Student Performance Prediction Using Machine Learning

Project report submitted to
Visvesvaraya National Institute of Technology, Nagpur
in partial fulfillment of the requirements for the award of the degree

Bachelor of Technology
in
Computer Science and Engineering

by

Komma Srinivas	**BT18CSE121**
Aitipamula Manish Chandra	**BT18CSE020**
Arangi Shiva Shankar	**BT18CSE048**
Maddela Sumanth	**BT18CSE127**
Mudhireddy Hrushikesh Reddy	**BT18CSE131**

under the guidance of

Dr. A. S. Mokhade

Department of Computer Science and Engineering
Visvesvaraya National Institute of Technology
Nagpur 440 010 (India)

2021-22

Student Performance Prediction Using Machine Learning

Project report submitted to
Visvesvaraya National Institute of Technology, Nagpur
in partial fulfillment of the requirements for the award of the degree

Bachelor of Technology
in
Computer Science and Engineering

by

Komma Srinivas	**BT18CSE121**
Aitipamula Manish Chandra	**BT18CSE020**
Arangi Shiva Shankar	**BT18CSE048**
Maddela Sumanth	**BT18CSE127**
Mudhireddy Hrushikesh Reddy	**BT18CSE131**

under the guidance of

Dr. A. S. Mokhade

Department of Computer Science and Engineering
Visvesvaraya National Institute of Technology
Nagpur 440 010 (India)

2021-22

Department of Computer Science and Engineering
Visvesvaraya National Institute of Technology, Nagpur

Declaration

We, "Komma Srinivas, Aitipamula Manish Chandra, Arangi Shiva Shankar, Maddela Sumanth, Mudhireddy Hrushikesh Reddy", hereby declare that this project work titled "Student Performance Prediction" is carried out by us in the Department of Computer Science and Engineering of Visvesvaraya National Institute of Technology, Nagpur. The work is original and has not been submitted earlier whole or in part for the award of any degree/diploma at this or any other Institution / University.

Sr.No	Enrollment No	Name	Signature
1.	BT18CSE121	Komma Srinivas	
2.	BT18CSE020	Aitipamula Manish Chandra	
3.	BT18CSE048	Arangi Shiva Shankar	
4.	BT18CSE127	Maddela Sumanth	
5.	BT18CSE131	Mudhireddy Hrushikesh Reddy	

Date:

Certificate

This to certify that the project titled "**Student Performance Prediction**", submitted by "**Komma Srinivas, Aitipamula Manish Chandra, Arangi Shiva Shankar, Maddela Sumanth, Mudhireddy Hrushikesh Reddy**" in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology in Computer Science and Engineering**, VNIT Nagpur. The work is comprehensive, complete and fit for final evaluation.

Dr. A. S. Mokhade
Assistant Professor
Department of Computer Science and Engineering
VNIT, Nagpur

Dr. P. S. Deshpande
Head, Department of Computer Science and Engineering
VNIT, Nagpur
Date:

ACKNOWLEDGEMENT

We would sincerely like to thank our project guide Dr. A. S. Mokhade, Associate Professor, Department of Computer Science and Engineering for his immense help and valuable guidance. We are extremely grateful for his insights, resourcefulness, and timely support and for his alacrity to help us whenever we needed help. It was his constant motivation that helped us complete this journey. We have gained immense knowledge under his patronage and are obliged to him for providing not only his valuable guidance, but also the necessary facilities provided to complete the project.

We take this opportunity to thank our Head of Department Dr. P. S. Deshpande for his support and for providing institute access to required research papers for our project experimentation. The resources provided by VNIT, Nagpur were an asset and made sure we met all our needs.

We would like to thank all the administrative and technical staff of the department for who have advised and helped us in their respective roles. We would like to thank all those who have directly or indirectly contributed to the completion of the project. Their role and support were of vital importance.

We would like to thank the lab assistants and non-technical staff for their support and making sure we have access to required research papers.

Komma Srinivas	**BT18CSE121**
Aitipamula Manish Chandra	**BT18CSE020**
Arangi Shiva Shankar	**BT18CSE048**
Maddela Sumanth	**BT18CSE127**
Mudhireddy Hrushikesh Reddy	**BT18CSE131**

Date:

ABSTRACT

To attain the best results and lower the failure rate, educational systems require new strategies to improve its quality. Student Performance Analysis has boomed in the educational systems in recent days as it enables to predict and analyze the performance of the student. It assists students in identifying their weaknesses and enhancing scores. Immense efforts have been made in order to improve the prediction process. However, Due to limited datasets, poor prediction accuracy, and inappropriate attribute analysis the existing approaches are neither fair nor satisfactory.

The main objective of this paper is application of machine learning techniques in predicting student performance considering all the factors that affect it.

To achieve this goal an experimental dataset is used in this work which will pass through Three stages. Firstly, Dataset Preprocessing is done, Then best attributes are selected that will help in predicting the student performance in an efficient way. Finally, various Supervised Machine Learning Algorithms (Linear Regression, Elastic Net Regression, Random Forest, Gradient Boosted, Decision Tree) are employed for prediction. Then Proposed ML Algorithm is integrated with Web Application to show the predicted results.

Preliminary results suggest that the "**Random Forest**" model achieved the best results among others.

LIST OF FIGURES

LIST OF TABLES

NOMENCLATURE

ML	Machine Learning
SVM	Support Vector Machine
MAE	Mean Absolute Error
RMSE	Root Mean Squared Error
R2	R Squared

TABLE OF CONTENTS

CHAPTER - 1

INTRODUCTION

An Overview of Definitions and Uses

Introduction

Educational institutions are an integral element of our society, and they play a crucial role in any country's growth and development. The primary goal of education institutions is to offer students a high-quality education. Education has encountered numerous challenges over the years. To improve learning quality, several teaching and learning strategies are suggested [3]. One of the most crucial jobs in any educational institution is to evaluate student performance.

Institutes have adopted Continuous Evaluation systems these days which are beneficial to the students in improving their performance [5]. Predicting student performance in advance can help students and teachers to keep track of their progress. So earlier, In many institutions teachers have done it manually to predict the student’s performance. However, Learning and teaching have improved by incorporating Technology into traditional educational methods [4].

In prior studies, it has been observed that many researchers have attempted to extract the key indicators that affect the scores [9]. Taking those key indicators into account, final scores are computed.

Predicting accurate student performance is still a challenging task due to various issues involved in it. Inefficiency and use of improper attributes are some key difficulties with performance prediction systems. This research explores performance prediction using machine learning algorithms considering best attributes.

Machine Learning

Machine learning algorithms create a mathematical model with the help of sample historical data, referred to as training data, that aids in making predictions or judgments without being explicitly programmed. In order to create predictive models, machine learning combines computer science and statistics [1]. Machine learning is the process of creating or employing algorithms that learn from past data. The more information we supply, the better our performance will be.

In this work, Regression based machine learning algorithms are employed to predict a student's score based on his previous test results, failures, attendance etc. This falls into Supervised Learning where labeled data is provided for training the machine learning model and based on that, the model predicts the output for new data.

1.1 Description of the problem statement

The aim of this project is to predict student marks using supervised machine learning algorithms (Regression Models) while taking into account all the variables that influence a student's performance. The model will be given past data of the student which include his previous test results. This data is given as input to employed ml algorithms to predict results. Efficient algorithm is proposed for final prediction from the comparative analysis carried out of all the outcomes.

1.2 Why did we select Student Performance Prediction ?

Many research papers discuss how Student Performance Prediction is carried out and how their technique might help to solve a specific problem, such as using Student Performance Prediction as a tool for predicting marks. However, we found only a few research publications that discussed predicting marks considering all the factors that affect it, which leads us to the rationale behind our study on this topic.

Our goal was to employ a regression model that could predict student's accurate scores and that were helpful to students in improving in areas where they were trailing.

1.3 State of the Art

Traditional approaches which relied majorly more on data mining and extracting key indicators have changed a lot with advancements in machine learning which outperformed the previously used techniques.

Some major previous work are :

- Linear Discriminant Analysis and SVM algorithms are used to predict student marks. Modeling a small dataset to get a credible accuracy rate [1].
- Neural Network is used as a clustering and classification layer on top of the data and helps to cluster the unlabelled data [2].
- Used DataMining techniques to evaluate impact of different attributes. Ensemble filtering technique is used to get better accuracy[3].

Recent Solutions after emerged evolutions in Machine Learning :

Author	**Areas of Interest**
Lubna Mahmoud Abu Zohair	Extracting key Attributes & Predicting Student performance on small Dataset
Yogesh Gupta	Deriving correlations of student's performance with psychographic attributes using Deep Learning
Rabiul Islam	Evaluating impact of various Attributes on student Performance Using Data Mining Techniques
Gamal Alkawsi	Extracting key Attributes to predict Student performance

Table. 1.1 Solutions till date

1.4 Overview

Following this introduction chapter in this thesis next we talked about our proposed approach where we explained about the different modules we have implemented and the flow of our implementation. Then, we have deeply explained the concept behind what all our used modules are, how they work till chapter-3. In the next chapter we have also discussed the details of the dataset on which we have trained our model . Then, we explained how Data Preprocessing is done, how our model is prepared which also contains some code screenshots, after that we have added some output snippets which we took while testing our model. At the last we mentioned what were the challenges and what else we can do to make our product more better and useful to students.

CHAPTER - 2

THE PROPOSED APPROACH

A walkthrough of the implemented model

Our Approach typically consists these major stages :

Data Pre-processing, Attribute Selection, Selecting Machine Learning Model, Integration with web application.

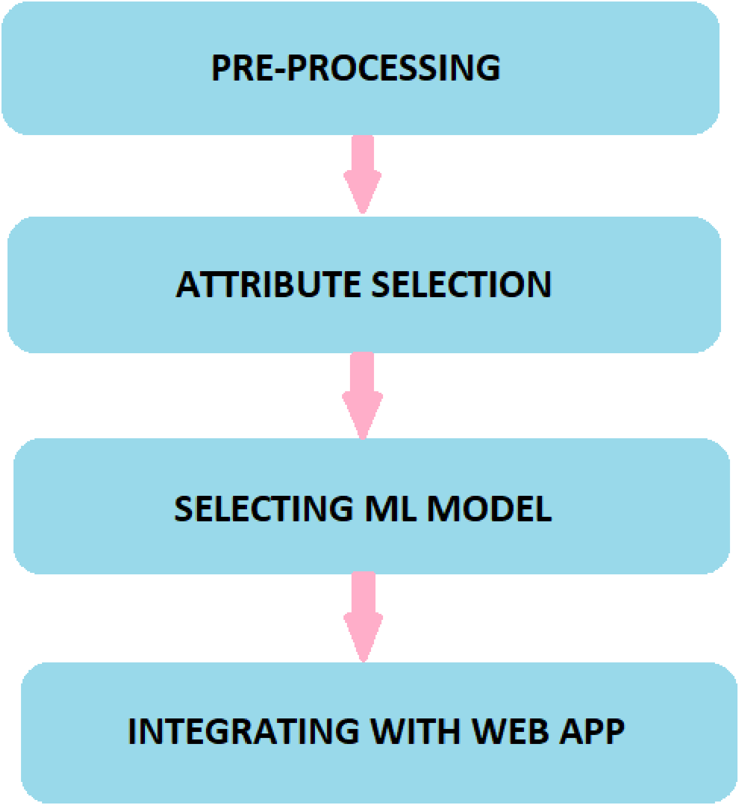


Fig. 2.1 Basic Model

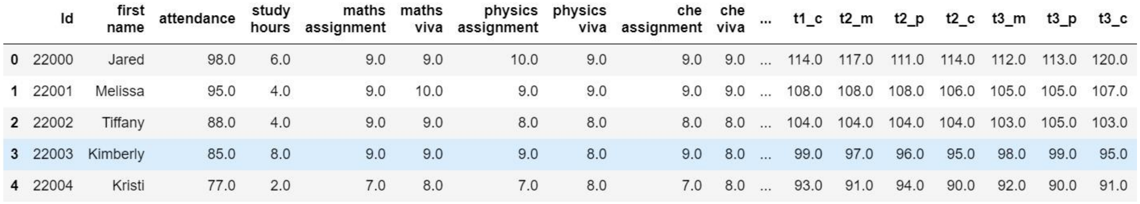
2.1 Description of the model

Student Performance Prediction involves four major steps. Here is the detailed description of them.

2.1.1 Data Pre-Processing

Data preprocessing is a process of preparing the raw data and making it suitable for a machine learning model. It is not always the case that you come across clean and formatted data. If you train your model using faulty or dirty data, you'll end up with an inadequately trained model that won't be useful in your research [10]. As a result, we do data preprocessing tasks. Machine learning model's accuracy and efficiency are improved using Data Preprocessing. Unnecessary attributes from the dataset will be removed in this phase. Pre-processing phase includes: Getting appropriate dataset, Data quality assessment, Data cleaning, Features encoding, Normalization [2,6].

Dataset used in this paper named ‘Student Marks’ which is collected from Kaggle has 10000 instances and 32 features. In this dataset, student's attributes are categorized into three groups. These are demographic attributes, academic background features and parent's participation in student's learnings.



	Id	first name	attendance	study hours	maths assignment	maths viva	physics assignment	physics viva	che assignment	che viva	...	t1_c	t2_m	t2_p	t2_c	t3_m	t3_p	t3_c
0	22000	Jared	98.0	6.0	9.0	9.0	10.0	9.0	9.0	9.0	...	114.0	117.0	111.0	114.0	112.0	113.0	120.0
1	22001	Melissa	95.0	4.0	9.0	10.0	9.0	9.0	9.0	9.0	...	108.0	108.0	108.0	106.0	105.0	105.0	107.0
2	22002	Tiffany	88.0	4.0	9.0	9.0	8.0	8.0	8.0	8.0	...	104.0	104.0	104.0	104.0	103.0	105.0	103.0
3	22003	Kimberly	85.0	8.0	9.0	9.0	9.0	8.0	9.0	8.0	...	99.0	97.0	96.0	95.0	98.0	99.0	95.0
4	22004	Kristi	77.0	2.0	7.0	8.0	7.0	8.0	7.0	8.0	...	93.0	91.0	94.0	90.0	92.0	90.0	91.0

Fig. 2.2 Dataset

Demographic features of a student are their roll no, place, dob etc., academic features of a student are their academic curriculum's records like student attendance, study hours, previous test scores, viva scores etc., and parent's participation on student's learnings are parent's involvement in the student’s study [9].

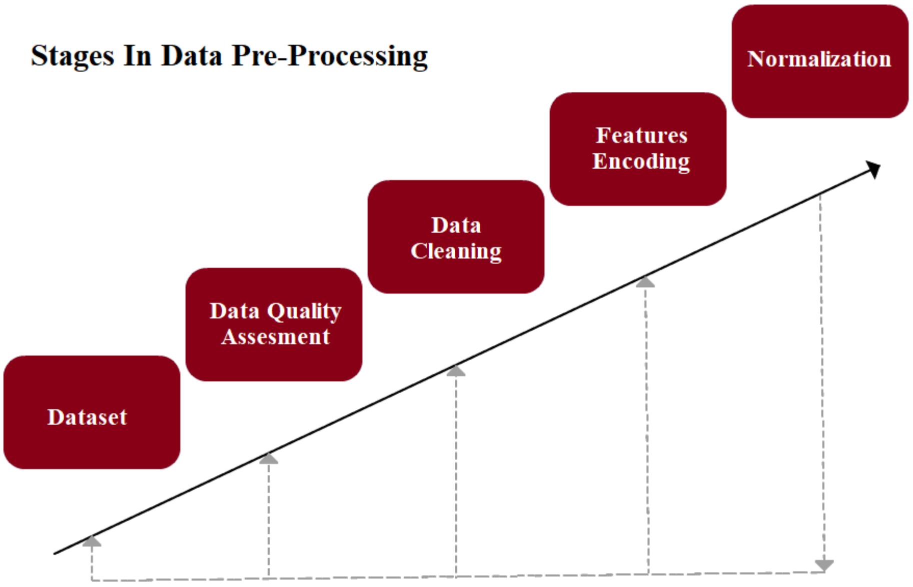


Fig. 2.3 Data Pre-processing Phases

Dataset is passed through various pre-processing phases to get clean & well-formatted data.

- **Data quality assessment**

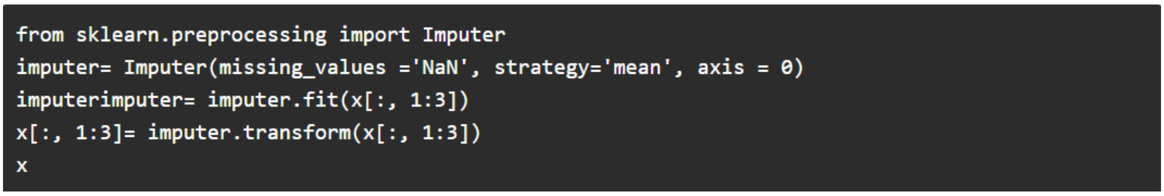
  The aim of a data quality assessment is to identify incorrect data. Mismatched data types, Missing Data, Data Outliers etc., are some data anomalies looked after.

- **Data cleaning**

  Data cleaning is the process of fixing or removing incorrect, corrupted, incorrectly formatted, duplicate or incomplete data within a dataset. Data cleaning will correct all the inconsistent data identified in data quality assessment.

  Irrelevant attributes to this study (like: Date Of Birth, mail id, Academic Year, and Phone no) were removed. Mismatched data types are reformatted. Missing values can be manually filled, however this is not suggested if the dataset is large.

When the data is normally distributed, the attribute's mean value can be used to replace the missing value; when the data is not normally distributed, the attribute's median value can be utilized. Duplicate observations will happen most often during data collection. So, Deduplication is carried out to remove duplicates.



```
from sklearn.preprocessing import Imputer
imputer= Imputer(missing_values ='NaN', strategy='mean', axis = 0)
imputerimputer= imputer.fit(x[:, 1:3])
x[:, 1:3]= imputer.transform(x[:, 1:3])
x
```

Fig. 2.4 Code Snippet of Missing Values replaced by Mean in Normal Distribution

- **Features encoding**

  It is the process of converting categorical data in a dataset into numerical data. As most machine learning models can only comprehend numerical data and not data in written form, feature encoding is essential.

- **Normalization**

  Normalization scales your data into a regularized range, making it easier to compare.

2.1.2 Attributes Selection

When creating a predictive model, attributes selection is the process of decreasing the number of input variables.

The number of input variables should be reduced to lower the computational cost of modelling and to improve the model's performance [2].

It's the process of selecting appropriate characteristics for the machine learning model based on the sort of problem we are attempting to solve automatically. We accomplish this by retaining or removing critical attributes without altering them. It aids in the reduction of noise in our data as well as the size of our input data. Attribute selection limits the model's input variable by only using relevant data in order to reduce overfitting [10].

Heat map visualization and hierarchical clustering approaches were utilized to aid in seeing the relationships between variables and determining the key qualities that could aid with Marks prediction.

- **Hierarchical clustering**

  Hierarchical clustering is an unsupervised clustering method that involves creating groups with a dominating top-to-bottom ordering. Based on their similarity, the algorithm separates objects into clusters [2]. The endpoint is a collection of clusters, each of which is distinct from the others but has broadly comparable things within it.

- **Heat map visualization**

  Heat Maps are the graphical representations of data that utilize color-coded systems. Heatmaps use color variations to visualize the data. Heat Maps are primarily used to better represent the amount of events within a dataset and to guide users to the most important sections on data visualizations.

  They are used to cross-examine multivariate data in a tabular format by inserting variables in the rows and columns and coloring the cells inside the table. Heatmaps are useful for showing variance across multiple variables, exhibiting whether variables are comparable to one another, and detecting whether there are any correlations between them [6].

2.1.3 Selecting Machine Learning Model

After Pre-Processing & Attribute Selection, Dataset is passed for Training & Testing. 80% of the dataset goes into the training set and 20% of the dataset goes into the testing set. Dataset is trained by various Machine Learning Models (Elastic net regression, Linear Regression, Random Forest, Decision Tree & Gradient Boost). Best performing model is considered for final prediction.

It is important to achieve accuracy on training data, but it is equally important to obtain a real and approximate result on unknown data, or the model will be useless. So, in order to construct and deploy a generalized model, we must evaluate it using various metrics, which allows us to better optimize, fine-tune, and acquire a better outcome.

Metrics used for evaluating Machine Learning models to Predict Marks:

i) Mean Absolute Error

Calculates the absolute difference between actual and predicted values.

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n} \blacksquare \mid x_i - x \mid$$

where,

xi: Actual value for the ith observation

x: Calculated value for the ith observation

n: Total number of observations

ii) Root Mean Squared Error

RMSE is the simple root of mean squared error (MSE calculates squared difference between actual and predicted value).

$$\text{RMSE} = \frac{1}{n}(\sum_{i=1}^{n} \blacksquare \mid y_i - x_i \mid^2)\frac{1}{2}$$

where,

xi: Actual value for the ith observation

yi: Calculated value for the ith observation

n: Total number of observations

iii) R Squared(R2)

R2 score tells the performance of the model, not the loss in an absolute sense that how many wells did the model perform.

2.1.4 Integrating with web application

Web Application is designed using Django. Our proposed Machine Learning Model is integrated with this web application which takes student past data as input and predicts final marks.

CHAPTER - 3

REGRESSION

An Overview

Regression Analysis in Machine Learning

Regression analysis is a statistical method to model the relationship between a dependent (target) and independent (predictor) variables. It is used for predictive modelling in machine learning, in which an algorithm is used to predict continuous outcomes and to find the trends in data. By performing the regression, we can confidently determine the most important factor, the least important factor, and how each factor is affecting the other factors.

Elastic Net Regression

Linear Regression

Decision Tree

Random Forest

Gradient Boost

Fig. 3.1 Regression Models for Experimentation

3.1 Linear Regression

Linear regression is a supervised learning machine learning algorithm. It carries out a regression task. Based on independent variables, regression models a target prediction value. It is mostly used in forecasting and determining the relationship between variables [7]. Different regression models differ in terms of their relationship between dependent and independent variables, as well as the amount of independent variables they employ [8].

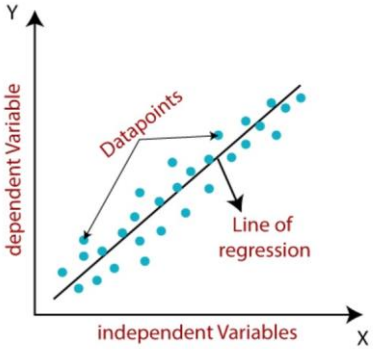


Fig. 3.2 Linear Regression

Linear regression is used to predict the value of a dependent variable (y) based on the value of an independent variable (x). As a result of this regression technique, a linear relationship between x (input) and y (output) is discovered (output). As a result, the term Linear Regression was coined.

Mathematically, we can represent Linear Regression as:

$$y= a_0+a_1x+ \varepsilon$$

where,

y = Dependent Variable (Target Variable)

x = Independent Variable (predictor Variable)

a_0 = intercept of the line (Gives an additional degree of freedom)

a_1 = Linear regression coefficient (scale factor to each input value).

ε = random error

Although linear regression is a useful tool for analyzing correlations between variables, it is not recommended for most practical applications since it oversimplifies real-world situations by assuming a linear relationship between variables.

3.2 Elastic Net Regression

Elastic-Net Regression is a modification of Linear Regression which shares the same hypothetical function for prediction.

where,

$$\frac{1}{m}\sum_{i=1}^{m}\left(y^{(i)}-h\left(x^{(i)}\right)\right)^{2}$$

m = Total number of training examples in the dataset

h(x(i)) = Hypothetical prediction function is denoted

y(i) = value of the target variable for the i^{th} training example

Overfitting is a problem with linear regression, and it can't handle collinear data. When the dataset contains a large number of features, some of which are irrelevant to the predictive model.

This complicates the model since the test set prediction is too imprecise (or overfitting). A high-variance model like this does not generalize to fresh data. To address these concerns, we combine L-2 and L-1 norm regularization in order to get the benefits of both Ridge and Lasso at the same time. The resulting model outperforms Lasso in terms of predictive power. It selects features while also simplifying the hypothesis.The following is the adjusted cost function for

Mathematically, we can represent Elastic-Net Regression as :

$$\frac{1}{m}\left[\sum_{l=1}^{m}\left(y^{(i)}-h\left(x^{(i)}\right)\right)^{2}+\lambda_{1}\sum_{j=1}^{n}w_{j}+\lambda_{2}\sum_{j=1}^{n}w_{j}^{2}\right]$$

where,

w(j) = weight for the jth feature

n = number of features in the dataset

λ_1 = regularization strength for the L-1 norm

λ_2 = regularization strength for the L-1 norm

The penalties from both the lasso and ridge approaches are used to regularize regression models in elastic net linear regression. The strategy combines the lasso and ridge regression methods by learning from their flaws to better statistical model regularization. The Elastic-Net approach overcomes lasso's constraints, such as when high-dimensional data requires only a few samples. The elastic net combines the best aspects of both lasso and ridge regression.

3.3 Decision Tree

Decision Tree is a type of supervised learning technique that can be used to solve classification and regression problems. Internal nodes represent dataset attributes, branches represent decision rules, and each leaf node represents the output of those decisions and do not contain any more branches. It's a graphical depiction for obtaining all feasible solutions to a problem/decision depending on certain parameters [5]. A decision tree simply asks a question and divides the tree into subtrees based on the answer (Yes/No).

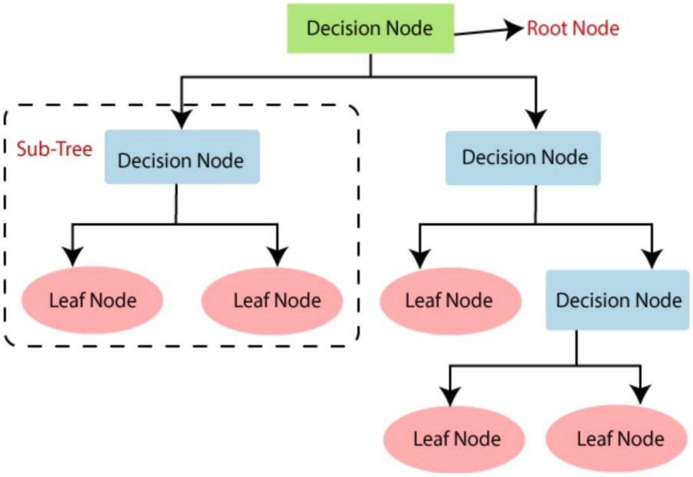


Fig. 3.3 Decision Tree

In a decision tree, the mechanism for deciding the class of a given dataset begins at the root node. Based on the comparison, this algorithm compares the values of the root attribute to the values of the record (actual dataset) attribute, then follows the branch and jumps to the next node. The method compares the value of the attribute with the values of the other sub-nodes before moving on to the next node. It continues in this manner until it reaches the leaf node of the tree.

Decision trees need less effort for data preparation during pre-processing than other algorithms, yet a slight change in the data might cause a big change in the structure of the decision tree, resulting in instability [5]. Because of the intricacy and time required, decision tree training is relatively costly.

3.4 Random Forest

Random Forest is a well-known machine learning algorithm that uses the supervised learning method. In machine learning, it can be used for both classification and regression problems. It is based on ensemble learning, which is a method of integrating several classifiers to solve a complex problem and increase the model's performance.

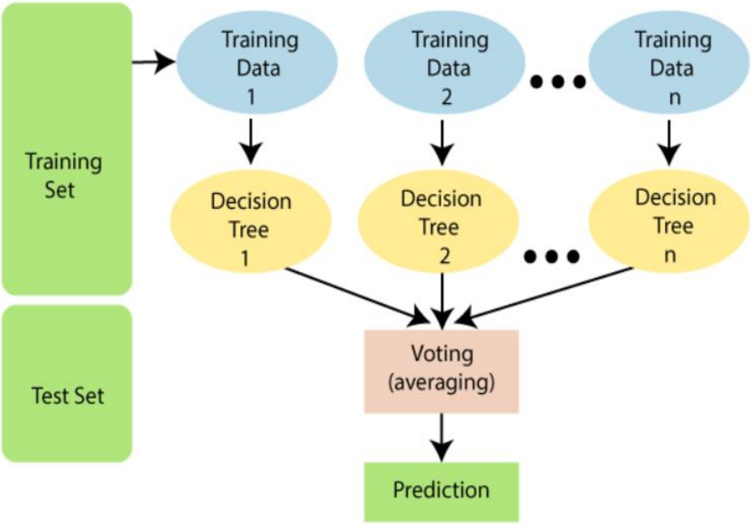


Fig. 3.4 Random Forest

It creates decision trees from various samples and averages the results to increase the dataset's predicted accuracy. The bigger the number of trees in the forest, the more accurate it is and the problem of overfitting is avoided. Random Forest algorithm does not support null values therefore the missing values will be handled in order to maintain the accuracy of a large proportion of the data. Random forest utilizes bootstrapping to train each decision tree with different sub-samples of data.

Unlike curve-based algorithms, non linear parameters have no effect on the performance of a Random Forest. So, if the independent variables have a lot of non-linearity, Random Forest may beat other curve-based algorithms, but it builds a lot of trees (unlike a decision tree, which only has one tree) and combines their results [5]. In the Python sklearn library, it builds 100 trees by default. This approach necessitates a significant increase in processing power and resources.

3.5 Gradient Boosted

Gradient boost is a popular boosting algorithm. Each predictor in gradient boosting corrects the error of its predecessor. Unlike Adaboost, the training instance weights are not adjusted; instead, each predictor is trained using the predecessor's residual errors as labels. The diagram below shows how regression problems are solved using gradient boosted trees.

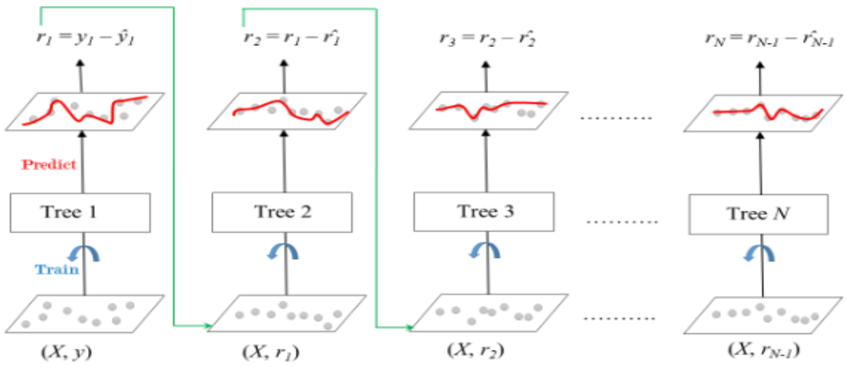


Fig. 3.5 Gradient Boosted

N trees make up the ensemble. The feature matrix X and the labels y are used to train Tree1. The training set residual errors r_1 are calculated using the predictions labeled $\widehat{y1}$. The feature matrix X and the residual errors r_1 of Tree-1 are then used as labels to train Tree-2. The residual r_2 is calculated using the projected results $\widehat{r1}$. The technique is continued until all of the ensemble's N trees have been trained.

Each tree predicts a label and final prediction is given by the formula,

$$y = y_1 + (eta * r_1) + (eta * r_2) + ... + (eta * r_N)$$

where ,

eta is the learning which ranges between 0 and 1.

As every classifier is required to rectify the errors of the predecessors, boosting is sensitive to outliers. As a result, the technique is too reliant on outliers. Another drawback is that scaling up the process is very impossible. Because each estimator is based on prior predictors, the technique is difficult to simplify.

CHAPTER - 4

DATASET

Dataset used in this work is “Student Marks” by Ashish Mehra collected from Kaggle. It comprises 32 different attributes and data of 10000 students.

There are three categories of marks in this dataset, as well as additional factors such as attendance, study hours, viva marks, parental involvement, failures etc. This dataset is multivariate, which is one of its characteristics. A multivariate dataset is one that has two or more than two variables.

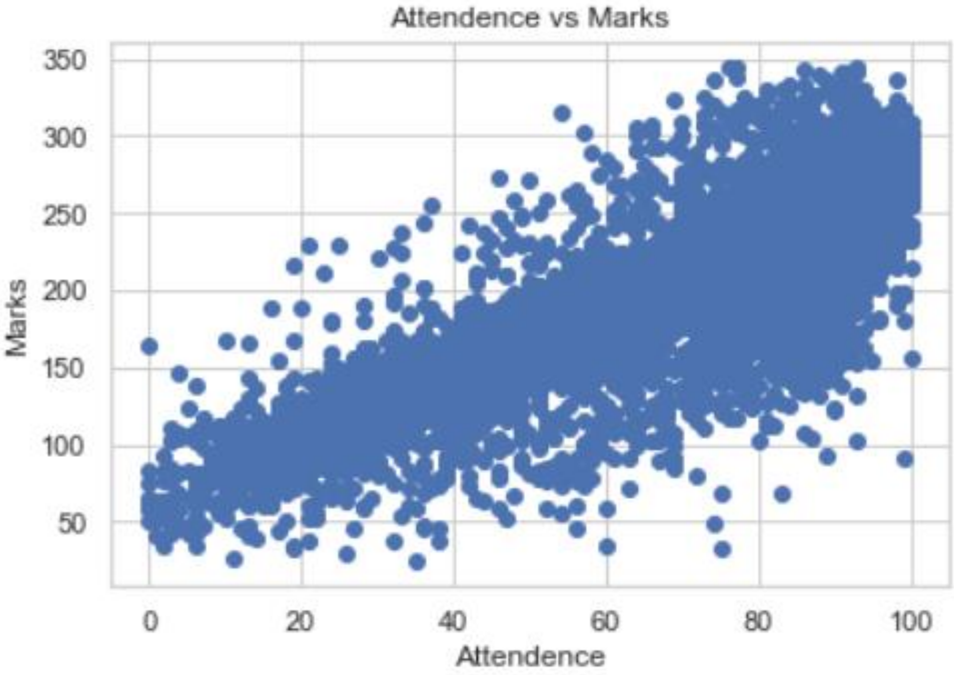


Fig. 4.1 Attendance vs Marks

According to the above data visualization, we can say that attendance and the marks obtained have a linear relationship. This means the more students attend the classes, the better they can score. Also several other factors that influence student marks, Attendance is one of the most correlating attributes.

From Fig 4.1, It is evident that a vast set of students has marks ranging 150-250 & attendance 75-100. Also, some low scored candidates have high attendance and v.v.

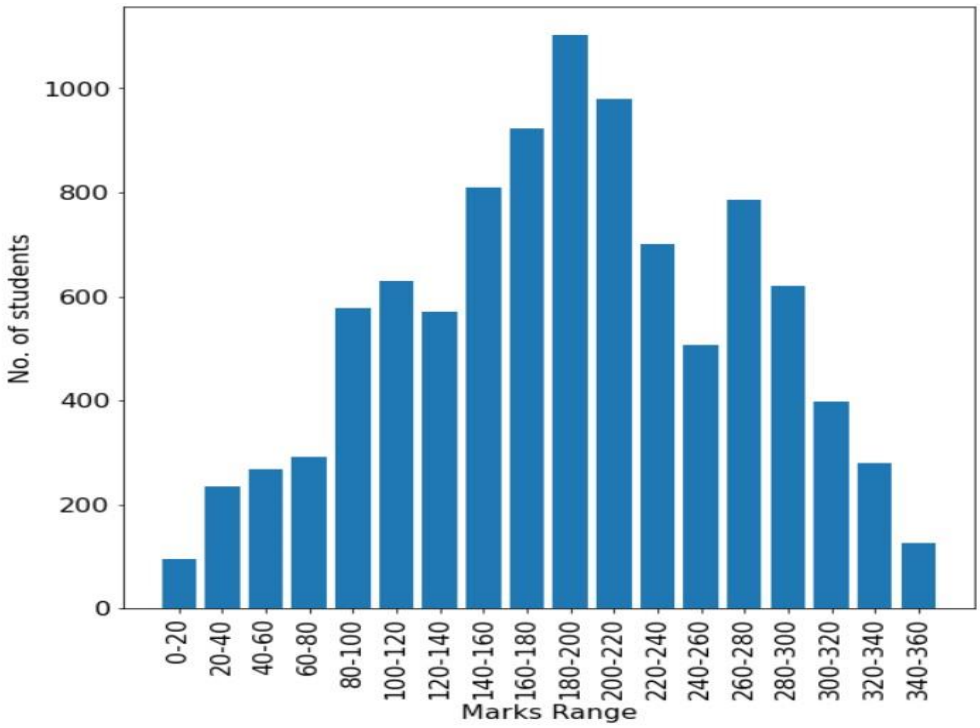


Fig. 4.2 Students vs Marks

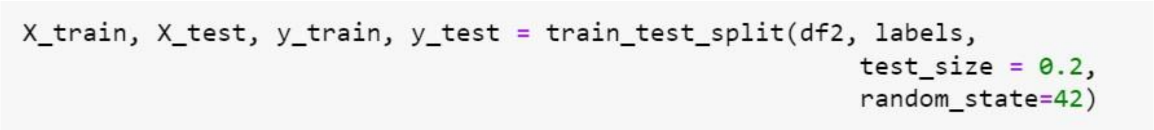
Fig 4.2 shows, Large group of students has average marks (140-240) and Students with high & low marks are few.

Marks is our target variable, So this falls under Supervised Regression ML. Because, We have a collection of training data with known targets, and we want our model to learn to predict marks based on the other factors during training.

In the Data Pre-Processing Phase, we balanced the dataset by imputing missing values, normalizing a few attributes like parental involvement where we scaled it from 1-10. Also removed some irrelevant features like his personal data (dob, phone no, mail id etc.,) & features encoding is carried out to convert categorical data to numerical data.

We used correlation values to perform attribute selection (also used hierarchical clustering Techniques). Finally we retained just the six attributes that had the greatest impact on the final marks.

Attendance, Study Hours, Viva Scores, Assignment Scores, Parental Involvement & Failures are the attributes with high correlation value i.e they have huge impact on marks. So these attributes are considered for training the model against marks. We also split the dataset into a training and testing set in 80:20 intuition. Splitting data into training and testing sets is necessary because we need to have a hold-out test set to evaluate our model and make sure it is not overfitting to the testing data.



```
X_train, X_test, y_train, y_test = train_test_split(df2, labels,
                                                     test_size = 0.2,
                                                     random_state=42)
```

Fig. 4.3 Dataset Splitting

After splitting the Dataset we applied different regression algorithms like Random forest, Linear regression, Gradient Boost, ElasticNet Regression & Decision Tree.

Software Requirements:

● Operating system	:	Windows XP/7/10/11
● Coding Language	:	Python
● Development environment	:	Anaconda, Jupyter
● Dataset	:	Students Marks
● IDE	:	Jupyter Notebook

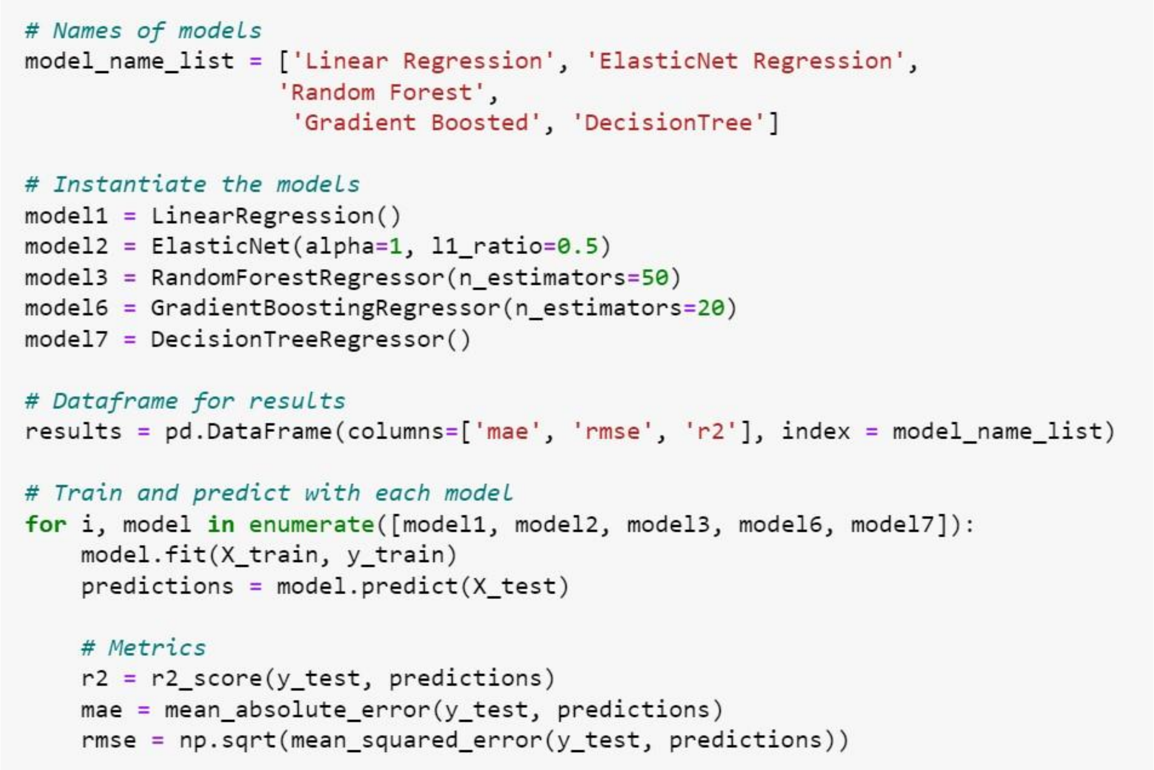
CHAPTER - 5

RESULTS

Comparative Analysis

5.1 Training & Testing

We have trained our dataset with Linear Regression, Elastic Net Regression, Decision Tree, Random Forest and Gradient Boosted. Then evaluated the models using the metrics MAE, MSE, R2. Models are evaluated for every individual subject i.e. Maths, Physics and Chemistry.



```
# Names of models
model_name_list = ['Linear Regression', 'ElasticNet Regression',
                  'Random Forest',
                   'Gradient Boosted', 'DecisionTree']

# Instantiate the models
model1 = LinearRegression()
model2 = ElasticNet(alpha=1, l1_ratio=0.5)
model3 = RandomForestRegressor(n_estimators=50)
model6 = GradientBoostingRegressor(n_estimators=20)
model7 = DecisionTreeRegressor()

# Dataframe for results
results = pd.DataFrame(columns=['mae', 'rmse', 'r2'], index = model_name_list)

# Train and predict with each model
for i, model in enumerate([model1, model2, model3, model6, model7]):
    model.fit(X_train, y_train)
    predictions = model.predict(X_test)

    # Metrics
    r2 = r2_score(y_test, predictions)
    mae = mean_absolute_error(y_test, predictions)
    rmse = np.sqrt(mean_squared_error(y_test, predictions))
```

Fig. 5.1 Code Snippet of Training and Testing

5.2 Linear Regression

Table 5.1 shows the results of the linear regression model on our dataset :

	MAE	RMSE	R2 Score
Math	2.663	4.641	0.976
Physics	2.635	4.630	0.976
Chemistry	2.627	4.594	0.977

Table. 5.1 Linear Regression Results

5.3 Elastic Net Regression

Table 5.2 shows the results of the Elastic Net regression model on our dataset :

	MAE	RMSE	R2 Score
Math	2.657	4.658	0.975
Physics	2.640	4.654	0.970
Chemistry	2.634	4.628	0.976

Table. 5.2 Elastic Net Regression Results

5.4 Decision Tree

Table 6.3 shows the results of the decision tree model on our dataset :

	MAE	RMSE	R2 Score
Math	2.733	4.303	0.979
Physics	2.790	4.454	0.978
Chemistry	2.820	4.501	0.977

Table. 5.3 Decision Tree Results

5.5 Random Forest

Table 6.4 shows the results of the random forest model on our dataset :

	MAE	RMSE	R2 Score
Math	2.151	3.208	0.988
Physics	2.120	3.189	0.988
Chemistry	2.233	3.310	0.988

Table. 5.4 Random Forest Results

5.6 Gradient Boosted

Table 6.5 shows the results of the gradient boosted model on our dataset :

	MAE	RMSE	R2 Score
Math	4.121	5.272	0.969
Physics	4.099	5.272	0.969
Chemistry	4.283	5.526	0.966

Table. 5.5 Gradient Boosted Results

5.7 Comparative Analysis

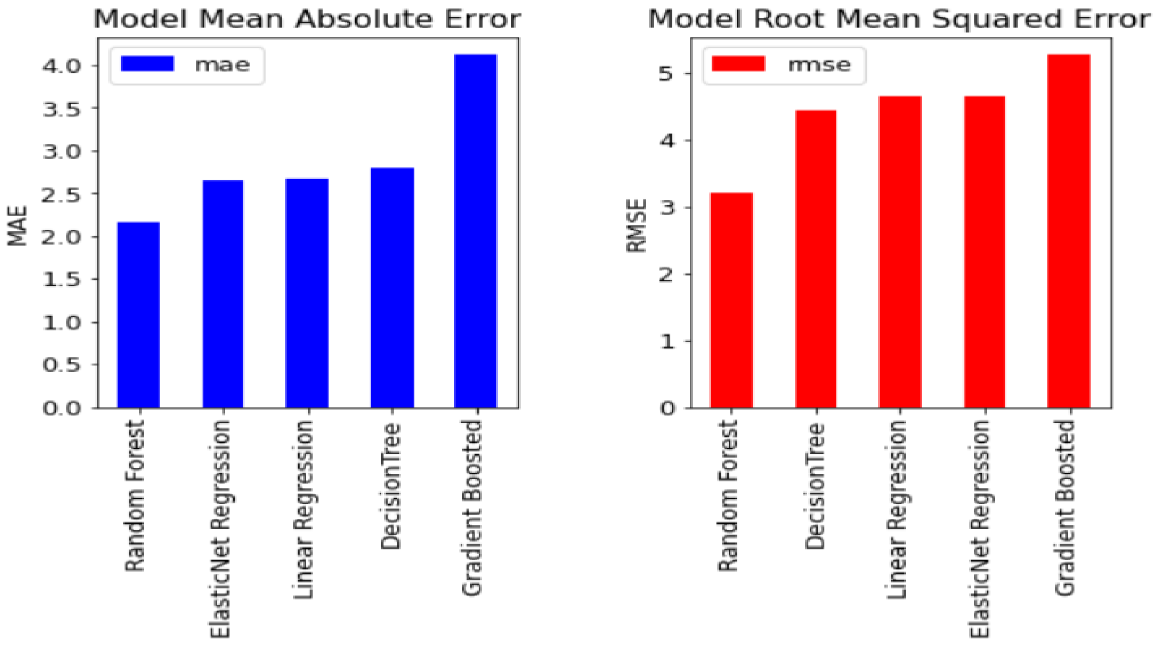
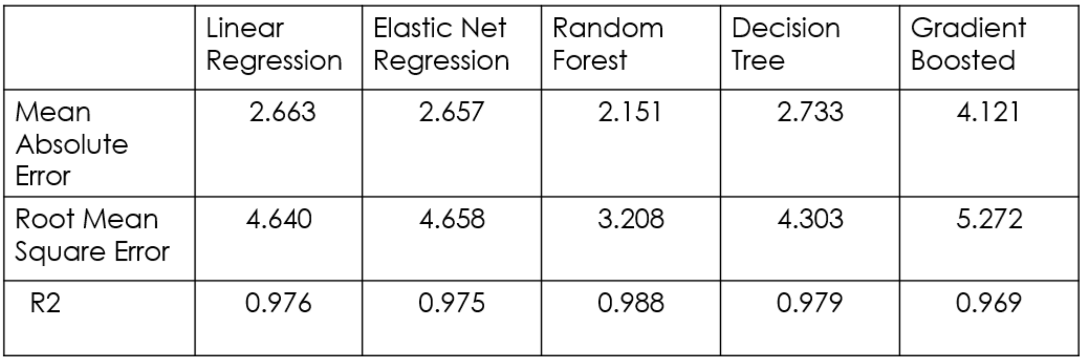


Fig. 5.2 Comparison of Results

These are the results we got after implementing all these regression models. Comparing the results, we can say that **Random Forest** is performing better than other regression models. So, Random Forest is our proposed model for web application.



	Linear Regression	Elastic Net Regression	Random Forest	Decision Tree	Gradient Boosted
Mean Absolute Error	2.663	2.657	2.151	2.733	4.121
Root Mean Square Error	4.640	4.658	3.208	4.303	5.272
R2	0.976	0.975	0.988	0.979	0.969

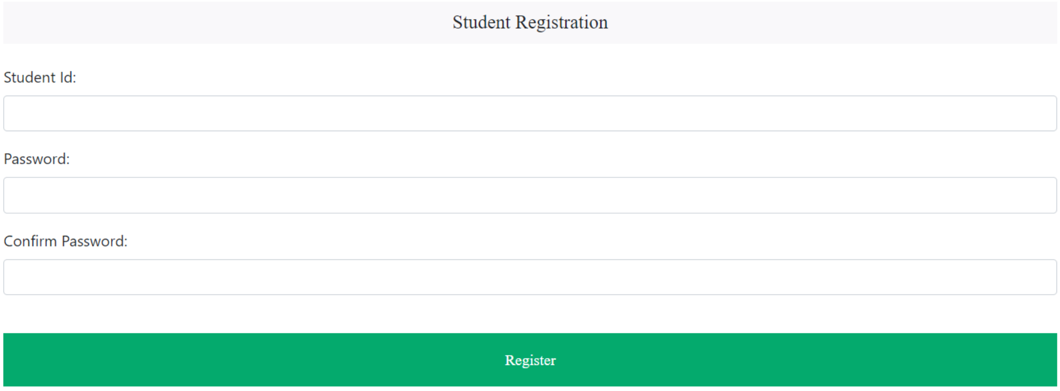
Table. 5.6 Comparison of results

CHAPTER - 6

WEB APPLICATION

We aimed to create an end-to-end solution for institutions. So, we have developed a web application using Django. This acts as an interface between students and teachers. Our application has two types of users - Institution & Student. Students who are in the Institute's database can only register. Where institutions can upload & update student data. Our ML model will take it as input and predict student's marks. These predicted marks will be shown to students in their end.

Some Web Application Snapshots :

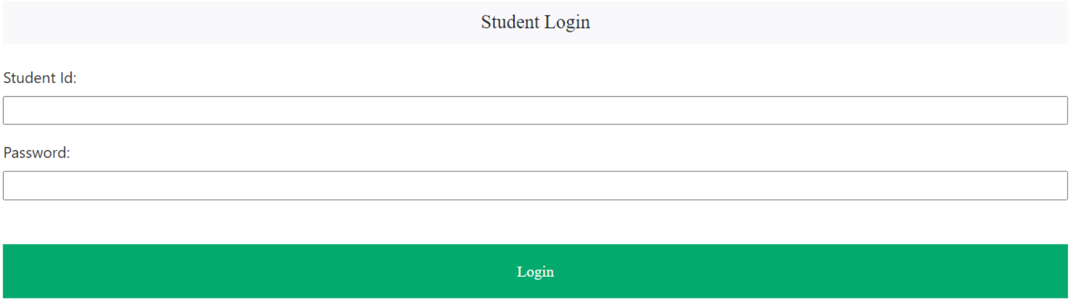

Student Registration

Student Id:

Password:

Confirm Password:

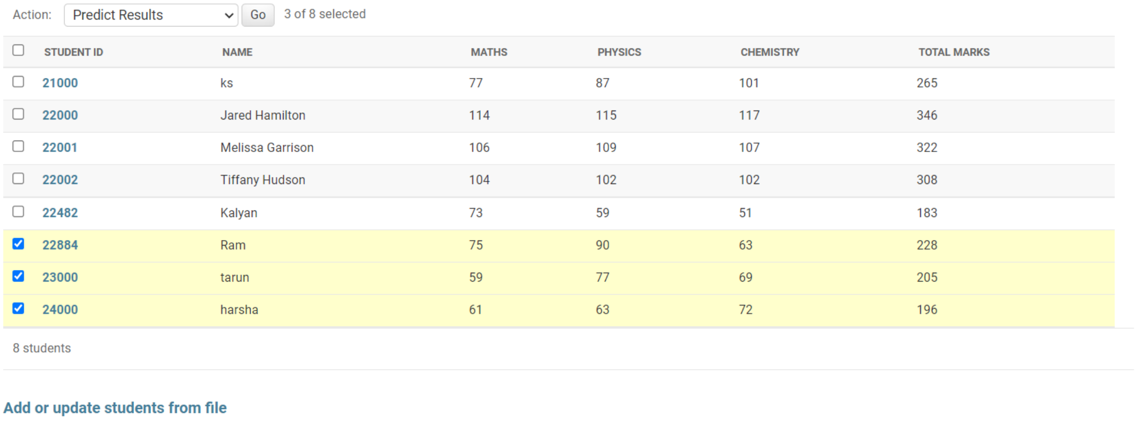
Fig. 6.1 Registration Page


Student Login

Student Id:

Password:

Fig. 6.2 Login Page



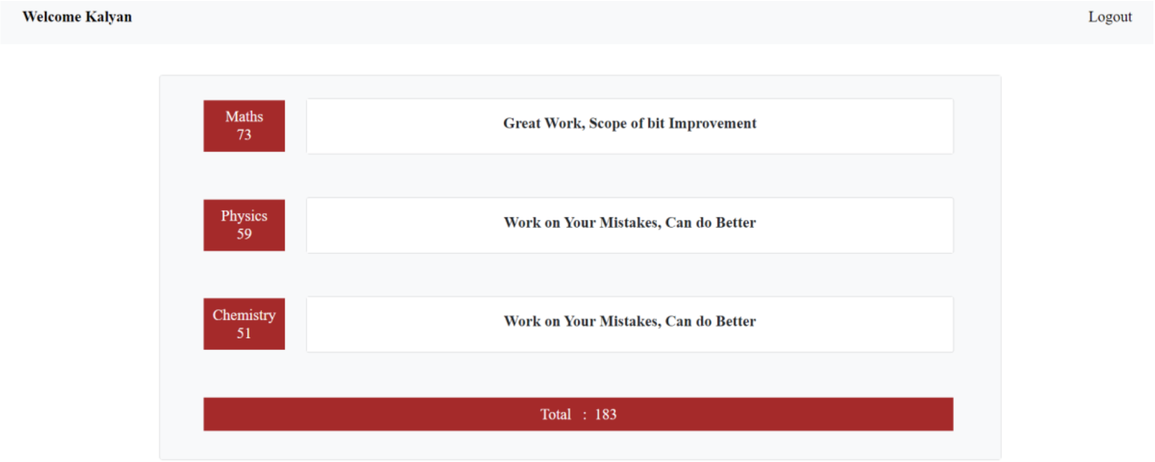
Action: Predict Results Go 3 of 8 selected

□	STUDENT ID	NAME	MATHS	PHYSICS	CHEMISTRY	TOTAL MARKS
□	21000	ks	77	87	101	265
□	22000	Jared Hamilton	114	115	117	346
□	22001	Melissa Garrison	106	109	107	322
□	22002	Tiffany Hudson	104	102	102	308
□	22482	Kalyan	73	59	51	183
☑	22884	Ram	75	90	63	228
☑	23000	tarun	59	77	69	205
☑	24000	harsha	61	63	72	196

8 students

Add or update students from file

Fig. 6.3 Student's Predicted Results in Database

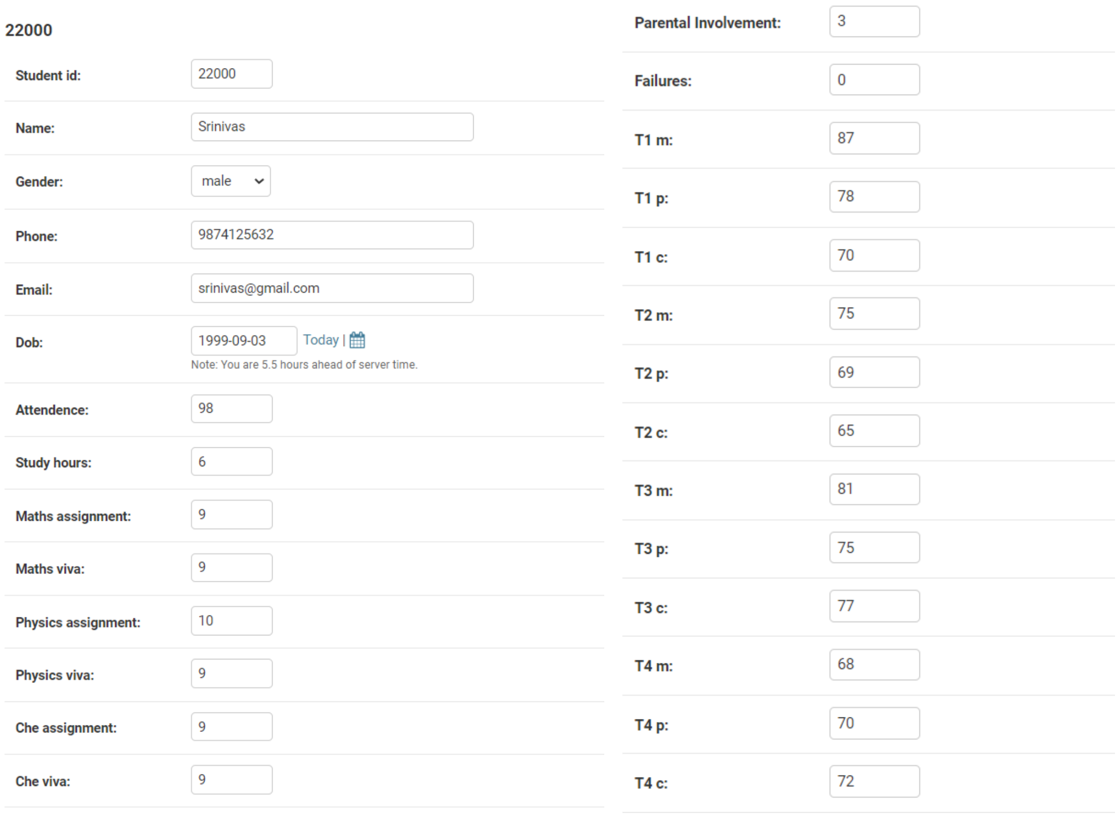


Welcome Kalyan Logout

Maths 73	**Great Work, Scope of bit Improvement**
Physics 59	**Work on Your Mistakes, Can do Better**
Chemistry 51	**Work on Your Mistakes, Can do Better**

Total : 183

Fig. 6.4 Predicted Marks on Student's end



22000

Student id:	22000
Name:	Srinivas
Gender:	male
Phone:	9874125632
Email:	srinivas@gmail.com
Dob:	1999-09-03 Today | 📅 Note: You are 5.5 hours ahead of server time.
Attendence:	98
Study hours:	6
Maths assignment:	9
Maths viva:	9
Physics assignment:	10
Physics viva:	9
Che assignment:	9
Che viva:	9

Parental Involvement:	3
Failures:	0
T1 m:	87
T1 p:	78
T1 c:	70
T2 m:	75
T2 p:	69
T2 c:	65
T3 m:	81
T3 p:	75
T3 c:	77
T4 m:	68
T4 p:	70
T4 c:	72

Fig. 6.5 Student Data

CONCLUSION

The prediction of student performance is getting difficult day by day. Student’s performance prediction and Its analysis are necessary to help educators in the identification of student’s weaknesses and to improve their scores and learning activities. The aim of this study was to predict student marks based on their previous scores and demographics . We worked towards making an end-to-end platform solution for colleges. This project aimed to get better results using Supervised Regression ML models in the prediction of a student's performance.

We used 10,000 students' data to train the models. The mean absolute error in predicting the result of one subject is 2.151 for the best model, and 4.121 for the worst. We also used two other metrics to compare the machine learning models namely Root Mean Square Error and R2 Score. Based on the results our proposed model is Random Forest.

FUTURE SCOPE

- Behavioral Attributes like a student's emotional state will be considered to get more accurate results.
- Our model can also be used to visualize in which topic a student is lacking in a particular subject.

REFERENCES

To understand the workflow we used some research papers few are listed here :-

1. Prediction of Student's performance by modeling small dataset size : Lubna Mahmoud Abu Zohair International Journal of Educational Technology in Higher Education.
   https://educationaltechnologyjournal.springeropen.com/articles/10.1186/s41239-0190160-3

2. Predict Student's Academic Performance and Evaluate the Impact of Different Attributes on the Performance Using Data Mining Techniques.
   https://ieeexplore.ieee.org/document/8412892/authors#authors

3. Student Performance Assessment and Prediction System using Machine Learning.
   https://ieeexplore.ieee.org/abstract/document/9036250

4. Predicting student's performance using machine learning and neural network methods.
   https://ieeexplore.ieee.org/document/9497185

5. Predicting at-Risk Students at Different Percentages of Course Length for Early Intervention Using Machine Learning Models
   https://ieeexplore.ieee.org/document/9314000

6. A data mining based survey on student performance evaluation system.
   https://ieeexplore.ieee.org/document/7238389

7. Student final grade prediction based on linear regression.
   http://www.ijcse.com/docs/INDJCSE17-08-03-080.pdf

8. Predicting Students' Final Exam Scores from their Course Activities
   https://ieeexplore.ieee.org/document/7344081

9. Predicting Students' State Examination Results based on Previous Grades and Demographics
   https://ieeexplore.ieee.org/document/9284401

10. Student's Academic Performance Evaluation Method Using Fuzzy Logic System
    https://ieeexplore.ieee.org/abstract/document/8934496