APPLICATION OF ENSEMBLE LEARNING TECHNIQUES

FOR PREDICTION MODELS

A DISSERTATION

SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR

THE AWARD OF DEGREE

OF

MASTER OF TECHNOLOGY IN

SOFTWARE ENGINEERING

SUBMITTED BY:

SUMAN NANDI

2K20/SWE/23

UNDER THE SUPERVISION OF

Dr. RUCHIKA MALHOTRA
HEAD OF DEPARTMENT
DEPARTMENT OF SOFTWARE ENGINEERING
DELHI TECHNOLOGICAL UNIVERSITY

DEPARTMENT OF SOFTWARE ENGINEERING

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Bawana Road, Delhi-110042

MAY 2022

DEPARTMENT OF SOFTWARE ENGINEERING

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Bawana Road, Delhi-110042

CANDIDATE'S DECLARATION

I, Suman Nandi, Roll No. 2K20/SWE/23 student of MTech (Software Engineering), hereby declare that the project Dissertation titled "Application of Ensemble Learning Techniques for Prediction Models" which is submitted by me to the Department of Software Engineering, Delhi Technological University, Delhi in partial fulfillment of the requirement for the award of the degree of Master of Technology, is original and not copied from any source without proper citation. This work has not previously formed the basis for the award of and Degree, Diploma Associateship, Fellowship or other similar title or recognition.

S. Nandi

Place: Delhi

Suman Nandi

Date:

2K20/SWE/23

DEPARTMENT OF SOFTWARE ENGINEERING

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Bawana Road, Delhi-110042

CERTIFICATE

I hereby certify that the Project Dissertation titled **"Application of Ensemble Learning Techniques for Prediction Models"** which is submitted by Suman Nandi, 2K20/SWE/23 Department of Software Engineering, Delhi Technological University, Delhi in partial fulfillment of the requirement for the award of the degree of Master of Technology, is a record of the project work carried out by the students under my supervision. To the best of my knowledge, this work has not been submitted in part or full for any Degree or Diploma to this University or elsewhere.

Ruchika Malhotra

Place: Delhi

Date:

Dr. RUCHIKA MALHOTRA

SUPERVISOR

HEAD OF DEPARTMENT

Department of Software Engineering,

Delhi Technological University

(Formerly Delhi College of Engineering)

Bawana Road, Delhi - 110042

ACKNOWLEDGEMENT

The success of Major project II requires the assistance and input of numerous people and the organization. I am grateful to everyone who helped in shaping the result of the project.

I express my sincere thanks to **Dr. Ruchika Malhotra**, my project guide and mentor, for providing the opportunity to undertake this project under her guidance. Her constant support and encouragement have made me realize that it is the process of learning which weighs more than the end result. I am highly indebted to the panel faculties during all the progress evaluations for their guidance, constant supervision and formotivating me to complete my work. They helped me throughout with new ideas, provided information necessary and pushed me to complete the work.

I also thank all my fellow students and my family for their continued support.

S. Nandi

Suman Nandi
2k20/SWE/23

ABSTRACT

With the exponential proliferation of malware, it has become a big concern in our everyday lives, which are largely reliant on computers running a variety of different types of software to function properly. Malware authors produce dangerous software by inventing new variations, new innovations, new infections, and more obfuscated malware through the use of tactics such as packaging and encrypting techniques, amongst other methods. Malicious software categorization and detection are critical components of cyber security research, and they represent a significant problem. Because of the rising number of false alarms, proper categorization and detection of malware has become a major issue that must be addressed in the near future. In this study, eight malware families were identified and classified according to their family members. The research presents four feature selection techniques for use in multiclass classification problems, each of which is designed to choose the best feature. Then the top 100 characteristics of these algorithms are picked for performance assessments and they are found. In order to determine the best models, five machine learning methods are compared. Then, using the feature ranking of the best model, the frequency distribution of features is determined. Finally, it is stated that the frequency distribution of each character in an API call sequence may be utilized to classify malware families.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

1. SVM: Support Vector Machine
2. KDD: Knowledge Discovery in Databases
3. EDM: Educational Data Mining
4. CBE: Computer Based Education
5. DM: Data Mining
6. ML: Machine Learning
7. LA: Learning Analytics
8. CSEDM: Computer Science Education in Data Mining

CHAPTER 1

INTRODUCTION TO ML

1.1. SUPERVISED LEARNING

Supervised learning is a type of machine learning, where the models are trained using labelled data. By labelled data we mean that the output is given beforehand, and we are just trying to divide the entire dataset into training set and test set and thereafter we are calculating the accuracy of the various machine learning models on the test data. There are two main types of Supervised learning. They are classification and regression.

Classification is used when we have a categorial labelled dataset. By categorial here means that the final output consists of output in the form of binary values. For example: The target variable has output as True and False, Yes and No etc.

Regression is a part of supervised machine learning algorithm and we use Regression when there is a continuous relationship between input variable and output variable. There are many kinds of Regression algorithms. Some of the examples of Regression algorithms are Linear Regression, Naïve Bayes etc. We use regression when we deal with datasets related to weather forecasting, trends in Market etc.

LOGISTIC REGRESSION

Logistic Regression is a type of Supervised machine learning algorithm and is used when the outcome has binary values such as true or false, win or lose etc. It anticipates the final outcome based on the prior observations of the records of the dataset. It also helps in data preparation by making the records of the dataset fall into specifically predefined buckets during the extract, transform and load process.

DECISION TREE

Decision Tree is a kind of Logistic Regression Machine Learning Algorithm and is used when the outcome has binary values such as true or false, win or lose etc. We Represent the features of the dataset by the help of internal nodes, Decision rules are represented by the branches whereas the outcome is represented in the leaves. The decision tree uses recursive partitioning for splitting the main datasets into various subsets. We don’t need any parameter setting or domain knowledge for the construction of the decision tree classifier.

Decision Tree is a supervised learning method that can be used for both planning and retrospective problems, but is often preferred in solving planning problems. It is a tree-shaped divider, where the internal nodes represent the elements of the database, the branches represent the rules of decision and each leaf node represents the result. In the Decision Tree, there are two nodes, namely the Resolution Node and Leaf Node. Decision Nodes are used to make any decision and have many branches, while Leaf nodes are the result of those decisions and have no other branches. Decisions or tests are made on the basis of the characteristics of the data provided. It is called a pruning tree because it resembles a tree, it begins with a root node, extends to the upper branches and forms a tree-like structure.

1.2. UNSUPERVISED LEARNING

Unsupervised machine Learning has a totally different goal towards the fabrication of the dataset. The goal of the machine learning is to find the similar structures as well as hidden patterns from the given input dataset. Here only the input data will be provided to the Unsupervised Machine Learning Model, and it the task of the model to find hidden patterns and useful insights by showing the results.

They are mainly of two types: Clustering and Association

Clustering is a type of Unsupervised machine Learning Model, where it groups the data showing similar traits into clusters. So, there will be two main groups in total. One group where the data features showing similarity are collected together, whereas on the other hand, the datasets having less or no similarities are grouped together.

Association is a rule-primarily based technique for locating relationships among variables in a given dataset. those strategies are regularly used for market basket evaluation, allowing organizations to better recognize relationships among one-of-a-kind products. know-how consumption conduct of clients enables organizations to broaden better recommendation engines and cross-selling strategies. Examples of this may be seen in Amazon’s “clients Who bought This item also sold” or Spotify’s "find out Weekly" playlist. while there are some distinct algorithms used to generate association guidelines, which include many algorithms such as Eclat, Apriori and FP-Boom.

1.3. ENSEMBLE TECHNIQUES USED

This is the process by which we integrate various machine learning models to solve a particular problem of computer intelligence. It is mainly used to improve the performance of the model and to reduce the chances of choosing the wrong fixed feature. Mendis Moreira et al. gave the Ensemble study description, "Collaborative learning is a process that uses a set of models, each of which is based on a learning process in a particular problem. This set of models (ensemble) is integrated in some way to obtain a final prediction."

There are 2 types of ensemble methods: -

- **Sequential** - Base learners are generated sequentially.
- **Parallel** - Base learners are generated simultaneously or in parallel manner.

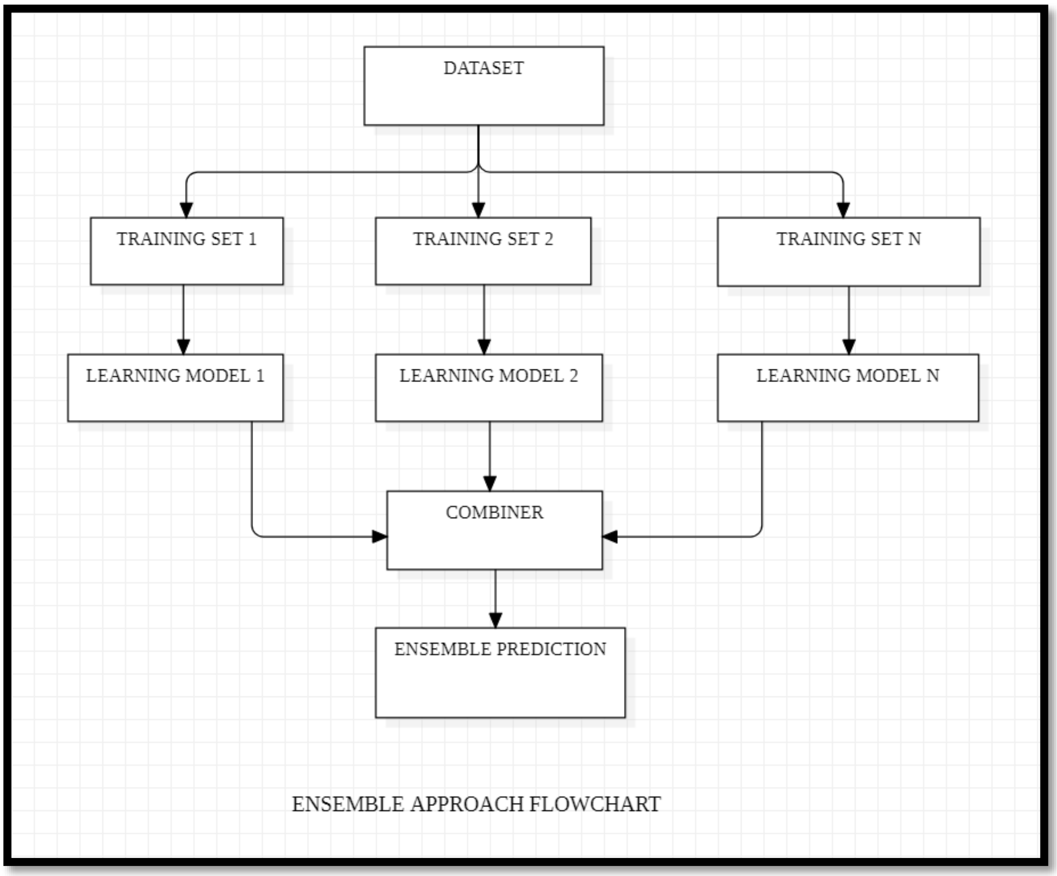


Fig 1.1. Ensemble Approach Flowchart

There are three main kinds of Ensemble techniques that are used in this project. They are:-

- **Bagging:** Another name for Bagging is Bootstrap Aggregation. It is used to minimize variations in data having noisy values. In these methods the random data sample is selected by rotation which means that individual data points can be selected more than once. After the production of several data points these weak models are then trained independently and depending on the type of operation whether retrieval or separation, most of these predictions produce the most accurate result.

- **Boosting:** This process of integration attempts to create a strong divider by combining a few weak dividers. Initially build a model from training data. Now build the following model by correcting the existing errors in the first model. This process of construction model continues until a high number of models are added or complete training data is accurately predicted.

- **Stacking:** It is considered a weak student and comes under the Parallel ensemble learning process. Includes student meta to generate predictions based on different weak student predictions. The meta reader is trying to learn how to better integrate input data to make a better output prediction.

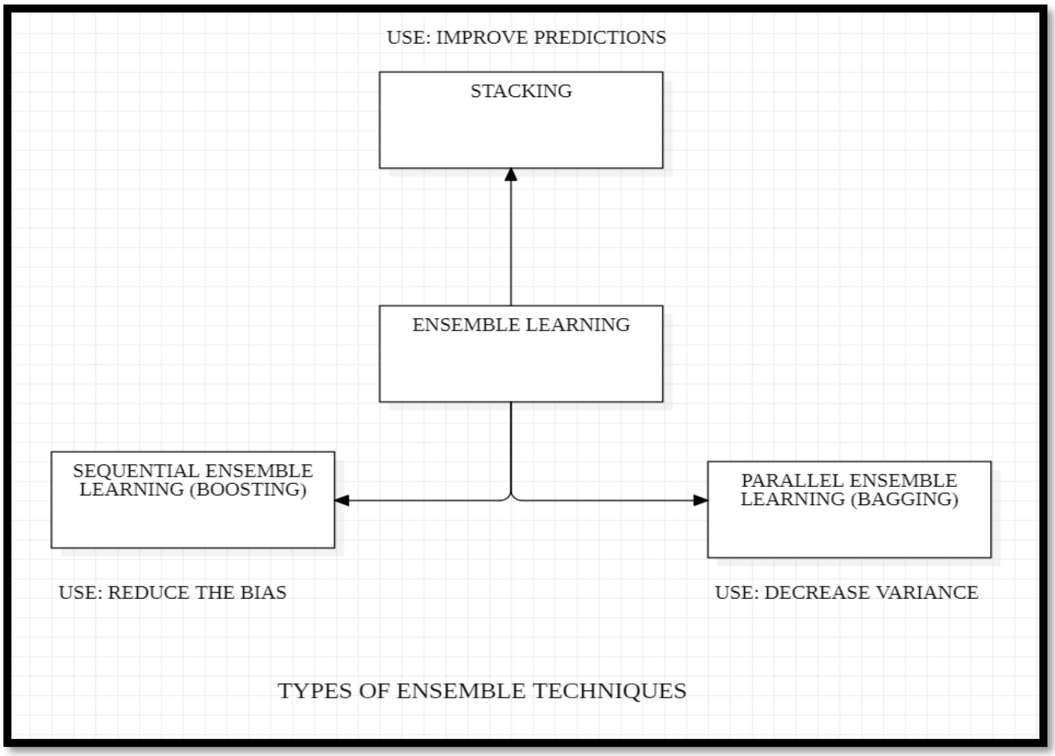

USE: IMPROVE PREDICTIONS
STACKING
ENSEMBLE LEARNING
SEQUENTIAL ENSEMBLE LEARNING (BOOSTING)
PARALLEL ENSEMBLE LEARNING (BAGGING)
USE: REDUCE THE BIAS
USE: DECREASE VARIANCE
TYPES OF ENSEMBLE TECHNIQUES

Fig 1.2. Types of Ensemble Techniques

CHAPTER 2

PRIOR WORK

2.1. INTRODUCTION

In the discipline of data mining, or in Knowledge Discovery of Databases (KDD), new and potentially usable information is discovered from big datasets (1999). To account for (and take use of) educational data's inherent non-independence and multi-level hierarchy, it has been suggested that educational data mining methods typically differ from ordinary data mining approaches. As a result, educational data mining publications are increasingly turning to models taken from the psychometrics field. [1]

In educational research and teaching practice, educational data mining technology combines the ideas and techniques of education, computer science, psychology, and statistics to address issues. EDM technology may help educators and students learn more effectively by evaluating and mining educational data. It can aid management in making choices, instructors in refining their courses, and students in learning more effectively. The multidisciplinary aspect of EDM in its data sources, data features, research methods, and application aims demonstrates its distinctiveness in terms of educational difficulties

In the last several years, both education and information have undergone a radical shift. EDM research has been bolstered by a wide range of online learning platforms, mobile apps, and social networks. When it comes to online education, MOODLE [3] is a good example. More than 60 million students and instructors have benefited from its services as of 2013[4]. In June 2012, there were more than 1 billion people using smartphones [5] over the world, while more than 2.2 billion people were using Facebook. A person [6] MOOCs, or massive open online courses, are a relatively recent development in the world of education. On Coursera, more than 10 million people had signed up for MOOCs by year-end 2014. EDM is, of course, also living in the "big data" era. Due of its unique context, EDM research is expected to grow quickly in the next years.

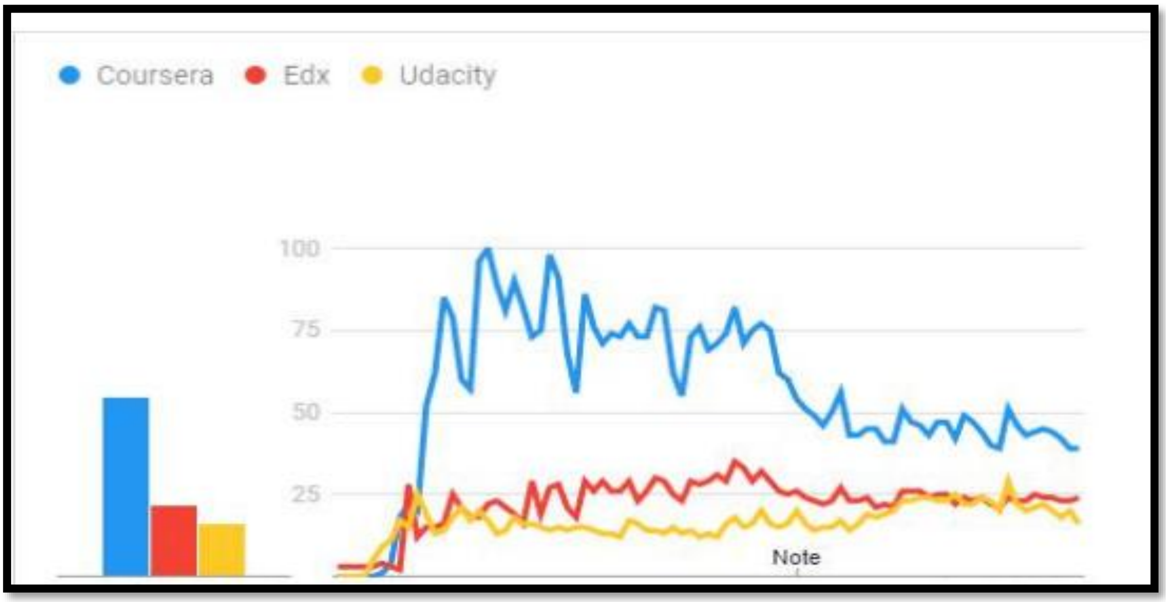


Fig 2.1. The E-Learning Trend

Teachers may use EDM and LA data analysis to gain a deeper understanding of their students and their learning processes, and then use that information to determine the most effective teaching techniques and instructional sequences for a variety of pupils. Using the data from a single course, we want to create an EDM statistical model that can illustrate how well students are doing and where they may improve. Instructors can also benefit from this research, which helps them alter their course schedules in time.

2.2. BACKGROUND INFORMATION

2.2.1. Educational Data Mining

Machine learning and Data mining (ML & DM), computer-based education (CBE), and learning analytics (LA) are the most closely connected fields to EDM. The interaction between each two areas has developed DM&ML, CBE, and LA (LA). It is possible to see the features of EDM by comparing it to these three fields.

When comparing EDM research to those conducted under the auspices of DM&ML, these are some of the key differences to keep in mind. Education, psychology, and sociology ideas and approaches are commonly used to analyze EDM data. For example, pedagogy, learning experiences, teaching evaluations, and motives all play a role in EDM data. Researchers need to be able to grasp the ideas and methods used to measure and evaluate this sort of data.

Multi-level: Structure of educational institutions and instructional materials is the source of EDM data's multi-level nature If pupils are arranged by school district or class or course or chapter or knowledge point, then the teaching content may be structured in the same way.

EDM data is often multi-precision since it comprises time scales. The length of a teaching study might range from a few months to a few decades, or it can be documented to the millisecond. As a result, data may be analysed with varying degrees of precision in terms of time. EDM data has the ability to be used in a variety of different circumstances because of the nature of the educational field itself. When we consider learning, a student's experience is shaped by a variety of factors including the circumstances of the lesson, the teacher's personality and the atmosphere in which the lesson was delivered. It's possible that new circumstances will provide a different kind of educational experience.

EDM data is multi-semantic because of several factors, including the ambiguity of instructors and students' conduct, the abstruseness of natural language used by teachers and students, the noisy data in the educational environment, or the missing data. An educational theory's interpretation of the same evidence might lead to uncertainty. There is a major distinction between EDM and general CBE research because of the difference in the intended use of the data. Rather than supplementing or replacing conventional education, EDM focuses on functions that are either missing or difficult to implement in traditional instruction.

2.2.2. Recent work in educational data mining

Data pretreatment, data mining, and assessment are all part of a typical EDM workflow. An educational perspective is that this is information obtained in the data provided by the educational environment and then utilized to improve the educational environment itself.6 Research in educational data mining may be divided into three categories: statistics and visualization, web mining, or a combination of the three.

Preprocessing, data mining, and assessment are all parts of the standard EDM procedure. Knowledge obtained in educational data created by the educational environment and utilized to enhance the educational environment is what we mean from an educational point of view. Work in educational data mining is divided into two areas by Romero and Ventura (2007): statistics and visualization, and web mining.

Researchers Partho Mandal and I-Han Hsiao (2018) apply differential mining [7] from the recent Educational Data Mining in Computer Science Education (CSEDM) Workshop to investigate students' problem-solving tactics. Computer science students' multiple-choice problem-solving activities were collected from a semester-long course in 2016 Fall semester. To construct the problem-solving sequences, the frequent behavioral patterns were extracted based on the accuracy, difficulty, subject, and time of each question. Pupils' meta-cognitive skills and mental processes were revealed by comparing high- and low-performing students in terms of these patterns. Both a panel of human experts and a data-driven technique are used to generate these attributes. Students advance through state spaces that have been created by experts as well as those that have been generated by data-driven features. Compared to typical code-states, both techniques significantly decrease the state-space, with the data-driven approach having a high degree of overlap with the expert feature.

CHAPTER 3

TERMINOLOGIES USED

3.1. CORRELATION MATRIX

As the name suggests it is going to be a 2- Dimensional data model, where the rows as well as the columns both comprise the features of the dataset, and each feature is compared with every other feature and is given a numerical value based on the calculations with resect to all other features.

Every cell in the table shows the correlation among two variables. A correlation matrix is used to summarize data, as an enter right into a more superior evaluation, and as a diagnostic for superior analyses.

The values of a typical feature lie between -1 to 1. If two features in a dataset are inversely proportional then a negative correlation is shown between the two features and the greater the intensity the value lies close to -1. Similarly, if two features in a dataset are directly proportional, then a positive correlation is shown between two features and greater the intensity the value lies close to 1. And if there is no correlation between the two features in a dataset then the value lies close to 0.

3.2. PRECISION

Precision is described as the ratio of efficiently labeled fine samples (real fantastic) to a total number of classified fine samples (both effectively or incorrectly). Precision facilitates us to visualize the reliability of the machine studying model in classifying the model as fine.

$$\text{Precision} = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

Fig 3.1. Precision Formula

3.3. RECALL

Recall take into account absolutely is how most of the real positives have been recalled (decided), i.e., how lots of an appropriate hit had been additionally found. It is the fraction of applicable times that have been retrieved. each precision and don't forget are therefore based totally on relevance.

$$\text{Recall} = \frac{True\ Positive}{True\ Positive + False\ Negative}$$

Fig 3.2. Recall Formula

3.4. F1 SCORE

The F1 score is basically an average (to be more specific, average of Precision and Recall) which is going to be harmonic in nature. A high F1 score is obtained when both the recall and precision values are high similarly, a low F1 score is obtained when both the recall and precision scores are lower. The F1 rating is described because the harmonic implies of precision and recollect.

$$F1\ score = 2 * \frac{Precision\ * Recall}{Precision + Recall}$$

Fig 3.3 F1 Score Formula

3.5. CONFUSION MATRIX

A confusion matrix is a kind of 2*2 two-dimensional array, where we write binary values in the rows as well as columns. Now each cell has a significance as well as meaning. The first cell in the matrix having values in both rows and columns as zero signifies true negative. It means that we predicted false and it is actually true. Similarly, the cell in the first row and second column has a value of zero and one, where zero is the value of the row and one is the value of the column, signifies false negative. This means that we predicted the outcome of the target variable as Tue but it is actually false. Similarly, the cell in the second row and first column has a value of one and zero, where one is the value of the row and zero is the value of the column, signifies false positive. This means that we predicted the final outcome of the target variable as false but in reality, the final outcome of the target variable is false. The last cell which has a value of both the row and column as one, signifies true positive. This means that we predicted the final outcome of the target variable as true and the real value of the final outcome of the target value is positive.

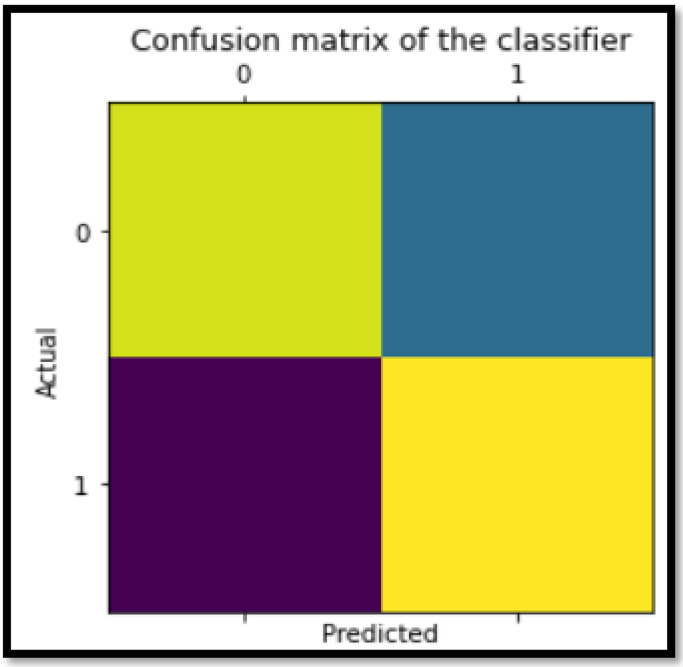


Fig 3.4. Confusion Matrix

CHAPTER 4

PROPOSED WORK

4.1. PROBLEM STATEMENT

Many elements affect a student's performance in tests like family problems or alcohol intake, and via the use of machine learning models the following goals are achieved:

- To predict whether or not a student will skip his final exam or not.
- To understand which all factors create a positive impact as well as which all factors create a negative impact in student's performance.

So, teachers and parents will be able to intervene before students reach the exam stage and solve the problems.

4.2. PROPOSED SOLUTION

One of the most significant aspects of machine learning is the design of the features that will be used. Feature creation, feature extraction, and feature selection are all sub-modules of this process. An essential part of feature construction is the identification of real-world physical features. Auto-feature extraction builds on existing features to create an entirely new collection of attributes with important physical, statistical, or kernel properties. As an example, a time stamp, geometric elements, textures, and so forth. A dimension reduction effect may be achieved by picking a set of statistically important feature subsets from the feature set and removing the unnecessary ones.

A timestamp feature may be broken down into 'Remaining time,' 'Delay times, and 'Total Submissions' in the original dataset. Finally, we can determine the number of submissions each day and the number of compiler errors depending on the current date. We may use a binary bit to determine if a submission has four or five parts in order to integrate submissions with varied numbers of grade sections.

We're starting out with just five features. Making data as good as possible is essential. Feature engineering allows us to enhance existing ones while also creating new ones. In all, we gain 15 features in the training dataset and 18 features after one-hot encoding by combining and correlating features. After feature engineering, the names and descriptions of the features are shown in figure 4.1.

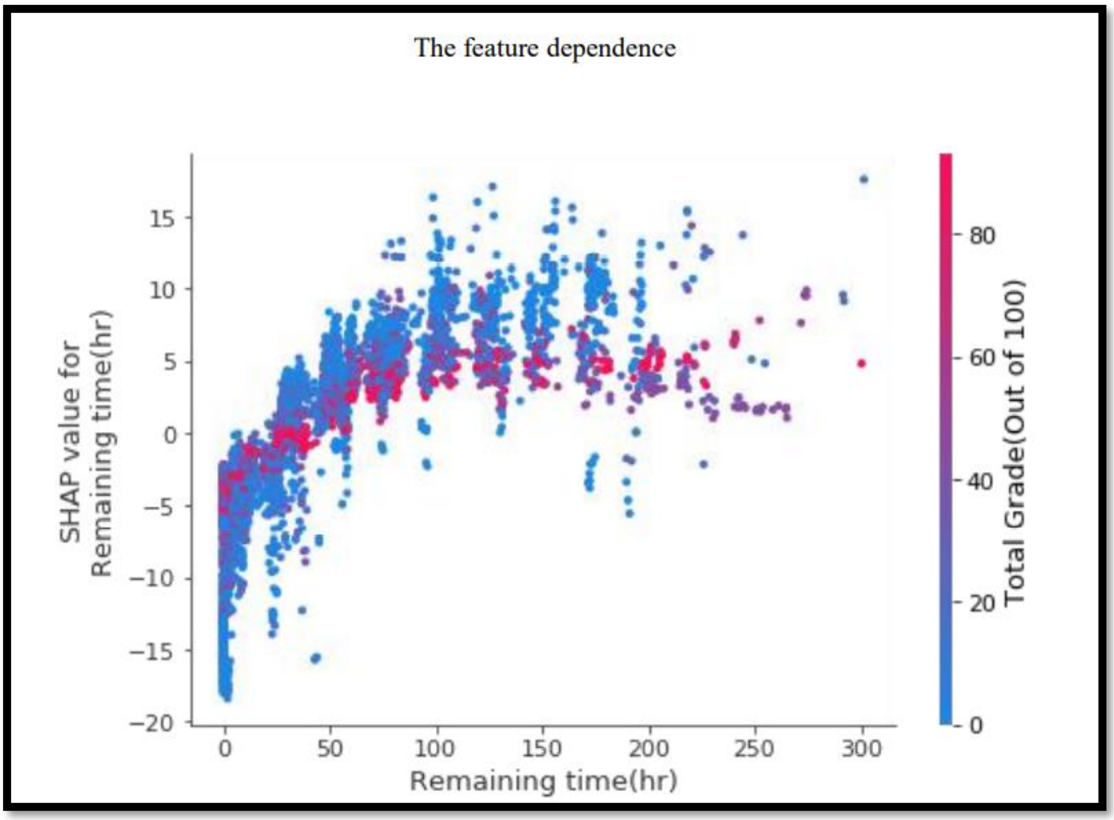


Fig 4.1. Dependence contribution between Reaming hours and total grades

Based on basic heuristics, statistical principles, random rules, or previously utilized algorithms, the Baseline Model predicts recognized issues and associated data sets. Before the official work begins, this is frequently done to provide a foundation for evaluating how well the new model performs in comparison to what was previously used. As a starting point, we'll utilize a simple linear regression model. The least square’s function, referred to as a linear regression equation, is used to represent the connection between a set of independent variables and a set of dependent variables in linear regression. One or more model parameters known as regression coefficients are used to create this function. Single-variable regression is referred to as simple regression, whereas multiple-variable regression refers to regression using more than one independent variable.

4.3.DATASET OVERVIEW

The dataset that is currently utilized in this project comprises of the records of students which are residents of Portugal and are going to schools where their primary conversation language is Portugal. Now the records of the dataset are derived by carefully segregating the score cards of the various students, and by studying the manner in which a student responds to various questions. Now there are various kinds of subjects that are being taught in the school and in order to isolate and conduct the project, two primary subjects are taken into consideration and those subjects are Portuguese and Mathematics. Now in order to give an overview of each and every feature of the dataset, a short one-line description is given and the description is self-explanatory. Below is the list of all the various features:

1. ‘_school_’: This feature gives the name of the school in which a particular student has taken admission.
2. ‘_sex_’: This column basically gives information about the gender of an individual.

3. ‘_address_’: This column gives information about how far does a student reside with respect to his/her home.
4. ‘_age_’: This column gives description about age of a particular student.
5. ‘_famsize_’: This column tells about how many members are present in a particular student’s family.
6. ‘_PStatus_’: This column gives an idea about the relationship between the guardians. This column consists of two indicators namely: ‘A’ which means both the guardians are living under one roof/ together and ‘T’ which means both the guardians are living separately.
7. ‘_MEdu_’: This column gives an idea about the education qualification of a student’s mother. It is represented by four numerical values and those are zero which signifies that a student’s mother has no education at all, one which signifies that a student’s mother has done primary education, two which signifies that a student’s mother has done secondary education and last but not the least four which signifies that a student’s mother has done higher education.
8. ‘_FEdu_’: This column gives an idea about the education qualification of a student’s father. It is represented by four numerical values and those are zero which signifies that a student’s father has no education at all, one which signifies that a student’s father has done primary education, two which signifies that a student’s father has done secondary education and last but not the least four which signifies that a student’s father has done higher education.

9. ‘_Mjob_’: This column gives an idea about whether a student’s mother is working or not.
10. ‘_Fjob_’: This column gives an idea about whether a student’s father is working or not.
11. ‘_Reason_’: This column tells the primary reason for a student to choose a particular school
12. ‘_Guardian_’: This column tells that who is the primary guardian of a particular student
13. ‘_TravelTime_’: This attribute tells us that how much time it takes for a student to reach to school from his/her home.
14. ‘_StudyTime_’: This attribute gives an idea about how much hours a particular student spends per week on studies.
15. ‘_Failures_’: This column tells that how many number of times a particular student has failed in the past.
16. ‘_SchoolSup_’: This column tells that whether a particular student receives support from external sources other than school.
17. ‘_Famsup_’: This column tells that how much a student’s family is giving support to a student.
18. ‘_Paid_’: This attribute tells that whether a particular student takes extra classes or not outside of school premises and from other external sources.
19. ‘_Activities_’: This column tells that whether a student has an interest in participating in other extracurricular activities or not.
20. ‘_Nursery_’: This column tells that whether a student has attended nursery school or not.

21. ‘_Higher_’: This attribute tells that whether a student has an interest in pursuing higher education or not.
22. ‘_Internet_’: This attribute tells that whether a student has internet access in his/her home or not.
23. ‘_Romantic_’: This attribute tells that whether a student is involved in a relationship or not.
24. ‘_Famrel_’: This attribute gives an idea about a student’s relationship with his/her other family members
25. ‘_Freetime_’: This attribute gives an idea about the amount of free time or break from studies a student takes after coming back from school

4.4.DATA PREPROCESSING

Each and every field of the two datasets are processed thoroughly in order to check that the data in the dataset doesn’t comprise of null values, as well as invalid values and as well as duplicate values.

However luckily, none of the above-mentioned contortion is found in these datasets and thereby we can skip all the data preprocessing steps. Each dataset is merged vertically, which will make the dataset powerful and growth the dataset, this procedure supposedly key thing of information preprocessing.

4.5. FLOWCHART OF PROPOSED MODEL

DATA PREPROCESSING AND SPLITTING

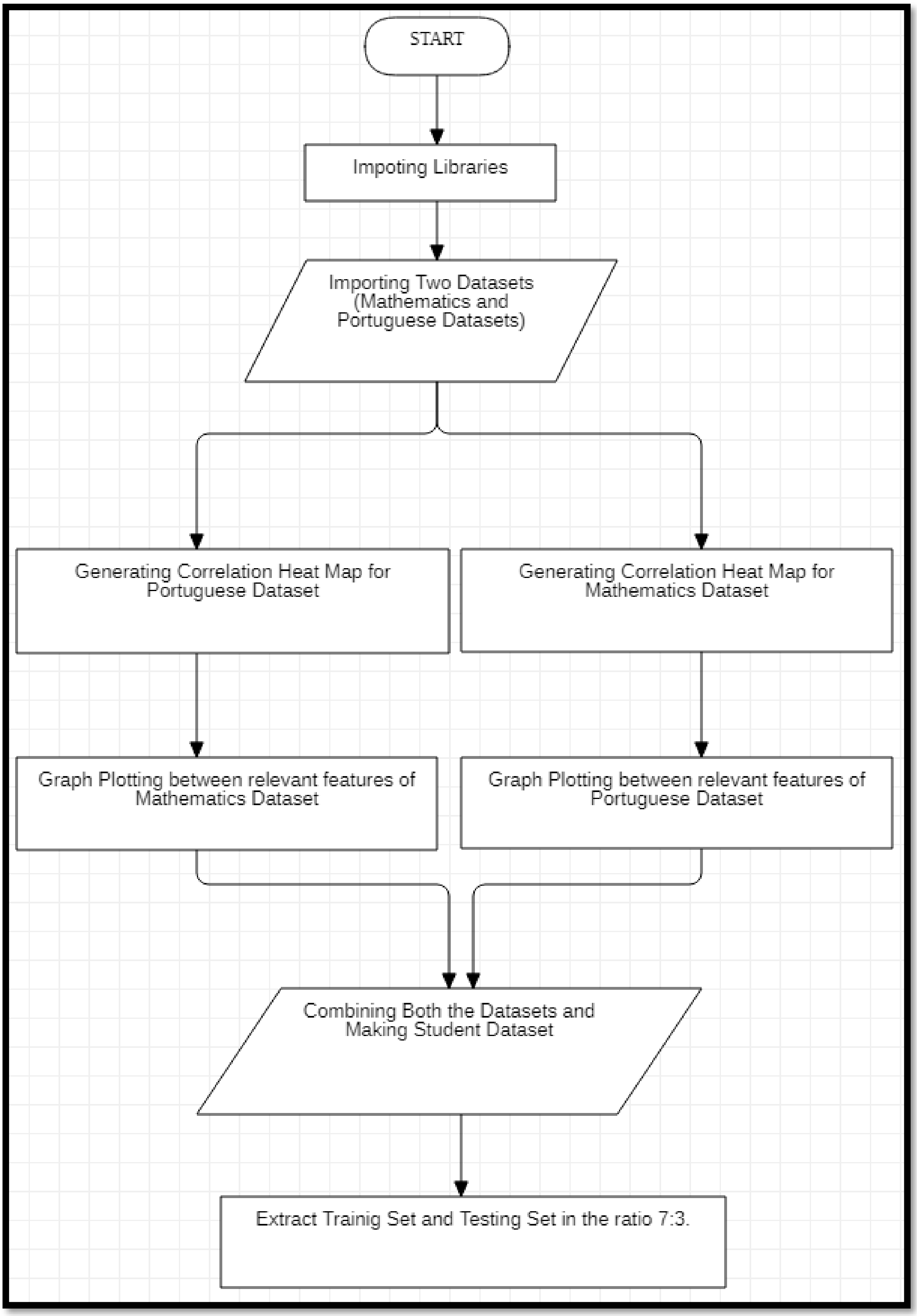

START
Impoting Libraries
Importing Two Datasets (Mathematics and Portuguese Datasets)
Generating Correlation Heat Map for Portuguese Dataset
Generating Correlation Heat Map for Mathematics Dataset
Graph Plotting between relevant features of Mathematics Dataset
Graph Plotting between relevant features of Portuguese Dataset
Combining Both the Datasets and Making Student Dataset
Extract Trainig Set and Testing Set in the ratio 7:3.

Fig 4.2. Data Preprocessing and Splitting

APPLYING MACHINE LEARNING ALGORITHMS ON TEST DATA

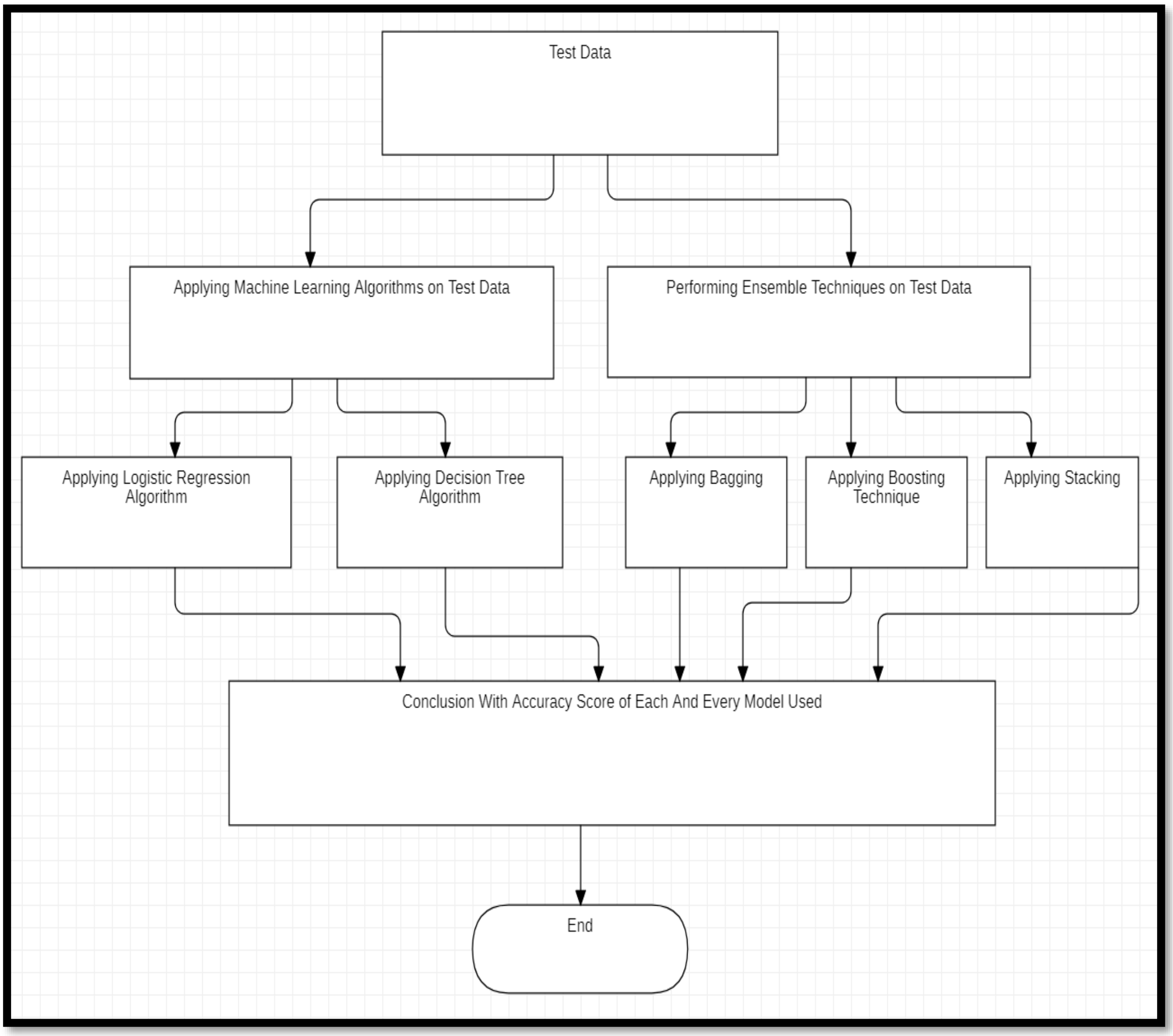

Test Data
Applying Machine Learning Algorithms on Test Data
Performing Ensemble Techniques on Test Data
Applying Logistic Regression Algorithm
Applying Decision Tree Algorithm
Applying Bagging
Applying Boosting Technique
Applying Stacking
Conclusion With Accuracy Score of Each And Every Model Used
End

Fig 4.3. Applying Machine Learning Algorithms on Test Data

CHAPTER 5

WORKING AND ANALYSIS

To understand which factors are affecting the student performance we need to draw a correlation matrix. By the help of correlation heat map, we will be able to analyze which all features are causing a negative impact, positive impact and no impact at all on the student's performance. We will consider both the negative as well as positive impact features and will drop the null impact features on the student's performance.

5.1. CORRELATION HEAT MAP OF MATHS DATASET

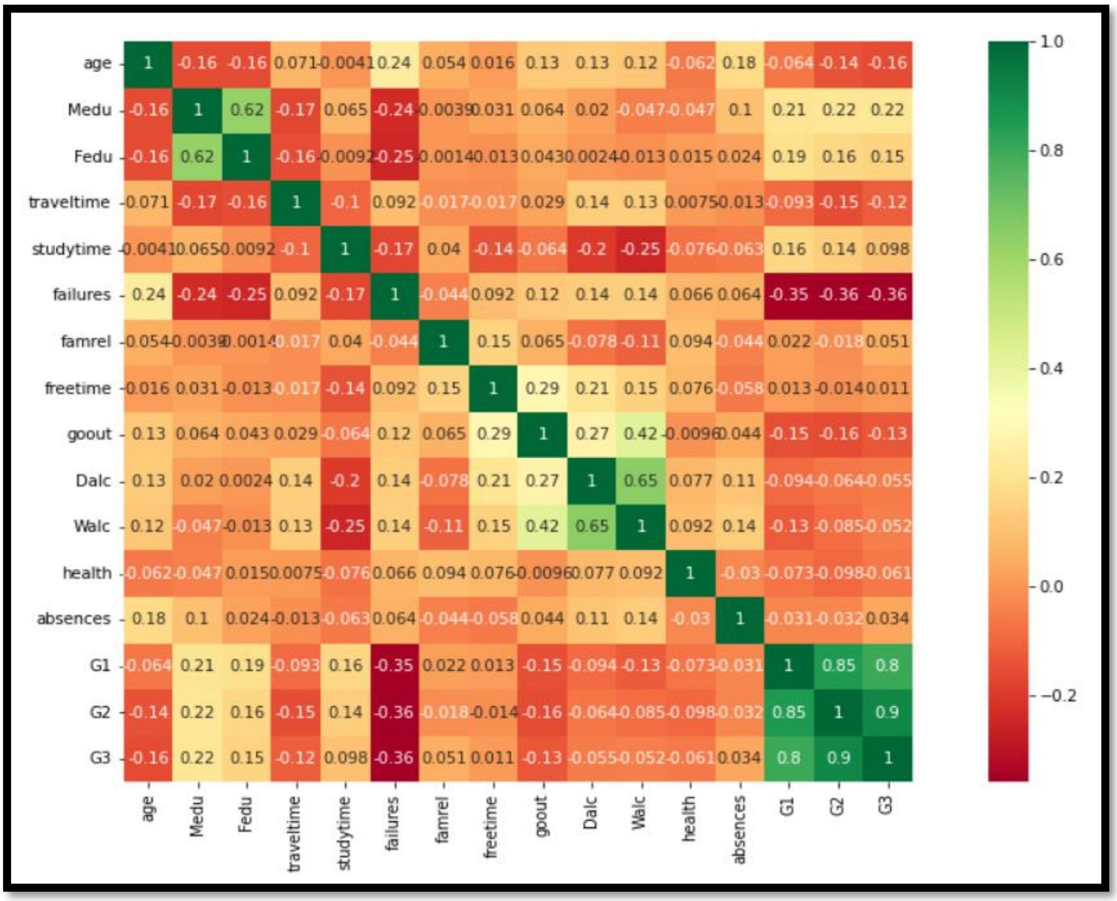


Fig 5.1. Correlation Heat Map of Mathematics Dataset

5.2. CORRELATION MAP FOR PORTUGUESE DATA

We can clearly see from the below Heat Map that mother's education, father's education, study-time, family relationship attributes have a positive impact on the student's performance. And the factors which have a negative impact are travel-time, free time, going out and absences.

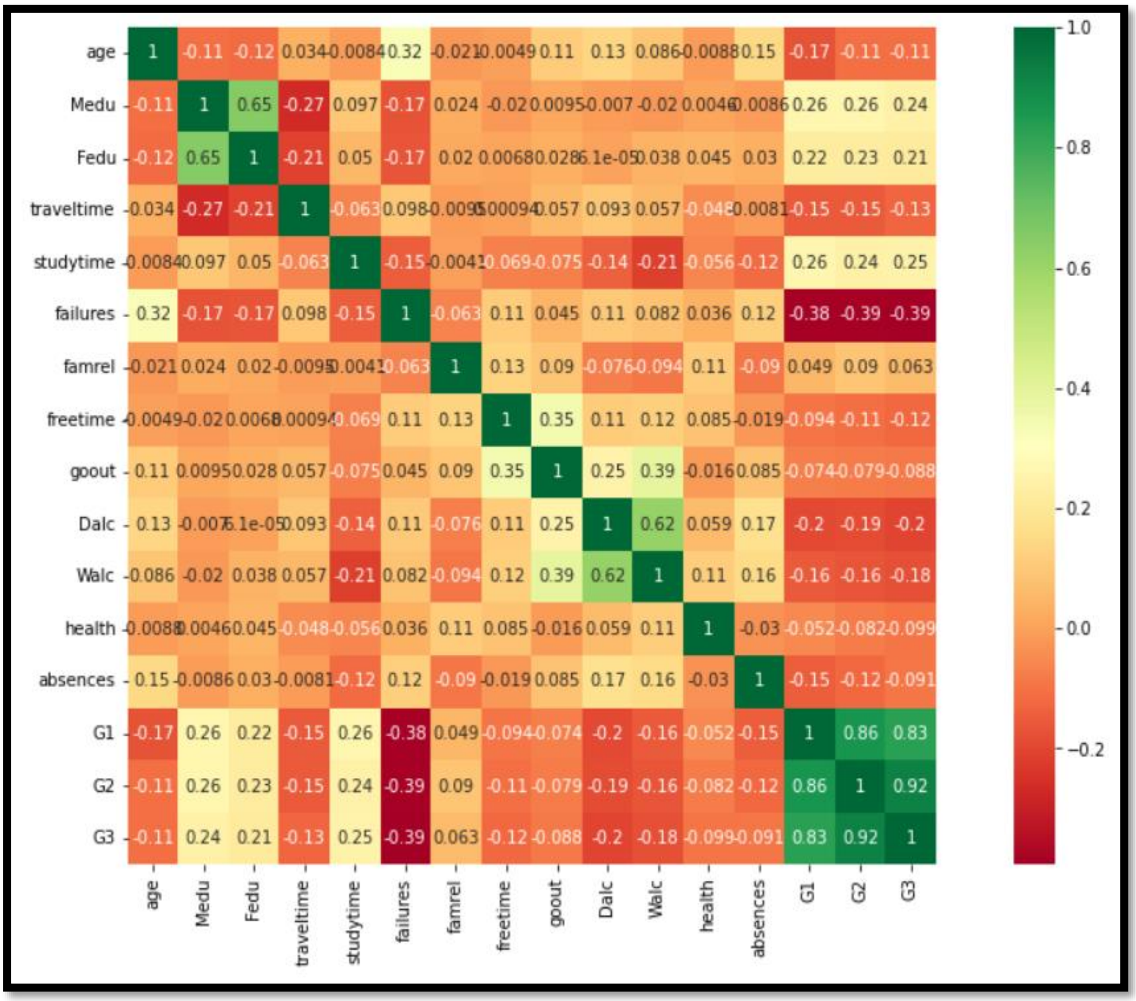


Fig 5.2. Correlation Heat Map of Portuguese Dataset

5.3. GRAPH ANALYSIS OF BOTH DATASETS

Now we are going to analyze two attributes namely ‘Age’ and ‘Study-Time’ in-order to determine the interest towards studies with respect to age.

If we consider both the mathematics as well as Portuguese dataset, we can see that the children in the age group 15 – 17 spend only two hours of their time in reading these subjects. Whereas only, very few children in the age group of 18-22 spend around 4 hours or more of their time in reading these subjects. The graph analysis is shown below:

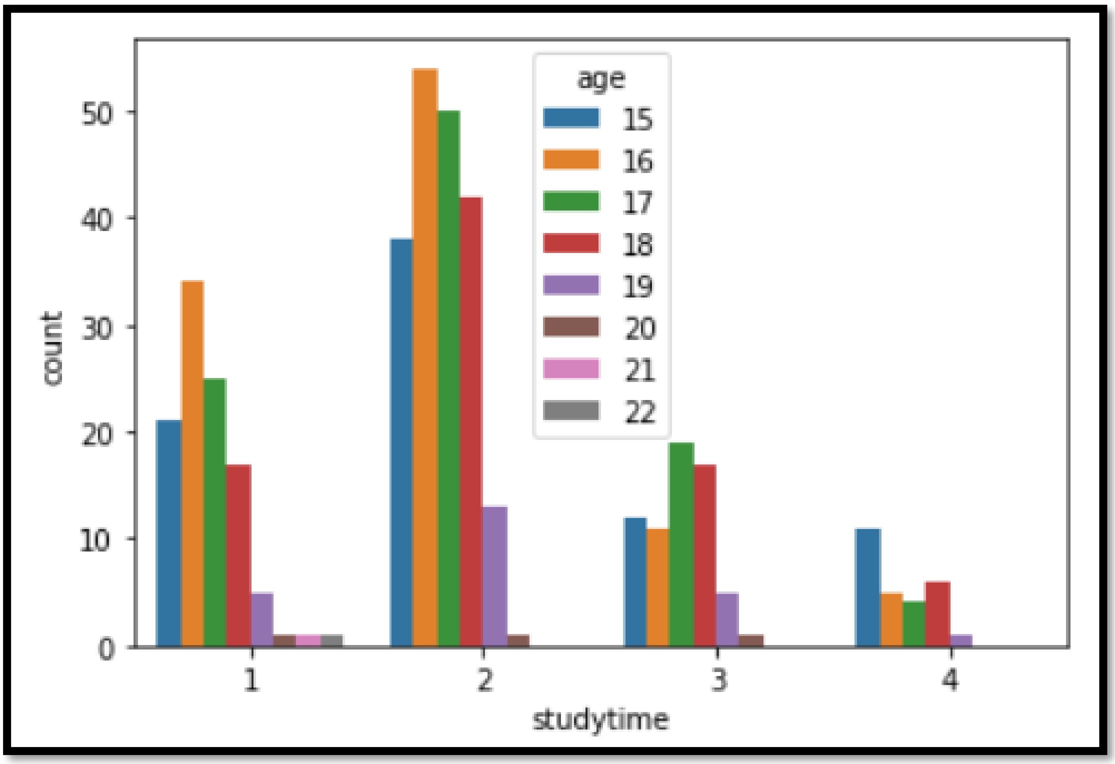


Fig 5.3. Bar graph Analysis b/w Age and Study time of Math Dataset

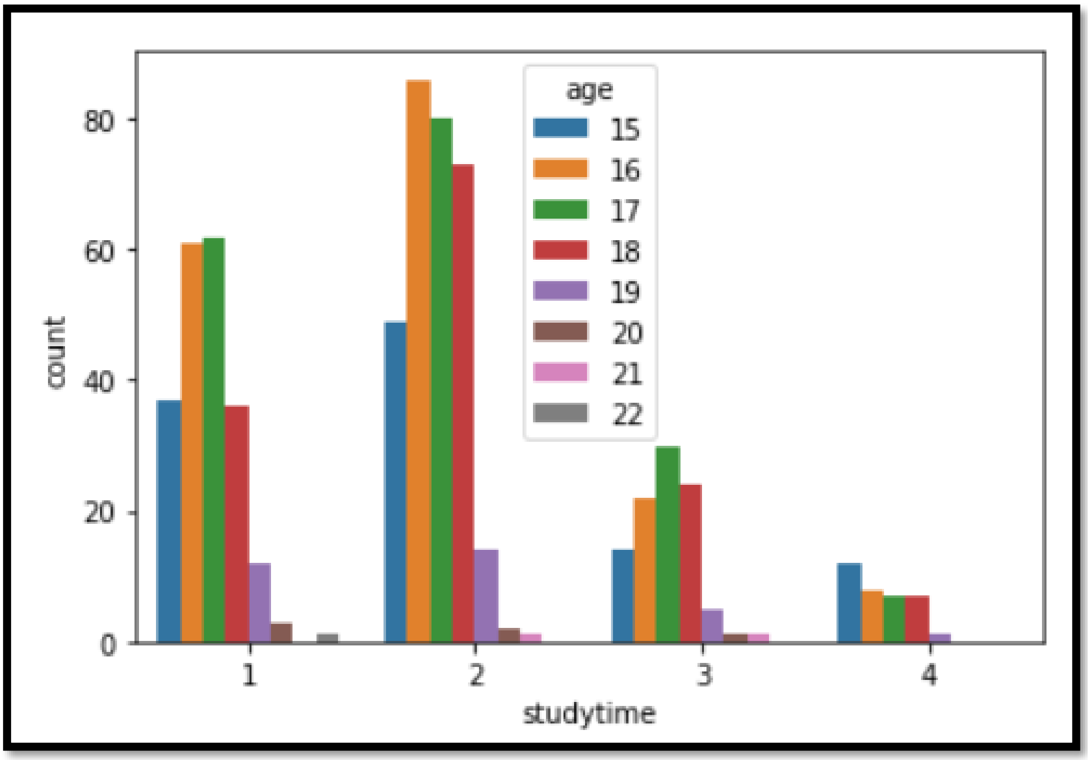


Fig 5.4. Bar graph Analysis b/w Age and Study time of Portuguese Dataset

Now we will visualize the graph between age group of the children who consume alcohol on the weekends.

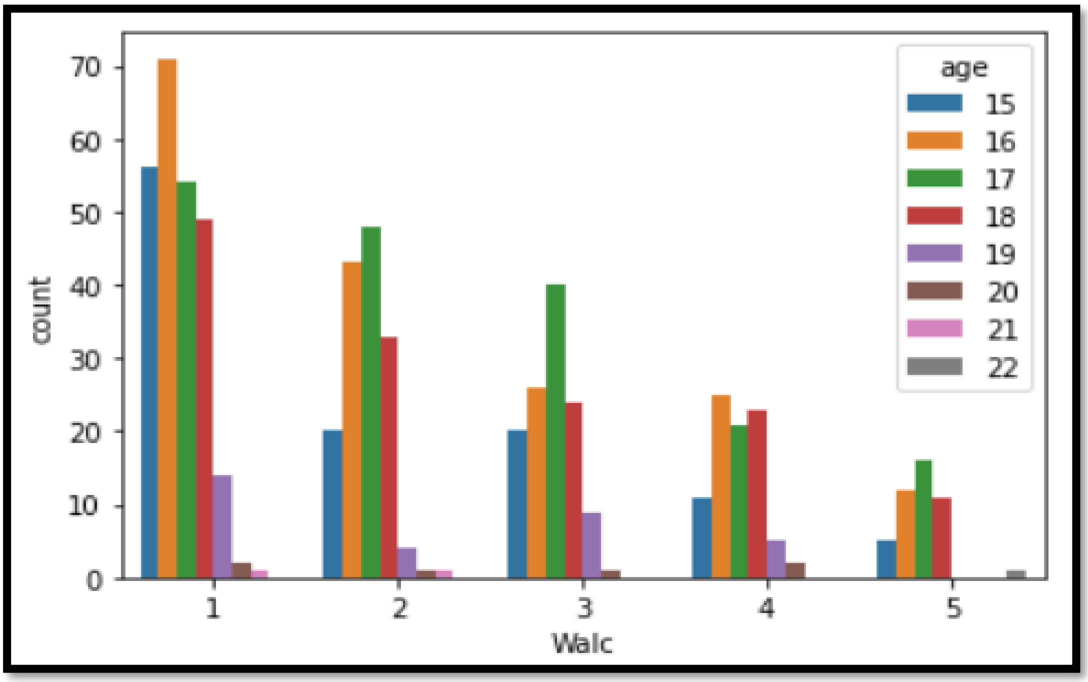
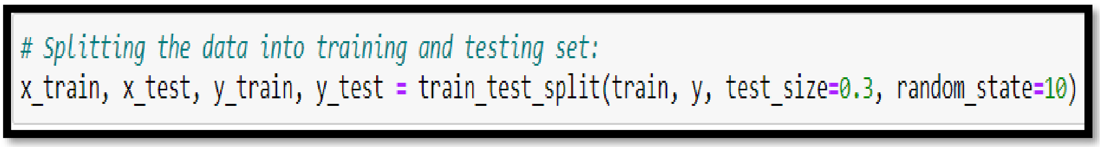


Fig 5.5. Bar graph Analysis b/w Age and Walc of Portuguese Dataset.

5.4. DATASET SPLITTING

From the heat map we can clearly observe that the features like study time, mother’s education, father’s education and family relation attributes are highly correlated features with the Target variable. Here I am splitting the dataset in the **ratio 70:30** in order to create training and testing sets.

CODE:



```
# Splitting the data into training and testing set:
x_train, x_test, y_train, y_test = train_test_split(train, y, test_size=0.3, random_state=10)
```

Fig 5.6. Code for training and test set Split.

CHAPTER 6

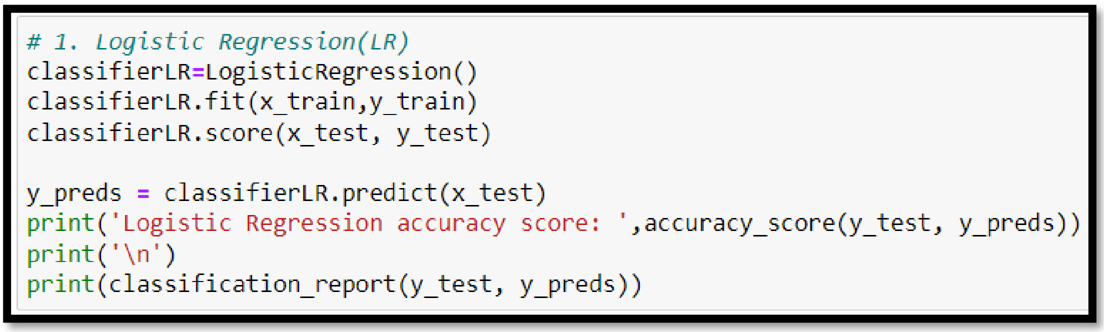
EXPERIMENT AND RESULTS

6.1. APPLICATION OF BASE LEARNERS

Base learners are the normal machine learning algorithms. In this project Logistic Regression and Decision Tree are regarded as the base learners. Classification report is generated in order to show the performance of these classifiers.

6.1.1. LOGISTIC REGRESSION

The logistic regression function is used from sklearn. And the accuracy achieved from logistic regression is around 75.4%.



```
# 1. Logistic Regression(LR)
classifierLR=LogisticRegression()
classifierLR.fit(x_train,y_train)
classifierLR.score(x_test, y_test)

y_preds = classifierLR.predict(x_test)
print('Logistic Regression accuracy score: ',accuracy_score(y_test, y_preds))
print('\n')
print(classification_report(y_test, y_preds))
```

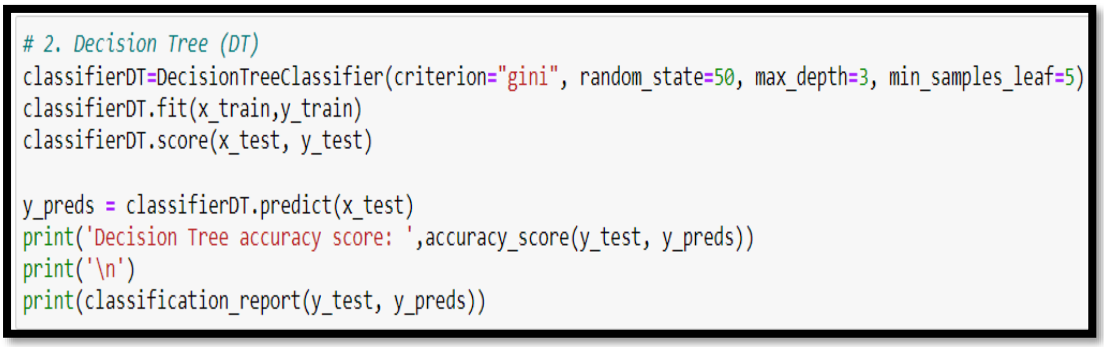
Fig. 6.1. Code for logistic Regression

Table 6.1. Classification Report for Logistic Regression

_~_	_Precision_	_Recall_	_F1-Score_	_Support_
0	0.4	0.67	0.5	6
1	0.5	0.5	0.5	14
2	0.61	0.49	0.54	51
3	0.77	0.75	0.76	106
4	0.83	0.89	0.86	137

6.1.2. DECISION TREE

The decision tree function is used from sklearn. And the accuracy achieved from logistic regression is around 84%.



```
# 2. Decision Tree (DT)
classifierDT=DecisionTreeClassifier(criterion="gini", random_state=50, max_depth=3, min_samples_leaf=5)
classifierDT.fit(x_train,y_train)
classifierDT.score(x_test, y_test)

y_preds = classifierDT.predict(x_test)
print('Decision Tree accuracy score: ',accuracy_score(y_test, y_preds))
print('\n')
print(classification_report(y_test, y_preds))
```

Fig. 6.2. Code for Decision Tree

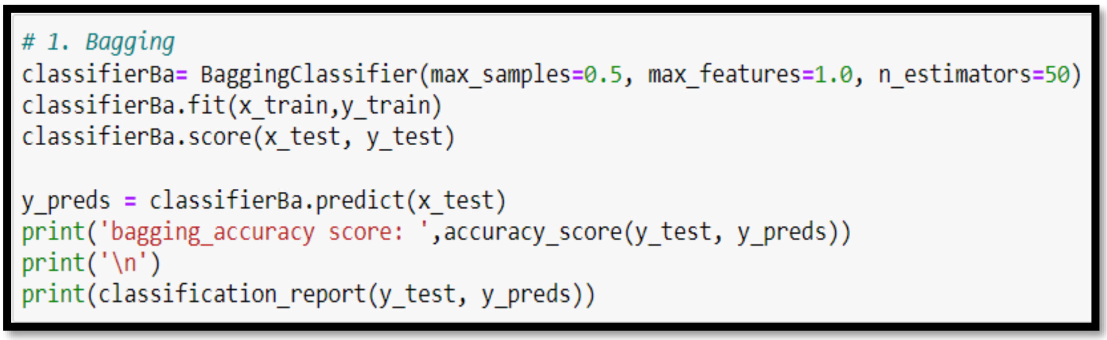
Table 6.2. Classification Report for Decision Tree

_~_	_Precision_	_Recall_	_F1-Score_	_Support_
0	0	0	0	6
1	0.45	0.71	0.56	14
2	0.93	0.76	0.84	51
3	0.82	0.92	0.87	106
4	0.89	0.86	0.88	137

6.2. APPLICATION OF ENSEMBLE TECHNIQUES

6.2.1. BAGGING

The bagging function is used from sklearn. And the accuracy achieved from logistic regression is around 80.8%.



```
# 1. Bagging
classifierBa= BaggingClassifier(max_samples=0.5, max_features=1.0, n_estimators=50)
classifierBa.fit(x_train,y_train)
classifierBa.score(x_test, y_test)

y_preds = classifierBa.predict(x_test)
print('bagging_accuracy score: ',accuracy_score(y_test, y_preds))
print('\n')
print(classification_report(y_test, y_preds))
```

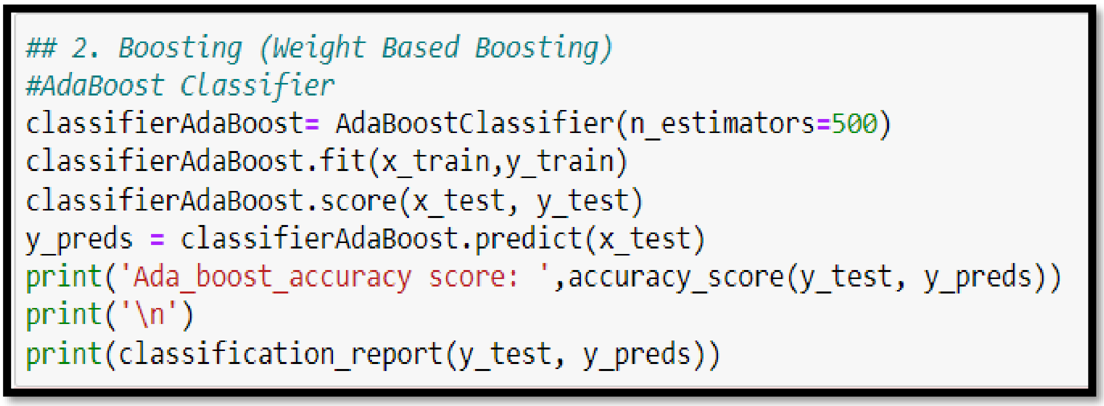
Fig. 6.3. Code for Bagging Ensemble Technique

Table 6.3. Classification Report for Bagging

_~_	_Precision_	_Recall_	_F1-Score_	_Support_
0	0.5	0.83	0.62	6
1	0.67	0.71	0.69	14
2	0.89	0.67	0.76	51
3	0.82	0.81	0.82	106
4	0.82	0.87	0.84	137

6.2.2. BOOSTING

The boosting function is used from sklearn. And the accuracy achieved from logistic regression is around 81.5%.



```
## 2. Boosting (Weight Based Boosting)
#AdaBoost Classifier
classifierAdaBoost= AdaBoostClassifier(n_estimators=500)
classifierAdaBoost.fit(x_train,y_train)
classifierAdaBoost.score(x_test, y_test)
y_preds = classifierAdaBoost.predict(x_test)
print('Ada_boost_accuracy score: ',accuracy_score(y_test, y_preds))
print('\n')
print(classification_report(y_test, y_preds))
```

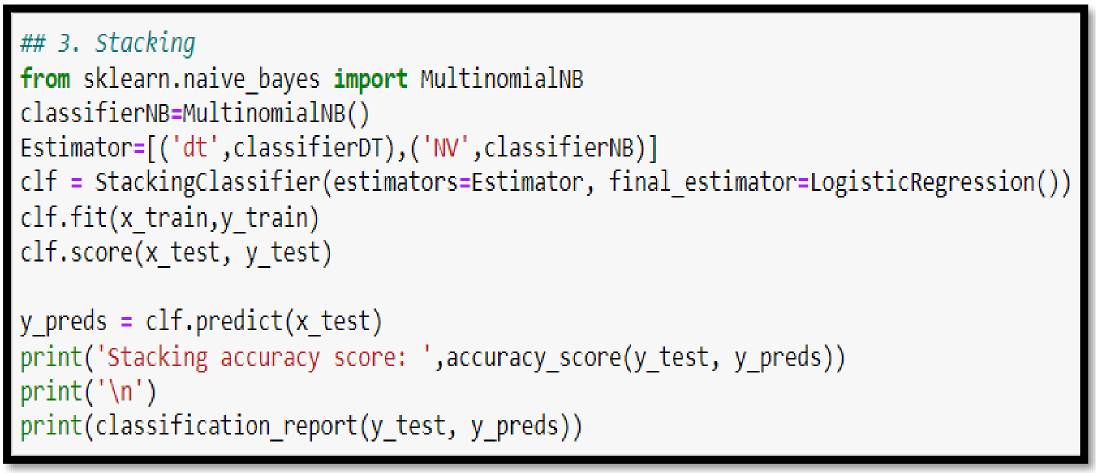
Fig. 6.4. Code for Boosting Ensemble Technique

Table 6.4. Classification Report for Boosting

_~_	**_Precision_**	**_Recall_**	**_F1-Score_**	**_Support_**
0	0.23	0.83	0.36	6
1	0	0	0	14
2	0.95	0.69	0.80	51
3	0.82	0.92	0.87	106
4	0.87	0.87	0.87	137

6.2.3. STACKING

The stacking function is used from sklearn. And the accuracy achieved from logistic regression is around 84.7%.



```
## 3. Stacking
from sklearn.naive_bayes import MultinomialNB
classifierNB=MultinomialNB()
Estimator=[('dt',classifierDT),('NV',classifierNB)]
clf = StackingClassifier(estimators=Estimator, final_estimator=LogisticRegression())
clf.fit(x_train,y_train)
clf.score(x_test, y_test)

y_preds = clf.predict(x_test)
print('Stacking accuracy score: ',accuracy_score(y_test, y_preds))
print('\n')
print(classification_report(y_test, y_preds))
```

Fig. 6.5. Code for Stacking Ensemble Technique

Table 6.4. Classification Report for Stacking

_~_	**_Precision_**	**_Recall_**	**_F1-Score_**	**_Support_**
0	0.5	0.5	0.5	6
1	0.62	0.71	0.67	14
2	0.93	0.75	0.83	51
3	0.82	0.92	0.87	106
4	0.89	0.86	0.87	137

CHAPTER 7

CONCLUSION AND FUTURE WORK

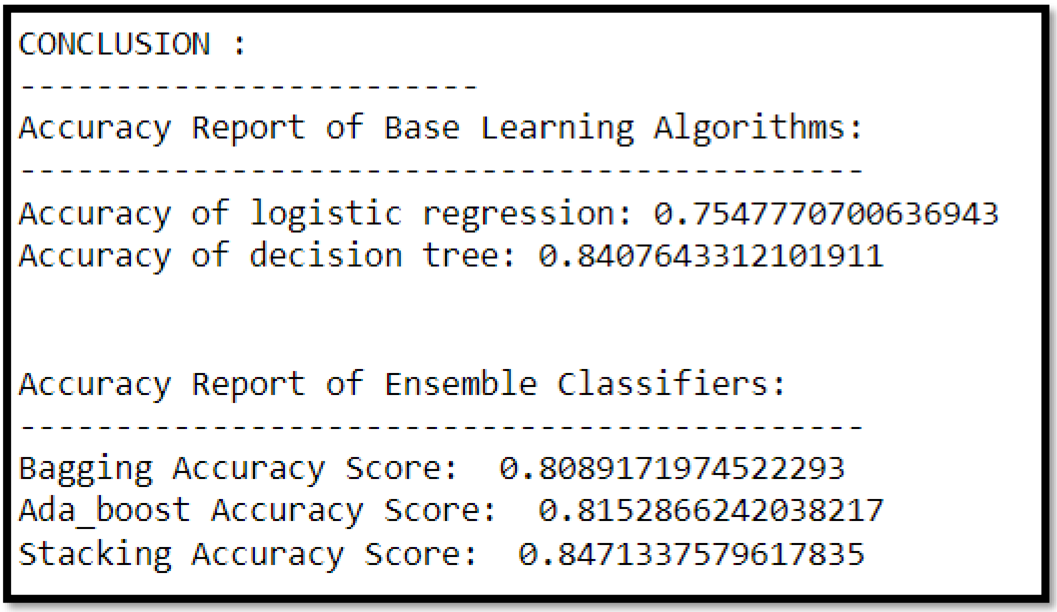
This project attempts to predict the potential factors which actually affect the student’s performance. It depicts that by the help of using ensemble techniques the accuracy of predicting the target value can be increased.

In this project two base learners namely Logistic regression and decision tree are used. After that by the help of ensemble techniques namely bagging, boosting and stacking I have tried to increase the efficiency.

Observing the results, it appears that the 'Remaining Time' feature has a more negative influence on low scores near the due date, whereas the score increases as time passes. On the other hand, data points in the range of 0 to 50 have a substantially greater negative influence than other data points, as shown in Figure 4.6. Beginners may not put forth enough effort to demonstrate their high-grade potential as one explanation.

When a student makes mistakes or receives a lower mark, the overall number of submissions would have a favorable impact. A forecast and explanation given by our experiment enables instructors to identify at-risk pupils early, especially for large programming classrooms. Educators are also able to give timely advice because of this. Scalability is included into this data mining project's data processing pipeline. Object-oriented programming and Java programs, for example, contain comparable grading and time elements, therefore this approach may be used to other programming tasks as well.

The results are as follows: -



```
CONCLUSION :
------------------------
Accuracy Report of Base Learning Algorithms:
--------------------------------------------
Accuracy of logistic regression: 0.7547770700636943
Accuracy of decision tree: 0.8407643312101911


Accuracy Report of Ensemble Classifiers:
--------------------------------------------
Bagging Accuracy Score:  0.8089171974522293
Ada_boost Accuracy Score:  0.8152866242038217
Stacking Accuracy Score:  0.8471337579617835
```

Fig. 7.1. Accuracy Report

REFERENCES

[1] Baker, R. S., & Yacef, K. (2009). The state of educational data mining in 2009: A review and future visions. JEDM| Journal of Educational Data Mining, 1(1), 3-17.

[2] Anjewierden A, Kolloffel B, Hulshof C. Towards educational data mining: Using data mining methods for automated chat analysis to understand and support inquiry learning processes. In: Proc. of the Int’l Workshop on Applying Data Mining in e-Learning (ADML 2007). 2007.

[3] Cole J, Foster H. Using Moodle: Teaching with the Popular Open-Source Course Management System. 2nd ed., O’Reilly Media, Inc., 2007.

[4] Lara JA, Lizcano D, Martínez MA, Pazos J, Riera T. A system for knowledge discovery in e-learning environments within the European higher education area — Application to student data from open university of madrid. UDIMA. Computers & Education, 2014,72:23-36. [

[5] Worldwide smartphone user base hits 1 billion. 2012.

[6] Facebook users reach 2.2 billion, one third of the global population. 2014.

[7] Partho Mandal and I-Han Hsiao. (2018) Using Differential Mining to Explore Bite-Size Problem Solving Practices. Educational Data Mining in Computer Science Education (CSEDM) Workshop, 2018.

[8] Mohammed Alzaid I-Han Hsiao. (2018) Personalized Self-Assessing Quizzes in Programming Courses. Educational Data Mining in Computer Science Education (CSEDM) Workshop, 2018.

[9] Yancy Vance Paredes, David Azcona, I-Han Hsiao, Alan F. Smeaton. (2018) Predictive Modelling of Student Reviewing Behaviors in an Introductory Programming Course. Educational Data Mining in Computer Science Education (CSEDM) Workshop, 2018.

[10] Rui Zhi, Thomas W. Price, Nicholas Lytle, Yihuan Dong and Tiffany Barnes. (2018) Reducing the State Space of Programming Problems through Data-Driven Feature Detection. Educational Data Mining in Computer Science Education (CSEDM) Workshop, 2018.

[11] Coursera. https://www.coursera.org/ [12] Romero C, Ventura S. Data mining in education. Wiley Interdisciplinary Reviews-Data Mining and Knowledge Discovery, 2013, 3(1):12-27.

[12] Hand DJ, Mannila H, Smyth P. Principles of Data Mining. The MIT Press, 2001.

[13] Peng Y, Kou G, Shi Y, Chen Z. A descriptive framework for the field of data mining

[14] Islam, R., & Altas, I. (2012, October). A comparative study of malware family classification. In International Conference on Information and Communications Security (pp. 488-496). Springer, Berlin, Heidelberg.

[15] Khammas, B. M., Monemi, A., Bassi, J. S., Ismail, I., Nor, S. M., & Marsono, M. N. (2015). Feature selection and machine learning classification for malware detection. Jurnal Teknologi, 77(1), 234-250.

[16] Kwon, I., & Im, E. G. (2017, September). Extracting the representative API call patterns of malware families using recurrent neural network. In Proceedings of the International Conference on Research in Adaptive and Convergent Systems (pp. 202-207).

[17] Mays, M., Drabinsky, N., & Brandle, S. (2017). Feature Selection for Malware Classification. In MAICS (pp. 165-170).

[18] Taheri, L., Kadir, A. F. A., & Lashkari, A. H. (2019, October). Extensible Android Malware Detection and Family Classification Using Network-Flows and API-Calls. In 2019 International Carnahan Conference on Security Technology (ICCST) (pp. 1-8). IEEE.

[19] San, C. C. (2019). Effective Malicious Features Extraction and Classification for Incident Handling Systems (Doctoral dissertation, University of Computer Studies, Yangon).

[20] Banin, S., & Dyrkolbotn, G. O. (2018). Multinomial malware classification via low-level features. Digital Investigation, 26, S107-S117.