Ministry of Science and Higher Education of the Republic of Kazakhstan

Suleyman Demirel University

Bakdaulet Tolbassy

Analysis of effective machine learning techniques for improvement of student performance

THESIS

Presented in Partial Fulfilment for the

Master of Technical Sciences Degree in Computer Science

(degree code: 7M06102)

Department of Computer Science

Faculty of Engineering and Natural Sciences

Supervisor: **Associate Professor, Dr. Phys.-Math. Sc. Lyazzat Atymtayeva**

Kaskelen 2023

Suleyman Demirel University
Faculty of Engineering and Natural Sciences
Department of Computer Science

Dean of Faculty

Associate Professor

PhD Azamat Zhamanov

«06» 06 2023

Topic of the thesis:
Analysis of effective machine learning techniques for improvement of student performance

Thesis submitted as part of the requirements for the award of the MSc in "7M06102 - Computer Science" SDU, 2021-2023

Head of Department Assistant Professor, PhD Mukash Zh.

Academic Supervisor Associate Professor, Dr. Phys.-Math. Sc. Atymtayeva L.

Master student Tolbassy B.

Kaskelen 2023

Declaration

I confirm that this is my own work and the use of all material from other sources has been properly and fully acknowledged.

Bakdaulet Tolbassy

2023

Acknowledgements

I would like to thank Dr. Lyazzat Atymtayeva, my supervisor, whose direction, knowledge, and continuous support have greatly influenced my academic work. Her extensive understanding in the sciences of computer, physics and mathematics has been extremely helpful in guiding my studies and igniting my interest in machine learning.

Dedication

I dedicate this dissertation to my coworker Duman Telman for his important contributions, teamwork, and contagious excitement throughout this project. Your commitment and wisdom have greatly raised the caliber of this effort.

I want to convey my sincere gratitude to the Suleyman Demirel University faculty and staff for giving me the space and resources I needed to do this research in a stimulating intellectual atmosphere. Your dedication to academic achievement has had a significant impact on how I approach my research.

I will always be grateful to my family and friends for their continuous support, love, and confidence in me. Your unwavering help and compassion have been my rock, keeping me going in the face of difficulties.

The last thing I want to say is that I dedicate this dissertation to all the students whose academic experiences have inspired me. May this research help to improve powerful machine learning approaches that will empower both instructors and students, enhance learning results, and promote a more promising future for future generations.

Abstract

It is advantageous to use machine learning (ML) algorithms to assess student performance based on their prior performances and current behaviour because they can project both positive and negative outcomes at different educational levels. Learning outcomes can be improved by early performance prediction for students. The prediction of a student's academic performance is important because it shapes changes in university academic policies, guides instructional strategies, evaluates the efficacy and efficiency of learning, provides teachers and students with pertinent feedback, and modifies learning environments. All of these elements support higher graduation rates. There is currently no clear winner among the various machine learning techniques for predicting student performance while enhancing learning outcomes. Therefore, this study is going to present the most effective machine learning techniques using and analysing open-source data from the Kaggle platform.

Аңдатпа

Студенттердің бұрынғы үлгерімдері мен оқу уақытындағы болмыстарына негізделген оқу іс-әрекетін бағалау үшін машиналық оқыту алгоритмдерін пайдалану әртүрлі білім беру деңгейлеріндегі оң және теріс жетістіктерді болжау арқылы оқушылардың үлгерімін арттырудың тиімді әдісі болып табылады. Оқушылардың өнімділігін ерте болжау оқу нәтижелерін жақсартуға көмектеседі. Студенттің оқу үлгерімінің болжамы өте маңызды, себебі ол университеттің академиялық ережелеріндегі өзгерістерге әсер етеді, оқу тәжірибесін хабардар етеді, оқудың тиімділігі мен тиімділігін бағалайды, мұғалімдер мен студенттерге тиісті кері байланыс береді және оқу параметрлерін реттейді. Осы факторлардың барлығы бітіру көрсеткіштерінің жоғарылауына ықпал етеді. Оқыту нәтижелерін жақсарту кезінде оқушылардың өнімділігін жобалауға арналған машиналық оқытудың оңтайлы алгоритмі әртүрлі машиналық оқыту әдістерінің арасында әлі айқын емес. Сондықтан, бұл зерттеу Kaggle платформасынан ашық бастапқы деректерді пайдалана отырып, оны талдайтын машиналық оқытудың ең тиімді әдістерін ұсынбақ.

Аннотация

Использование алгоритмов машинного обучения для оценки успеваемости учащихся на основе их прошлых достижений и текущего поведения является полезным способом повышения успеваемости учащихся путем прогнозирования положительных и отрицательных успехов на различных уровнях образования. Раннее прогнозирование успеваемости учащихся помогает улучшить результаты обучения. Прогноз успеваемости учащегося имеет решающее значение, поскольку он влияет на изменения в академических правилах университетов, информирует о методах обучения, оценивает эффективность и действенность обучения, дает преподавателям и учащимся соответствующую обратную связь и корректирует условия обучения. Все эти факторы способствуют более высокому уровню выпускников. Оптимальный алгоритм машинного обучения для прогнозирования успеваемости учащихся при одновременном улучшении результатов обучения среди различных методов машинного обучения еще не очевиден. Поэтому в этом исследовании будут представлены наиболее эффективные методы машинного обучения с использованием и анализом данных из открытых источников с платформы Kaggle.

Abbreviations

1. **ML** - Machine Learning
2. **SPR** - Statistical Pattern Recognition
3. **NB** - Naive Bayes
4. **SVM** - Support Vector Machines
5. **MLP-ANN** - Multilayer Perceptron Artificial Neural Network
6. **CFS** - Correlation-Based Feature Selection
7. **IG** - Information Gain
8. **GR** - Gain Ratio
9. **SU** - Symmetrical Uncertainty
10. **CHI** - Chi-Squared Attribute Evaluation
11. **SVM** - Support Vector Machines
12. **KNN** - K-Nearest Neighbours
13. **SGD** - Stochastic Gradient Descent
14. **Linear SVC** - Linear Support Vector Classifier

Table of Contents

Chapter 1

Introduction

This is a master's thesis that has been influenced by the work of Bakdaulet Tolbassy. To improve education through more accurate prediction of student performance on assessments, this study will critically examine machine learning methods. Suleyman Demirel University's esteemed academic Lyazzat Atymtayeva has assiduously supervised the research endeavour.

1.1 Background and motivations

After the first year, some students leave their higher education programs. Research claims that more than a million students leave college each year. According to this study [1], motivation and a sense of progress are the two main causes of this phenomenon. There are a lot of other things that could have happened, though. In addition, in order to raise the proportion of students who successfully complete university programs, the study investigates a potential strategy for offering a helpful critique and, in a way, a humorous approach to the learning process.

Through machine learning (ML), this behaviour can be researched. This is undoubtedly a big issue with numerous problems and difficulties that need to be overcome. The main subject of this study is classification as a whole. Classification in machine learning is defined [2] as the process of figuring out which category an observation belongs to. It basically involves a process that establishes a correlation between an independent variable and a dependent variable (which

could be categorical or numerical).

A wide variety of decision-making issues are now included in the classification category. According to the study [3], this kind of work is categorized using two empirical learning methodologies. Statistical pattern recognition (SPR) is the first method, while machine learning (ML), which generates decision trees and inference rules, is the second. This study examines several machine learning approaches, with an emphasis on the latter group.

1.2 Problem statement

The issue at hand takes into account the well-known pattern of college students dropping out of bachelor's degree programs at institutions at a certain rate. While there may be a number of causes for this problem, this research especially focuses on two of them: motivation and a sense of progress. For student performance and retention rates to increase, it is essential to recognize and address these concerns.

Evaluating the effectiveness of various machine learning (ML) techniques within ML frameworks is crucial to achieving this goal. This research attempts to determine the model that performs best in terms of forecasting student outcomes based on pertinent parameters by comparing and assessing several ML models. As a result, this study will examine and evaluate the potential of two or more ML techniques while taking into account their time- and labour-intensive nature.

This study will make use of student information from university education credit data to carry out a thorough examination. The project will make use of this dataset to conduct a comparative analysis of various ML models, offering insights into how well they predict and enhance student performance.

Given that it affects students and educational institutions all around the world, this issue is significant beyond specific schools. This problem has sociological and economic repercussions, such as the possibility of social embarrassment, fewer job options, and lower pay for students who do not achieve academically.

The final objective of this thesis is to investigate whether it is possible to identify students who might need extra assistance to improve their academic performance

at the university level using student data and machine learning techniques. This project aims to contribute to the creation of efficient interventions and methods that can improve student performance and, ultimately, lower the attrition rate in bachelor's degree programs by utilizing the power of ML.

1.3 Purpose/Aims/Rationale

This study's main goal is to use an analysis of credit data from higher education to pinpoint students who need study help, leading to better academic results. In order to accomplish this, the project intends to create a framework that is ML-modelled that university instructors and students may use to improve student performance and raise program completion rates.

This research aims to identify commonalities among students who are more likely to drop out or experience difficulties finishing their studies, in addition to offering support for students. The likelihood of at-risk pupils succeeding academically can be improved by recognizing these trends and implementing interventions and customized tactics to address the particular problems they encounter.

This study's scientific investigation attempts to contrast the effectiveness of two or more machine learning models and algorithms based on important assessment criteria like accuracy, precision, recall, f1 score, and prediction. These measures offer a thorough evaluation of how well machine learning models predict student outcomes and spot trends in student accomplishment. A further discussion of these criteria can be found in the methodology part of this study.

The creation of an ML model that can recognize trends in student achievement using historical data from university education credit records is the study's intended goal. In order to estimate academic outcomes, such as dropout and completion rates, based on students' present performance, this model will be an invaluable tool for university students and staff.

Technically speaking, this study evaluates the effectiveness of different machine learning algorithms in relation to the aforementioned elements. This study advances machine learning methods in the realm of education by assessing and con-

trasting the performance of different algorithms. In the end, the research's findings will help to increase the precision and dependability of prediction models, giving educational institutions crucial information for putting supportive programs for struggling students into place.

1.4 Research Questions

How well can machine learning models perform in terms of accuracy, recall, precision, F1 score, and prediction when using freely available student datasets? This is the main research question that forms the basis of this work [4]. The following linked research questions have been developed in light of this core research question:

1. Are machine learning algorithms adequate for assessing and forecasting student outcomes? Can they successfully discover patterns in student data?

2. What are the best methods for putting machine learning models into practice? Which particular model performs best in terms of accuracy, precision, recall, and F1 score when used with student data from higher education credit records?

3. How much information must be present in the dataset in order to produce accurate classifications and predictions? What effect does the dataset size have on how well the machine learning models perform? These new research inquiries seek to expand on the applicability and efficiency of machine learning techniques for studying student data. The purpose of this study is to answer these questions in order to gain knowledge about the suitability of ML techniques for spotting trends in student performance, identify the best ML model for this task, and comprehend how the size of the dataset affects the classification and prediction abilities of the models. Additionally, in addition to the ones already mentioned, the following other study questions can be looked into:

4. How do various feature selection methods affect how well machine learning models succeed at forecasting student outcomes?

5. Can actionable interventions or support strategies be developed using the recognized patterns and predictions from machine learning models to enhance student performance and retention rates?

6. Are there any ethical issues or potential biases that should be taken into account when utilizing machine learning to analyse student data, and how might these issues be resolved?

This study intends to increase knowledge of the capabilities and constraints of machine learning approaches in the context of analysing student performance. By doing so, it will aid in the creation of efficient educational interventions and strategies.

Chapter 2

Literature Review

2.1 Scientific research

How to better prepare students for the shifting needs of society and the job market is one of the issues that many colleges across the world face. The term "applicability" describes how well students can use their knowledge and abilities to solve issues and situations in the actual world. In order to solve this issue, a great deal of scientific study has been done to pinpoint the variables that affect students' learning outcomes and job possibilities as well as to create efficient treatments and pedagogies.

The study "EMT: A metadata ensemble-based tree model for predicting student achievement" is an illustration of this kind of research. It suggests a novel approach for forecasting student performance based on different metadata elements, such as gender, age, academic background, learning style, etc. The accuracy and robustness of the prediction are increased by the authors through the use of both standalone approaches and ensemble algorithms, which integrate several classifiers. To choose the best classifier for each metadata characteristic, they compare various approaches, such as decision trees, naive Bayes, support vector machines, etc. They also use random voting. They found that the ensemble approaches beat the single methods after running numerous trials on a real-world dataset, particularly NBTree and Adaboost_J48, which have an accuracy of roughly 98.5% in

forecasting student performance [5].

In order to improve students' learning outcomes and career preparedness, this study shows how data mining and machine learning techniques can be used to provide timely, individualized feedback and advice. Additionally, it demonstrates how metadata elements can be used as gauges of student performance and potential, assisting educators and legislators in creating curricula and regulations that are more successful.

Higher education institutions face a significant challenge when it comes to the topic of student development and retention because it has an impact on the institutions' reputation, finances, and overall quality. As a result, a significant number of academics have endeavoured to forecast the academic performance of students and identify those who are at risk of failing their classes or dropping out entirely. Because they can analyse huge and complicated datasets and uncover important patterns and insights, data mining and machine learning techniques have seen widespread application for this purpose in recent years. The implementation of these strategies, on the other hand, calls for careful consideration of a variety of elements, including the definition of student performance, the selection of important features, the choice of appropriate methodologies, and the evaluation of the outcomes.

This issue is addressed in the context of a Saudi Arabian university in the paper "Prediction of Student's performance by modeling small dataset size, International Journal of Educational Technology in Higher Education" by Lubna Mahmoud and Abu Zohair. In this setting, the number of students attending the institution is on the lower end, and the data available is limited and unreliable. Using a variety of data mining and machine learning algorithms, such as linear discriminant analysis (LDA), k-nearest neighbours (KNN), support vector machines (SVM), naive Bayes (NB), and multilayer perceptron artificial neural network (MLP-ANN), the authors hope to forecast the final grades of fifty students who are currently enrolled in a computer science class. In addition to this, they provide a comprehensive explanation of the data pre-processing stages, which include the management of categorical features, the imputation of missing values, and the standardization of the data. This article's primary contribution

is that it demonstrates how data mining techniques may be efficiently applied to tiny datasets by utilizing suitable feature selection and filtering procedures. This is the article's primary contribution since it is the most important part of the article. The authors evaluate and contrast several distinct filtering algorithms, including correlation-based feature selection (CFS), consistency-based feature selection (CONSISTENCY), information gain (IG), gain ratio (GR), relief attribute evaluation (RELIEF), symmetrical uncertainty (SU), and chi-squared attribute evaluation (CHI). When utilizing CFS or IG as a filtering mechanism, they discover that KNN and LDA obtain the maximum accuracy in forecasting student performance (74–79%) [6].

This study is significant to my research since it demonstrates how approaches from data mining may be used to predict student performance in a demanding environment in which the data is restricted and noisy. In addition to this, it offers a detailed summary of the data mining process, including an explanation of the several methods and factors that are utilized. In addition to that, it sheds light on the significance of selecting relevant features and using filters as a means of enhancing the accuracy and effectiveness of prediction models. Nevertheless, this essay does have a few shortcomings, which can be remedied by additional investigation in the future. For the evaluation of their models, for instance, the authors only use a single set of data from one particular course, which may restrict the generalizability of their conclusions. In addition, they do not take into account other aspects that might influence student performance, such as the level of student engagement, feedback, or motivation. In addition, they do not describe how their models might be utilized to support children who are in danger of having low academic performance or dropping out of school and to intervene on their behalf.

Namoun and Alshanqiti's (2021) article titled "Predicting Student Performance Using Data Mining and Learning Analytics Techniques: A Systematic Literature Review" provides a comprehensive overview of the intelligent techniques used for predicting student performance in an academic setting in which academic success is strictly measured using student learning outcomes. The authors review a decade's worth of research work that was carried out between 2010 and November

2020 and then synthesize and analyze a total of 62 relevant papers with a focus on three perspectives:

- the forms in which the learning outcomes are predicted;
- the predictive analytics models developed to forecast student learning; and
- the predominant factors that have an impact on student outcomes.

The authors do a synthesis of the findings and summarize the most important findings by making use of the best techniques for carrying out systematic literature reviews, such as PICO and PRISMA. According to the findings of the paper, the achievement of learning outcomes was primarily evaluated based on performance class standings (also known as ranks) and achievement scores (also known as grades). In the process of classifying student performance, regression and supervised machine learning models were utilized quite frequently. In conclusion, the most obvious determinants of student learning outcomes were the students' academic moods, their grades on term assessments, and their participation in online learning activities. The report also identifies many major research challenges and presents a summary of critical recommendations to promote future studies on this subject. Both of these aspects are included to encourage future research on this subject. In the essay, consideration is given to both of these facets [7].

This article provides a comprehensive and up-to-date assessment of the current state of the art in predicting the academic performance of students through the application of data mining and learning analytics techniques which are can be helful for my research work. In addition to this, it draws attention to the limitations and gaps that are present in the existing literature and offers valuable insights and proposals for the direction of future study. In addition to this, it points out the deficiencies and restrictions that are present in the existing literature. In addition to this, it analyzes the advantages and disadvantages of a wide variety of techniques for predicting the performance of students, such as ranking, regression, binary classification, and multiclass classification, amongst others. Binary classification, multiclass classification, regression, and ranking are some of the methods included in this category. In addition to this, it investigates the operation of a number of different types of predictive analytics models, including

linear regression, logistic regression, decision trees, random forests, support vector machines, and neural networks, amongst others, to investigate how effective these models are in predicting outcomes.

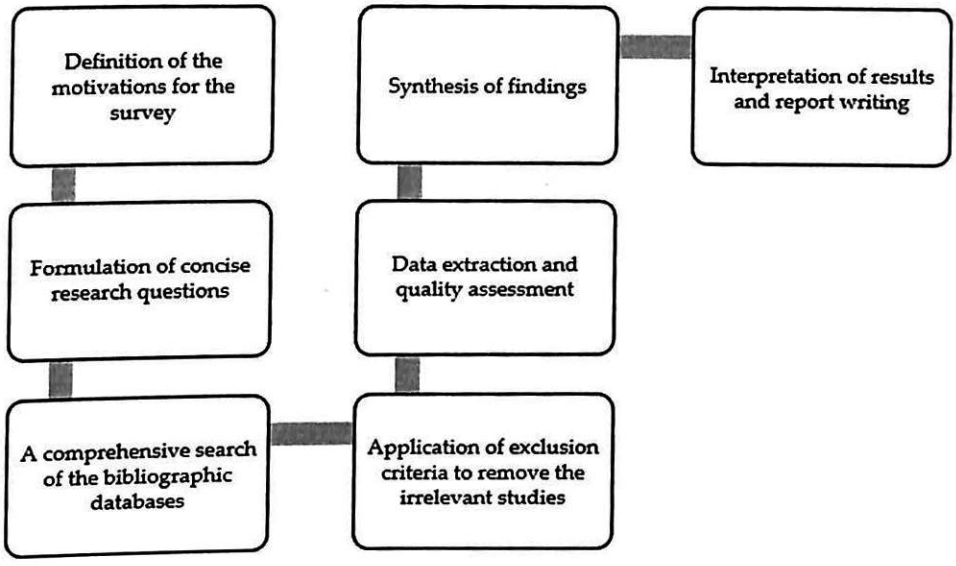

Figure 2.1: Major steps of the methodology [7]

According to the findings of these studies, data mining and machine learning approaches are effective methods for forecasting the academic success of students in a variety of educational environments. In addition to this, they offer a comprehensive assessment of the most recent advancements in this subject as well as a discussion of the opportunities and obstacles that lie ahead for future study.

2.2 Information systems for monitoring progress

A significant number of educational institutions employ research data to develop a variety of educational systems that are intended to increase students' academic performance and graphically show students their current level of education.

For instance, the English University of Nottingham Trent developed a system that monitors not only student success but also student engagement in the process of learning. This system was explained in great detail in the article that was written by Foster et al. (2019). This system was established in order to improve students' academic performance, lower the percentage of kids who drop

out of school, and examine the level of participation pupils have in extracurricular activities. This study's principal objective was "to provide students with a tool that would motivate them", as stated in the introduction to the study [8]. 72% of first-year students said that the utilization of this service led to an improvement in their overall academic performance, which was measured three years after the initial implementation of the system.

It is interesting to hear about the experiences of a university located in another country. Course Signals is a predictive analytics system that was developed by Purdue University in the United States. This system utilizes student academic performance, digital learning activity, and demographic data to calculate the risk of dropout for each student. This information is then sent to the instructor as well as the student himself. The university has been able to increase learning results and minimize the number of students dropping out of school thanks to the interactive system [9].

It is reasonable to infer, given the experiences of other countries' educational institutions, that the introduction of such a system might improve the overall academic performance of students at Russian universities while simultaneously lowering the number of students who are kicked out of those institutions.

Nevertheless, information systems for tracking progress are not exclusive to colleges located in other countries. In addition, several examples of such systems may be found in Russian educational institutions. For example, [10] describe a neural network model for forecasting student success at Perm State Research University based on pre-university resources. This model uses neural networks. The authors construct and train neural network models with the help of the STATISTICA Automated Neural Networks program. These models have the ability to predict the average score that students receive in four different core classes. In order to enhance the total number of observations and fine-tune the accuracy of the models, the authors make use of a technique that generates synthetic data by combining real data with new data. The authors conduct an analysis of student data obtained from a variety of sources, including online courses, electronic journals, and social networking sites, using data mining techniques. In addition to this, the writers employ methods of machine learning in order to divide the

students into distinct groups according to both their academic success and their overall demeanor. These groups are defined by the authors' analysis of the students' overall performance. Students and teachers can improve their learning and teaching processes by using the system's visualizations and recommendations, which are provided by the system.

According to the findings of these studies, information systems that track progress can also be useful tools for improving student performance in educational institutions located in foreign countries. In addition to this, they demonstrate how techniques such as data mining and machine learning may be used to analyze student data in order to provide feedback and guidance to the students.

Article	**University**	**Data sources**	**Data analysis methods**	**Data visualization methods**	**Main results**
Foster et al. (2019)	Nottingham Trent University	Student engagement dashboard	Descriptive statistics	Charts and graphs	72% of first-year students reported improved academic performance
Arnold and Pistilli (2012)	Purdue University	Course Signals system	Predictive analytics	Traffic light indicators	21% increase in retention rate

Table 2.1: Table of research articles on student performance

Chapter 3

Methods and Methodology

3.1 Machine learning Models

Machine learning (ML), a subfield of artificial intelligence, makes predictions or judgments based on data by using computer algorithms to learn from it. supervised learning, unsupervised learning, and reinforcement learning are the three primary divisions of ML.

Learning from labeled data where the desired outcome of each input is known in advance is referred to as supervised learning. It is comparable to having a teacher oversee the educational process. While unsupervised learning concentrates on learning from unlabeled data, when the desired output is either unknown or irrelevant for each input, it focuses on learning from unlabeled data. It resembles examining data structures and patterns haphazardly rather than with a predetermined end in mind. Finally, trial-and-error learning, where the desired outcome is established by a reward or penalty system, is known as reinforcement learning. Similar to a learning process, it involves doing something and getting feedback to make improvements.

In our study, we especially focus on supervised learning approaches to forecast student achievement utilizing a variety of indicators, such as student characteristics, behaviors, activities, and evaluations. Our goal is to comprehend the connection between these elements and academic success. Educators can iden-

tify students who may be at risk of failing or dropping out with the use of the ability to anticipate student performance. It enables effective interventions and teaching approaches, prompt and tailored feedback and guidance, better learning outcomes, and higher retention rates.

Support vector machines (SVM), k-nearest neighbors (KNN), logistic regression, random forest, naive Bayes, perceptron, stochastic gradient descent (SGD), and linear support vector classifier (Linear SVC) are just a few of the supervised learning techniques we use to predict student performance. These methods are our choice since they are popular and meet the criteria for our prediction task. They deliver accurate results and are perfect for the task at hand.

3.1.1 Support Vector Machines (SVM)

The support vector machine (SVM) looks for a hyperplane that divides the data into two classes with the greatest possible margin. Margin is defined as the distance between the hyperplane and the data points that belong to each class that are closest to it. These data points are what are known as support vectors, and the reason for this name is that they either support or determine the position and orientation of the hyperplane. SVM is capable of dealing with data that is both high-dimensional and sparse, and it achieves excellent levels of accuracy and generalization. The original data are transformed into a higher-dimensional space, where it is possible for a linear separation to occur, so that the SVM can model complex and nonlinear interactions. This is accomplished through the use of kernel functions. The radial basis function (RBF) is a particular kernel that we use in our method to improve the performance of our models. The following is a description of the RBF kernel:

$$K(x_i, x_j) = e^{-\gamma \|x_i - x_j\|^2} \tag{3.1.1}$$

where x_i and x_j are the coordinates of x_i and x_j, respectively; *gamma* controls how broad the kernel is; and x_i and x_j are two data points. By applying methods like cross-validation and grid search, we adjust the parameter *gamma* as well as another parameter C, which controls the trade-off between increasing the margin

and decreasing the error. Using Python's Scikit-learn module, we create Support Vector Machines (SVM), and we assess their performance using a range of metrics like accuracy, precision, recall, and F1-score. We compare SVM's effectiveness to a number of different classification techniques, such as KNN, logistic regression, random forest, naive Bayes, perceptron, SGD, and linear SVC [11].

The statistical learning theory, often known as the VC theory, was first proposed by Vapnik and Chervonenkis in 1974 [12]. SVM are derived from this theory. This theory offers a framework for examining the trade-off between a learning model's level of complexity and the degree to which it can be generalized. The variable capacity (VC) dimension is the central idea underlying this theory. This dimension assesses the ability or adaptability of a model to accommodate various configurations of input. A model with a high VC dimension has the ability to match complicated patterns of data; however, it also runs the risk of overfitting the data and performing badly when applied to new data. A model that has a low VC dimension can prevent overfitting, but it also runs the risk of not fitting the data well enough and missing some significant patterns in the process. The best model is the one that reduces the VC dimension as much as possible while simultaneously increasing the margin between the different classes.

SVMs are widely recognized as one of the most effective and widely used machine learning algorithms currently available. Image recognition, text classification, emotion analysis, face identification, handwriting recognition, bioinformatics, and anomaly detection are just some of the many applications that have found widespread use for them. They have also been expanded to handle multi-class classification and regression problems by making use of various methodologies, such as one-vs-one, one-vs-all, or error-correcting output codes. This was accomplished by employing the one-vs-one, one-vs-all, and error-correcting output code approaches. SVM does, however, have some challenges and limitations, such as choosing an appropriate kernel function and tuning its parameters, scaling and preprocessing the data, computing and storing large kernel matrices, providing probability estimates rather than binary predictions, and handling imbalanced data. These challenges and limitations can be overcome, however, through the use of SVM.

3.1.2 K-nearest neighbors (KNN)

K-nearest neighbors (KNN) is a supervised learning method that we utilize for forecasting student performance as a categorical variable with numerous classes, such as A, B, C, D, or F. KNN is an abbreviation for "k-nearest neighbors," which stands for "k-nearest neighbors" and "k-nearest neighbors algorithm." An instance is assigned a classification by KNN based on the consensus of the votes cast by its k nearest neighbors in the training data. Different metrics, such as the Euclidean distance or the Manhattan distance, can be utilized to determine the distance that exists between two instances. Powerful algorithms like K-nearest neighbors (KNN) can handle high-dimensional and sparse data while still maintaining excellent accuracy and generalization skills. It does this by making predictions while taking into account nearby data points. KNN's adaptability to model complicated and nonlinear connections is one of its strengths. KNN may successfully identify complex patterns in the data by modifying the value of k or using other weighting approaches for the neighbors, such as inverse distance or similarity metrics. In order to do this, the ideas of "similarity" and "inverse distance" are crucial. They enable the KNN algorithm to find hidden associations that may be present in the dataset by measuring the proximity or relatedness between data points.

KNN are built on the premise that points with similar properties can be discovered nearby one another. The cluster hypothesis and homogeneity hypothesis are other names for this supposition. Fix and Hodges in 1951 [13] published a paper in which they offered a method for distinguishing between two normal distributions that was based on the closest neighbor rule. This paper is where the concept of using neighbors as a means of categorization can be traced back to. Later on, Cover and Hart demonstrated in 1967 [14] that the error rate of the nearest neighbor classifier is bounded by twice the Bayes error rate, which is the optimal error rate that can be reached by any classifier. When they compared the nearest neighbor classifier's error rate to the Bayes error rate, they were able to demonstrate that this was the case. In addition to this, they demonstrated that the error rate of the nearest neighbor classifier tends to converge to the Bayes error rate as the number of examples grows.

K-nearest neighbors (KNN) is widely recognized as one of the most straightfor-

ward and user-friendly machine learning techniques available today. It is renowned for its uncomplicated and organic approach to problem-solving. They have found widespread application in a variety of domains, including image recognition, text classification, recommendation systems, pattern recognition, data mining, financial market prediction, intrusion detection, and a great many other areas. They have also been improved to manage regression issues by employing the average of the values of k nearest neighbors as the output. This improvement was made possible through further development. KNN does, however, face certain limitations and challenges, like selecting a suitable value for the distance measure and scaling the data, handling missing values and noisy data, handling imbalanced data, storing and searching all of the training data, and more. One of the difficulties KNN encounters is selecting an appropriate value for the distance metric and k.

3.1.3 Logistic Regression

Logistic regression calculates the likelihood of an occurrence, such as passing a test, based on a specified group of independent factors, such as the amount of time spent studying, the number of absences, the number of unsuccessful attempts, and so on. The odds of the event are subjected to a logit transformation during the logistic regression procedure, which may be thought of as the ratio of the probability of success to the probability of failure. The logit function is described as follows:

$$\text{logit}(p) = \ln\left(\frac{1-p}{p}\right) \tag{3.1.2}$$

where the probability of success is p. As a linear combination of the independent variables, the logit function may also be expressed as follows:

$$\text{logit}(p) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_n x_n \tag{3.1.3}$$

where $x_1, x_2, \ldots, x_n$ are the independent variables, β_0 is the intercept term, and $\beta_1, \beta_2, \ldots, \beta_n$ are the coefficients of the independent variables. Maximum likelihood estimation, often known as MLE, can be used to estimate the coeffi-

cients. Using the observed data, this approach finds the values that maximize the likelihood function. The probability function is described as follows:

$$L(\beta) = \prod_{i=1}^{n} p_i^{y_i}(1-p_i)^{1-y_i} \tag{3.1.4}$$

where n is the overall number of instances, y_i is the result that was seen (either 0 or 1), and p_i is the anticipated probability for occurrence i. The anticipated probability is obtained by using the logistic function, which is the inverse of the logit function:

$$p = \frac{1}{1+e^{-\text{logit}(p)}} \tag{3.1.5}$$

Following the estimation of the coefficients, the following formula can be used to determine the prediction for a fresh instance x:

$$\hat{y} = \text{round}\left(\frac{1}{1+e^{-(\beta_0+\beta_1 x_1+\beta_2 x_2+\ldots+\beta_n x_n)}}\right) \tag{3.1.6}$$

where $\hat{y}$ is the outcome that was forecasted. By using alternative methods, such as one-vs-one, one-vs-all, or multinomial logistic regression, logistic regression may also handle several classes. One-to-one logistic regression, for instance, compares each class to its own.

Logistic regression is based on the statistical theory of generalized linear models, or GLM. In order to incorporate non-normal distributions and non-linear relationships, GLMs are an extension of linear models [15]. A binomial distribution and a logit link function are features of the Generalized Linear Model (GLM) class, which includes logistic regression. The logit link function ensures that the predicted probability is always confined between 0 and 1 and bears the appearance of a sigmoid distribution. Following Joseph Berkson's initial idea in 1944, David Cox further expanded the concept of logistic regression in 1958. One of the machine learning algorithms that has been used the most and that has drawn the greatest interest from researchers is logistic regression. It has a wide range of

advantages and can be used in various contexts. It is simple to learn and simple to put into practice. It can handle variables of both a numerical and categorical form. It can offer probability estimates rather than just yes or no predictions. It may do feature selection by utilizing regularization techniques like Ridge or Lasso. By using kernel methods or polynomial characteristics, it can also be expanded to handle nonlinear relationships. It could be prolonged as a result of this. Two of the challenges and restrictions of logistic regression are selecting a suitable set of independent variables and preventing multicollinearity. Scaling and prepping the data, dealing with missing values and noisy data, and handling imbalanced data are additional difficulties and restrictions [16].

3.1.4 Random Forest

Random forest is a type of learning that takes place in a tree structure. The Random Forest algorithm is an example of ensemble learning, which aggregates the findings of several different decision trees into a single conclusion. Creating an uncorrelated forest of decision trees is the goal of the random forest technique, which employs bagging in addition to feature randomness. Bagging is a method that involves generating random samples of data from the training set in order to train a decision tree on each individual sample. A technique known as feature randomness chooses a different group of characteristics at each split in the decision tree. Feature randomness is used to improve classification accuracy. The random forest model may accommodate numerical as well as categorical variables. In addition to that, it is able to produce estimations of out-of-bag errors and measures of the relevance of variables.

The concept behind random forest is that a collection of weak learners can be transformed into a strong learner through the use of voting or the averaging of their predictions. Ho first proposed this idea in 1995. He used the random subspace technique to create a collection of decision trees with a set variance. Later, Breiman expanded on this idea in 2001 by combining the bagging strategy with the random subspace method to create random forests. Breiman also applied to register "Random Forests" as a trademark in 2006. Numerous applications, including as classification, regression, clustering, feature selection, anomaly de-

tection, and many more, have shown the effectiveness and dependability of the Random Forest technique [17].

The utilization of random forests has numerous potential advantages and uses in a variety of industries. It is simple to learn and simple to put into practice. It can handle high-dimensional and sparse data, and it achieves a high level of generalization and accuracy. By averaging the results of numerous decision trees, it can help reduce variation and overfitting. By utilizing bootstrap samples, it is also able to deal with missing values as well as noisy data. It is capable of carrying out feature selection by making use of feature importance scores. It is also possible to adapt it such that it can deal with nonlinear interactions by employing a variety of splitting criteria or kernel approaches. Choosing an adequate number of trees and characteristics, scaling and preprocessing the data, computing and storing a large number of trees, providing probability estimates instead of class labels, and managing imbalanced data are some of the challenges and limits of random forest [18].

3.1.5 Naive Bayes Classifier

We utilize a supervised learning method called naive bayes to predict student performance as a categorical variable with numerous classes. This method is called "naive" since it is not very sophisticated. Naive Bayes is a probabilistic classifier that applies Bayes' theorem with the naïve assumption of conditional independence between every pair of features given the value of the class variable. Naive Bayes is sometimes referred to as "naive Bayes" or "naive Bayes classification." The naive bayes model calculates the likelihood of an occurrence, like receiving an A grade, based on a predetermined group of independent variables, such the amount of time spent studying, the number of absences, and so on. The posterior probability of each class is calculated using the naive Bayes method, which involves first multiplying the prior probability of the class by the likelihood of the features given the class, and then normalizing the result by the evidence. A mathematical expression for the posterior probability looks like this:

$$P(C_k|x) = \frac{P(x)P(C_k|x)}{P(x)} \tag{3.1.7}$$

For each class C_k, the prior probability is denoted by $P(C_k)$, the likelihood of features is denoted by $P(x|C_k)$, and the evidence or marginal probability of features is denoted by $P(x)$. The class variable is denoted by C_k, while the feature vector is denoted by x.

Typically, the category with the highest posterior probability is chosen as the predicted class. The naive bayes algorithm can process both numerical and categorical information. And instead of providing hard-and-fast labels for each category, it can instead provide probabilistic estimations. Naive Bayes is based on inverting conditional probabilities using Bayes' theorem to make predictions in light of available evidence. It was Thomas Bayes, in 1763, who first proposed this idea by proving a special case of Bayes' theorem for binary occurrences. Pierre-Simon Laplace, building on Bayes's work, applied it to a wide range of astronomical and physical problems in 1812. The term "naive bayes" was coined by John Duchi in 2003. He did this to refer to a group of classifiers that erroneously assume feature independence under certain conditions. This assumption simplifies the computation of the likelihood term by allowing us to multiply the probability of each characteristic given the class.

$$P(x|C_k) = \prod_{i=1}^{n} P(x_i|C_k) \tag{3.1.8}$$

where x_i is the feature corresponding to the i-th position in the list and n is the total number of features. The Naive Bayes method has been shown to be effective and trustworthy for a range of tasks, including text classification, spam filtering, sentiment analysis, document clustering, face recognition, and many more tasks.

The naive Bayes algorithm has many benefits and can be applied in a wide variety of settings. It is straightforward and straightforward to execute and understand. It is able to deal with data that is both high-dimensional and sparse, and it achieves a high level of accuracy and generalization. Utilizing the feature

relevance scores, it is able to carry out the feature selection process. It is also possible to expand it such that it can deal with nonlinear interactions by employing a variety of distributions or kernel approaches. Choosing an adequate prior probability and likelihood function, scaling and preparing the data, dealing with missing values and noisy data, and handling imbalanced data are some of the challenges and limits of the naive Bayes method [19].

3.1.6 Perceptron

The perceptron is the most basic type of artificial neural network, which consists of only one layer of artificial neurons. Every neuron receives a vector of input values, computes a weighted sum of those values, and then adds a bias term to the result. After that, the weighted sum is pushed through an activation function like a step function, a sigmoid function, or a ReLU function so that an output value can be generated. The result that is given as the output can either be regarded as the probability of belonging to a particular class or as the label for the class itself. Variables of both a category and numerical nature can be handled by perceptrons. They can also discover the ideal weights and bias by utilizing a learning algorithm such as gradient descent or stochastic gradient descent. This allows them to learn the optimal values for the weights and bias.

Modeling the biological neurons in the brain and the connections between them is the fundamental premise behind the concept of perceptrons. The term "perceptron" was first used in 1957 by Frank Rosenblatt, who also developed a rudimentary model of an artificial neuron with adjustable synaptic weights and a threshold. He also devised a learning method for perceptrons that was based on the delta rule and updated the weights based on the difference between the desired output and the actual output. This technique was one of his many contributions to the field of artificial intelligence. Rosenblatt demonstrated that perceptrons are capable of learning any function that can be partitioned linearly. On the other hand, in 1969, Marvin Minsky and Seymour Papert demonstrated that perceptrons have limitations and are unable to learn certain straightforward operations, such as XOR. Because of this, interest in neural networks began to wane until the 1980s, when multilayer perceptrons and the backpropagation method were first

developed [20].

In the realm of machine learning, perceptrons are among the algorithms that are considered to be among the most fundamental and fundamental. They offer a wide range of benefits and can be applied in a variety of contexts. They are straightforward and easy to comprehend as well as implement. They are capable of working with data that can be separated linearly and reach great levels of accuracy and generalization. They also have the capability of carrying out feature selection by making use of feature relevance scores. They can also be made capable of dealing with nonlinear data through the utilization of several layers or kernel approaches. Choosing an appropriate activation function and learning rate, scaling and preparing the data, dealing with missing values and noisy data, and handling imbalanced data are some of the challenges and constraints associated with perceptrons [21].

3.1.7 Stochastic Gradient Decent (SGD)

The term "Stochastic Gradient Descent" (abbreviated as "SGD") refers to a supervised learning method that we implement for the purpose of forecasting student performance as a binary variable, such as "pass" or "fail." SGD is an iterative method that updates the weight vector and the bias term of a linear classifier by making use of a single or a small batch of randomly selected training instances at each step of the process. SGD is able to achieve high levels of accuracy and generalization, despite the fact that it can manage data on a huge scale. Regularization strategies, such as L1 or L2 regularization, can be utilized in SGD's feature selection process for additional functionality. SGD is also capable of dealing with nonlinear data thanks to kernel approximation approaches like Nystroem and RBFSampler, among others. The SGD algorithm is based on the concept of minimizing a convex loss function over training data by utilizing gradient descent in order to do so. An optimization approach known as gradient descent works by moving in the direction of the steepest descent of the loss function until it finds a local minimum. This process continues in an iterative manner. On the other hand, gradient descent necessitates computing the gradient over the entirety of the training data at each step, which can be both computationally expensive and

slow when applied to problems of a large scale. SGD is able to produce an approximation of the gradient by employing a single or a small batch of training instances at each step. This decreases the amount of computing that is required and makes online and incremental learning possible. Bottou in 1998 was the first person to suggest using SGD, and he applied it to solving large-scale learning issues. In later research, Zhang in 2004 demonstrated that SGD can, under some circumstances, accomplish optimal convergence rates. SGD has seen widespread use across a variety of domains, including bioinformatics, image recognition, text classification, sentiment analysis, face identification [22] [23].

3.1.8 Linear Support Vector Classifier (Linear SVC)

A primal-dual algorithm is used in the linear SVC approach to identify the best weight vector and bias term. By doing this, the margin between classes is improved. The distance between the decision boundary and the support vectors—the data points within each class that are closest to the boundary—is used to determine the margin. Both numerical and categorical variables can be taken into consideration by the linear SVC model. Additionally, rather than probability estimates, it includes the ability to provide class designations.

In order to categorize data using convex loss functions like logistic loss and hinge loss, linear SVC uses linear models. Based on this idea, linear SVC was created. A linear model is a function that uses a linear decision boundary, which is defined by a weight vector and a bias term, to split the data into two classes. One way to think of linear models is as representations of the data. It is possible to learn both the bias term and the weight vector by minimizing the loss function over the training data. For each training instance, the loss function calculates a value that represents the degree of difference between the predicted class and the actual class. Convex loss functions are functions that have only a single minimum at the global level and no minima at the local level. This makes it much simpler to optimize convex loss functions.

An optimization algorithm known as the primal-dual algorithm can solve both the primary problem and the secondary problem at the same time. To minimize

the loss function in addition to a regularization term while taking into account the weight vector and bias term is the primary problem. A penalty for large weights is what the regularization term is called. This penalty helps prevent overfitting and promotes generalization. The dual problem consists of optimizing a function of Lagrange multipliers while taking into account certain limitations. The Lagrange multipliers are different coefficients that are used for each training instance to ensure that the margin restrictions are met. The weight vector, bias term, and Lagrange multipliers are all subject to iterative modifications by the primal-dual method until such time as they converge on the values that are optimal.

It was first proposed by Fan et al. in 2008. It was developed using liblinear rather than libsvm, making it more adaptable and scalable. Fan et al. made the initial argument for linear SVC. For large-scale linear classification, Liblinear is a library that supports a number of penalty and loss functions. Support for a wide range of kernels and parameters is provided by the support vector machine library Libsvm. Many different sectors and industries, including bioinformatics, picture identification, text classification, sentiment analysis, face recognition, and more have found extensive use using linear SVC [23].

There are several advantages to using linear SVC, and it may be used in many different situations. It is simple to learn and simple to put into practice. It can handle high-dimensional and sparse data, and it achieves a high level of generalization and accuracy. By using regularization techniques like L1 or L2 regularization, it may carry out feature selection. It may also work with nonlinear data using kernel techniques like polynomial or radial basis function kernels [24]. Scaling and preparing the data, dealing with missing values and noisy data, and handling imbalanced data are some of the obstacles and limits of linear SVC [21]. Another challenge is choosing a proper penalty, loss function, and regularization parameter.

Briefly, Support Vector Machines (SVM), k-Nearest Neighbors (KNN), Logistic Regression, Random Forest, Naive Bayes, Perceptron, Stochastic Gradient Descent (SGD), and Linear Support Vector Classifier (Linear SVC) are the models that we employ for predicting the performance of students. These models were selected because they are suited for binary classification problems, such as

passing or failing, and because they are able to deal with categorical as well as numerical variables. In addition, each of these models has a unique set of benefits and drawbacks, which can be contrasted with one another and evaluated in light of our findings.

The support vector machine (SVM) is a classification technique that utilizes a kernel function to map the data into a high-dimensional feature space and locate the optimal hyperplane that differentiates the classes with the highest possible margin. In addition to its superior accuracy and generalization, SVM can handle high-dimensional and sparse data. In addition, it can return labels for classes rather than probabilities. SVM, on the other hand, may be delicate to adjustments to the regularization parameter and kernel function. It may also be challenging to train and test, as well as expensive in terms of processing resources.

To classify a new instance, KNN takes into account the opinions of its nearest k neighbors in the training data. A voting system is used to achieve this goal. High-dimensional and sparse data are no problem for KNN, as it still manages to provide excellent precision and generalization. Moreover, it can provide probability estimates in place of class labels. However, KNN may be affected by missing values and noisy data, and it may be delicate to the distance metric and the number of neighbors. It's possible that KNN is sensitive to the number of neighbors as well.

In order to represent the likelihood of an instance belonging to a specific class, logistic regression makes use of logistic functions. You may sometimes hear this strategy referred to as "logistic analysis." The accuracy and generalization of logistic regression are excellent, and it may be used with both numerical and categorical data. Moreover, it can provide probability estimates in place of class labels. However, logistic regression may not fare as well if the data are not linearly separable, and its performance can be delicately dependent on the regularization parameter and the solver technique.

The ensemble learning technique known as Random Forest combines the forecasts from several separate decision trees. In spite of the high dimensionality and sparsity of the data, Random Forest is still capable of producing very accu-

rate and generalizable results. In addition, it may generate measurements of the importance of variables and out-of-bag error estimates. However, from a computational aspect, training and testing with Random Forest can be costly and time-consuming, and its output can be difficult to interpret.

Naive Bayes is a probabilistic classifier that uses Bayes' theorem on the naive assumption that all pairs of characteristics are unrelated to one another regardless of the value of the class variable. The term "naive Bayes" is sometimes used interchangeably with "naive Bayes classification." The Naive Bayes algorithm can handle high-dimensional data with ease and achieve high precision and generalization. Moreover, it can provide probability estimates in place of class labels. However, if the conditional independence criteria is not met, the Naive Bayes algorithm's performance may deteriorate because the prior probability and likelihood function may introduce bias.

A single layer of artificial neurons makes up the perceptron, the simplest form of artificial neural network. Perceptron models can achieve excellent accuracy and generalization when given information that can be split linearly. It can also use feature relevance scores to make feature selection decisions. However, perceptrons have a limited ability to process nonlinear input because of their sensitivity to the activation function and the learning rate.

SGD is an iterative method that uses a single or a limited set of randomly selected training instances to update the weight vector and bias term of a linear classifier. One such technique is known as stochastic gradient descent, or SGD. Despite SGD's ability to process massive amounts of data, it is still capable of producing highly accurate and generalized results. It can also perform feature selection with the help of regularization techniques. It's possible that missing values and noisy data, as well as the penalty, loss function, and regularization parameter, could have an impact on SGD. It could also be particularly delicate in this respect.

To find the optimal weight vector and bias term that maximizes the gap between the classes, the linear support vector classifier (SVC) employs a primal-dual algorithm. You may also hear it referred to as "linear SVC" or "linear support

vector classification." High-dimensional data is no problem for the linear SVC method, and it can also achieve excellent precision and flexibility. Additionally, it has the capability of providing class labels rather than probability estimations. However, Linear SVC may be influenced by missing values and noisy data, and it may be sensitive to the penalty, loss function, and regularization parameter. In addition, Linear SVC may be sensitive to these factors.

3.2 Dataset

This study made use of the dataset known as the Student Performance Dataset [25], which can be found on Kaggle. This dataset includes information regarding the academic accomplishments of students attending secondary school in two different Portuguese schools. The data was gathered through the use of school reports and questionnaires, and the elements of the data include student grades in addition to demographic, social, and school-related characteristics.

The dataset is representative of the following results in the Math subject: The file has a total of 649 records and 33 attributes. Table 1 has a description of the characteristics.

Attribute	Description
school	student's school (binary: 'GP' - Gabriel Pereira or 'MS' - Mousinho da Silveira)
sex	student's sex (binary: 'F' - female or 'M' - male)
age	student's age (numeric: from 15 to 22)
address	student's home address type (binary: 'U' - urban or 'R' - rural)
famsize	family size (binary: 'LE3' - less or equal to 3 or 'GT3' - greater than 3)
Pstatus	parent's cohabitation status (binary: 'T' - living together or 'A' - apart)
Medu	mother's education (numeric: 0 - none, 1 - primary education (4th grade), 2 – 5th to 9th grade, 3 – secondary education or 4 – higher education)

Fedu	father's education (numeric: 0 - none, 1 - primary education (4th grade), 2 – 5th to 9th grade, 3 – secondary education or 4 – higher education)
Mjob	mother's job (nominal: 'teacher', 'health' care related, civil 'services' (e.g. administrative or police), 'at_home' or 'other')
Fjob	father's job (nominal: 'teacher', 'health' care related, civil 'services' (e.g. administrative or police), 'at_home' or 'other')
reason	reason to choose this school (nominal: close to 'home', school 'reputation', 'course' preference or 'other')
guardian	student's guardian (nominal: 'mother', 'father' or 'other')
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)
studytime	weekly study time (numeric: 1 - <2 hours, 2 - 2 to 5 hours, 3 - 5 to 10 hours, or 4 - >10 hours)
failures	number of past class failures (numeric: n if 1<=n<3, else 4)
schoolsup	extra educational support (binary: yes or no)
famsup	family educational support (binary: yes or no)
paid	extra paid classes within the course subject (Math or Portuguese) (binary: yes or no)
activities	extra-curricular activities (binary: yes or no)
nursery	attended nursery school (binary: yes or no)
higher	wants to take higher education (binary: yes or no)
internet	Internet access at home (binary: yes or no)
romantic	with a romantic relationship (binary: yes or no)
famrel	quality of family relationships (numeric: from 1 - very bad to 5 - excellent)
freetime	free time after school (numeric: from 1 - very low to 5 - very high)
goout	going out with friends (numeric: from 1 - very low to 5 - very high)
Dalc	workday alcohol consumption (numeric: from 1 - very low to 5 - very high)

Walc	weekend alcohol consumption (numeric: from 1 - very low to 5 - very high)
health	current health status (numeric: from 1 - very bad to 5 - very good)
absences	number of school absences (numeric: from 0 to 93)
G1	first period grade (numeric: from 0 to 20)
G2	second period grade (numeric: from 0 to 20)
G3	final grade (numeric: from 0 to 20, output target)

Table 3.1: Student Performance Dataset Attributes and Descriptions

3.3 Evaluation Metrics

The performance of machine learning models for forecasting student performance is evaluated using four metrics: accuracy, precision, recall, and F1 score. These metrics are used to measure how well the models perform. These metrics are frequently applied in classification tasks, with the purpose of evaluating the accuracy, completeness, and overall efficacy of the model in terms of its ability to make predictions [26].

- Accuracy can be defined as the proportion of accurately anticipated instances relative to the total number of instances. It assesses the degree to which the model is capable of producing accurate predictions throughout the whole dataset. Calculating accuracy looks like this:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{3.3.1}$$

where TP represents the number of true positives, TN represents the number of true negatives, FP represents the number of false positives, and FN represents the number of false negatives.

- Precision can be defined as the ratio of accurately anticipated positive instances to the total number of positive instances that have been forecasted.

It determines how accurately the model predicts good outcomes and quantifies the frequency with which it does so. The following factors determine precision:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{3.3.2}$$

- The proportion of accurately anticipated positive cases in relation to the total number of actual positive instances is referred to as recall. It assesses the degree to which the model is capable of producing accurate forecasts for all favorable cases. The formula for recall is as follows:

$$\text{Recall} = \frac{TP}{TP + FN} \tag{3.3.3}$$

- The F1 score represents the mathematical middle ground between precision and recall. It assesses how well precision and recall are balanced against one another. The following factors make up the F1 score:

$$F1_{\text{Score}} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3.3.4}$$

These metrics can be used to compare various models and determine which one is the most suitable solution to the problem at hand. If one of these indicators has a greater value, it implies that the model is performing more effectively [26].

Chapter 4

Results and Discussion

Using the Student Performance Dataset from Kaggle [25], the purpose of this research was to investigate and assess how well different machine learning approaches were able to forecast student performance. The dataset was meticulously cleansed, and then extra columns were added so that we could find the primary characteristics that determine whether or not a student is successful. For the purpose of gaining insights into the dataset, data visualization techniques were utilized.

In order to make the process of machine learning easier, essential columns such as school, sex, and family size were initially transformed into their binary equivalents after the data had been preprocessed. After then, the dataset was divided into training and testing sets with an 80:20 proportion between the two. Support Vector Machines (SVM), K-Nearest Neighbors (KNN), Logistic Regression, Random Forest, Naive Bayes, Perceptron, Stochastic Gradient Descent (SGD), Linear SVC, and Decision Tree were the eight distinct machine learning models that were used to predict student performance.

This table displays the results of an evaluation of the performance of various machine learning models based on the Student Performance Dataset in attempting to predict student performance. Accuracy, precision, recall, and the F1 score were the four primary metrics that were used in the evaluation of the models. The findings reveal that each model is effective in classifying students and identifying

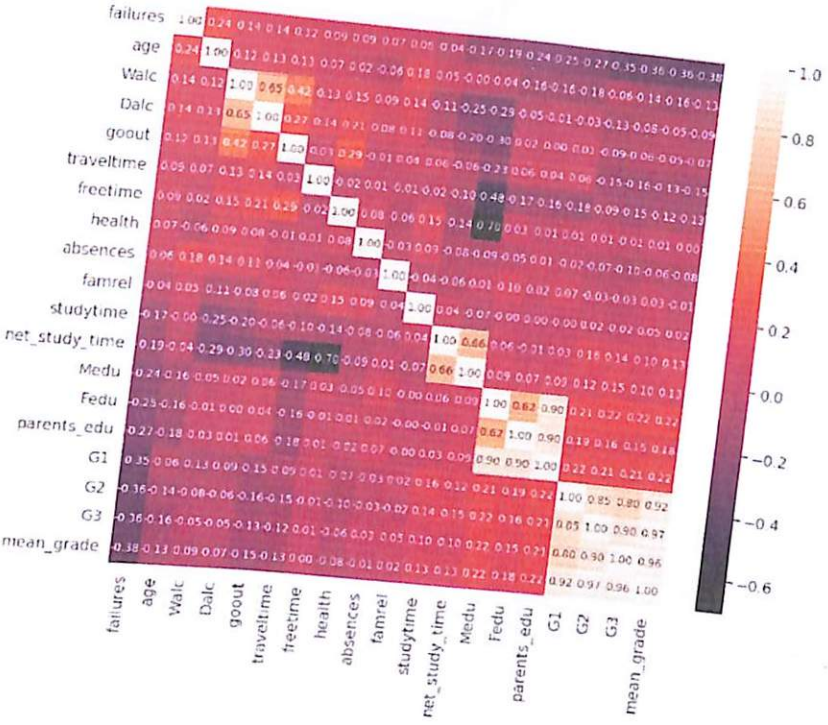

Figure 4.1: Correlation

those who are at risk of failing the class as well as those who are likely to achieve higher grades.

The accuracy, precision, recall, and F1 score were the four major measures that were used in the evaluation of these models' performance.

Naive Bayes accomplished outstanding performance, displaying a perfect 100% accuracy, precision, and recall score, as well as an F1 value. It's possible that

Model	Accuracy	Precision	Recall	F1_Score
Naive Bayes	100.00	100.00	100.00	100.00
Decision Tree	100.00	100.00	100.00	100.00
Logistic Regression	97.69	97.31	97.69	97.29
Random Forest	97.69	96.62	97.69	96.97
Linear SVC	97.69	96.37	97.69	96.98
Stochastic Gradient Decent	92.31	94.78	92.31	91.63
Perceptron	90.00	86.73	90.00	88.33
Support Vector Machines	86.92	75.56	86.92	80.84
KNN	84.62	81.20	84.62	82.77

able 4.1: Performance Evaluation of Machine Learning Models for Predicting udent Performance

categorical data both contributed to its excellent performance in this instance.

The Decision Tree model, which is quite similar to the Naive Bayes model, received flawless scores across all criteria, which indicates that it properly predicted student success. As a result of their capacity to manage non-linear interactions and the fact that they are easily interpretable, decision trees are useful in the context being discussed here.

The logistic regression model performed wonderfully in predicting student performance, with an accuracy of 97.69%. It performed quite well in terms of precision, recall, and F1 scores, which demonstrates how well it was able to categorize children. Because of its ease of use and clarity of explanation, logistic regression is a technique that is frequently chosen for binary classification jobs.

The Random Forest model likewise did quite well, with an accuracy of 97.69% along with competitive precision, recall, and F1 scores.

Linear SVC proved trustworthy performance in forecasting student performance, with an accuracy of 97.69% thanks to its use of linear regression. It obtained a high recall score, which indicates that it is able to identify a considerable proportion of children who are at danger of failing.

The model developed using stochastic gradient descent (SGD) achieved an accuracy score of 92.31% and a relatively good precision score. In spite of the fact that it did not perform quite as well as some of the other models, it was still able to provide insightful information regarding the forecast of student performance.

The accuracy of the Perceptron model was 90.00%, and it had a satisfactory recall score as well. Despite the fact that it had a lower precision score, it was able to demonstrate the potential to detect a considerable percentage of students who would be at danger of failing.

The SVM displayed acceptable performance in forecasting student performance, with an accuracy of 86.92%. When compared to other models, it exhibited poorer precision and recall scores, which suggests that it may have problems appropriately classifying students in some circumstances.

The KNN model exhibited encouraging results and reached an accuracy of

84.62%, despite having poorer precision, recall, and F1 scores than other models.

Chapter 5

Conclusion

5.1 Conclusion

Using the Student Performance Dataset that was available on Kaggle, this dissertation carried out an investigation into productive machine learning strategies that can be utilized to raise students' overall performance. In order to conduct the research, a complete procedure was utilized, which included data cleaning, feature engineering, data visualization, data preprocessing, and the application of eight distinct machine learning models.

The findings of the study showed that a number of the machine learning models exhibited high accuracy and did a good job of forecasting the performance of the students. Specifically, the Naive Bayes and Decision Tree models both achieved flawless scores across all evaluation measures, which demonstrates the remarkable prediction ability of both of these models. Both Logistic Regression and Random Forest had good performance, which is a sign that these methods are helpful in classifying pupils into a variety of performance groups.

The findings emphasize the potential of machine learning approaches in identifying students who are likely to receive higher scores or who are at risk of failing the course. These kinds of findings can be helpful for educational institutions, policymakers, and individual educators as they work to enhance student outcomes through the implementation of focused interventions and support systems.

It is essential to recognize that attaining flawless scores in all models raises issues regarding overfitting and indicates the requirement for additional validation and testing on a variety of datasets. In addition, even though the primary focus of this investigation was on analyzing the Student Performance Dataset, it is possible that a more in-depth comprehension of the factors that influence student performance could be attained by including additional datasets and taking into account a wider variety of aspects, such as socioeconomic status and the instructional strategies that are utilized.

In general, this dissertation makes a contribution to the expanding body of study on the subject of utilizing the methods of machine learning to improve educational practices. The examination of a variety of models and evaluation measures yields insightful information regarding the applicability and efficiency of these tools in forecasting the performance of students. The findings of this study have the potential to both support the creation of tailored interventions to improve student outcomes as well as act as a platform for future research.

When deploying predictive models in educational settings, it is essential to take into account ethical considerations, data protection, and transparency as the area of machine learning continues to experience rapid advancement. To address these problems and investigate the potential of machine learning to personalize education and raise the level of academic achievement among students, more research is required.

This study highlights the potential of machine learning approaches in assessing student performance data and generating significant insights that may be used for educational decision-making, as the conclusion of the study states. The findings that were acquired through the examination of a variety of models highlight the significance of making use of sophisticated analytics tools in order to motivate evidence-based interventions and enhance the results of education. We can work towards making the educational environment that all students are exposed to more welcoming and encouraging for all of them by combining the power of data analysis with good pedagogical practices.

5.2 Future work

In spite of the fact that this dissertation has provided useful insights into the examination of successful machine learning algorithms for increasing student performance, there are a variety of potential pathways for future research and development in this topic. To achieve even better results from this research, additional investigation could be directed at the following areas:

Incorporation of more Data Sources: It would be advantageous to incorporate more data sources in order to acquire a more thorough view of student performance. This could include things like a student's socioeconomic background, the educational level of their parents, the extracurricular activities they participate in, and how engaged they are in school. When forecasting student performance, the models can become more robust and accurate when a greater variety of variables are incorporated into the equation.

Advanced Predictive Models The development of new technology in the foreseeable future may make it possible to create more advanced predictive models. The employment of neural networks, which are capable of capturing complicated relationships and patterns in student performance data, is one potential option that could be pursued. It is feasible to improve the accuracy and prediction capacity of the models by making use of deep learning approaches, such as recurrent neural networks or convolutional neural networks, for example. These models can be taught to analyze past student records and offer staff members with early warnings when kids are at risk of performing poorly on assessments.

Individualized Education Interventions: In the future, research could focus on the development of individualized education interventions, building upon the predictive capabilities of machine learning models. By taking into account the unique qualities of each student, such as their preferred modes of instruction, areas in which they excel, those in which they struggle, and their areas of interest, it is possible to develop and implement individualized improvement methods. Students can be provided with tailored recommendations and adaptive learning routes that can be established using adaptive learning platforms that can be developed. This will allow for a more individualized and effective learning experience for the stu-

dents.

Ethical Considerations and Concerns Regarding Data Privacy It is absolutely necessary to address ethical considerations and concerns regarding data privacy as machine learning models become more implemented in educational environments. The focus of work to be done in the future should be on building frameworks and rules to ensure the ethical and responsible use of data regarding student performance. To preserve the confidentiality of student information, this includes putting in place effective data anonymization procedures, gaining informed consent, and following to data protection rules.

Integration with Learning Management Systems: Future research can investigate the possibility of integrating the machine learning models with pre-existing learning management systems (LMS) in order to maximize the practical usefulness of the machine learning models. Educators can gain real-time information and practical recommendations to enhance their instructional decision-making if they smoothly integrate the predictive capabilities within the LMS. This integration can make it easier to identify kids who are having difficulty in a timely manner and give teachers the ability to provide more targeted interventions.

Longitudinal Analysis: In the future, research could concentrate on longitudinal analysis in order to acquire a more in-depth understanding of the trajectories of student performance. It is feasible to uncover patterns, trends, and long-term elements that contribute to academic success if one tracks the performance of students over a prolonged period of time. The results of a longitudinal study can shed light on the efficiency of various interventions and make it possible to adapt one's approach in response to changing circumstances over time.

In conclusion, the focus of further research in this field should be on integrating additional data sources, investigating advanced predictive models, providing individualized therapies, addressing ethical problems, integrating with existing educational systems, and undertaking longitudinal analysis. These developments will further boost the ability of machine learning approaches to raise the bar for student accomplishment and support educators in the provision of educational experiences that are both individualized and effective [27] [28].

Bibliography

[1] M. Kantrowitz. Shocking Statistics About College Graduation Rates. https://www.forbes.com/sites/markkantrowitz/2021/11/18/shocking-statistics-about-college-graduation-rates/, nov 18 2021.

[2] S. Bansal. Classification in machine learning: Types and methodologies. https://www.analytixlabs.co.in/blog/classification-in-machine-learning/, jan 25 2023.

[3] S. M. Weiss and I. Kapouleas. An Empirical comparison of pattern recognition, neural nets, and machine learning classification methods. https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.81.9576 rep=rep1type=pdf, 2021.

[4] A. Mishra. Metrics to Evaluate your Machine Learning Algorithm. https://towardsdatascience.com/metrics-to-evaluate-your-machine-learning-algorithm-f10ba6e38234, may 28 2020.

[5] A. Almasri, E. Celebi, and R. S. Alkhawaldeh. Emt: Ensemble Meta-Based Tree Model for Predicting Student Performance. Scientific Programming, 2019:1–13, feb 24 2019.

[6] Abu Zohair and Lubna Mahmoud. Prediction of Student's performance by modelling small dataset size. International Journal of Educational Technology in Higher Education, 16(1), aug 2 2019.

[7] A. Namoun and A. Alshanqiti. Predicting Student Performance Using Data Mining and Learning Analytics Techniques: A Systematic Literature Review. Applied Sciences, 11(1):237, dec 29 2020.

[8] Learning Analytics in Higher Education. CASE STUDY I: Predictive Analytics at Nottingham Trent University. 2015.

[9] Purdue University. Student support interventions and predictive analytics. https://www.purdue.edu/idata/documents/White_Papers/Student_Support_Interventions_and_Predictive_Analytics.pdf.

[10] S.V. Rusakov, O.L. Rusakova, and K.A. Posokhina. A neural network model for predicting a risk group based on the progress of first-year students. Modern information technologies and IT education, pages 815–822, 2018.

[11] C. Cortes and V. Vapnik. Support-vector networks. Machine learning, 20(3): 273–297, 1995.

[12] V. Vapnik and A. Chervonenkis. Theory of pattern recognition: Statistical problems of learning. 1974.

[13] E. Fix and J. L. Hodges Jr. Discriminatory analysis-nonparametric discrimination: consistency properties. Technical report, Technical Report 4, 1951.

[14] T. Cover and P. Hart. Nearest neighbor pattern classification. IEEE Transactions on Information Theory, 13(1):21–27, 1967.

[15] P. McCullagh and J. A. Nelder. Generalized linear models, volume 37. CRC press, 1989.

[16] D. W. Hosmer Jr, S. Lemeshow, and R. X. Sturdivant. Applied logistic regression, volume 398. John Wiley & Sons, 2013.

[17] A. Liaw and M. Wiener. Classification and regression by randomforest. R news, 2(3):18–22, 2002.

[18] C. Chen, A. Liaw, and L. Breiman. Using random forest to learn imbalanced data. In University of California, Berkeley, volume 110, pages 24–32, 2004.

[19] P. Domingos and M. Pazzani. On the optimality of the simple bayesian classifier under zero-one loss. Machine learning, 29(2-3):103–130, 1997.

[20] M. Minsky and S. Papert. Perceptrons: An Introduction to Computational Geometry, volume 165. MIT Press, 1969.

[21] A. Géron. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems. O'Reilly Media, 2019.

[22] L. Bottou. Online learning and stochastic approximations. On-line learning in neural networks, 17(9):142, 1998.

[23] T. Zhang. Solving large scale linear prediction problems using stochastic gradient descent algorithms. In Proceedings of ICML, volume 21, pages 116–123, 2004.

[24] B. Schölkopf and A. J. Smola. Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond. MIT Press, 2001.

[25] Student performance dataset. `https://www.kaggle.com/datasets/devansodariya/student-performance-data`.

[26] Cyril Goutte and Eric Gaussier. A probabilistic interpretation of precision, recall and f-score, with implication for evaluation. volume 3408, pages 345–359, 04 2005. ISBN 978-3-540-25295-5. doi: 10.1007/978-3-540-31865-1_25.

[27] Bakdaulet Tolbassy. Classification of reviews, error reports and product feature requests using machine learning methods. Suleyman Demirel University Bulletin: Natural and Technical Sciences, 62(1):54–64, 2023. ISSN 2709-2631. doi: 10.47344/sdubnts.v62i1.934. URL `https://journals.sdu.edu.kz/index.php/nts/article/view/934`.

[28] Bakdaulet T. and Duman T. Predicting student performance: An svm - based approach for high school students. Limited liability company "OMEGA SCIENCE", apr 2023. URL `https://elibrary.ru/item.asp?id=50745485`.

Appendix A

Code snippets showcasing Data preprocessing and Prediction

A.1 Creating and Modifying a Binary Feature in the Dataset

```
for dataset in combine:
    dataset['fjob_binary'] = 0
    dataset.loc[dataset['Fjob'] == 'service', 'fjob_binary'] = 1
    dataset.loc[dataset['Fjob'] == 'at_home', 'fjob_binary'] = 1
del perform["Fjob"]
```

A.2 Train-Test Split in Machine Learning using sklearn

```
from sklearn.model_selection import train_test_split
X = target.drop(["paid","activities","nursery","higher",
                 "internet", "romantic"], axis=1)
```

```
Y = X["failures"]
X_train, X_test, Y_train, Y_test = train_test_split(X, Y,
                        test_size=0.2, random_state=101)
X_train.shape, Y_train.shape, X_test.shape
```

A.3 Training and Evaluating a SGD Classifier in Machine Learning

```
from sklearn.linear_model import SGDClassifier
from sklearn.metrics import accuracy_score, precision_score,
                                    recall_score, f1_score

sgd = SGDClassifier()
sgd.fit(X_train, Y_train)
Y_pred = sgd.predict(X_test)
acc_sgd = round(sgd.score(X_train, Y_train) * 100, 2)
acc_sgd

acc_sgd = round(accuracy_score(Y_test, Y_pred) * 100, 2)
prec_sgd = round(precision_score(Y_test, Y_pred,
                 average='weighted') * 100, 2)
rec_sgd = round(recall_score(Y_test, Y_pred,
              average='weighted') * 100, 2)
f1_sgd = round(f1_score(Y_test, Y_pred,
         average='weighted') * 100, 2)

print("Accuracy: {}%".format(acc_sgd))
print("Precision: {}%".format(prec_sgd))
print("Recall: {}%".format(rec_sgd))
print("F1 Score: {}%".format(f1_sgd))
```

Appendix B

Mathematical formulations of ML models

B.1 Naive Bayes Classifier

$$P(C_k|x) = \frac{P(x)P(C_k|x)}{P(x)} \tag{B.1.1}$$

B.2 Logistic Regression

$$\text{logit}(p) = \ln\left(\frac{1-p}{p}\right) \tag{B.2.1}$$

B.3 Support Vector Machine

$$K(x_i, x_j) = e^{-\gamma\|x_i - x_j\|^2} \tag{B.3.1}$$