

**School of Computer Science and Engineering**

VIT Chennai

Vandalur - Kelambakkam Road, Chennai - 600 127

**Final Report**

**Programme:** BTech

**Course:** CSE2004-DBMS

**Slot:** D2

**Faculty:** Prof M. Premalatha

**Component:** J

**Title: STUDENT MARKS PREDICTION**

**Team Member(s):**

DOLLY AGARWALA (20BCE1863)

NAYAN KHEMKA (20BCE1884)

HARSH CHAUHAN (20BCE1886)

**Abstract**

This project attempts to predict grade of students for Maths and Portuguese subjects based on data collected from two Portuguese schools using classification and prediction models. The goal is to go beyond knowing what has happened to providing a best assessment of what will happen in the future. Predictive models use known results to develop (or train) a model that can be used to predict values for different or new data. Modelling provides results in the form of predictions that represent a probability of the target variable based on estimated significance from a set of input variables.

Keywords: Data Mining, Machine Learning, Regression, Prediction, Data Visualisation

**Introduction**

The ultimate goal of any educational institution is offering the best educational experience and knowledge to the students. Identifying the students who need extra support and taking the appropriate actions to enhance their performance plays an important role in achieving that goal. In this research, four machine learning techniques have been used to build a classifier that can predict the performance of the students The models have been compared using the ROC index performance measure and the classification accuracy. In addition, different measures have been computed such as the classification error, precision, recall, and the F measure.

1. **Data Set Description**:

**Source**: -<https://archive.ics.uci.edu/ml/datasets/student+performance>

This data approach student achievement in secondary education of two Portuguese schools. The data attributes include student grades, demographic, social and school related features) and it was collected by using school reports and questionnaires. Two datasets are provided regarding the performance in two distinct subjects: Mathematics (mat) and Portuguese language (por).

Attribute Information: # Attributes for both student-mat.csv (Math course) and student-por.csv (Portuguese language course) datasets:

1. school - student's school (binary: 'GP' - Gabriel Pereira or 'MS' - Mousinho da Silveira)
2. sex - student's sex (binary: 'F' - female or 'M' - male)
3. age - student's age (numeric: from 15 to 22)
4. address - student's home address type (binary: 'U' - urban or 'R' - rural)
5. famsize - family size (binary: 'LE3' - less or equal to 3 or 'GT3' - greater than 3)
6. Pstatus - parent's cohabitation status (binary: 'T' - living together or 'A' - apart)
7. Medu - mother's education (numeric: 0 - none, 1 - primary education (4th grade), 2 â€“ 5th to 9th grade, 3 â€“ secondary education or 4 â€“ higher education)
8. Fedu - father's education (numeric: 0 - none, 1 - primary education

(4th grade), 2 â€“ 5th to 9th grade, 3 â€“ secondary education or 4 â€“ higher education)

1. Mjob - mother's job (nominal: 'teacher', 'health' care related, civil 'services' (e.g. administrative or police), 'at\_home' or 'other')
2. Fjob - father's job (nominal: 'teacher', 'health' care related, civil 'services' (e.g. administrative or police), 'at\_home' or 'other')
3. reason - reason to choose this school (nominal: close to

'home', school 'reputation', 'course' preference or 'other')

1. guardian - student's guardian (nominal: 'mother', 'father' or

'other')

1. traveltime - home to school travel time (numeric: 1 - 1 hour)
2. studytime - weekly study time (numeric: 1 - 10 hours)
3. failures - number of past class failures (numeric: n if 1<=n
4. schoolsup - extra educational support (binary: yes or no)
5. famsup - family educational support (binary: yes or no)
6. paid - extra paid classes within the course subject (Math or Portuguese) (binary: yes or no)
7. activities - extra-curricular activities (binary: yes or no)
8. nursery - attended nursery school (binary: yes or no)
9. higher - wants to take higher education (binary: yes or no) 22) internet - Internet access at home (binary: yes or no)
10. romantic - with a romantic relationship (binary: yes or no)
11. famrel - quality of family relationships (numeric: from 1 - very bad to 5 - excellent)
12. freetime - free time after school (numeric: from 1 - very low to 5 - very high)
13. goout - going out with friends (numeric: from 1 - very low to 5 - very high)
14. Dalc - workday alcohol consumption (numeric: from 1 - very low to 5 - very high)
15. Walc - weekend alcohol consumption (numeric: from 1 - very low to 5 - very high)
16. health - current health status (numeric: from 1 - very bad to 5 - very good)
17. absences - number of school absences (numeric: from 0 to 93)
18. G1 - first period grade (numeric: from 0 to 20)
19. G2 - second period grade (numeric: from 0 to 20)
20. G3 - final grade (numeric: from 0 to 20)

**Data Pre-processing:**

Both datasets are processed to check

1. Null Values
2. Duplicates &
3. Invalid values

but fortunately, there are no such irregularities in the dataset, implying that data is already clean and processed. Since both datasets have same set of attributes and has similar kind of data, they are merged vertically, so as to make the dataset effective and increase the dataset, this process supposedly key aspect of data pre-processing.

**Modification of Dataset:**

A new column called ‘FinalGrade’ is asserted which reflects ‘Grade3’ and a broader level view of ‘Grade3’. The column is inferred broadly as five categories under given conditions as follows:

* 'Excellent'[(data.G3 >= 18) & (data.G3 <= 20)] ● 'Good' [(data.G3 >= 15) & (data.G3 <= 17)]
* 'Satisfactory' [(data.G3 >= 11) & (data.G3 <= 14)]
* 'Poor' [(data.G3 >= 6) & (data.G3 <= 10)]
* 'Failure' [(data.G3 >= 0) & (data.G3 <= 5)]

New columns s\_no, m\_id and f\_id are made to form keys.

Feature Extraction:

Exploring and observing the data through visualizations resulted in some major conclusions, which made primary contribution to extract and structure important features.

And it is stored in a new file named **‘Features’**

1. Consider the schema alone and normalize it till BCBF using schema decomposition

FEATURES DATABASE

features(s\_no, school, sex, age, address, famsize, Pstatus, M\_id, Medu, F\_id, Fedu, Mjob, Fjob, reason, guardian, traveltime, studytime, failures, schoolsup, famsup, paid, activities, nursery, higher, internet, romantic, famrel, freetime, gout, Dalc, Walc, health, absences, G1, G2, G3, subject, FinalGrade, Regularity, Grade1, Grade2)

1NF – Identify Key, FDs

s\_no -> school, sex, age, address, famsize, Pstatus, reason, guardian, traveltime, studytime, failures, schoolsup, famsup, paid, activities, nursery, higher, internet, romantic, famrel, freetime, gout, Dalc, Walc, health, absences, G1, G2, G3, subject, Regularity

M\_id -> Medu, Mjob

F\_id -> Fedu, Fjob

G1 -> Grade1

G2 -> Grade2

G3 -> FinalGrade

Primary Key Selected: **sno, M\_id, F\_id**

Features is already in 1NF

|  |
| --- |
| **2NF – No Partial Dependency – If table has Partial Dependency decompose it.**  There is a partial dependency since:   1. Medu, Mjob depend on M\_id only 2. Fedu, Fjob depend on F\_id only 3. And rest of the attributes depend only on s\_no   Hence ‘features’ is not in 2NF and is decomposed to:  Students (**s\_no** (primary key), school, sex, age, address, famsize, Pstatus, reason, guardian, traveltime, studytime, failures, schoolsup, famsup, paid, activities, nursery, higher, internet, romantic, famrel, freetime, gout, Dalc, Walc, health, absences, G1, G2, G3, subject, Regularity, Grade1, Grade2, FinalGrade)  Mothers (**M\_id** (primary key), Medu, Mjob)  Fathers (**F\_id** (primary key), Fedu, Fjob)  Parents (**s\_no** (foreign key), **M\_id** (foreign key), **F\_id** (foreign key)) |

|  |
| --- |
| **3NF – There should be No Transitive Dependency – No Non key to Non key dependency, if present decompose.**  The tables Mothers, Fathers, Parents have no transitive dependency but Students table has partial dependency since (G1 -> Grade1, G2 -> Grade2, G3 -> FinalGrade) and G1, G2 and G3 are non-keys in the functional dependencies.  Hence students table needs to be decomposed.  Students (**s\_no** (primary key), school, sex, age, address, famsize, Pstatus, reason, guardian, traveltime, studytime, failures, schoolsup, famsup, paid, activities, nursery, higher, internet, romantic, famrel, freetime, gout, Dalc, Walc, health, absences, subject, Regularity)  Grades (**Grade** (primary key), Remark)  Stud\_Grades (**s\_no** (foreign key), **G1** (foreign key), **G2** (foreign key), **G3** (foreign key))  Where all G1, G2 and G3 refer to the same primary key of Grades table.  So now tables after normalised to 3NF, which also are in **BCNF** as well since all pks have only 1 attribute, are:  Students (**s\_no** (primary key), school, sex, age, address, famsize, Pstatus, reason, guardian, traveltime, studytime, failures, schoolsup, famsup, paid, activities, nursery, higher, internet, romantic, famrel, freetime, gout, Dalc, Walc, health, absences, subject, Regularity)  Grades (**Grade** (primary key), Remark)  Mothers (**M\_id** (primary key), Medu, Mjob)  Fathers (**F\_id** (primary key), Fedu, Fjob)  Parents (**s\_no** (foreign key), **M\_id** (foreign key), **F\_id** (foreign key))  Stud\_Grades (**s\_no** (foreign key), **G1** (foreign key), **G2** (foreign key), **G3** (foreign key)) |

1. Draw the ER diagram for the final decomposed schema stating the key attributes, mapping cardinalities, participation constraint, and so on

The final decomposed schema to the BCNF form is:

Students:

**s\_no** (primary key), school, sex, age, address, famsize, Pstatus, reason, guardian, traveltime, studytime, failures, schoolsup, famsup, paid, activities, nursery, higher, internet, romantic, famrel, freetime, gout, Dalc, Walc, health, absences, subject, Regularity

Grades:

**Grade** (primary key), Remark

Mothers:

**M\_id** (primary key), Medu, Mjob

Fathers:

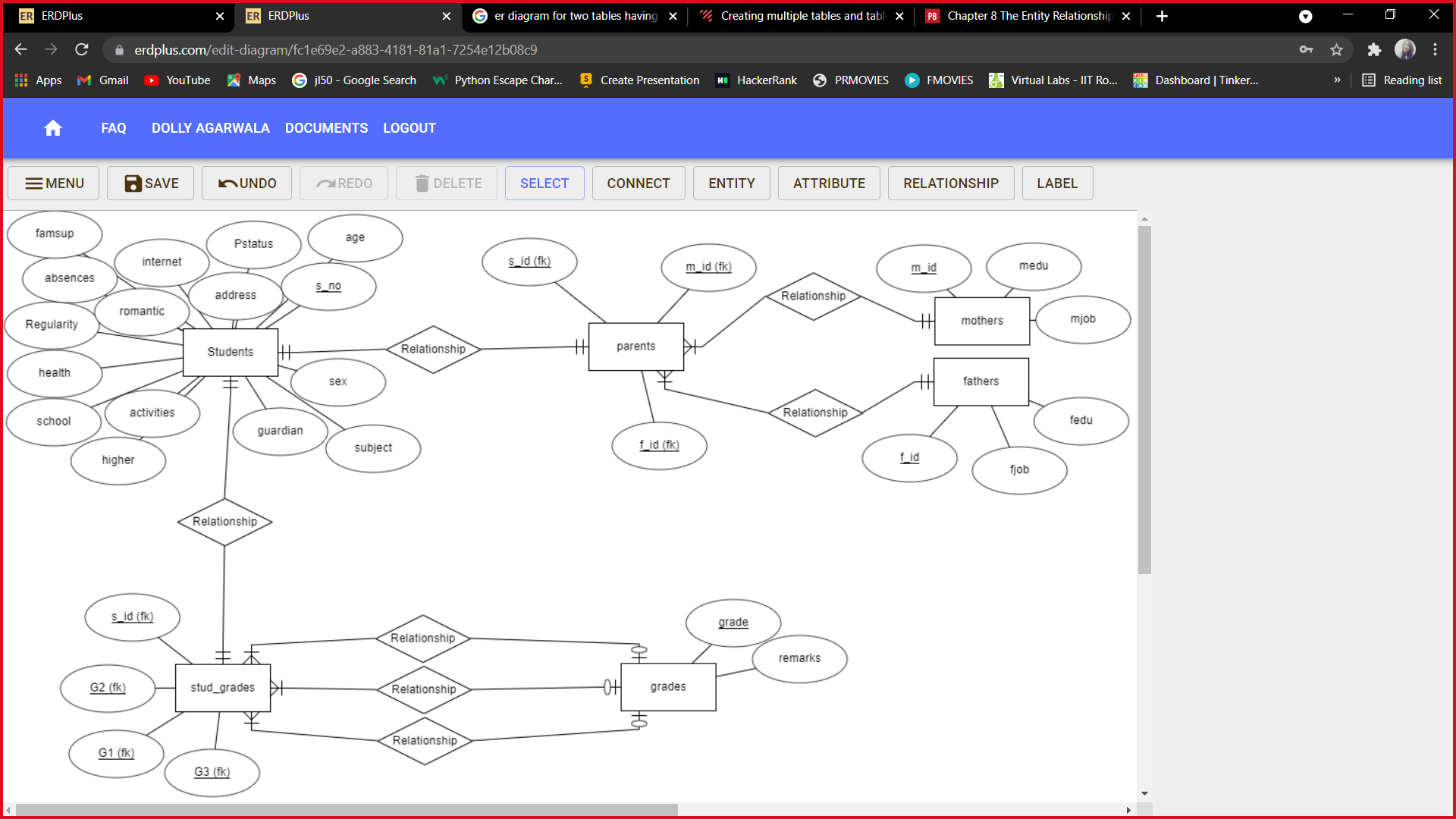
**F\_id** (primary key), Fedu, Fjob

Parents:

**s\_no** (foreign key), **M\_id** (foreign key), **F\_id** (foreign key)

Stud\_Grades:

**s\_no** (foreign key), **G1** (foreign key), **G2** (foreign key), **G3** (foreign key)

The ERD for the above schema:

There is **one-to-one** relationship and **total** participation between ‘*Students*’ and ‘*parents*’.

There is **one-to-one** relationship and **total** participation between ‘*mothers*’ and ‘*parents*’.

There is **one-to-one** relationship and **total** participation between ‘*fathers*’ and ‘*parents*’.

There is **one-to-one** relationship and **total** participation between ‘*students*’ and ‘*stud\_grades*’.

There are **three many-to-one** relationships between ‘*stud\_grades*’ and ‘*grades*’, where the ‘*grades*’ table have **partial** participation.

1. **Methodology and Algorithm used:**

This project presents main characteristics of data sets in concise form. It covers following two basic data mining techniques:

• REGRESSION: Regression analysis is a statistical method to model the relationship between a dependent (target) and independent (predictor) variables with one or more independent variables. More specifically, Regression analysis helps us to understand how the value of the dependent variable is changing corresponding to an independent variable when other independent variables are held fixed. It predicts continuous/real values such as temperature, age, salary, price, etc.

• PREDICTION: The word prediction in machine learning refers to the output of a trained model, representing the most likely value that will be obtained for a given input. Prediction in machine learning has a variety of applications, from chatbot development to recommendation systems. The model is trained with historical data, and then predicts a selected property of the data for new inputs. Prediction is used in lots of different areas, since it allows us to make highly accurate guesses about many things, such as predicting what the stock markets will do on any given day, predict results in sports, or even help the medical industry predict diseases. The algorithms for prediction are classified as supervised learning algorithms since they need a training dataset with correct examples to learn from them.

• CLASSIFICATION: As the name suggests, Classification is the task of “classifying things” into sub categories. But, by a machine. If that doesn’t sound like much, imagine your computer being able to differentiate between you and a stranger. Between a potato and a tomato. Between an A grade and a F. In Machine Learning and Statistics, Classification is the problem of identifying to which of a set of categories (sub populations), a new observation belongs to, on the basis of a training set of data containing observations and whose categories membership is known.

**Algorithms used: -**

i. LOGISTIC REGRESSION: Logistic regression is the appropriate regression analysis to conduct when the dependent variable is dichotomous (binary). Like all regression analyses, the logistic regression is a predictive analysis. Logistic regression is used to describe data and to explain the relationship between one dependent binary variable and one or more nominal, ordinal, interval or ratio-level independent variables.

Sometimes logistic regressions are difficult to interpret; the Intellects Statistics tool easily allows you to conduct the analysis, then in plain English interprets the output. Logistic regression measures the relationship between the categorical dependent variable and one or more independent variables by estimating probabilities using a logistic function, which is the cumulative distribution function of logistic distribution. Thus, it treats the same set of problems as probit regression using similar techniques, with the latter using a cumulative normal distribution curve instead. Equivalently, in the latent variable interpretations of these two methods, the first assumes a standard logistic distribution of errors and the second a standard normal distribution of errors.

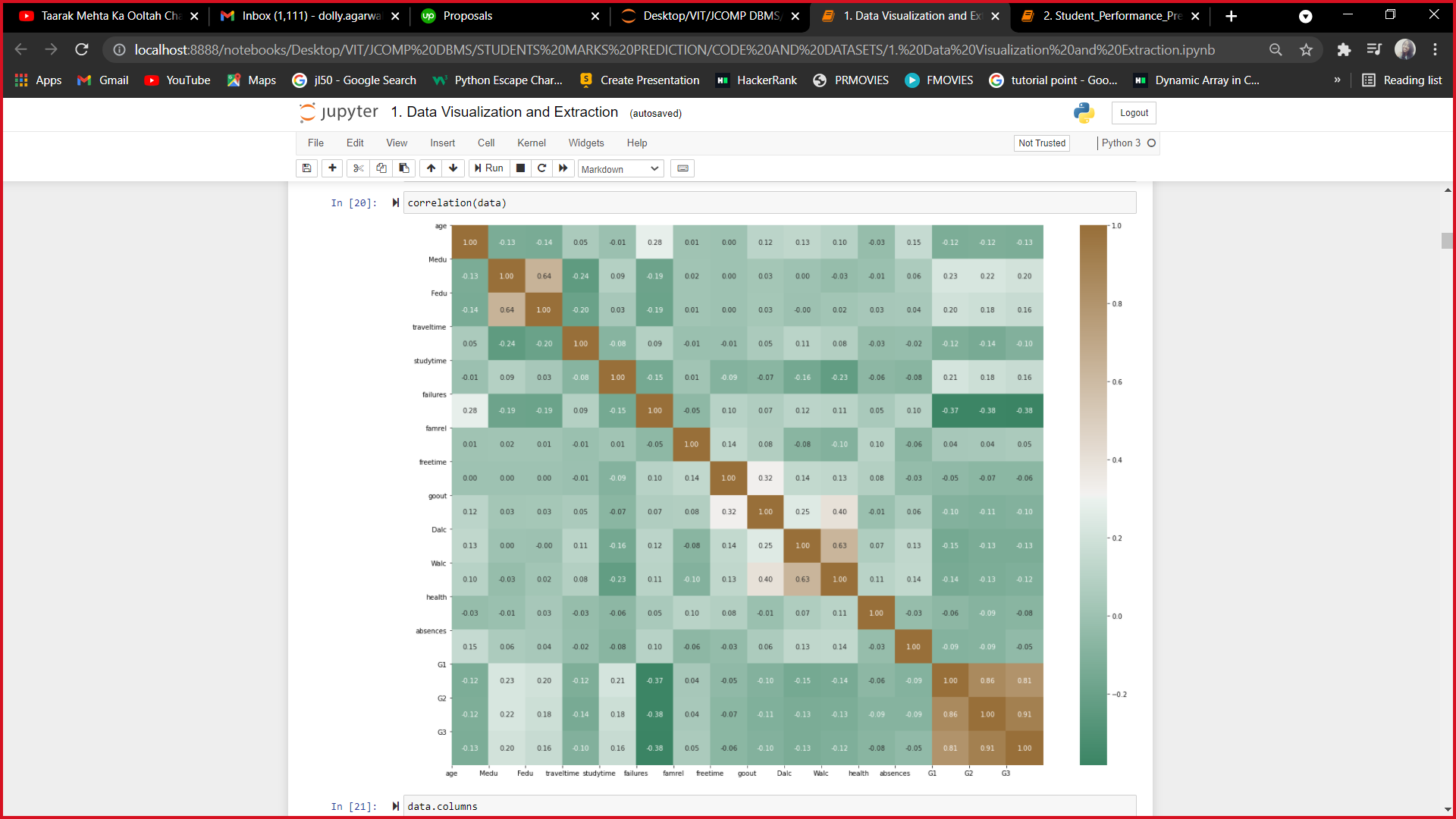
ii. XG BOOST: XGBoost is an open-source software library which provides the gradient boosting framework for C++, Java, Python, R, and Julia. It works on Linux, Windows, and macOS. From the project description, it aims to provide a "Scalable, Portable and Distributed Gradient Boosting (GBM, GBRT, GBDT) Library". Other than running on a single machine, it also supports the distributed processing frameworks Apache Hadoop, Apache Spark, and Apache Flink. It has gained much popularity and attention recently as it was the algorithm of choice for many winning teams of a number of machine learning competitions.

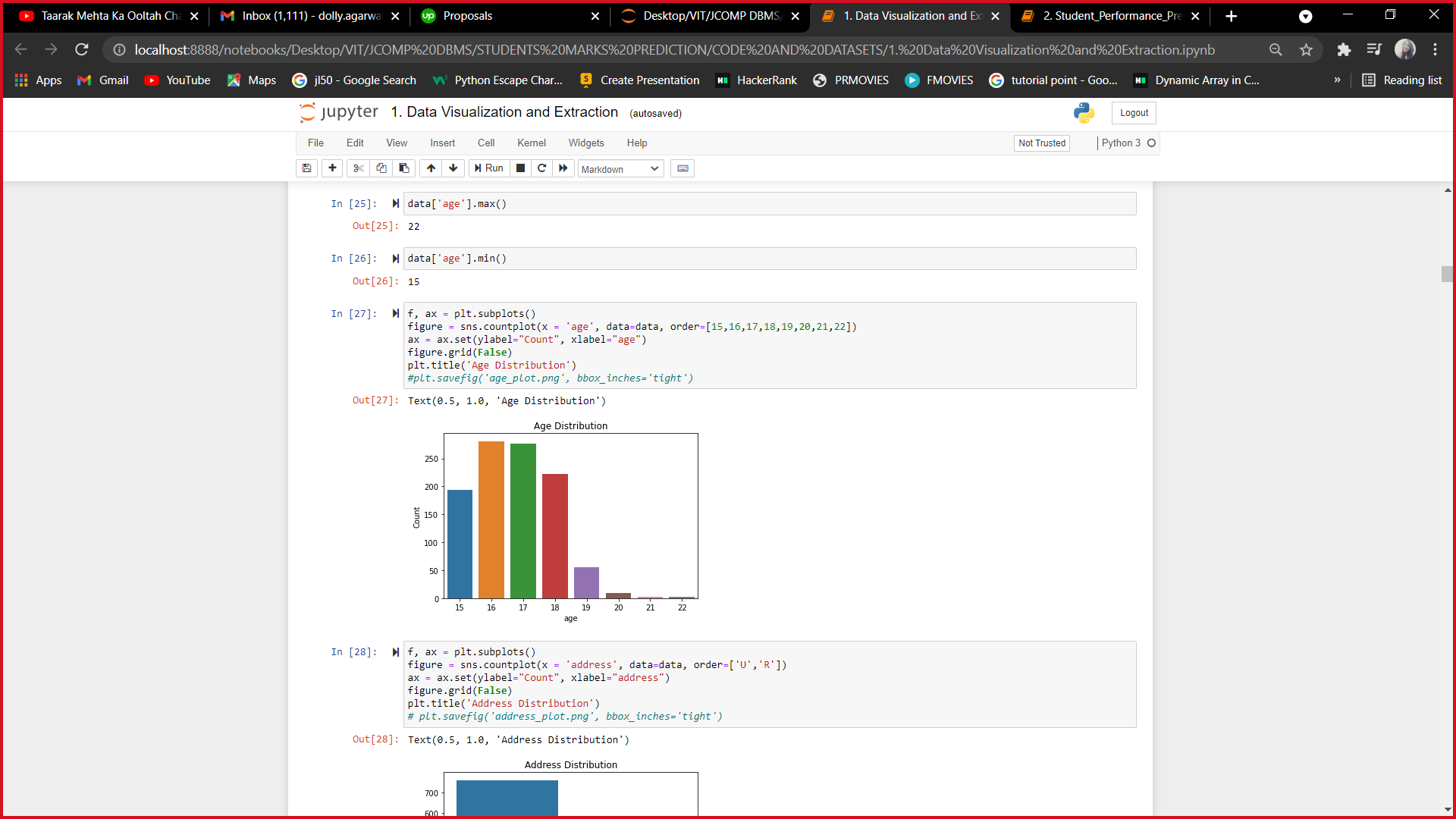
iii. SUPPORT VECTOR MACHINE: Support Vector Machine (SVM) is a supervised machine learning algorithm which can be used for both classification or regression challenges. However, it is mostly used in classification problems. In this algorithm, we plot each data item as a point in n dimensional space (where n is number of features you have) with the value of each feature being the value of a particular coordinate. Then, we perform classification by finding the hyper-plane that differentiate the two classes very well

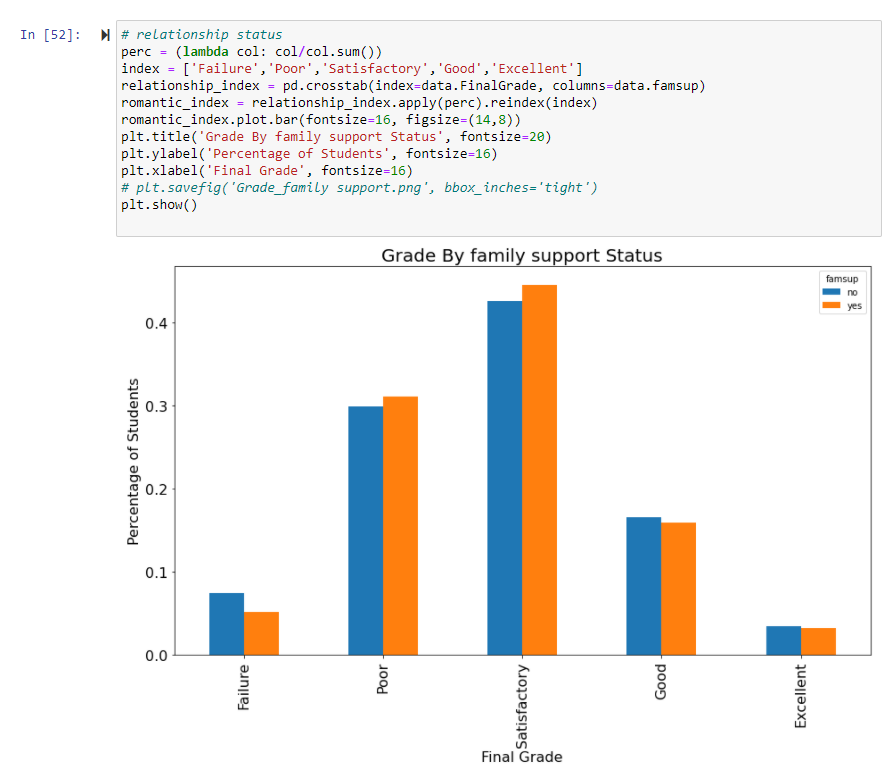
*After Data Visualisation and implementation of the given models we decide to use* ***XGBoost*** *model for our predictive analysis because it gives us the most accurate value for the Final Grade Prediction in accordance with our dataset.*

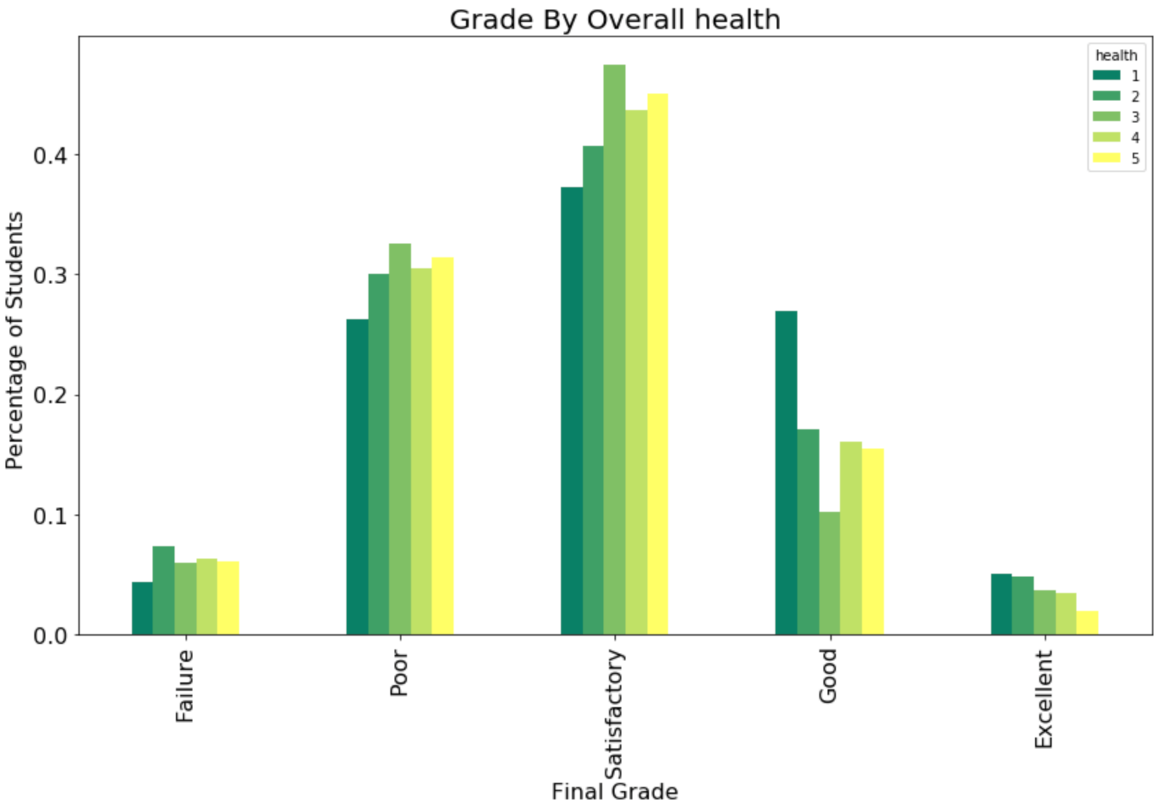
1. **Implementation**

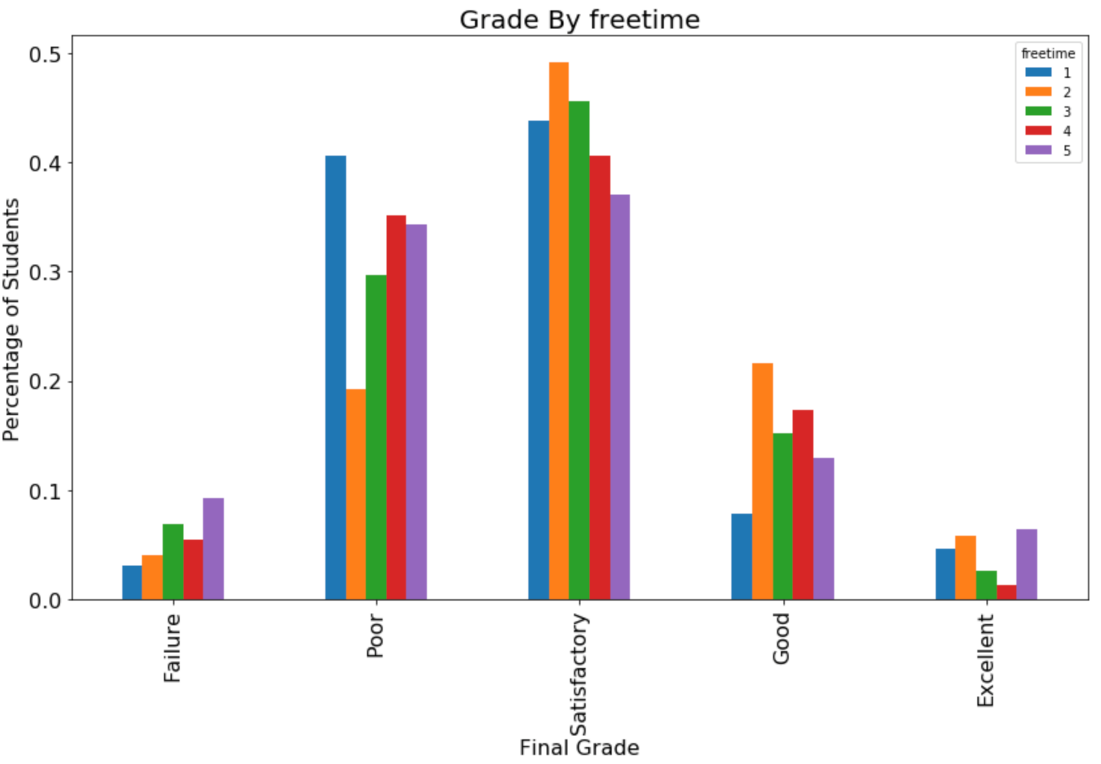
First, we visualize our data set to see which features impact our prediction the most. And here are few of the results of our data visualization.

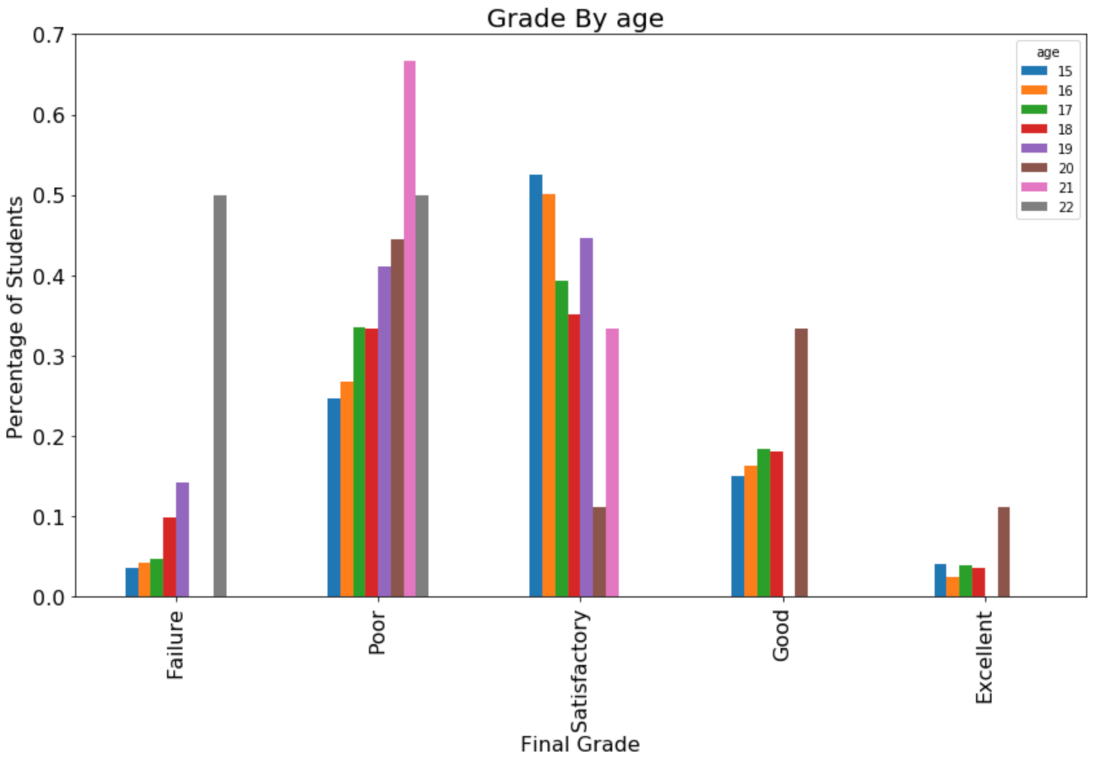


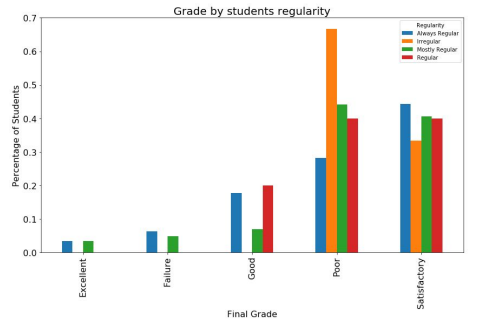


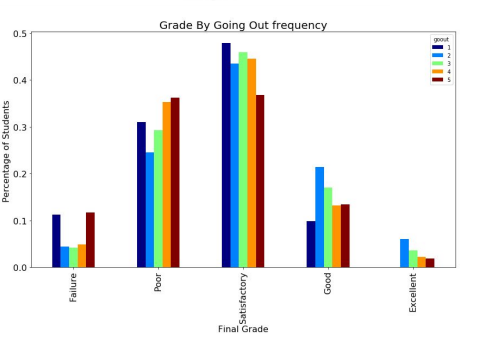


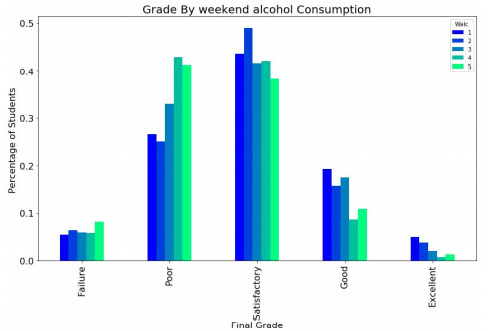






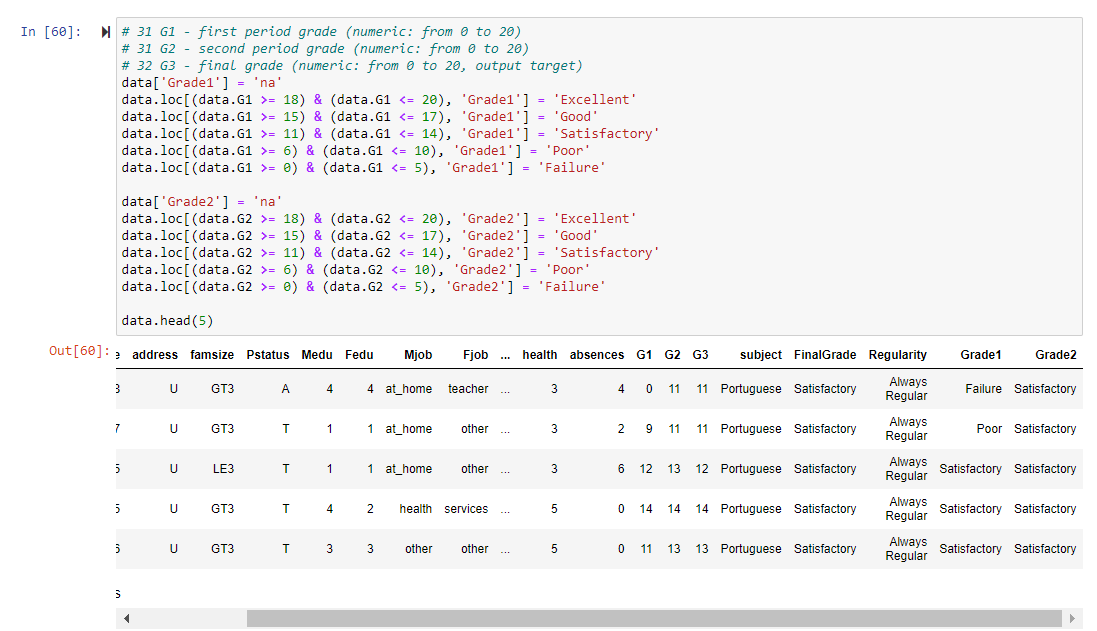






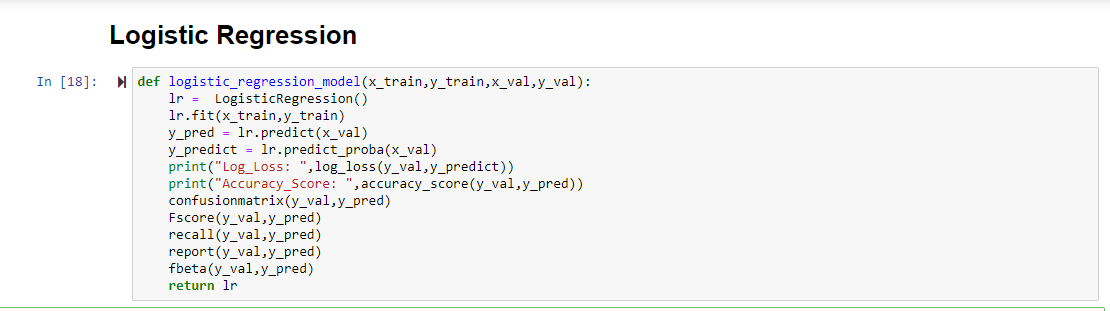
We add few columns like regularity grade1, grade2 and final grade. And we extract it in a file called features.csv



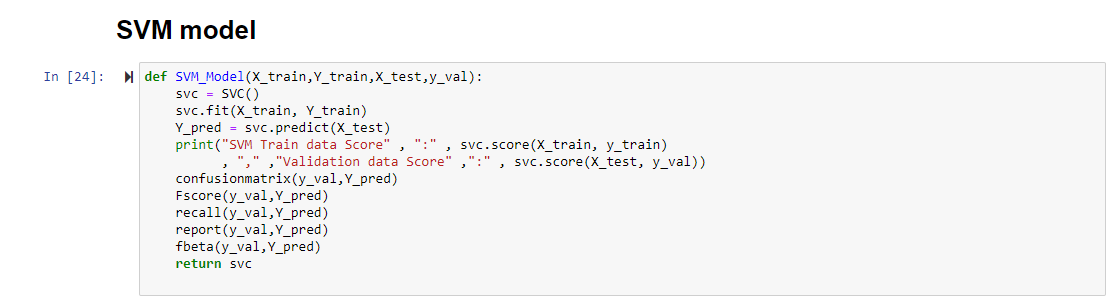


After extracting our dataset into features.csv we implement few models on it such as

* 1. Logistic Regression



* 1. SVM model



* 1. XGBoost Model

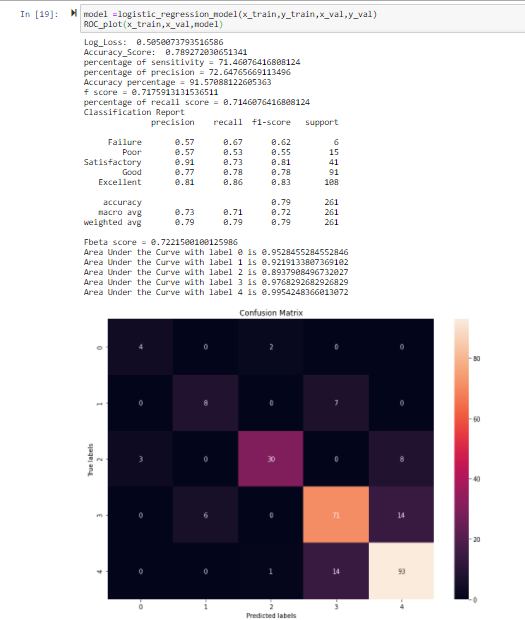


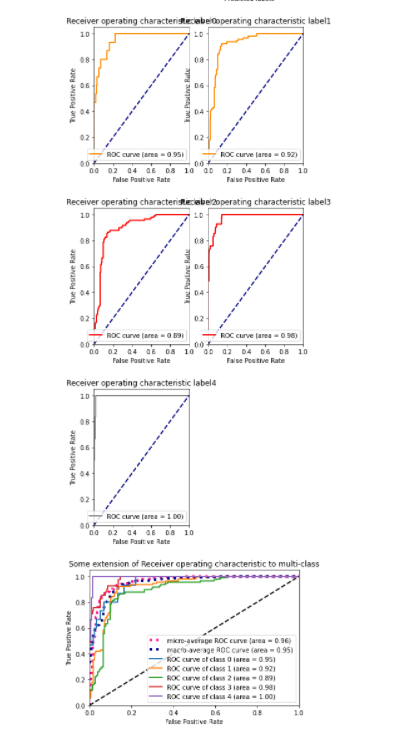
1. **Results and Discussion**

Here are all the results of the model we applied one by one to predict the final grade. The data is trained and tested with all three algorithms.

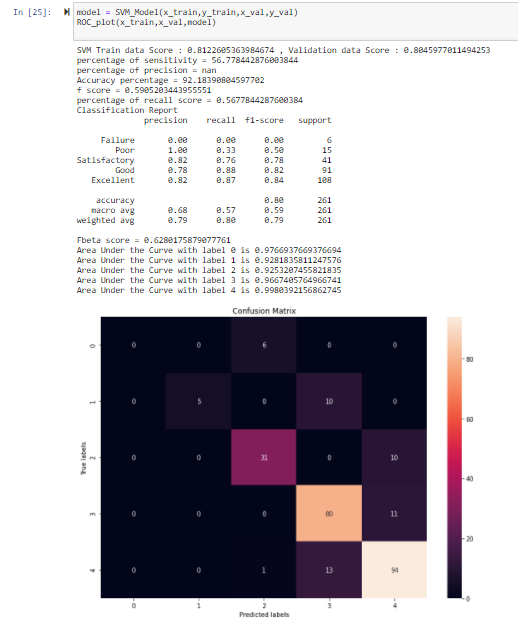
We have tried to find out accuracy score, accuracy percentage, fscore, classification report and ROC plot of each of these models to find out which model is best suited for our given data set to predict the final grade

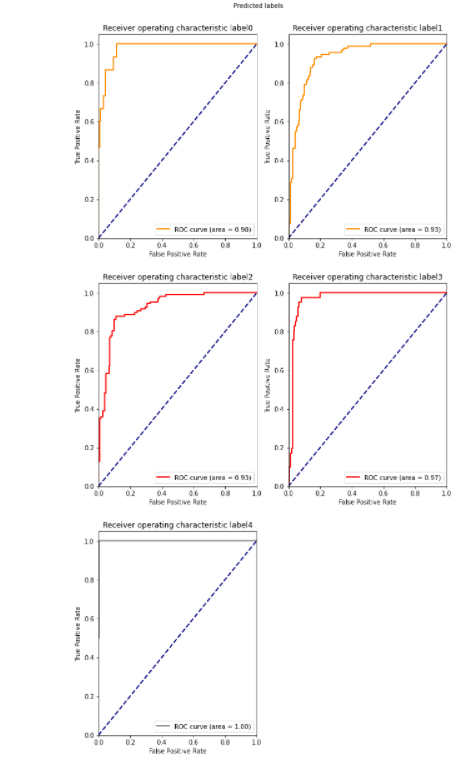
1. LOGISTICAL REGRESSION



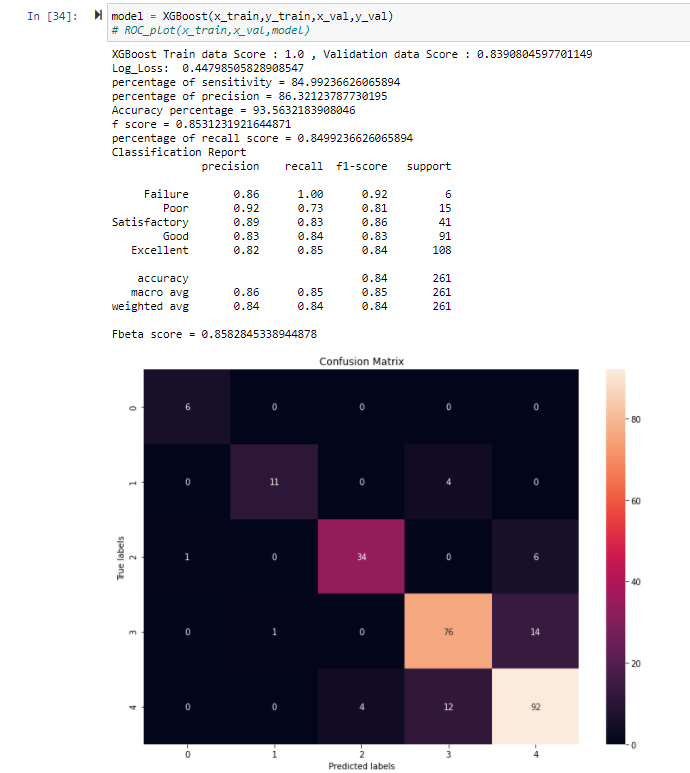


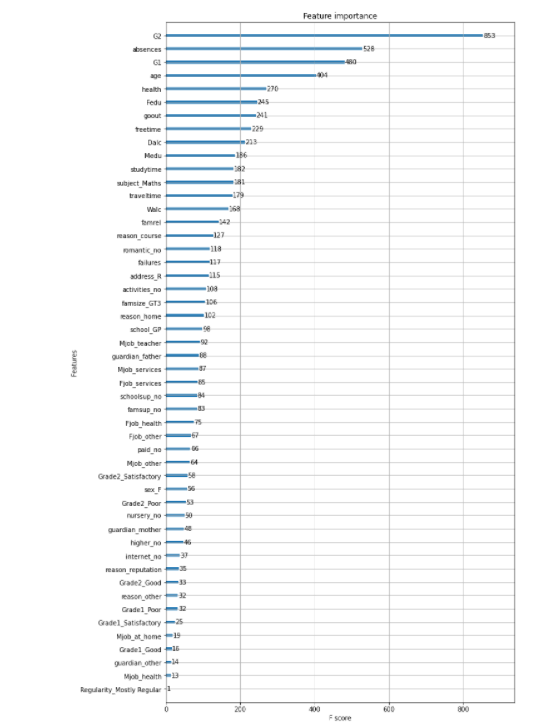
1. SVM MODEL





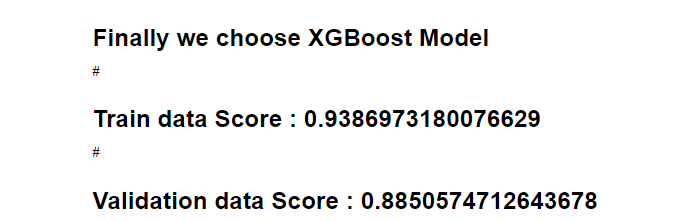
1. XGBOOST





We build a classification task using 3 informative features. Then we build a forest and compute the feature importances and then we plot it.

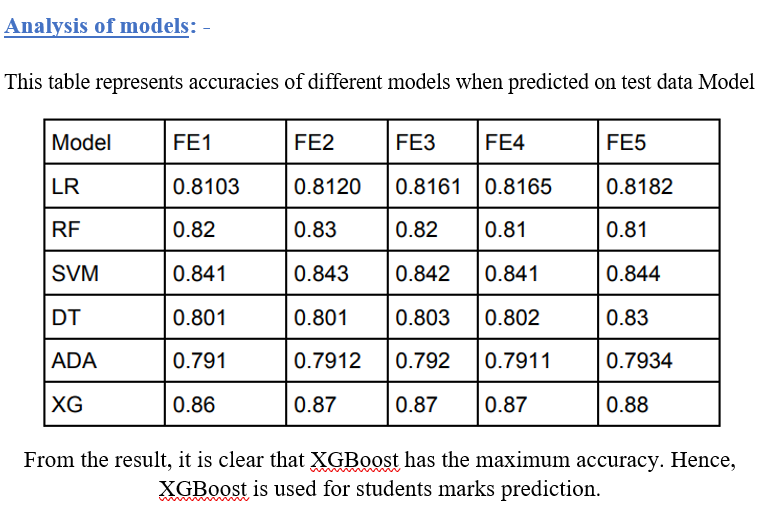




After analysing our results from above we can see that xgboost has a train data score of 0.9387 and validation score of 0.8851. So we choose xgboost model for our prediction of final grade.

1. **Conclusion**

All these results can be taken into consideration and even more reliable and more accurate algorithms can be used. Then the project will be more powerful to depend upon and even more efficient to depend upon. Here is the conclusion that we draw from our project for the dataset and the final model that we used:



From the result, it is clear that XGBoost has maximum accuracy. Hence it is the best suited classifier to predict the final grade for the given dataset.