VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

**School of Computer Science and Engineering**
VIT Chennai
Vandalur - Kelambakkam Road, Chennai - 600 127

# Review Report - 1

**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.

# 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)
9) Mjob - mother's job (nominal: 'teacher', 'health' care related, civil 'services' (e.g. administrative or police), 'at_home' or 'other')
10) Fjob - father's job (nominal: 'teacher', 'health' care related, civil 'services' (e.g. administrative or police), 'at_home' or 'other')
11) reason - reason to choose this school (nominal: close to 'home', school 'reputation', 'course' preference or 'other')
12) guardian - student's guardian (nominal: 'mother', 'father' or 'other')
13) traveltime - home to school travel time (numeric: 1 - 1 hour)
14) studytime - weekly study time (numeric: 1 - 10 hours)
15) failures - number of past class failures (numeric: n if 1<=n
16) schoolsup - extra educational support (binary: yes or no)
17) famsup - family educational support (binary: yes or no)
18) paid - extra paid classes within the course subject (Math or Portuguese) (binary: yes or no)
19) activities - extra-curricular activities (binary: yes or no)
20) nursery - attended nursery school (binary: yes or no)
21) higher - wants to take higher education (binary: yes or no)
22) internet - Internet access at home (binary: yes or no)
23) romantic - with a romantic relationship (binary: yes or no)
24) famrel - quality of family relationships (numeric: from 1 - very bad to 5 - excellent)
25) freetime - free time after school (numeric: from 1 - very low to 5 - very high)
26) goout - going out with friends (numeric: from 1 - very low to 5 - very high)
27) Dalc - workday alcohol consumption (numeric: from 1 - very low to 5 - very high)
28) Walc - weekend alcohol consumption (numeric: from 1 - very low to 5 - very high)
29) health - current health status (numeric: from 1 - very bad to 5 - very good)
30) absences - number of school absences (numeric: from 0 to 93)
31) G1 - first period grade (numeric: from 0 to 20)
32) G2 - second period grade (numeric: from 0 to 20)

33) 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, the both datasets 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)]

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'**.

# DATA VISUALISATION:

So major conclusions from visualizations are-:

● 'G1' and 'G2' play a vital role in prediction of 'FinalGrade', so structuring these columns would help.

● 'absences' (number of absent days) has good weightage among features to decide the prediction.Student Grade Prediction

- Information about Students whose father is working at home has least importance as per data analysis.
- Parental status information has negligible influence or effect on 'FinalGrade' prediction

Following feature extraction is done as part of implementing the results from above conclusions: -

- New features called 'Grade1' and 'Grade2' is included in feature set, to add weight to the significance of 'G1' ,'G2' columns.
- Similarly, a new feature called 'Requirement' is extracted out of 'absences' column to subsequently add importance to the attribute. Similarly

Following deductions are made on feature set: -

- ''Fjob_teacher' feature dropped from feature set.
- 'Pstatus_A' is removed from feature set.
- 'Pstatus_T' is dropped from feature set.

because their presence is not making any significant difference, in fact their absence making the model more effective and less sensitive.

## **MODEL BUILDING**:

So as per our analysis of data, our choices of model are-:

- Logistic Regression.
- Decision tree.
- Random forest.
- XG Boost.
- Support vector Machine (SVM)
- Ada boost

### 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. DECISION TREES:

Tree based learning algorithms are considered to be one of the best and mostly used supervised learning methods. Tree based methods empower predictive models with high accuracy, stability and ease of interpretation. Unlike linear models, they map nonlinear relationships quite well. They are adaptable at solving any kind of problem at hand (classification or regression). Methods like decision trees, random forest, gradient boosting are being popularly used in all kinds of data science problems. Hence, for every analyst it’s important to learn these algorithms and use them for modelling. Decision tree is a type of supervised learning algorithm (having a pre-defined target variable) that is mostly used in classification problems. It works for both categorical and continuous input and output variables. In this technique, we split the population or sample into two or more homogeneous sets (or sub-populations) based on most significant splitter / differentiator in input variables.

### iii. RANDOM FORESTS:

Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks that operates by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean/average prediction (regression) of the individual trees. Random decision forests correct for decision trees' habit of overfitting to their set. Random forests generally outperform decision trees, but their accuracy is lower than gradient boosted trees. However, data characteristics can affect their performance.

Random forest classifier creates a set of decision trees from randomly selected subset of training set. It then aggregates the votes from different decision trees to decide the final class of the test object. Basic parameters to Random Forest Classifier can be total number of trees to be generated and decision tree related parameters like minimum split, split criteria etc.

### iv. XG BOOST:

XG Boost 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.

### v. 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

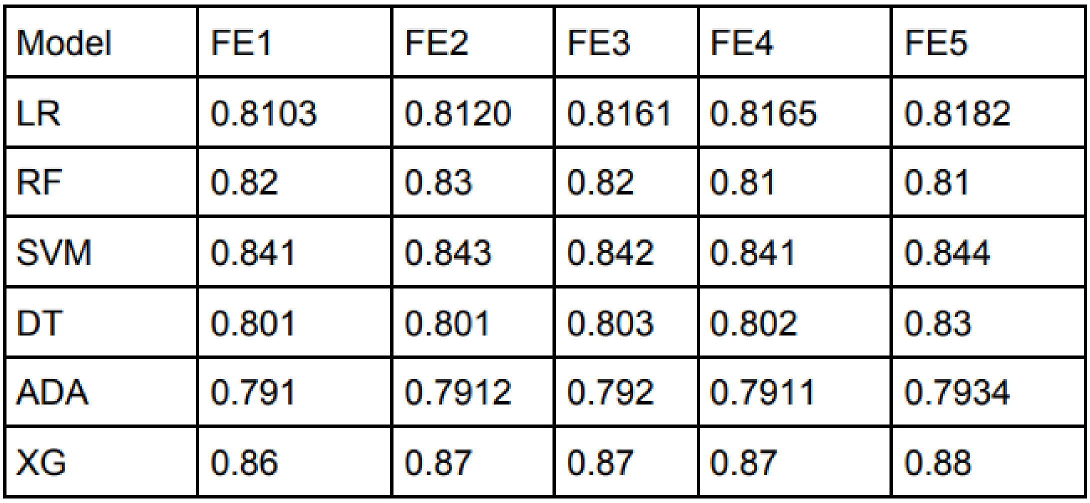
### vi. ADABOOST:

AdaBoost algorithm**,** short for Adaptive Boosting, is a Boosting technique that is used as an Ensemble Method in Machine Learning. It is called Adaptive Boosting as the weights are re-assigned to each instance, with higher weights to incorrectly classified instances. Boosting is used to reduce bias as well as the variance for supervised learning. It works on the principle where learners are grown sequentially. Except for the first, each subsequent learner is grown from previously grown learners. In simple words, weak learners are converted into strong ones. Adaboost algorithm also works on the same principle as boosting, but there is a slight difference in working. Let’s discuss the difference in detail.

# RESULT

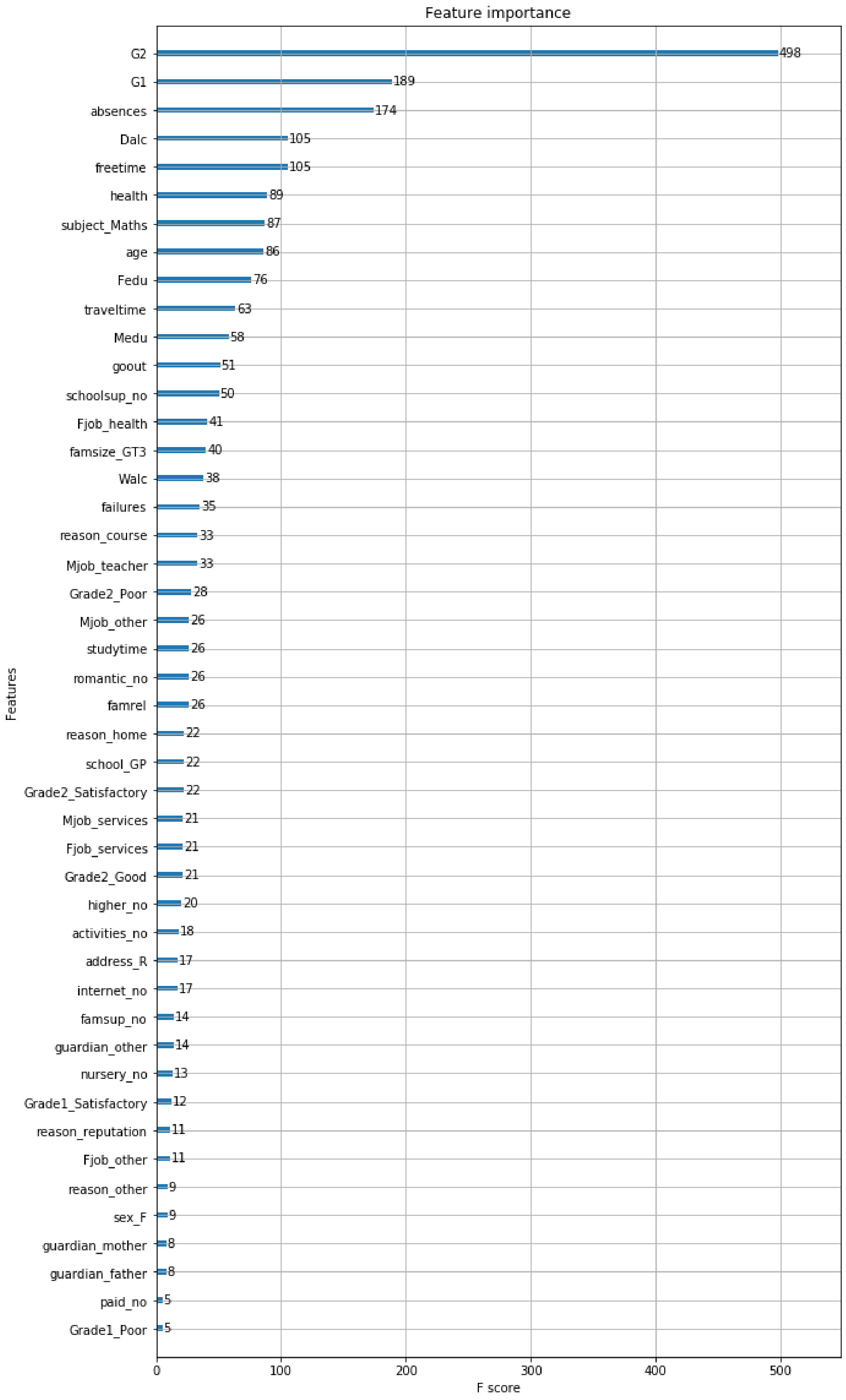
**Analysis of models: -**

This table represents accuracies of different models when predicted on test data Model



| Model | FE1 | FE2 | FE3 | FE4 | FE5 |
|---|---|---|---|---|---|
| LR | 0.8103 | 0.8120 | 0.8161 | 0.8165 | 0.8182 |
| RF | 0.82 | 0.83 | 0.82 | 0.81 | 0.81 |
| SVM | 0.841 | 0.843 | 0.842 | 0.841 | 0.844 |
| DT | 0.801 | 0.801 | 0.803 | 0.802 | 0.83 |
| ADA | 0.791 | 0.7912 | 0.792 | 0.7911 | 0.7934 |
| XG | 0.86 | 0.87 | 0.87 | 0.87 | 0.88 |

From the result, it is clear that XGBoost has the maximum accuracy. Hence, XGBoost is used for students marks prediction.


Feature importance
Features
F score
0
100
200
300
400
500
G2 498
G1 189
absences 174
Dalc 105
freetime 105
health 89
subject_Maths 87
age 86
Fedu 76
traveltime 63
Medu 58
goout 51
schoolsup_no 50
Fjob_health 41
famsize_GT3 40
Walc 38
failures 35
reason_course 33
Mjob_teacher 33
Grade2_Poor 28
Mjob_other 26
studytime 26
romantic_no 26
famrel 26
reason_home 22
school_GP 22
Grade2_Satisfactory 22
Mjob_services 21
Fjob_services 21
Grade2_Good 21
higher_no 20
activities_no 18
address_R 17
internet_no 17
famsup_no 14
guardian_other 14
nursery_no 13
Grade1_Satisfactory 12
reason_reputation 11
Fjob_other 11
reason_other 9
sex_F 9
guardian_mother 8
guardian_father 8
paid_no 5
Grade1_Poor 5

The plot of ‘Feature importance’ shows that the most important feature in determining G3 is G2

The following are different iterations over feature sets obtained by feature engineering:-

FE1: Contains all features

FE2: removing column name “Fjob_at_home”

FE3: also removing column name “Fjob_teacher”

FE4: also removing column name “Pstatus_A”

FE5: also removing column name “Pstatus_T”

So as per observation over above models and it is clear that xg-boost dominates all other models and this model is more generalized based on our observations over train score and test score which has small difference between their predictions.