Analyzing Student Performance in Programming Education Using Classification Techniques

Team Members:

Sneha Teja Sree Reddy Thondapu	- AP19110010124
Nandini Thimmireddy Gari	- AP19110010128
Likhitha Tadikonda	- AP19110010006

I. Abstract

Programming Skills are very important for any computer engineering student to get good marks in exams, apply the concepts to solve any real-world problem, crack any job interview, etc. But the only way they can know about their performances is to analyze and improve their skills regularly by seeing the statistics of their results in that semester regularly. With the help of this project, one can analyze their scores regularly, introspect, and can deliberately practice for better scores. This reduces the students' stress, anxiety, and depression about getting good scores in their academics. This analysis helps even professors to improvise the learning outcomes of students and increase their performance in whatever field they are working in.

Engineering has a vast division of marks in each subject like the internals, externals, and lab components. Every student has his/her comfort zone in each of these three components. But when they step out of their comfort zone there comes the problem. In this research, we aggregated the department of computer science students' data from the Web Technology subject (CSE202) which we took in the 4th Semester from our university SRM AP, Amaravati. We implemented classification algorithms like KNN, Decision tree, Logistic regression, Naive Bayes, and Adaboost algorithms to analyze the data of the students. We compared all the ML algorithms based on 200 classification instances. This analysis helps us understand how many students are performing well and how many could not perform well in all these internal, external, and lab exams and also helps students find their weak areas that have to be focussed on to improve their performance before they take up next tests.

II. Introduction

Data analysis is very important for students, teachers, educationalists, etc to understand, evaluate, introspect and discover useful information. With Data analysis we can solve many problems- Once said by Einstein that if you could find the problem or understand the question, you have the 90% of the solution, and the same applies to data analysis also. When we can sort the data out according to the problem in an understandable way, then we can fix any problem or get any answer related to the data objects. Since computer science subjects give more weightage to the students' problem-solving skills, the evaluation and division of the subject into sub-components is also done by the faculty such as lab performance, internal, etc.

This data analysis classifies all the huge data and gathers them into groups to make the evaluation simpler using Machine Learning Algorithms, Graphs, Classifiers, and Outliers, etc. Since covid-19 has come, there is a necessity to make smart learning or online learning more efficient. Over the last decade, many programmers over the country have used these algorithms to analyze the data.

III. Related Work

Data mining in higher education is a recent research field and this area of research is gaining popularity because of its potential for educational institutes. Data Mining can be used in the educational field to enhance our understanding of the learning process to focus on identifying, extracting, and evaluating variables related to the learning process of students as described by Alaa el-Halees.

Pandey and Pal conducted a study on student performance by selecting 600 students from different colleges of Dr. R. M. L. Awadh University, Faizabad, India. By means of Bayes Classification on category, language, and background qualification, it was found whether newcomer students will perform or not.

Hijazi and Naqvi conducted a study on student performance by selecting a sample of 300 students (225 males, 75 females) from a group of colleges affiliated with the Punjab university of Pakistan. The hypothesis that was stated "Student's attitude towards attendance in class, hours spent in a study on daily basis after college, students' family income, students' mother's age, and mother's education are significantly related with student performance" was framed. By means of simple linear regression analysis, it was found that the factors like mother’s education and the student’s family income were highly correlated with the student's academic performance.

Khan [7] conducted a performance study on 400 students comprising 200 boys and 200 girls selected from the senior secondary school of Aligarh Muslim University, Aligarh, India with the main objective to establish the prognostic value of different measures of cognition, personality, and demographic variables for success at higher secondary level in science stream. The selection was based on a cluster sampling technique in which the entire population of interest was divided into groups or clusters, and a random sample of these clusters was selected for further analyses. It was found that girls with high socioeconomic status had relatively higher academic achievement in the science stream and boys with low socioeconomic status had relatively higher academic achievement in general.

Al-Radaideh, et al [9] applied a decision tree model to predict the final grade of students who studied the C++ course at Yarmouk University, Jordan in the year 2005. Three different classification methods namely ID3, C4.5, and the NaïveBayes were used. The outcome of their results indicated that the Decision Tree model had better prediction than other models.

IV. Proposed Model

Data Preparation:

We extracted data from the students from the fourth-semester computer science programming course which was held for three months. The study for this semester was conducted online due to Covid-19. We have taken the course titled "Web Technology" which explains the aspects of web technology front and backend. This course includes markup languages such as HTML, CSS, and text programming languages like Javascript. The course outline contains, for example, an overview of fundamental web concepts, client-side programming, and server-side programming, using colors, fonts, images, etc. Four hours of weekly classes were held for this subject. It is compulsory for computer science students but it can be taken as a minor for some students in other departments. A lab class is conducted for this subject which is considered a separate subject and was given marks. Two hours of weekly classes were held in which we had practiced different topics that were taught to us. The marks for the theory subject are given and a total grade for the subject is given at the end of the semester. The grading is divided as follows: Assignments, polls, mid marks, and sem marks. The grading for the lab is decided on the lab assignment and viva. For a student to pass a particular course, such a student should possess at least 40% of the total score.

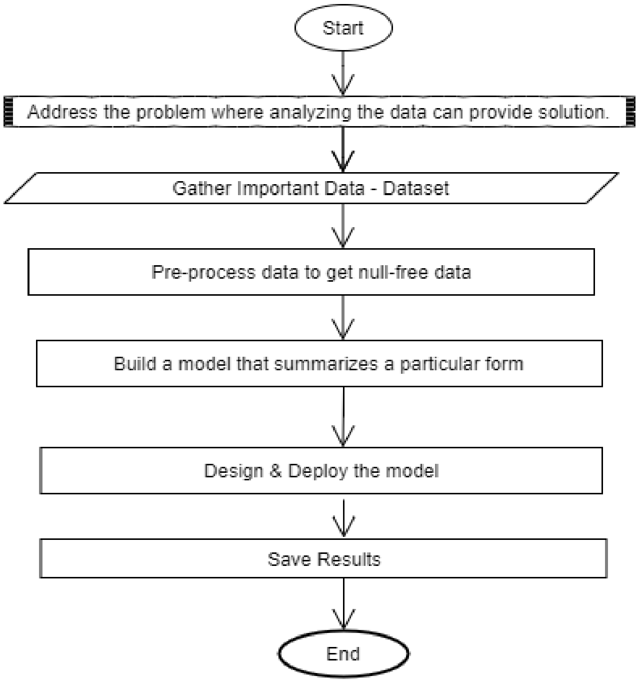
Start
Address the problem where analyzing the data can provide solution.
Gather Important Data - Dataset
Pre-process data to get null-free data
Build a model that summarizes a particular form
Design & Deploy the model
Save Results
End

Fig. 1 Data Mining Steps

Selection & Transformation of Data:

Here, we selected the required fields for data mining. Furthermore, in Table 1, we gave an overview of all the response variables as well as the predictor variable for reference purposes.

Table 1: Variables related to Dataset

VARIABLES	VALUES
Your Roll Number	{AP1911001xxx(000,006,124etc)}
Your Section	{CSE (A,B,D)}
CLA	{27,37,45,...50}
Mid	{20,37,45,...50}
Lab Internal	{36,40,49,45,....,50}
Theory Total	{35,50,48,37,.....,50}
Attendance	{80 to 85%,90 to 95%,96 to 100%}
Performance	{Poor, Good, Very Good, Excellent, Average}

The domain values are defined below:

- Your Roll Number - Students are given their respective roll number during admission.
- Your Section - This indicates the class section of the student.
- CLA - This stands for Cumulative Learning Assessment.
- Mid - It is the exams conducted for a small portion of the syllabus.
- Lab Internal - This is the internal marks given by the professor according to lab assignments and viva.
- Theory Total - Total marks given for the subject according to the sem marks, mid marks, and assignments.
- Attendance- This regards the student participation in attending the classes. A student should have a minimum of 80% attendance to attend end sem examinations.
- Performance - According to the grade given at the end of the sem, the categories Poor, Good, Very Good, Excellent, and Average are given.

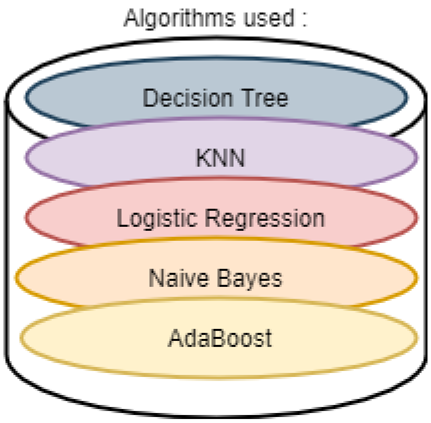

Fig. 2 Algorithms used in our Project

Decision Tree algorithm:

Because of its powerful features, this algorithm is widely used for classification and prediction in both Machine Learning and Data Mining applications. One of the advantages of choosing this algorithm is that the Decision Tree represents rules that are readily understood and can be interpreted due to its simplicity and comprehensibility to uncover large or small data structures and predict them.

A Decision Tree Classifier is usually a flowchart classifier consisting just like a Tree Structure where

- A test on an attribute is denoted by a non-leaf node.
- An outcome of the test is represented by the tree branch.
- A value of the target attribute indicates a terminal node.
- The topmost node in a tree is the root node.

We decided to choose the decision tree algorithm because of the following strong features:

- High dimensional data can be easily handled with the decision tree
- Small-sized trees can easily be interpreted
- The steps to be followed to properly classify decision tree induction are fast.

KNN (K-Nearest Neighbors):

The KNN algorithm believes that objects that are similar are close together. To put it another way, related items are close together.

KNN Algorithm:

1. Load the data
2. Initialize K to your chosen number of neighbors
3. For each example in the data

3.1 Calculate the distance between the query example and the current example from the data.

3.2 Add the distance and the index of the example to an ordered collection

4. Sort the ordered collection of distances and indices from smallest to largest (in ascending order) by the distances
5. Pick the first K entries from the sorted collection
6. Get the labels of the selected K entries
7. If regression, return the mean of the K labels
8. If classification, return the mode of the K labels

Choosing the appropriate K value:
We run the KNN algorithm numerous times with different values of K to find the K that decreases the number of errors we encounter while retaining the algorithm's capacity to generate correct predictions when it's given data it hasn't seen before.

Logistic Regression:

Logistic regression is a statistical analysis approach for predicting a data value based on previous data set observations. In the machine learning field, logistic regression has become an efficient process. The method enables a machine learning application to classify incoming data using an algorithm based on historical data. The algorithm should get better at guessing classes within data sets as more relevant data comes in. Logistic regression can also help with data preparation by allowing data sets to be placed into predefined categories.

Logistic regression has become particularly popular in online advertising, enabling marketers to predict the likelihood of specific website users who will click on particular advertisements as a yes or no percentage. Logistic regression can also be used in healthcare to identify risk factors for diseases and plan preventive measures, weather forecasting apps, and voting apps.

Naive Bayes:

The Naive Bayes method is a supervised learning technique for addressing classification issues that is based on the Bayes theorem. It is mostly utilized in text classification tasks that need a large training dataset.

The Naive Bayes Classifier is a simple and effective classification method that aids in the development of rapid machine learning models capable of making quick predictions.
It's a probabilistic classifier, which means it makes predictions based on an object's likelihood. Spam filtration, sentiment analysis, and article classification are all common uses of the Naive Bayes Algorithm.

Bayes' theorem is also known as Bayes' Rule or Bayes' law, which is used to determine the probability of a hypothesis with prior knowledge. It depends on conditional probability. The formula for Bayes' theorem is given as

$$P(A|B)=\frac{P(B|A)P(A)}{P(B)}$$

Where,

P(A|B) is Posterior probability: Probability of hypothesis A on the observed event B.

P(B|A) is Likelihood probability: Probability of the evidence given that the probability of a hypothesis is true.

AdaBoost Classifier:

Adaptive Boosting, short for Adaptive Boosting, is a Boosting approach used in Machine Learning as an Ensemble Method. The weights are re-allocated to each instance, with higher weights applied to improperly identified instances. This is termed Adaptive Boosting. In supervised learning, boost is used to reduce bias and variation. It is based on the notion of successive learning. Each subsequent student, with the exception of the first, is grown from previously grown learners.

As the first decision tree/model is made, the incorrectly classified record in the first model is given priority. Only these records are sent as input for the second model. The process goes on until we specify the number of base learners we want to create. Remember, repetition of records is allowed with all boosting techniques.

V. Results

We present the data set containing the results of n = 200 students in the programming course-Web Technology (CSE202) which were obtained from the CSE Students of Batch 2023, SRM University, Amaravati. The dataset used in this study covers the 2020-2021 academic session (4th Semester).

This is the link to the form that we used to take data from students:
https://docs.google.com/forms/d/e/1FAIpQLSfwCILq0mUqE1t2uEJjmKOfVOG1oYgMQKk-KmfXn_KSuj1GzA/viewform
=> Table 2 presented below shows the sample data and Table 2 shows the frequency of the occurrence of each grade.

Table 2: First 20 records of Dataset

Your Roll Number	**Your Section**	**CLA**	**Mid**	**Lab Internal**	**Theory total**	**Attendance**	**Performance**
19110010001	CSE A	7	12	42	38	90 to 95%	Good
19110010002	CSE A	6	10	35	32	96 to 100%	Average
19110010003	CSE A	6	13	38	38	90 to 95%	Good
19110010004	CSE A	7	11	36	36	96 to 100%	Good
19110010005	CSE D	6	9	49	30	96 to 100%	Poor
19110010006	CSE A	7	10	44	34	96 to 100%	Average
19110010007	CSE A	6	8	45	28	96 to 100%	Poor
19110010008	CSE D	7	8	50	30	90 to 95%	Poor
19110010009	CSE D	8	12	46	40	90 to 95%	Good
19110010010	CSE D	8	13	39	42	90 to 95%	Very Good
19110010011	CSE D	6	10	40	32	90 to 95%	Average
19110010012	CSE D	8	15	41	46	90 to 95%	Excellent
19110010013	CSE A	7	14	50	42	96 to 100%	Very Good
19110010014	CSE D	9	10	37	38	90 to 95%	Good
19110010015	CSE A	10	10	37	40	96 to 100%	Good
19110010016	CSE A	10	15	48	50	96 to 100%	Excellent
19110010017	CSE D	6	14	36	40	96 to 100%	Good
19110010018	CSE A	9	11	47	40	90 to 95%	Good
19110010019	CSE A	6	12	44	36	96 to 100%	Good

Table 3: Frequency of Grade Occurrence

Performance	**Frequency**	**Percentage (%)**
Excellent	29	14.5
Very Good	47	23.5
Good	76	38
Average	33	16.5
Poor	15	7.5
Total	**200**	**100**

From table 3, we can infer that 7.5 % performed poorly while 92.5 % passed the course.

Table 4: Comparison of Accuracy Scores- Classification & Regression

ML Algo.	**Accuracy Score**
Decision Tree Classifier	100
KNN - K Nearest Neighbours	90 (Neighbours=7)
Logistic Regression	70
Naive Bayes	100
AdaBoost Classifier	82.5 ,SVC = 100

Calculating Classification Report

1. Decision Tree Classifier:

	Precision	**recall f1-score support**
Average	1.00	1.00 1.00 7
Excellent	1.00	1.00 1.00 3

Good	1.00	1.00 1.00 18
Poor	1.00	1.00 1.00 4
Very Good	1.00	1.00 1.00 8

2. KNN - K Nearest Neighbours:

	Precision	**recall f1-score support**
Average	0.88	1.00 0.93 7
Excellent	1.00	0.67 0.80 3
Good	1.00	0.89 0.94 18
Poor	1.00	0.75 0.86 4
Very Good	0.73	1.00 0.84 8

3. Logistic Regression:

	Precision	**recall f1-score support**
Average	0.80	0.57 0.67 7
Excellent	0.50	0.67 0.57 3
Good	0.78	0.78 0.78 18
Poor	1.00	0.75 0.86 4
Very Good	0.50	0.62 0.56 8

4. Naive Bayes:

	Precision	**recall f1-score support**
Average	1.00	1.00 1.00 7
Excellent	1.00	1.00 1.00 3

Good	1.00	1.00 1.00 18
Poor	1.00	1.00 1.00 4
Very Good	1.00	1.00 1.00 8

5. AdaBoost Classifier:

	Precision	**recall f1-score support**
Average	1.00	1.00 1.00 7
Excellent	1.00	1.00 1.00 3
Good	1.00	1.00 1.00 18
Poor	1.00	1.00 1.00 4
Very Good	1.00	1.00 1.00 8

Total Number of Instances considered for each ML Algorithm are 200. After seeing the Comparison Table 4, we can see that the Decision Tree Classifier and AdaBoost gave the best results. Thereby giving 100 as an accurate prediction.
We found the performance of the models by using Evaluation matrices:

- accuracy scores
- confusion matrix
- precision, recall, f1-score, and support

But the final comparison was done based on the Accuracy Score.

We took the training test in the same area (X, y, test_size=0.2, random_state=1), to predict which is the best-suited algorithm for our case.
The Decision Tree Method makes use of Gain Ratio & Information Gain to split the attributes.
Then we created a Visualization tree / Decision Tree.

AdaBoost Classification gave 82.5% accuracy in the first go, but when we Boosted by creating the 2nd Model it attempted to correct from the first model. Models are added until the training set is predicted perfectly or the max number of models is added. So when we used SVC to make the 2nd model, It gave us 100% accuracy there itself.

VI. Discussion

The studied outcome on student performance will be discussed in this part. This study is based on the most accurate categorization techniques as well as the most essential elements that may impact students' performance. Table 3 shows the classification accuracy of all 5 algorithms used in this study. From the table, it is evident that the Decision tree classifier, Adaboost, and Naive Bayes have the highest accuracy as compared to other algorithms. Looking at Table 2, a higher percentage of students failed the course titled "Web Technology" by 7.5%.

VII. Conclusion

The project demonstrates the value of data analysis in assessing undergraduate student performance, notably in the field of higher education. This information is collected from 4th-semester students of SRM University on a subject titled "Web Technology". In order to uncover buried knowledge, several data mining approaches were used. We utilized the decision tree approach, KNN, Logistic Regression, Naive Bayes, and Adaboost classifier to evaluate the data set and predict the student success or failure rate. Both professors and students will benefit from this research.

VIII.REFERENCES

1. **Alaa el-Halees**: https://arxiv.org/ftp/arxiv/papers/1201/1201.3417.pdf
2. **Pandey & Pal:**https://www.researchgate.net/publication/51989865_Mining_Educational_Data_to_Analyze_Students'_Performance
3. **Hijazi and Naqvi:**https://www.ijert.org/research/mining-parent-socio-economic-factors-to-predict-students-academic-performance-in-osun-state-college-of-technology-esa-oke-IJERTV2IS120345.pdf
4. **Khan:**https://www.ijert.org/mining-parent-socio-economic-factors-to-predict-students-academic-performance-in-osun-state-college-of-technology-esa-oke-2
5. **Al-Radaideh:**https://repensarlafisica.files.wordpress.com/2015/10/mining-student-data-using-decision-trees.pdf

Thank-you