A Practical Model for the Evaluation of High School Student Performance Based on Machine Learning

Name: JAWAHARLAL BANOTHU
Roll No.: CH21BTECH11037
Department: Chemical Engineering

Introduction

The research paper taken for this project is “A Practical Model for the Evaluation of High School Student Performance Based on Machine Learning”.

MLs are currently highly advanced and can do more than scoring exams with the answer key. They can provide information about student performance and even perform more conceptual assessments such as scoring the essays or students engagements. MLs can collect information about how students acted and can evaluate this performance. Machine learning and data mining are powerful tools for instructors and institutions to explore the educational database; this has increased with the assistance of ML, and enabled decision-makers to extract information from data for decisions and policies. The application that ML uses in the educational database is called Educational Data Mining (EDM). The educational database is available at different levels of detail for different tasks and often expands in several software systems over some time. EDM tries to use features and patterns in a database to make an effective analysis for predicting student performance. This produced information can be used by data scientists, instructors, administrators, a student’s parents, etc. The evaluation of student performance refers to how much a student approaches the educational goals and specifies how well a student learned, how motivated to learn a student is or how good the teaching method was. The information obtained from the evaluation gives teachers this insight so they can make the right decisions to improve the learning of the student and give appropriate feedback. The individual differences of each student, such as personality, motivation, self-efficacy, intelligence and self-control, have a close relationship to his/her performance, so in this research, all these differences were covered by choosing the proper features. The curriculum contains mathematics, science, foreign languages and so on. High school students aged 12 to 18 years old are divided into four fields (streams): humanism, science, and technical and vocational streams. Choosing their stream is based on his/her grades and the results of his/her examination, not based on their interest. Furthermore, grades are determined on a scale between 0 and 20 in all levels of education and students are assessed during the semester and end of the semester as well. The lowest score to pass a lesson is 10. Studies conducted to evaluate student performance with the machine learning approach are referred to in the next section. After that, the model selection process, and the design and development of models are discussed, and the overall definition of each algorithm is presented. Furthermore, in the results section, the performance of each model on datasets is shown and is discussed.

Data Creation

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). In [Cortez and Silva, 2008], the two datasets were models under binary/five-level classification and regression tasks. Important note: the target attribute G3 has a strong correlation with attributes G2 and G1. This occurs because G3 is the final year grade (issued at the 3rd period), while G1 and G2 correspond to the 1st and 2nd period grades. It is more difficult to predict G3 without G2 and G1, but such prediction is much more useful.

Data set link: [UCI Machine Learning Repository: Student Performance Data Set]

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 - <15 min., 2 - 15 to 30 min., 3 - 30 min. to 1 hour, or 4 - >1 hour)
14 studytime - weekly study time (numeric: 1 - <2 hours, 2 - 2 to 5 hours, 3 - 5 to 10 hours, or 4 - >10 hours)
15 failures - number of past class failures (numeric: n if 1<=n<3, else 4)
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)

these grades are related with the course subject, Math or Portuguese:

31 G1 - first period grade (numeric: from 0 to 20)
31 G2 - second period grade (numeric: from 0 to 20)
32 G3 - final grade (numeric: from 0 to 20, output target)

Mapping strings to numeric values:

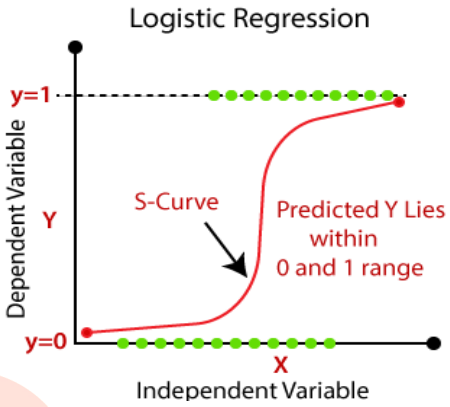
'school' --->> 'GP': 0, 'MS': 1
'sex' --->> 'M': 0, 'F': 1
'address' --->> 'U': 0, 'R': 1
'famsize' --->> 'LE3': 0, 'GT3': 1
'Pstatus' --->> 'T': 0, 'A': 1
'Mjob' --->> 'teacher': 0, 'health': 1, 'services': 2, 'at_home': 3, 'other': 4
'Fjob' --->> 'teacher': 0, 'health': 1, 'services': 2, 'at_home': 3, 'other': 4
'reason' --->> 'home': 0, 'reputation': 1, 'course': 2, 'other': 3
'guardian' --->> 'mother': 0, 'father': 1, 'other': 2
'schoolsup' --->> 'no': 0, 'yes': 1
'famsup' --->> 'no': 0, 'yes': 1
'paid' --->> 'no': 0, 'yes': 1
'activities' --->> 'no': 0, 'yes': 1
'nursery' --->> 'no': 0, 'yes': 1
'higher' --->> 'no': 0, 'yes': 1
'internet' --->> 'no': 0, 'yes': 1
'romantic' --->> 'no': 0, 'yes' : 1

Models used in Project:

Machine learning models were used to classify the performance of high school students. ML is a subset of computer science that is applied in mathematics and statistics. The main objective of supervised ML is to build an algorithm that can receive input data and use statistical analysis to predict outputs while being updated by adding new data. In the past decade, artificial intelligence and machine learning have been drawn toward education tasks, including the extraction of useful data and knowledge production to support decision making for students' progress. In this study, student performance was evaluated by supervised ML models. Studies that had similar data have used logistic regression (LRG), K-Nearest Neighbour (KNN), support vector machines (SVM) , and random forest (RFC) algorithms.

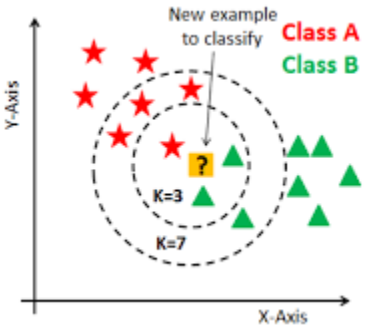
Logistic regression

Logistic Regression is **a "Supervised machine learning" algorithm that can be used to model the probability of a certain class or event**. It is used when the data is linearly separable and the outcome is binary or dichotomous in nature. That means Logistic regression is usually used for Binary classification problems.



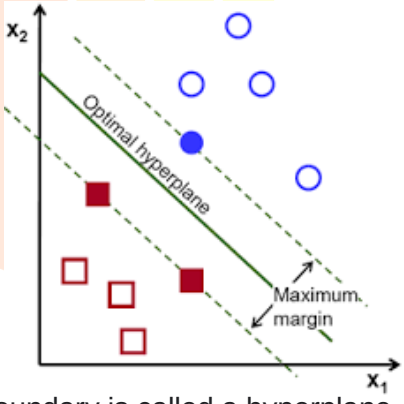
K-Nearest Neighbors

The abbreviation KNN stands for "**K-Nearest Neighbour**". It is a supervised machine learning algorithm. The algorithm can be used to solve both classification and regression problem statements. The number of nearest neighbours to a new unknown variable that has to be predicted or classified is denoted by the symbol 'K'.



Support vector machine

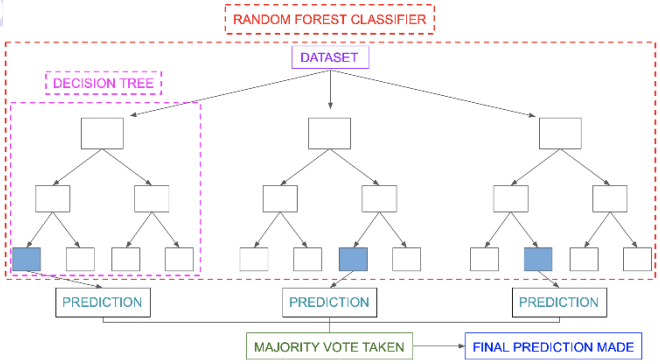
In machine learning, support-vector machines (SVMs, also support-vector networks) are **supervised learning models with associated learning algorithms that analyze data for classification and regression analysis**.



The goal of the SVM algorithm is **to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future**. This best decision boundary is called a hyperplane.

Random Forest Classifier

Random Forest Regression is a supervised learning algorithm for regression that use the ensemble learning method. Ensemble learning is the process of strategically generating and combining many models, such as classifiers or experts, to solve a computer intelligence problem.



A random forest produces good predictions that can be understood easily. It can handle large datasets efficiently. The random forest algorithm provides a higher level of accuracy in predicting outcomes over the decision tree algorithm.

Response Variable:

In this study student performance is taken as the response variable. In this case we have two type of outputs i.e. Pass(1) or Fail(0).

$$\text{Result} = \begin{cases} 0 \text{ (Fail)}, & \text{if } G3<10 \\ 1 \text{ (Pass)}, & \text{if } G3>=10 \end{cases}$$

Evaluation Metr or Machine Learning Models

Confusion Matrix in Machine Learning

Confusion Matrix helps us to display the performance of a model or how a model has made its prediction in Machine Learning.

Confusion Matrix helps us to visualize the point where our model gets confused in discriminating two classes. It can be understood well through a 2×2 matrix where the row represents the **actual truth labels**, and the column represents **the predicted labels.**

Truth	Fail	True Negative (TN)	False Positive (FP)
	Pass	False Negative (FN)	True Positive (TP)
		Fail	Pass
	Prediction		

- **True Positive:** This combination tells us how many times a model correctly classifies a positive sample as Positive.
- **False Negative:** This combination tells us how many times a model incorrectly classifies a positive sample as Negative.
- **False Positive:** This combination tells us how many times a model incorrectly classifies a negative sample as Positive.
- **True Negative:** This combination tells us how many times a model correctly classifies a negative sample as Negative.

Precision

Precision is the ratio of **true positives** to the total of the true positives and false positives. Precision looks to see how much junk positives got thrown in the mix. If there are no bad positives (those FPs), then the model had 100% precision. The more FPs that get into the mix, the uglier that precision is going to look.
To calculate a model's precision, we need the positive and negative numbers from the confusion matrix.

$$Precision = TP/(TP + FP)$$

Recall

Recall goes another route. Instead of looking at the number of false positives the model predicted, recall looks at the number of **false negatives** that were thrown into the prediction mix.

$$Recall = TP/(TP + FN)$$

The recall rate is penalized whenever a false negative is predicted. Because the penalties in precision and recall are opposites, so too are the equations themselves. Precision and recall are the yin and yang of assessing the confusion matrix.

Accuracy

Accuracy is the most intuitive performance measure and it is simply a ratio of correctly predicted observation to the total observations. One may think that, if we have high accuracy then our model is best. Yes, accuracy is a great measure but only when you have symmetric datasets where values of false positive and false negatives are almost same. Therefore, you have to look at other parameters to evaluate the performance of your model.

$$Accuracy = (TP+TN)/(TP+FP+FN+TN)$$

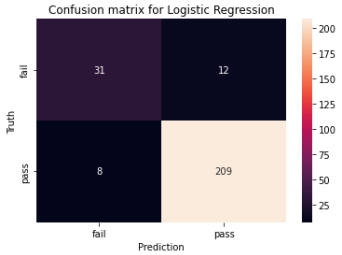
F1 score

F1 Score is the weighted average of Precision and Recall. Therefore, this score takes both false positives and false negatives into account. Intuitively it is not as easy to understand as accuracy, but F1 is usually more useful than accuracy, especially if you have an uneven class distribution. Accuracy works best if false positives and false negatives have similar cost. If the cost of false positives and false negatives are very different, it's better to look at both Precision and Recall. In our case, F1 score is 0.701.

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

Result of each Model

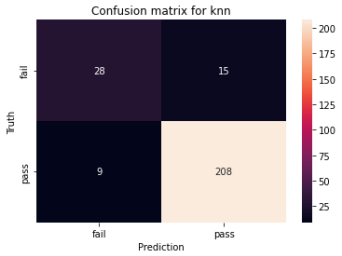
Logistic regression



Precision = TP/(TP + FP) = `0.946`
Recall = TP/(TP + FN)= `0.963`

```
Accuracy in % = 92.3076923076923
F1 score = 85.52177302594944
```

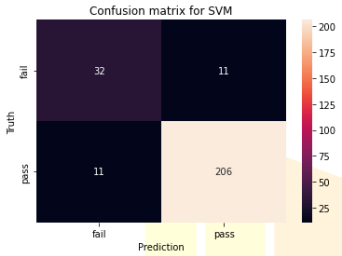
K-Nearest Neighbors



k = 3
Precision = TP/(TP + FP) = 0.933
Recall = TP/(TP + FN)= 0.959

Accuracy in % = 90.76923076923077
F1 score = 82.27272727272728

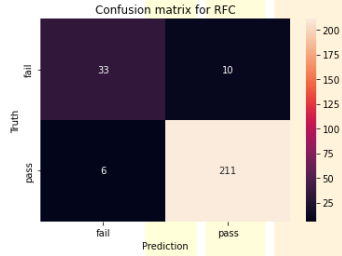
Support vector machine



Precision = TP/(TP + FP) = 0.949
Recall = TP/(TP + FN)= 0.949

Accuracy in % = 91.53846153846153
F1 score = 84.67474011359982

Random Forest Classifier



Max depth = 3
Precision = TP/(TP + FP) = 0.955
Recall = TP/(TP + FN)= 0.972

Accuracy in % = 93.84615384615384
F1 score = 88.41741842075955

Result of each Model After Implementation of BorutaPy Function

Result of BorutaPy

Due to the cost and time of collecting data and importing new data into the existing model to receive predictions by the model, the existence of many features may be problematic, and the utilization of an operational model or framework will be time-consuming and difficult. Therefore, the required features of the model are lower; firstly, the cost of collecting and training the model is reduced. Secondly, student performance prediction is done with less information which increases the utilization and comfort of the model. This may or may not affect the accuracy and performance of the model. To evaluate and calculate the effect of any of the features and their importance for the model, the Boruta algorithm was employed. The “Boruta algorithm is a wrapper built around the random forest classification algorithm”, and works similarly to the random forest in that it is an assembly approach based on several classifier votes. In this algorithm, a copy of all features adds to the randomness dataset and then a random forest model is fitted to the dataset and examines the importance of each feature. In each epoch, the features that have more importance are determined and features with low importance are eliminated.

Attributes = ['school', 'sex', 'age', 'address', 'famsize', 'Pstatus', 'Medu', 'Fedu', 'Mjob', 'F job', 'reason', 'guardian', 'traveltime', 'studytime', 'failures', 'schoolsup', 'famsup', 'paid ', 'activities', 'nursery', 'higher', 'internet', 'romantic', 'famrel', 'freetime', 'goout', 'Dalc ', 'Walc', 'health', 'absences', 'G1', 'G2']

The Boruta algorithm was used to obtain the most important feature of the dataset t o evaluate the performance of students. This algorithm was implemented on the pre processed original dataset using ten random forest estimators. The results of the Bor uta algorithm:

```
Ranks: [12 29  4 19 20 25 16  8 14 20  5 17 11 17  1 28 22 30 23
24  3 27 26 10  6  9 15 12  6  2  1  1]

Feature Selection: [False False False False False False False Fal
se False False False False False False  True False False False Fa
lse False False False False False False False False False False F
alse  True  True]
```

The importance for each feature, which is called ranking, is specified for each feature. The number of features that the model decides to be maintained can be specified by adjusting the algorithm threshol d. As the table indicates, the number of high-quality features which have a number 1 ranking are selec ted for keeping.

Selected Feature

Attribute Information:

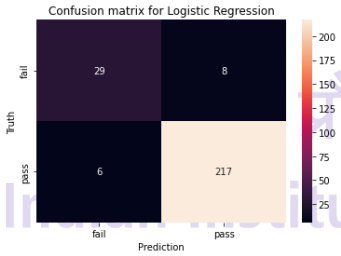
1 failures - number of past class failures (numeric: n if 1<=n<3, else 4)

2 G1 - first period grade (numeric: from 0 to 20)

3 G2 - second period grade (numeric: from 0 to 20)

4 G3 - final grade (numeric: from 0 to 20, output target)

Logistic regression



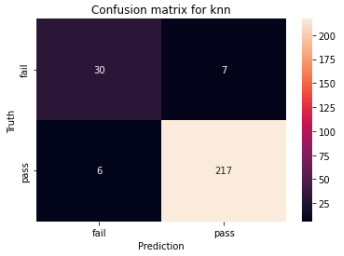
Precision = TP/(TP + FP) = 0.964

Recall = TP/(TP + FN)= 0.973

Accuracy in % = 94.61538461538461

F1 score = 88.71527777777777

K-Nearest Neighbours

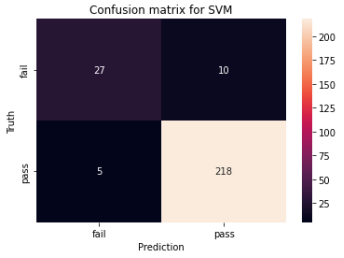


k=10

Precision = TP/(TP + FP) = 0.969
Recall = TP/(TP + FN)= 0.973

Accuracy in % = 95.0
F1 score = 89.64175170849806

Support vector machine

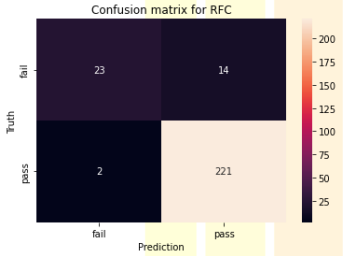


Precision = TP/(TP + FP) = 0.956
Recall = TP/(TP + FN)= 0.978

Accuracy in % = 94.23076923076923

F1 score = 87.4674636074424

Random Forest Classifier



Max depth = 3
Precision = TP/(TP + FP) = 0.940
Recall = TP/(TP + FN)= 0.991

Accuracy in % = 93.84615384615384

F1 score = 85.35004930271869

Comparison Of Results

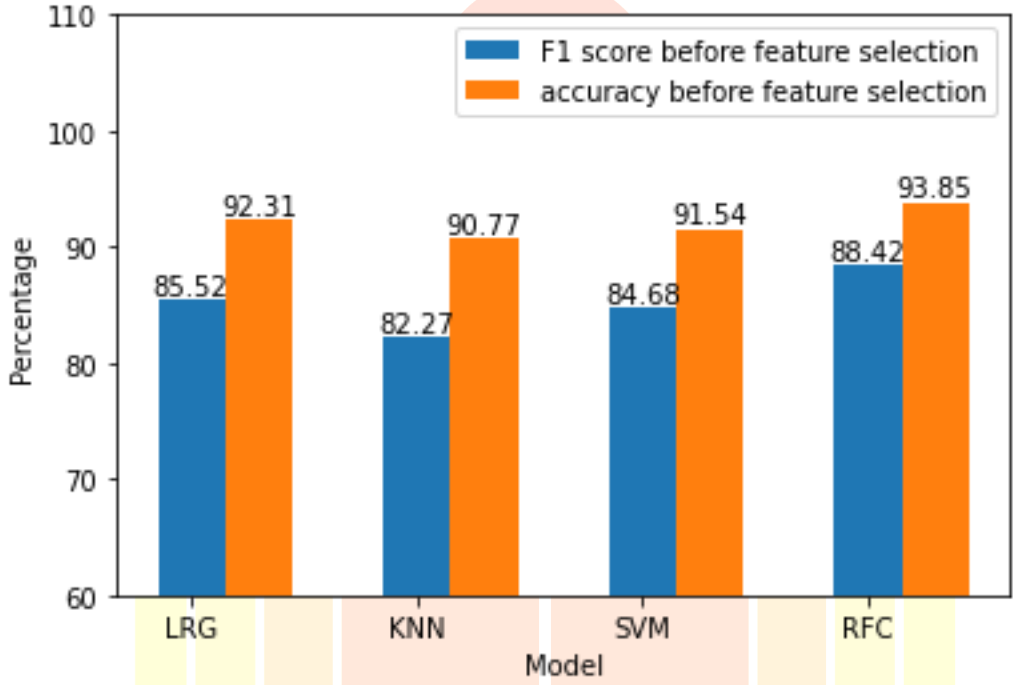
Accuracy and F1 score of all Models Before Feature Selection

metric	Logistic regression	KNN	SVM	RFC
f1 score	85.522	82.273	84.675	88.417
accuracy %	92.308	90.769	91.538	93.846
confusion matrix	[31 12] [8 209]	[28 15] [9 208]	[32 11] [11 206]	[33 10] [6 211]

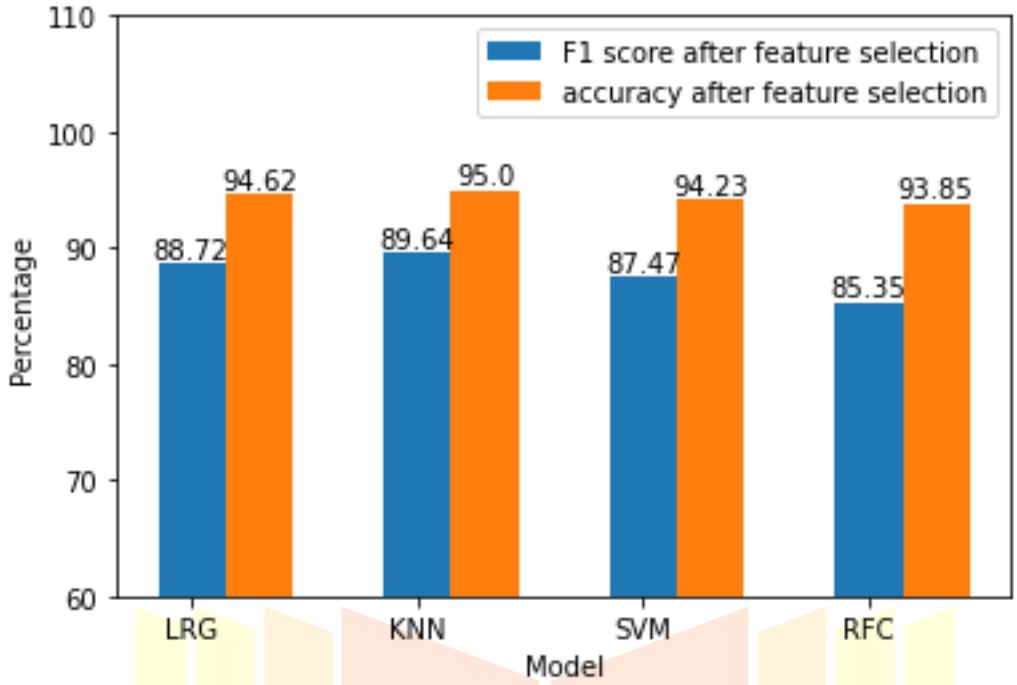
Accuracy and F1 score of all Models After Implementation of BorutaPy Function

metric	Logistic regression	KNN	SVM	RFC
f1 score	88.715	89.642	87.467	85.35
accuracy %	94.615	95.0	94.231	93.846
confusion matrix	[29 8] [6 217]	[30 7] [6 217]	[27 10] [5 218]	[23 14] [2 221]

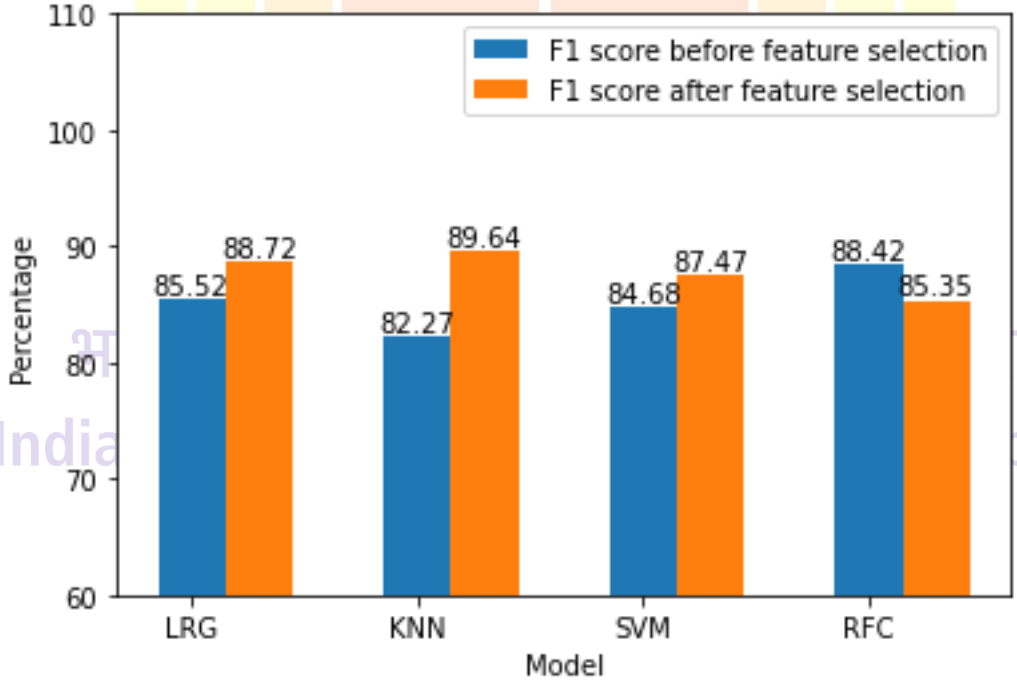
Graph for Accuracy and F1 score before feature selection



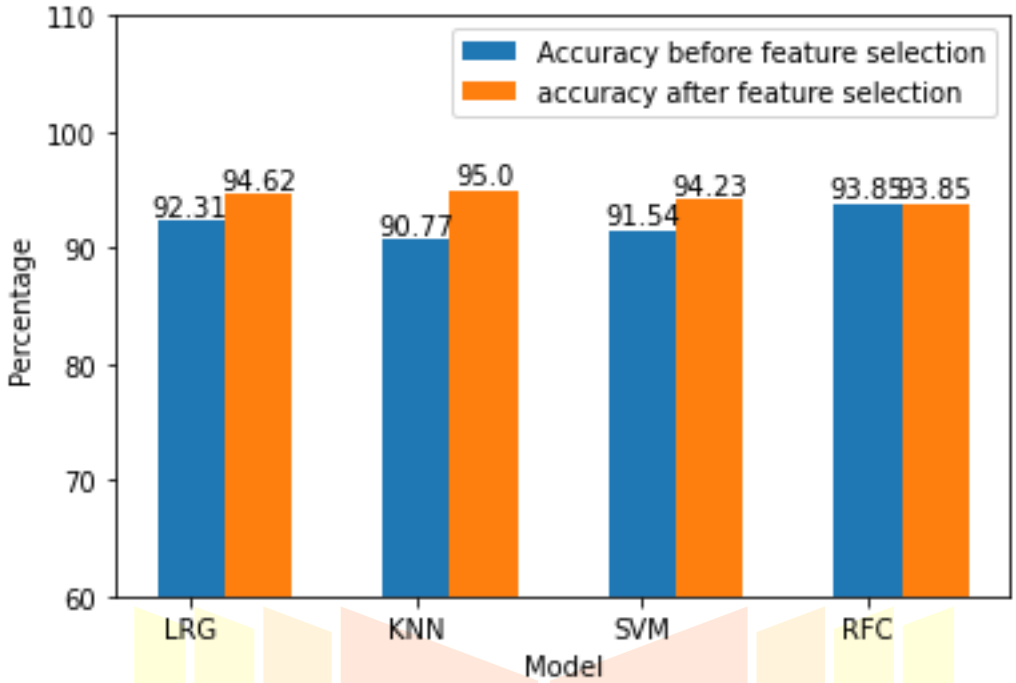
Graph for Accuracy and F1 score After feature selection



Comparison of F1 score with and without feature selection



Comparison of Accuracy with and without feature selection



After Implementation of bururaPy Algorithm, there is is an increase in accuracy and f1 score in logistic regression (LRG), K-Nearest Neighbour (KNN), support vector machines (SVM), and there is no change in Accuracy and decrease in f1 score of random forest (RFC).

Among all these ML models, we have good accuracy in the RFC model. We got good accuracy in the KNN model after the implementation of feature selection.

Conclusion

In this project, an ML-based system was designed and developed to evaluate high school students' performance across the state. This approach evaluates pupils on a variety of factors and divides them into two groups: those who passed and those who failed. Because there were no appropriate educational databases, the best information was gathered personally via online questionnaires and instructor participation.

The most influential features were found and studied, as well as the most efficient and effective machine learning models. Furthermore, the models were trained again using a smaller dataset including only the most significant attributes. As a result, more efficient models were created, and pupils were evaluated without the use of humans. By adding relevant labels to the data obtained, it can be used for additional research purposes such as dropout prediction and academic performances.

Students' responses in class can be tracked using an algorithm like deep learning, and their responses to class stimuli can be recognised by watching their eye angles. If students are not paying attention in class, the system will alert teachers or students, and pupils will be able to retain their focus in class.

भारतीय प्रौद्योगिकी संस्थान हैदराबाद

Indian Institute of Technology Hyderabad