trees.

Logistic regression: Logistic regression is a classification model. The model builds a classification model to predict the probability that a given data entry belongs to the category numbered as “1” or “0”. Just like Linear regression assumes that the data follows a linear function, Logistic regression models the data using the sigmoid function.

XG Boost: XG Boost is a decision-tree-based ensemble Machine Learning algorithm that uses a gradient boosting framework. In prediction problems involving unstructured data (images, text, etc.) artificial neural networks tend to outperform all other algorithms or frameworks. However, when it comes to small-to-medium structured/tabular data, decision tree-based algorithms are considered best-in-class right now.

4. Experiments and results:

We have run the backward elimination process and noticed out of 33 attributes only 5 attributes contribute towards the prediction, these 5 attributes have the P value less than the Significance level value. So, we will be using these 5 variables only for prediction.

Even though, first we are using all the variables for prediction and there after we are using only the best 5 variables, then comparing both the cases results and calculating the MAE, MSE, RMSE for both the cases.

We are predicting the grades of student by considering many variables and then distributing our data into training and testing.

First, we are implementing multiple linear regression and we compare each result:

Index	G3
329	14
318	10
317	9
65	15
59	16
287	12
212	14
60	11
394	9
324	15
375	10
74	11

The Original data from dataset

0	14.1889
1	10.731
2	9.39627
3	15.5454
4	16.5618
5	11.7355
6	13.2711
7	10.3054
8	8.14864
9	13.5894

Predictions with all variables

	0
0	13.981
1	10.4329
2	9.45362
3	15.5901
4	16.2359
5	11.9822
6	12.5536
7	10.4212
8	7.62376
9	14.7354
10	6.95917

Predictions with best variables

Second, we are implementing Support Vector regression and we compare each result:

G3
6
6
10
15
10
15
11
6
19
15
9
12
14

The Original data from dataset

0
5.7238
5.01102
7.92623
14.0956
9.73056
15.6098
12.126
4.85132
18.3104
15.4825

Predictions with best variables

Third, we are implementing Random forest regression and we compare each result:

6
6
10
15
10
15
11
6
19
15

The Original data from dataset

6.14
5.97
9.2
14.71
9.77
15.23
11.1
6
18.64
15.1

Predictions with best variables

Fourth, we are implementing classification modelling such as Logistic regression and we compare each result:

The Original data from dataset
1
1
1
1
1
1
1
1
0
1
0
1
1
0

Predictions with best variables
1
0
0
1
1
1
1
1
0
1
0
1
0
0

Fifth, we are implementing classification modelling such as XG Boost and we compare each result:

The Original data from dataset
1
0
1
1
1
1
1
0
1
0
1
1
0
1

Predictions with best variables
1
0
1
1
1
1
1
0
1
0
1
0
0
1

Evaluation:

As we are using regression models for modelling, we have several techniques for modelling, such as R-squared, MAE, MSE, RMSE for evaluation. We calculate the MSE, MAE, R-SQUARED, RMSE values using the statistical formulas. The lower the MSE, MAE values the higher will be the accuracy percentage be.

Mean Square Error (MSE): MSE is the sum of squared distances between our target variable and predicted values.

Mean Absolute Error (MAE): MAE is the sum of absolute differences between our target and predicted variables. So, it measures the average magnitude of errors in a set of predictions, without considering their directions.

Root Mean Square Error (RMSE): (Root Mean Squared Error) is the error rate by the square root of MSE.

R-squared: R-squared (Coefficient of determination) represents the coefficient of how well the values fit compared to the original values. The value from 0 to 1 interpreted as percentages. The higher the value is, the better the model is.

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}|$$

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y})^2$$

$$RMSE = \sqrt{MSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y})^2}$$

$$R^2 = 1 - \frac{\sum(y_i - \hat{y})^2}{\sum(y_i - \bar{y})^2}$$

Where,

$\hat{y}$ – *predicted value of y*

$\bar{y}$ – *mean value of y*

So, the results for R-squared, MAE, MSE, RMSE for each algorithm are below:

Linear Regression:

```
#Model Evaluation Metrics for Regression
# calculate MAE, MSE, RMSE with all the variables
print(metrics.mean_absolute_error(y_test, ypred))
print(metrics.mean_squared_error(y_test, ypred))
print(np.sqrt(metrics.mean_squared_error(y_test, ypred)))
```

```
1.6103288191783278
5.845178266970442
2.4176803483857086
```

The model evaluation considering all the variables.

```
#Model Evaluation Metrics for Regression
# calculate MAE, MSE, RMSE with the best variables
print(metrics.mean_absolute_error(y_test, ypred_model))
print(metrics.mean_squared_error(y_test, ypred_model))
print(np.sqrt(metrics.mean_squared_error(y_test, ypred_model)))
```

```
1.3934764142640703
5.291720423625023
2.300373974732157
```

The model evaluation considering the 5 best variables obtained after backward elimination process.

We can compare the results, the MAE, MSE, RMSE is less when we consider only 5 variables as compared to predicting with all the variables.

```
#r-SQUARED
regressor_OLS=sm.OLS(endog=y, exog=X_Modeled).fit()
regressor_OLS.summary()
regressor_OLS.rsquared
```

```
0.9730797818073089
```

As we can see the R square is very close to 1, therefore it represents the coefficient have well fitted compared to the original values.

Support vector Regression:

```
#Model Evaluation Metrics for Regression
# calculate MAE, MSE, RMSE with the best variables
print(metrics.mean_absolute_error(sc_y.inverse_transform(y), y_pred))
print(metrics.mean_squared_error(sc_y.inverse_transform(y), y_pred))
print(np.sqrt(metrics.mean_squared_error(sc_y.inverse_transform(y), y_pred)))
```

```
1.0313974889294373
3.8274598952944645
1.9563895050051932
```

The model evaluation considering the 5 best variables obtained after backward elimination process and using support vector regression model.

```
#r-SQUARED
regressor_OLS=sm.OLS(endog=y, exog=X_Modeled).fit()
regressor_OLS.summary()
regressor_OLS.rsquared
```

```
0.8087373833539431
```

R-Squared value obtained here is 0.8087, which is not as close to 1.

Random Forest Regression:

```
#Model Evaluation Metrics for Regression
# calculate MAE, MSE, RMSE with all the variables
print(metrics.mean_absolute_error(y, y_pred))
print(metrics.mean_squared_error(y, y_pred))
print(np.sqrt(metrics.mean_squared_error(y, y_pred)))
```

```
0.35075027124773966
0.36362836648439967
0.6030160582309559
```

The model evaluation considering the 5 best variables obtained after backward elimination process and using Random Forest regression model.

```
#r-SQUARED
regressor_OLS=sm.OLS(endog=y, exog=X_Modeled).fit()
regressor_OLS.summary()
regressor_OLS.rsquared
```

```
0.9730797818073089
```

As we can see the R square is very close to 1, therefore it represents the coefficient have well fitted compared to the original values.

As we are using classification models for modelling, we have several techniques for modelling, such as accuracy, precision, recall and F1 score. We calculate the accuracy, precision, recall and F1 score values using the statistical formulas. The highest the accuracy is the better the model is. Even we are plotting the ROC and AUC cure for evaluation.

Accuracy: Accuracy is the most intuitive performance measure and it is simply a ratio of correctly predicted observation to the total observations.

$$\text{Accuracy} = TP+TN/TP+FP+FN+TN$$

Precision: Precision is the ratio of correctly predicted positive observations to the total predicted positive observations.

$$\text{Precision} = TP/TP+FP$$

Recall: Recall is the ratio of correctly predicted positive observations to the all observations in actual class

$$\text{Recall} = TP/TP+FN$$

F1 Score: F1 Score is the weighted average of Precision and Recall. Therefore, this score takes both false positives and false negatives into account.

$$\text{F1 Score} = 2*(\text{Recall} * \text{Precision}) / (\text{Recall} + \text{Precision})$$

AUC - ROC curve: AUC - ROC curve is a performance measurement for classification problem at various thresholds settings. ROC is a probability curve and AUC represent degree or measure of separability. It tells how much model is capable of distinguishing between classes. Higher the AUC, better the model is at predicting 0s as 0s and 1s as 1s. By analogy, Higher the AUC, better the model.

So, the results for accuracy, precision, recall and F1 score are below:

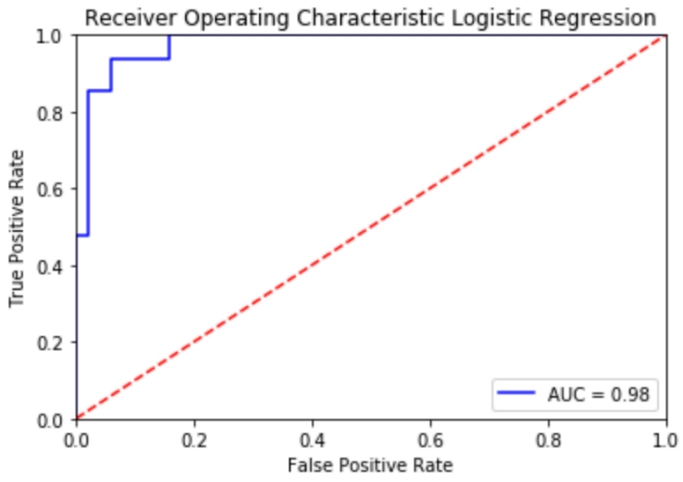
Logistic Regression:

```
        Confusion Matrix
-----------------------------
           Predict
            "1" |  "0"
Actual "1"| 47  | 1
       "0"| 8   | 43
-----------------------------

accuracy =  0.9090909090909091
precision =  0.8545454545454545
recall =  0.9791666666666666
F1 =  0.912621359223301
```

It gives up the accuracy of 90% which is a very good percentage.

AUC - ROC curve for Logistic Regression:



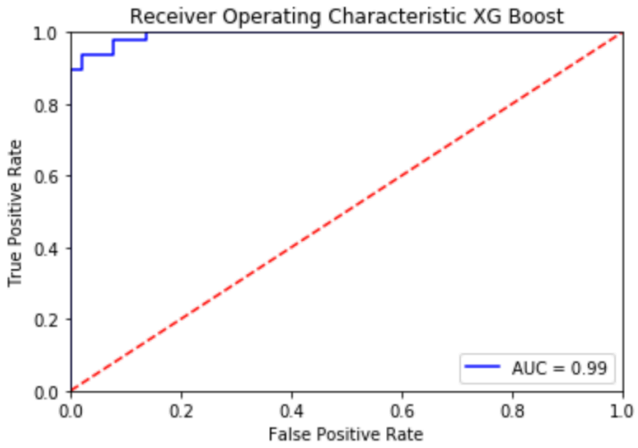
XG Boost:

```
        Confusion Matrix
-----------------------------
           Predict
            "1" |  "0"
Actual "1"| 47  | 1
       "0"| 5   | 46
-----------------------------

accuracy =  0.9393939393939394
precision =  0.9038461538461539
recall =  0.979166666666666
F1 =  0.9400000000000001
```

Here we get an accuracy of 93% which is far better than the logistic regression.

AUC - ROC curve for XG Boost:



6. Conclusion:

By implementing the three models such as multiple linear regression, support vector machine, random forest regression. And after evaluating these models using MAE, MSE, RMSE, R-square, we can conclude that for the given data set, random forest is working great as compared to multiple linear regression and support vector regression. The MAE, MSE, RMSE are very less and also R-square value is very high therefore it represents the coefficient have well fitted compared to the original values. The second-best model here is multiple linear regression.

By implementing the classification models such as logistic regression and XG boost, and after evaluation we can conclude that for our data the XG Boost is working good as compared to logistic regression as we are getting higher accuracy for XG Boost and also the area under curve is more in the case of XG Boost which implies, the XG Boost gives better performance.

7. Future Work:

- We can use other models while considering Regression, in order to get more better best fit line.
- We can use the other models while considering Classification, in order to get better accuracy and better ROC and AUC curves.
- We can also collect more data to be able to do more tests and more follow up to continue improving the prediction.

8. Bibliography:

1. https://www.geeksforgeeks.org/understanding-logistic-regression
2. https://towardsdatascience.com/https-medium-com-vishalmorde-xgboost-algorithm-long-she-may-rein-edd9f99be63d
3. https://blog.exsilio.com/all/accuracy-precision-recall-f1-score-interpretation-of-performance-measures/
4. https://towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5