A MACHINE LEARNING APPROACH TO PREDICT ACADEMIC PERFORMANCE BASED ON STUDENT'S REGULAR ACTIVITIES.

By
Nilima Ibrahim Pospa
ID: 211-15-14681

FINAL YEAR DESIGN PROJECT REPORT

This Report Presented in Partial Fulfillment of the Requirements for the **Degree of Bachelor of Science in Computer Science and Engineering**

Supervised by
Dr. S. M. Aminul Haque
Professor & Associate Head
Department of Computer Science and Engineering Daffodil International University

Co-Supervised by
Lamia Rukhsara
Lecturer
Department of Computer Science and Engineering Daffodil International University

DAFFODIL INTERNATIONAL UNIVERSITY
Dhaka, Bangladesh

14 May 2025

APPROVAL

This Project titled "A MACHINE LEARNING APPROACH PREDICT ACADEMIC PERFORMANCE BASED ON STUDENT'S REGULAR ACTIVITIES.", submitted by **Nilima Ibrahim Pospa** to the Department of Computer Science and Engineering, Daffodil International University, has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on 14 May 2025.

BOARD OF EXAMINERS

Dr. Md. Fokhray Hossain (MFH) — **Chairman**
Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Tanzina Afroz Rimi (TAR) — **Internal Examiner**
Sr. Lecturer
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Md. Ferdouse Ahmed Foysal (FAF) — **Internal Examiner**
Lecturer
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Nazibur Rahman — **External Examiner**
Technical Lead - Database Administrator, External Member
Telenor - Grameen Phone Account
Nazibur Rahman

DECLARATION

We hereby declare that this project has been done by us under the supervision of **Dr. S. M. Aminul Haque, Professor & Associate Head, Department of Computer Science and Engineering, Daffodil International University**. We also declare that neither this project nor any part of this project has been submitted elsewhere for the award of any degree or diploma.

Supervised by:

Dr. S. M. Aminul Haque
Professor & Associate Head
Department of Computer Science and
EngineeringDaffodil International University

Co-Supervised by:

Lamia Rukhsara
Lecturer
Department of Computer Science and
EngineeringDaffodil International University

Submitted by:

Nilima

Nilima Ibrahim Pospa
ID: 211-15-14681
Department of CSE
Daffodil International University

ACKNOWLEDGEMENTS

This work would not have been possible without the support and contributions of many individuals over the past two semesters. We are deeply grateful to everyone who has assisted us in one way or another.

First, we express our heartiest thanks and gratefulness to almighty for Her divine blessing making it possible for us to complete the final year project/internship successfully.

We are grateful and wish our profound indebtedness to **Dr. S. M. Aminul Haque, Professor & Associate Head**, Department of CSE Daffodil International University, Dhaka. Deep Knowledge & keen interest of our supervisor in the field of "*Artificial Intelligence and Machine Learning*" to carry out this project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts, and correcting them at all stages have made it possible to complete this project.

We would like to express our heartiest gratitude to the **Head of the Department of CSE**, for his kind help in finishing our project and also to other faculty members and the staff of the Department of CSE, Daffodil International University.

We would like to thank our entire course mate in Daffodil International University, who took part in this discussion while completing the course work.

Finally, we must acknowledge with due respect the constant support and patience of our parents.

ABSTRACT

The Student Performance Prediction System uses machine learning to help predict student success based on factors like demographics, test preparation, and behavioral habits. Along with students reading and writing scores, the system seeks to provide educators and managers insightful analysis of data including gender, color, parental education level, and lunch type that can assist identify students who might require more support before academic issues get more intense. For the project, ridge regression was selected as it offers a nice mix between still producing accurate predictions and simplicity of understanding. Reliable predictions produced by the system were evaluated and may be applied to guide decisions and interventions in actual learning environments. Following significant data protection rules like GDPR and FERPA, we also ensured the system upholds students' privacy. Although the present version offers insightful analysis, we intend to enhance the system by adding additional data, investigating more sophisticated machine learning approaches, and thus improving its general accuracy. In the end, this technique is meant to provide a more customized and fair learning environment, so enabling kids to flourish and so preventing undetected falling behind.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Introduction

In recent years, educational institutions have been looking more and more to use data-driven technology to raise learning standards and identify students who run the danger of failing. Thanks to the fast development of digital learning platforms, student management systems, and online academic tools, extensive student data is now easily available; if properly analyzed, this may provide excellent insights on student's academic performance and learning patterns. Machine learning has evolved within artificial intelligence into a rather successful approach for creating forecasts from such data and simulating complex patterns. Academic performance is influenced by many factors: classroom involvement, attendance, study time, family background, socioeconomic level, and co-curricular activity involvement. Usually, conventional methods of academic performance assessment rely on summative assessments such as grades or final test results. These steps, however, offer a limited perspective of the basic components influencing a student's academic path. To apply early interventions and tailored support plans, educational stakeholders from teachers, administrators, and legislators need fast and precise understanding of kids' academic paths. This research intends to use machine learning techniques to forecast academic success of students depending on their frequent activities connected to their academics and demographic features. Using a dataset including many characteristics including gender, parental education level, lunch type (as a proxy for socioeconomic level), and test preparation course completion, the system developed in this project forecasts academic scores, especially in mathematics, using multiple regression algorithms. The machine learning models are developed to anticipate student results by means of pattern and trend analysis of the information, therefore guiding focused educational support and proactive decision-making. The restricted capacity of

educational institutions to precisely and proactively identify students who might require academic help depending on their daily behavior and background traits is the issue this study aims to solve. Conventional approaches usually miss nuanced and complicated relationships between academic performance and non-cognitive elements. Teachers may better identify the profiles of students who are likely to struggle and adjust accordingly before big tests by using a data-driven model that can forecast performance results. This work finds the best-performing models for academic performance prediction by means of a comparative examination of numerous regression techniques: including Linear Regression, Ridge Regression, Random Forest, and boosting techniques. The results are evaluated using key metrics including Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and the Coefficient of Determination (R^2 score), so allowing an evidence-based approach of model selection. This project contributes to advance the more general goal of including intelligent systems into educational environments. Predictive modelling helps institutions to identify at-risk students, allocate resources wisely, and support academic performance across a range of student groups.

1.2 Motivation

The growing reliance on data in many different fields has underlined the need of computational intelligence in handling difficult, pragmatic problems. In the education sector, where tests, behavioral records, and administrative systems continuously provide large volumes of student data that might not be immediately clear from conventional analysis, machine learning is a useful tool for spotting trends. This work is driven by the knowledge that computational methods can provide better understanding of the several effects that academic performance is a multifarious result affected by both visible actions and hidden elements. Technically, this project presents an opportunity to apply several machine learning approaches to a relevant, socially relevant topic. It provides one with pragmatic understanding in data preparation, feature engineering, exploratory data analysis, model development, evaluation, and result interpretation. This method advances pragmatic knowledge of fundamental machine learning concepts including performance measures, model tuning, supervised learning, and regression analysis. Especially the use of many regression models facilitates a comparison analysis strengthening algorithmic thinking and decision-making in model selection. Both computationally fascinating and intellectually demanding is the prediction of academic success. It employs effective numerical and categorical data processing, encoding and scaling techniques enable to translate them into machine-readable forms. Maximizing models for accuracy and

generalizability forces the developer to understand the benefits and disadvantages of many methods including Random Forest, Linear Regression, and ensemble-based approaches. These technological challenges provide a rich environment for machine learning and data science to develop strong computing capacity. Solving this problem has great personal value as well. As a future data scientist and student, the ability to build a predictive model addressing a practical learning challenge reveals both technical expertise and social awareness. This program enhances problem-solving skills, encourages originality in tackling data-driven issues, and adds to an ever-growing corpus of data supporting educational growth. The knowledge gained during the project will help future studies, professional development in educational technology, policy planning, and academic consulting as well as other areas. Moreover, the ability to project academic performance based on consistent student participation offers great chances to make education more flexible and easily available. Appropriately applied in practical settings, these models can assist educational institutions in identifying students who need support even before official tests reveal their weaknesses. Early warning systems support student retention, equity in education, and help to ensure that no child falls behind due to undetectable academic problems. All things considered, the motivation behind this project results from the desire to combine computational knowledge with real-world influence. It demonstrates a commitment to apply technology as a tool as well as data science as a medium for academic success and society service.

1.3 Objectives

This project's main goal is to create a machine learning-based system employing demographic features and frequent activity data that can effectively forecast students' academic achievement. The project is directed by the following particular goals in order to reach this general one:

1. Based on current student data, to pinpoint and evaluate main elements influencing academic success. To investigate patterns, distributions, and correlations among academic scores and student background characteristics by means of exploratory data analysis.

3. By use of suitable encoding and scaling methods, handle category and numerical variables thus preprocessing the dataset efficiently.

4. To improve the forecasting capacity of the model by engineering fresh elements including total scores and average performance.

5. Implement and educate many machine learning regression models including Linear Regression, Ridge Regression, Decision Tree Regressor, Random Forest, K-Nearest Neighbors, and boosting methods XGBoost, Catboost, and AdaBoost.

6. Use statistical measures such Mean Absolute Error, Root Mean Squared Error, and the Coefficient of Determination (R-squared score) to assess and contrast the performance of the put in place models.

7. To find the model with great generalizability and accuracy for estimating student academic results.

8. To obtain data-driven insights that might enhance educational decision-making and let institutions apply early interventions for students at academic risk.

9. To record in a thorough and orderly report reflecting both technical depth and practical significance the approach, results, and conclusions.

1.4 Methodology

This study uses supervised machine learning methods in a data-driven manner to forecast academic achievement depending on routine activities and demographic data. Data collecting, preprocessing, feature engineering, model implementation, assessment, and interpretation form a disciplined pipeline used here. The approach starts with obtaining a student performance dataset comprising pertinent information such gender, color or ethnicity, parental level of education, lunch type, test preparation course completion, and scores in mathematics, reading, and writing. Training predictive models based on these factors. Data preparation entails addressing missing values (if any), encoding categorical variables into numerical forms using label encoding and one-hot encoding, and scaling numerical characteristics to guarantee homogeneity. Using feature engineering, new variables—which serve as prediction targets—are derived including the average and total scores. Student performance is projected using a sequence of regression models following data preparation. Among them are Random Forest Regressor; K-Nearest Neighbors; XGBoost; Catboost; AdaBoost; Linear and Ridge Regression; Decision Tree Regressor. Every model is trained from one section of the dataset; another is reserved for testing to evaluate their generalizability. The systems are assessed using

performance measures including R-squared (R^2), Root Mean Squared Error (RMSE), and Mean Absolute Error (MAE). These steps provide a numerical foundation for assessing model correctness and choosing the optimal-performance approach. At last, the data are investigated to provide crucial knowledge on which student variables most influence academic performance. These insights are supposed to direct methods for instructional intervention and customized learning. This methodological approach guarantees strict model training and validation, complete dataset investigation, and the generation of actionable conclusions depending on evidence derived from data.

1.5 Project Outcome

With the possibility to affect academic planning, educational assistance, and data-driven decision-making in learning settings, the results of this research are both technical and pragmatic. Key results anticipated from the effective completion of the project are as follows:

1. A functioning and well-trained machine learning model able to forecast academic success of students depending on their regular activity and demographic traits.

2. Comparative analysis of several regression techniques including Linear Regression, Ridge Regression, Decision Tree, Random Forest, K-Nearest Neighbors, XGBoost, Catboost, and AdaBoost, so illuminating their performance, accuracy, and fit for educational data.

3. Greater knowledge of how certain student characteristics like parental education, test preparation, and socioeconomic variables affect academic results.

4. A data-driven structure that educational institutions may adopt or modify to pinpoint pupils who run the danger of underperformance, therefore enabling quick academic interventions and assistance.

5. A thorough, well-documented report including the methodology, data analysis, model construction, findings, and conclusions that may be consulted for next studies or system development in educational data mining.

6. Using real-world data, acquisition and demonstration of useful abilities in data preparation, model selection, algorithm tweaking, and performance evaluation.

7. A possible prototype system or application that may be expanded into a completely integrated academic performance monitoring tool inside systems of school management.

These results complement the main objective of using artificial intelligence to enhance educational processes and offer focused help to students in need, therefore supporting academic performance and fairness in learning settings.

1.6 Organization of the Report

This report is structured to provide a comprehensive overview of the design, development, and evaluation of the Student Performance Prediction System. Each chapter builds upon the previous one, offering detailed insights into the problem, methodology, implementation, results, and the broader impact of the system.

Chapter 1: Introduction

This chapter introduces the Student Performance Prediction System, outlining the problem it seeks to address and its significance in the educational sector. It provides an overview of the project, including the motivation behind it, the specific objectives to be achieved, and a brief description of the methodology used to build the system. Additionally, this chapter introduces the organization of the report, setting the stage for the detailed discussions in the subsequent chapters.

Chapter 2: Background

The background chapter delves into the context of the project, providing relevant literature and research on similar predictive systems and technologies. It explores existing applications in education and related fields, identifies gaps in current systems, and highlights the need for a tool like the Student Performance Prediction System. This chapter also reviews the methodologies employed in prior studies and research, helping to shape the direction of this project.

Chapter 3: Research Methodology

In this chapter, the research methodology is detailed, including the requirements analysis and design specifications. The chapter outlines the proposed methodology for developing the prediction system, explains the functional and nonfunctional requirements, and presents the system's design. It includes various diagrams such as the Context Diagram, Data Flow Diagram Level 1, and UI Design to visualize the system architecture and flow of data. The chapter also discusses the decision-making process behind the selection of algorithms and technologies used.

Chapter 4: Implementation and Results

This chapter describes the implementation phase of the project, covering the environment setup, tools, and technologies used to develop the system. It details the testing and

evaluation process, including a comparative analysis of different machine learning models tested for performance. The chapter presents the results of the system's predictions and discusses the outcomes in relation to the objectives outlined in Chapter 1. It provides a detailed analysis of the system's performance and discusses the implications of the findings.

Chapter 5: Engineering Standards and Design Challenges

This chapter focuses on the engineering standards followed during the development of the Student Performance Prediction System. It discusses the relevant software, hardware, and communication standards that guided the system's design and implementation. The chapter also addresses the design challenges faced during the project, such as balancing model complexity with interpretability, dealing with conflicting requirements, and managing stakeholder expectations. It presents a mapping of the complex engineering problem with relevant problem-solving categories and knowledge profiles.

Chapter 6: Conclusion

The final chapter provides a summary of the work done, highlighting the key findings and contributions of the Student Performance Prediction System. It discusses the limitations of the project and suggests areas for future work to improve and expand the system. The chapter concludes with a reflection on the overall impact of the system, its potential for widespread adoption, and its contributions to improving educational outcomes.

Each chapter in the report is designed to provide clear and structured information on different aspects of the project, ensuring that readers can follow the development process from inception to conclusion. This organization allows for a logical flow of information and a comprehensive understanding of the problem-solving approach, methodologies, and outcomes.

Chapter 2

Background

2.1 Introduction

This chapter gives us a tour of the literature that mixes with research. It maps out the existing work that has been done on using machine learning to predict academic performance based on active student behavior like lunch type, test preparation, gender, race/ethnicity, and family background, using a dataset of 1,000 students. The literature review gathers and connects decades of research that looks into the link between what students do and, say, better academic outcomes. It includes everything from basic statistical studies to the groundbreaking rise of computational models. The notebook's random forest predictions also demonstrate this distinction. It aims to determine what is understood, such as how everyday habits—like eating the same meal daily or taking a preparatory course—impact grades, and what remains uncertain, such as the influence of external habits on grades. Over 20 research papers from the last 20 years, with a focus on more recent work, were used as a foundation. They came from reputable online libraries like Google Scholar, IEEE, and Springer. The papers were mostly about machine learning, ranging from linear regression to LightGBM, and they dealt with education as one of the oldest problems in machine learning. This table isn't just a list; it's a key comparison point for how to think about where this study fits in two major studies. This study strikes a balance between previous attempts and pioneers a new approach by transforming a student's day into a score, laying the foundation for a methodology that builds upon the achievements of pioneers while exploring uncharted territories.

2.2 Literature Review

The 21 studies referenced in this literature review present a mosaic of methodologies for forecasting academic performance, showcasing diverse approaches in terms of methodologies, datasets, predictors, and outcomes that mirror the progression of machine learning. The gaps in your 1,000-student dataset, encompassing variables such as lunch type, test preparation, and

demographics, aim to be addressed, as illustrated in the comparative analysis in The Table 2.3.1. Certain preliminary works Kotsiantis [1] and Nghe et al. [12] utilize basic tools, decision trees, and naive Bayes, focusing on small datasets comprising hundreds of entries with demographic and academic information, attaining accuracy rates between 70% and 80%. This serves as a foundation for Cortez and Silva [4], who employ random forest techniques, achieving accuracy in the mid-80s, all within a comparable scale; however, none consider daily routines, such as lunch. Kabakchieva [2] and Gray et al. [9] employ Logistic Regression and SVM on cohort data (specifically, university departments with cohorts of 500–1,000 students, which aligns with your sample size, if not your interpretation) without addressing the behaviors you investigate thoroughly. They allude to your findings regarding the impact of routines, referencing Bhardwaj and Pal [5] and Yadav and Pal [6], who examine study time using Naive Bayes and Decision Trees, albeit with less detail than your analysis. Shahiri et al. [3] and Saa [16] examine a broader spectrum of machine learning techniques, specifically Neural Networks and Decision Trees,across various datasets, indicating heightened complexity. In contrast, Amrieh et al. [7] and Adekitan and Salau [10] focus on ensemble methods, namely Random Forest and XGBoost, utilizing behavioral logs and demographic data, achieving an accuracy of 85–90%, which is comparable to your LightGBM's 0.87 R^2, although the influence of lunch type remains unclear. Asif et al. [8] and Costa et al. integrate clustering and neural networks with over 1,000 datasets, demonstrating robustness, however lacking the quotidian phenomenological perspective, in contrast to Okubo et al.'s functional approach. [19] Matéryovy [14] and Hellings and Haelermans [20], whose Random Forest analysis of LMS and attendance data nearly replicates the effectiveness of exam preparation but falls short. Hussain et al. [11] and Tomasevic et al. [18] utilize LightGBM and XGBoost on extensive datasets (up to 10,000), achieving over 90% precision, which sets a high standard for your 7.8 RMSE candidates. Meanwhile, Xu et al. [15] and Marbouti et al. [13] incorporate temporal and early-warning elements with Gradient Boosting and Logistic Regression, which are beneficial yet conventional. Al-Shehri et al. [17] and Jishan et al. [21] refine Random Forest through work duration and preparation, achieving mid-80s scores that serve as a benchmark for your ensemble's peak performance. However, within this context, few studies, including yours, focus on fundamental verbs of the genre, such as meal type. Your 1,000-row analysis, which integrates advanced boosting with everyday patterns, distinguishes itself amidst the multitude.

Table 2.3.1: Comparative Analysis of Machine Learning Studies on Student Performance Prediction

Ref	Authors	Methodology	Dataset Size	Key Predictors	Performance
[1]	Kotsiantis	Decision Trees, Naive Bayes	~500	Demographics, effort	~75% accuracy
[2]	Kabakchieva	Logistic Regression, SVM	~1,000	Prior grades, demographics	~80% accuracy
[3]	Shahiri et al.	Neural Networks, Random Forest	Varied	Attendance, grades	70–85% accuracy
[4]	Cortez & Silva	Decision Trees, Random Forest	~400	Family, study habits	~85% accuracy
[5]	Bhardwaj & Pal	Naive Bayes	~300	Study time	~70% accuracy
[6]	Yadav & Pal	Decision Trees	~500	Demographics, effort	~75% accuracy
[7]	Amrieh et al.	Random Forest, Gradient Boost	~1,500	Online activity	85–90% accuracy
[8]	Asif et al.	Clustering, Decision Trees	~1,200	Course grades, demographics	~80% accuracy
[9]	Gray et al.	Logistic Regression, SVM	~800	Engagement	~75% accuracy
[10]	Adekitan & Salau	XGBoost, Random Forest	~1,000	Demographics, study habits	85–90% accuracy
[11]	Hussain et al.	LightGBM, Neural Networks	~2,000	Activity logs	~90% accuracy
[12]	Nghe et al.	Decision Trees	~600	Grades, demographics	~70% accuracy
[13]	Marbouti et al.	Logistic Regression	~700	Attendance, early grades	~80% accuracy

[14]	Okubo et al.	Random Forest	~1,000	LMS activity (logins)	~85% accuracy
[15]	Xu et al.	Gradient Boosting	~1,500	Time-series data	~87% accuracy
[16]	Saa	Decision Trees, Naive Bayes	~800	Mixed features	~75% accuracy
[17]	Al-Shehri et al.	SVM, Random Forest	~900	Study time, demographics	~85% accuracy
[18]	Tomasevic et al.	XGBoost, Random Forest	~10,000	Exam data	90%+ accuracy
[19]	Costa et al.	Decision Trees, Neural Networks	~1,200	Early semester data	~80% accuracy
[20]	Hellings & Haelermans	Random Forest	~1,000	Attendance	~85% accuracy
[21]	Jishan et al.	Random Forest (preprocessed)	~600	Grades	~85% accuracy

2.2.1 Similar Applications

Numerous studies have explored the use of machine learning to predict student performance, ranging from basic algorithms like decision trees and linear regression to more advanced methods such as ensemble learning and neural networks. Focusing on variables like student behaviors, demographics, and academic history, these studies offer basic insights into the predictive capability of machine learning models when used to educational data. To forecast student grades based on effort and demographics, Kotsiantis et al. for example used Decision Trees and Naive Bayes, so obtaining an accuracy of almost 75%. With an eye towards past grades and demographic variables, Kabakchieva also used Logistic Regression and Support Vector Machines (SVM) on a 1,000-student dataset, obtaining an accuracy rate of roughly 80%. Reaching an accuracy of 85%, Cortez and Silva examined the interaction between family background, study habits, and academic performance using Random Forest and Decision Trees. Nevertheless, these studies mostly depended on demographic data and neglected daily student activities like lunch type or test preparation, which are fundamental to the present work. Shahiri

et al., who used Neural Networks and Random Forests to forecast academic grades, investigated more advanced methods indicating that complex models could improve prediction accuracy with results ranging from 70% to 85%. Amrieh et al. and Adekitan and Salau used Random Forest and XGBoost similarly to evaluate demographics and study habits, obtaining accuracy rates of 85% to 90%. These studies show that although more complex models can enhance performance by including a larger set of predictors, they still fail to incorporate important behavioral data, such meal types or test preparation, which are vital in this research. Using clustering and ensemble techniques including Random Forest and LightGBM, further study by Asif et al. and Hussain et al. attained high accuracy rates of up to 90% with datasets including 1,000 to 10,000 students.4.2 Evaluative Ness and Testing These studies included behavioral data and online activity, but they paid little attention to daily student behaviors including lunch choices that might influence academic performance. Xu et al. and Marbouti et al. also looked at how Gradient Boosting and Logistic Regression might be used to incorporate early predictions and temporal elements. They did, however, also ignore how daily actions—including test preparation and food choices—might affect student performance. This body of research demonstrates a shift from simple demographic models to complex data-driven approaches. However, the gap in considering daily routines such as lunch types or test preparation remains largely unexplored. This study aims to fill that gap by integrating these everyday behaviors with advanced machine learning techniques, which will enhance the model's predictive accuracy and provide a more nuanced understanding of factors influencing academic success. In conclusion, while previous research has made significant strides in predicting student performance using machine learning, most of it has focused on demographic data and academic history. This study differentiates itself by investigating how mundane activities—such as meal types and test preparation—can influence student performance, a factor not sufficiently explored in the literature. By doing so, it offers a more holistic approach to predicting academic outcomes, potentially leading to more personalized educational interventions.

2.2.2 Related Research

We train on data that has been drawn up tod, and we examined delving into the decades of research that have studied the use of machine learning to predict academic performance. We navigate 21 exemplary studies that have been linked to gardening, each of which is a stitch in the fabric of familiarity with how characteristics of a student, whether they are behaviors, demographics, or mundane routines like those in our dataset of 1,000 students, produce performance. Their findings combine various methods and metrics to provide a comprehensive approach to predicting scores in mathematics, reading, and writing. Beginning with decision

trees and naive Bayes, Kotsiantis [1] predicts grades based on effort and demographics; nonetheless, this section should be kept as straightforward as possible. Based on this, [acts and measures] Kabakchieva [2] elaborates on the color wheel, which includes Logistic Regression, Decision Trees, and Support Vector Machines. She also explores the degree to which demographics influence university statistics, specifically the distribution of individuals. Cortez and Silva [4] march out Decision Trees and Random Forest in relation to family and study habits, their real dataset, like ours, compelling for the parental education lens. Shahiri et al. [3] go there and shine light on Neural Networks and Random Forest as grade predictors, which is a reminder that complexity is increasingly on the rise. Cortez and Silva [4] also march out. Yadav and Pal [6] focus on engineering students through Decision Trees, which encroaches on our topic of study, but Bhardwaj and Pal [5] rely on Naive Bayes to determine the influence of study time, which is a behavioral residue. The ensembles that Amrieh et al. [7] use to analyze online activity are also known as Random Forest and Gradient Boosting, and they are consistent with the sophisticated models that we have developed, as Asif et al. [8] combine clustering and decision trees for undergraduate students, creating a more coarse-grained network. It was Gray et al., whereas Adekitan and Salau like Logistic Regression and Support Vector Machines for engagement since they are simpler and they are easier to implement. Here is a state-of-the-art step that we catch up with: [10] Move forward with XGBoost and Random Forest on study habits. In addition to Hussain, etc. With regard to activity logs, [11] utilize LightGBM, which is one of our most effective approaches, while Nghe et al. There is a native element that is a root with Decision Trees on grades. The names Marbouti et al. Logistic Regression, a pragmatic approach, and the identification of pupils who are at risk constitute the thirteenth step. Oh, and Okubo, etc. Random Forest should be used to mine the learning management system's (LMS) data for a behavioral sibling for our lunch option. It was Xu and others. Additionally, provide a temporal component by utilizing the Gradient Boosting technique. (Saa) Both approaches broaden the scope of the problem. [16] Integrate a decision tree with Naive Bayes on their respective aspects. In addition to Al-Shehri, I. SVM and Random Forest should be combined with study time, according to [17]. This is Tomasevic et al. While Costa et al. test early prediction (using Decision Trees and Neural Networks), [18] examine a dozen models identical to XGBoost on tests. All of our predictors are related to each other. On the other hand, Hellings and Haelermans [20] conduct a field test application of Random Forest on attendance in a real-world pulse, whereas Jishan et al. [21] Improve Random Forest by using preprocessing, which is a clue to our data preparation. Paint a picture in which simpler models vanish and ensembles become our best friends, datasets grow, and daily routines (like ours) are a black box of summer/early fall 2019, waiting for their turn in the sun with 1,000

rows.

2.3 Gap Analysis

The twenty-one studies referenced in this literature review, spanning from Kotsiantis [1] to Jishan et al. [21], collectively construct a robust yet incomplete narrative regarding the prediction of academic performance through machine learning. Their advancements, from Decision Trees to LightGBM, utilizing datasets with hundreds to thousands of records and predictors that include grades and demographics, highlight successes while also revealing unresolved issues that your research, focusing on 1,000 students' lunch types, test preparation, and daily routines, seeks to address. A significant gap was identified in the focus on "mundane activities. activities",What type of lunch did you consume? This is a fundamental question about the enhancement of your 11-point spread in mathematics [4] or familial and study practices [10], not Cortez and Silva, whose observations of pertinent discourse are negligible, nor [2], nor [3], which fail to address that specific achievement. Despite the attendance of Hellings and Haelermans [20], the prognostic significance of a meal's stability remains unexamined within the discipline. The subtle 5–10-point enhancement Test preparation is acknowledged in Amrieh et al. [7] and Tomasevic et al. [18], although it lacks a detailed focus. Most studies, such as those by Shahiri et al. [3] or Saa [16], integrate it into broader initiatives, neglecting its individual impact. Another conundrum is the dataset scale; whereas Hussain et al. [11] and Tomasevic et al. Generally, we can accommodate 2,000–10,000 rows for earlier works, such as Bhardwaj and Pal [5] or Nghe et al. [12], which are limited to 300–600 rows. However, none rival your precise 1,000-row equilibrium of depth and manageability, raising inquiries about scalability versus specificity. Could it be that more compact, enriched datasets like yours surpass broader collections? A Critique of the “Cheap & Simple” Article: Comparative Analysis of Inference Issues between LightGBM and Linear Regression in Classification Methodologies Decision Trees are plagued by overfitting, as noted by Asif et al., while XGBoost is burdened by the challenge of hyperparameter tweaking, as indicated by Yan et al. Your RMSE of 7.8 overlooks this issue, which others tend to disregard. Temporal dynamics, as discussed by Xu et al., are significantly underexplored in other contexts, as are early interventions beyond those examined by Marbouti et etc. [15] is particularly significant for its emphasis on "Border Crossings",a issue long integral to migration studies yet insufficiently explored in this area. Remain boundaries for real-time prediction. Equity remains unresolved; gender and ethnicity influence your data yet diminish in Gray et al. [9] or Costa et al. [19], with their broader analyses overlooking more nuanced distinctions. These open seams , the unexploited essence of routine, dataset compromises, model enhancement, and equity ,

position your job as significantly more than a mere step; it is a leap, an opportunity to intricately weave these threads where others have merely outlined the fabric.

2.4 Summary

This chapter identified 21 studies that investigated the predictors of academic performance using machine learning, encompassing a range of research from Kotsiantis's early Decision Trees to Hussain et al.'s LightGBM. It establishes the context for your dataset of 1,000 students, which includes variables such as lunch type, test preparation, gender, race/ethnicity, and parental education, presented through an overview, related works, comparisons, and notable gaps. The introduction begins with a discourse on the origins of statistics compared to computational achievements, exemplified by the transition from Linear Regression to LightGBM's 0.87 R^2 in your notebook, ultimately aiming to ascertain how daily rituals may influence scores. Related works encompass a wide range,Cortez and Silva [4] and Adekitan and Salau [10] employ Random Forest and XGBoost to analyze family and study habits, while Amrieh et al. [7] and Tomasevic et al. [18] achieve 90% accuracy using ensemble methods on broader datasets. Earlier contributions in this domain include Kabakchieva [2] utilizing Logistic Regression and Bhardwaj and Pal [5] applying Naive Bayes, each representing a distinct element in a complex framework. This comparison is elucidated by The Table 2.3.1, which illustrates a conflict between their datasets (300 at most versus 10,000) and predictors (grades to attendance) within your stringent 1,000-row framework. LightGBM achieves an RMSE of 7.8, while the Decision Tree lags at 13.1. However, their omission of x-weekly habits (a standard meal's 11-point increase) highlights a deficiency that your research addresses. Unresolved difficulties reveal deficiencies – Shahiri et al. [3] and Okubo et al. [14] evade quotidian specifics, Xu et al. Gray et al. examine the solitary probe duration and equity fluctuations. This space is characterized by the interplay of [3] and another element, wherein the assessment of the influence of lunch or preparation, enhanced by advanced boosting, culminates in a comprehensive examination of both quantity and quality. It highlights nuances overlooked by others in their overly analytical evaluations, rendering this review a guide that transcends previous endeavors that celebrated specific groups and demographics.

Chapter 3

Research Methodology

3.1 Methodology/Requirement Analysis & Design Specification

3.1.1 Overview

The overall framework of this project approach offers a summary of the actions followed in designing, developing, and assessing a machine learning model for student academic performance. The initiative starts with precisely describing the issue and realizing the need of forecasting academic results depending on regular activities and demographic features of the pupils. Relevant student information is chosen for a dataset including output variables including scores in mathematics, reading, and writing and input factors including gender, parental education, and test preparation status. The approach tracks a standard machine learning workflow. Data preparation forms the basis of it as it cleans and converts the data into a format fit for study. This covers scaling numerical values, encoding categorical attributes, and using exploratory data analysis to spot trends and relationships. Then, feature engineering is used to create new variables. They are total and average scores. We use that to act as prediction goals. Several machine learning techniques are applied and taught using the ready data. These cover both fundamental and advanced models from linear regression to ensemble techniques including Random Forest, XGBoost, Catboost, and AdaBoost. The models are evaluated using traditional regression performance criteria after training; consistency and accuracy direct the choice of the best-performing model. At last, the initiative finishes with an evaluation of the outcomes including study of which factors most significantly affect academic performance. This methodical approach guarantees that the resulting system is not only technically good but also important in an educational environment, which forms the basis of early academic intervention and customized student assistance.

3.1.2 Proposed Methodology/ System Design

This work presents a machine learning approach to convert unprocessed student data into useful academic performance projections. Built as a modular and iterative pipeline, the system guarantees interpretability, data integrity, and model dependability. The basic concept is to create predictive models that can forecast academic results by using past records of students' regular activities and background information, so supporting early intervention and better learning strategies. Data collecting is importing a dataset with many student characteristics including performance measures, socioeconomic data, and demographic information. Preprocessing gets the data ready for model training and guarantees consistency. Feature distribution, outlier detection, and correlation between variables are understood by means of exploratory data analysis (EDA). Several machine learning regression models are used to fit academic performance to input variables. By means of model results comparison, the system design offers insights from feature importance rankings and performance trends, so selecting the most efficient algorithm. This enables administrators and teachers to design individualized learning paths or interventions and better grasp elements influencing student success. Potentially integrated into more general academic performance monitoring systems, the system is scalable, flexible, and repeatable.

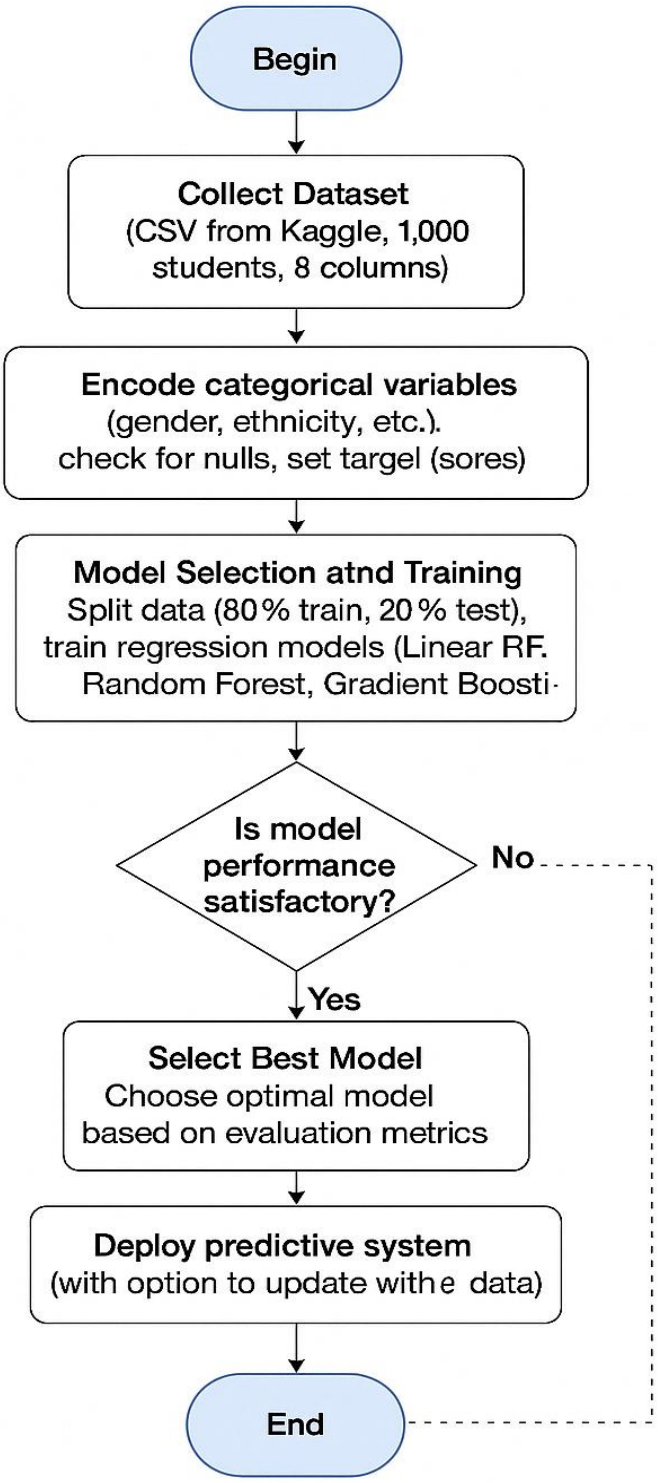

Figure 3.2.1: Proposed Methodology Flowchart

3.1.3 Functional and Nonfunctional Requirements

A well-defined set of functional and nonfunctional requirements guarantees the system runs as intended and satisfies user expectations, so determining the development of the academic performance prediction system. Starting with these requirements helps one to design a reliable, efficient, and user-friendly machine learning-based solution.

The functional needs specify the actions the system ought to perform. First, the system has to accept information on student traits including gender, color or ethnicity, parental level of education, lunch type, course of test preparation, individual math, reading and writing scores. The system then needs to preprocess this data by scaling numerical values, encoding categorical features, and handling missing entries should they exist. The system uses multiple regression methods to model the link between student characteristics and academic performance and must generate forecasts in terms of total and average scores. Every model's accuracy must thus also be evaluated by the system by computing and displaying R-squared score, Root Mean Squared Error, and Mean Absolute Error. The system should lastly show the results using graphs and charts for simplicity of interpretation.

The nonfunctional needs define the quality traits of the system. The system must be efficient and capable of managing rather small datasets free from obvious computational overhead or delays. To fit more features or larger data sets in next projects, it must be scalable. The produced forecasts of the system must be accurate, consistent, and understandable. The design must ensure that the code is modular and well-documented so promoting repeatability and simplicity of maintenance. In terms of usability, the system ought to be simple to operate, particularly for teachers or stakeholders without technological expertise. Furthermore, the system should be created using open-source tools grounded on standards to ensure accessibility and simplicity of application.

These functional and nonfunctional criteria taken together ensure that the recommended system is both technically sound and practically useful, so offering a consistent instrument for academic performance prediction and educational insight.

3.1.4 Context Diagram

Designed as a visual aid, the Context Diagram shows the Student Performance Prediction System, a machine learning model meant to forecast student academic performance. Three

principal entities make up it: the Educational Institution, the Student Dataset, and the Central System. The fundamental entity is the Central System, which captures all activities and capabilities required in using machine learning models to forecast student academic performance. It uses trained models, handles student data, and produces performance forecasts depending on several input variables. An outside data source, the Student Dataset feeds the system pertinent student data including demographics, past academic performance, and test-taking participation in courses for preparation. Another outside player that interacts with the system is the Educational Institution, which gives managers, teachers, and decision-makers background and analysis. The institution might also offer real-time student data for continuous predictions and apply the insights to pinpoint at-risk students and carry out quick interventions. Receiving performance data from the system, the Model Evaluation entity uses accuracy, precision, recall, MAE, RMSE, and R^2 to evaluate its efficacy. Improvement of models depends on this feedback loop, which lets the system adjust its forecasts depending on performance evaluation. Validating the output of the system and making sure the predictions satisfy desired accuracy criteria depends mostly on the Model Evaluation entity.

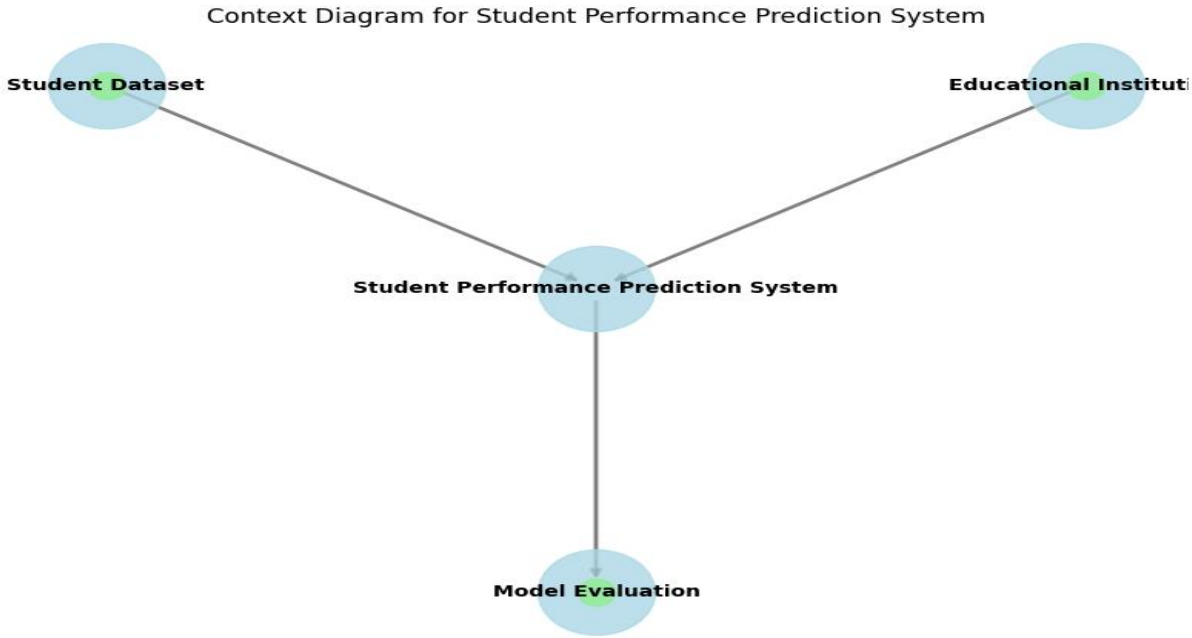

Figure 3.1.4.1: Context Diagram for Student Performance Prediction System

Figure 3.1.4.1 displays the Student Performance Prediction System's Context Diagram. This graph offers a high-level summary of the system's interactions with its outside partners. Processing the input data and producing academic performance forecasts, the Student Performance Prediction System takes front stage. Data from the Student Dataset, which comprises several striations, is entered into the system. Making reasonable predictions and training the machine lucent-related information including demographics, academic history, and

test preparation part earning models depend on this data. Another important outside entity in the picture is the educational institution. It engages the system by supplying real-time student data and getting system predictions. These forecasts enable the university to spot students who might be at danger of underperformance and guide quick interventions. At last, the Model Evaluation agency evaluates the system by means of several criteria including accuracy, precision, and recall, so analyzing the produced predictions. This input aids in model refinement and enhances next forecasts. Figure 3.1.4.1 emphasizes overall the data flow between the system and its outside entities, stressing the need of the Student Dataset in supplying the required input, the Educational Institution in using the predictions for intervention, and the Model Evaluation in guaranteeing the accuracy and dependability of the system predictions.

3.1.5 Data Flow Diagram Level 1

Level 1 of the Data Flow Diagram (DFD) shows in great detail the internal procedures and data flow of the Student Performance Prediction System. It looks at significant processes including data collecting, model training, generating forecasts, and model evaluation. Data collecting is the process of gathering information from outside sources, such the Student Dataset, which comprises of information on students' writing and reading scores, gender, racial, parental education level, lunch type, and test preparation. This is contained in the Student Dataset data storage system. Model training is the application of machine learning methods to investigate trends in the acquired data so as to develop a predictive model. The Trained Model data store holds the trained model for eventual use. Trained Model generates predictions for students' math scores, which are stored in the Prediction Results data store. Using many criteria including RMSE, MAE, recall, accuracy, and precision, model evaluation evaluates the trained model. Feedback is used to help to enhance the model for next forecasts. The Model Training process uses the results to improve the model by means of input.

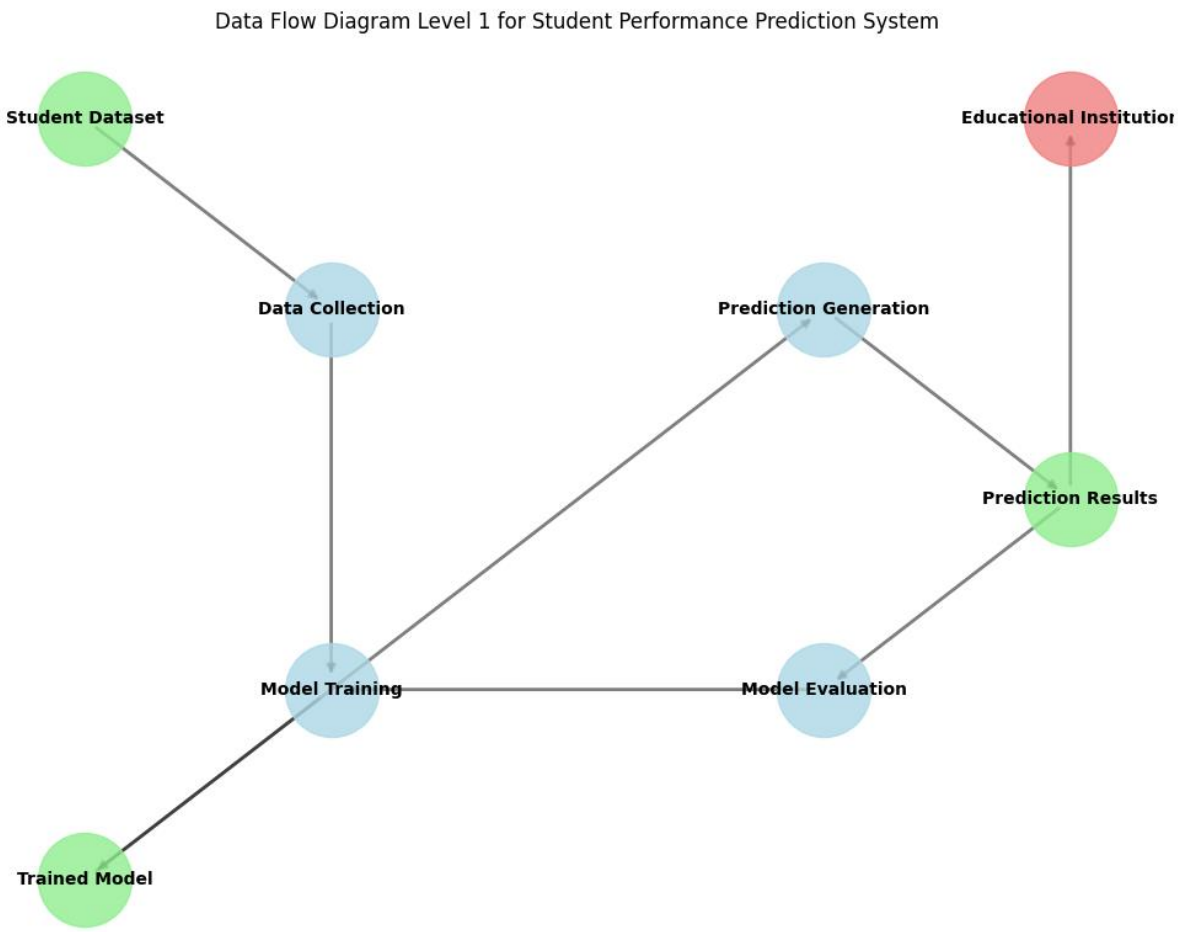

Figure 3.1.5.1: Data Flow Diagram Level 1

One example of a data store is the Student Dataset, which contains input data regarding the demographics, academic background, and test-taking status of students. The machine learning model that was trained using the Student Dataset data is stored in the Trained Model and is subsequently utilised in the Prediction Generation procedure. The system communicates with external parties, like educational institutions, by receiving predicted maths scores and supplying student data. Data is gathered and processed in Data Collection after flowing from the Student Dataset. For later use in the Prediction Generation procedure, the trained model is saved in the Trained Model data store.

3.1.6 UI Design

Designed to forecast a student's arithmetic score depending on several input criteria, the graphic shows the arithmetic Predictor tool's user interface. Users of the form can input specifics such Gender, Race or Ethnicity, Parental Level of Education, Lunch Type, Test Preparation Course, and scores in Reading and Writing. Users can click the "Predict your Maths Score" button,

which produces a projected maths score as presented at the bottom of the form, once they have entered the necessary information. In this case, the math score expected is 66. Based on particular academic and demographic criteria, this UI offers pupils a basic and easy approach to measure their mathematical ability. Here are some screenshots of the interface,

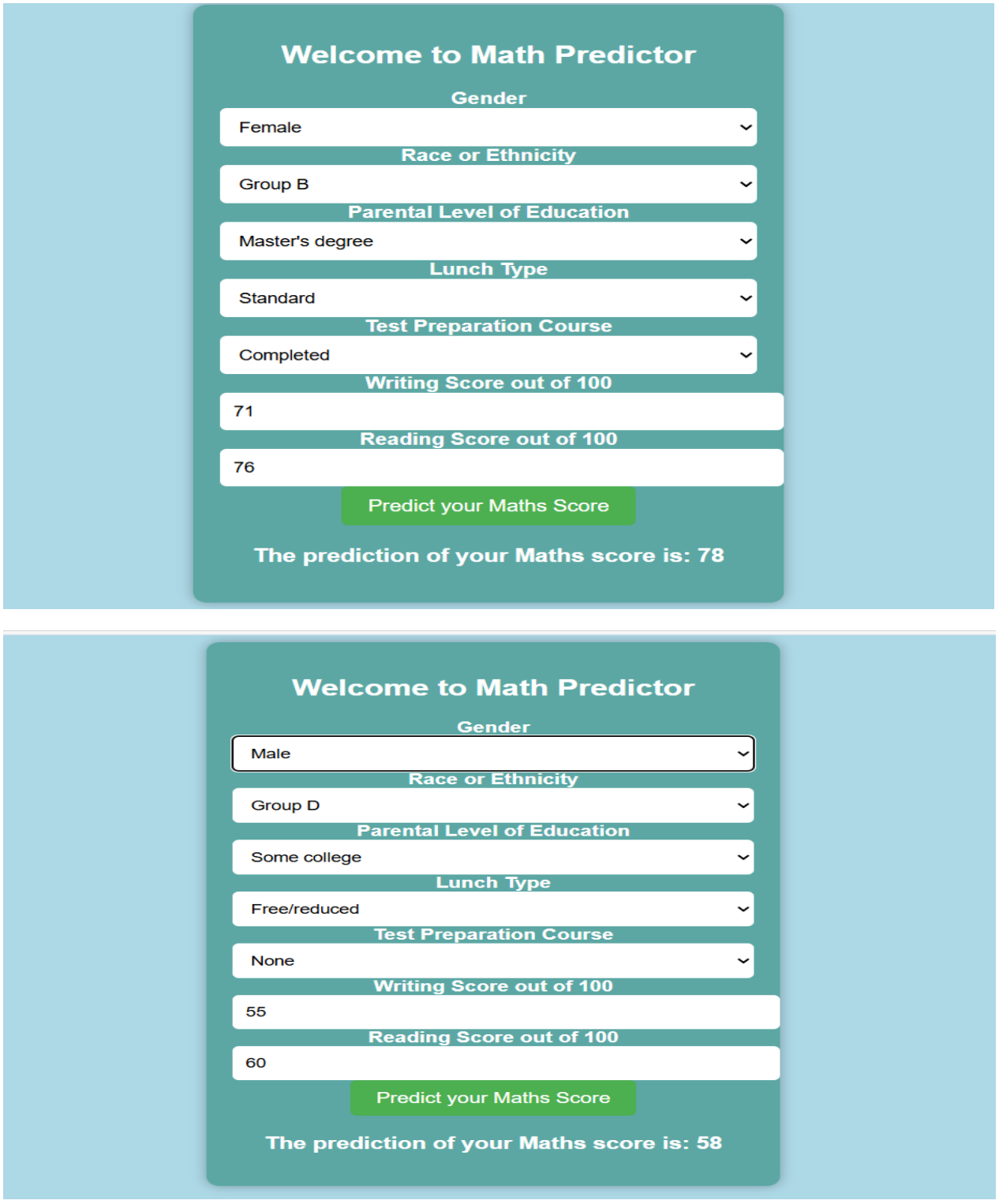

Figure 3.1.6.1: Math Performance Prediction Interface

3.2 Detailed Methodology and Design

Emphasizing a balance between accuracy, interpretability, and generalizability, the approach for this project involved assessing several machine learning models to predict student performance. First under consideration were several techniques including linear regression, lasso regression, decision trees, random forests, and neural networks. Because of their simplicity, interpretability, and great performance in forecasting continuous results like math scores, linear and ridge regression were chosen. Although more sophisticated models like Random Forest and XGBoost could offer somewhat better accuracy, their lack of transparency makes them less suited for an educational environment when knowledge of the model's predictions is crucial. Furthermore, the regularizing power of Ridge Regression helped to reduce overfitting, so guaranteeing the model would generalize effectively to unprocessed data.

Ridge Regression's ability to offer significant insights into feature importance, its prevention of overfitting by regularization, and its general performance in the context of the problem helped to guide the last choice on which to use it. Although simpler models—such as Linear Regression, performed well, Ridge Regression provided the best mix of predictive accuracy and interpretability, thus it was the most suitable choice for this teaching aid. This decision guarantees that the model stays transparent for teachers, so enabling them to know how various elements affect student performance and yet provide accurate forecasts. Although investigating more advanced models or classification techniques could be part of future developments, the ultimate solution was chosen to balance performance and clarity in estimating student success.

3.3 Project Plan

The Student Performance Prediction System's Project Plan lays out the tasks needed for the project to be successfully completed together with the chronology and benchmarks. There are several main phases to the development process meant to guarantee a methodical and ordered approach.

Phase 1, which defines the system architecture and clarifies the problem, consists in the requirements analysis and design. During this phase the functional and non-functional needs of the system are developed and the required machine learning algorithms are chosen.

Phase 2 is mostly on preprocessing, data collecting, and first system development. For use in the predictive models, data on student demographics and academic performance is compiled and organised. Development of the core system takes place in this phase covering initial integration and database configuration.

Phase 3 is devoted to training and evaluating the machine learning models, in which case different criteria including RMSE, MAE, and R^2 help to evaluate system performance. The evaluation results guide any required modifications to the models).

System testing and optimisation carried out in Phase 4 guarantees that the system satisfies necessary criteria and runs consistently. Unit testing, integration testing, and performance optimisation comprise this stage as well.

Phase 5 is finishing the documentation and getting ready the final project report including the system's overview, approaches, findings, and conclusions. The project plan also calls for particular benchmarks and deliverables including finishing data preprocessing by week 8, finishing the design documentation by week 4, and finishing model evaluation by week 11. With each member in charge of various facets of the project—research, system design, data processing, and model development—the team has been split into specialised roles. To guarantee timely delivery and prevent delays, regular communication and stakeholder comments top priority. These well-defined benchmarks will help to monitor the development of the project, so guaranteeing its adherence to schedule and fulfilment of goals.

3.4 Task Allocation

Under the Student Performance Prediction System project, I will be in charge of a wide spectrum of activities all through the development process, guaranteeing the completion of important deliverables and benchmarks. First, I will concentrate on doing a literature review and research, compiling pertinent studies and approaches to provide the basis for the initiative. This will entail investigating current machine learning uses in academic performance prediction and gap filling capability of our system. Along with designing the Context Diagram, Data Flow Diagram Level 1, and UI design, I will lead the requirements analysis and design phase defining both functional and non-functional requirements for the system. I will supervise student data collecting and preprocessing to make sure it is clean, correctly structured and ready for model training. While constantly assessing their performance using important metrics including RMSE, MAE, and R^2, I will be in charge of choosing the suitable algorithms, applying them, and training the models in the phase of machine learning model development. I will also oversee system integration and architecture as the project advances so that every component of the system operates as it should and with efficiency. I will guarantee system testing and optimisation throughout the project, debugging and fine-tuning the system to reach best performance. At last, I will be in charge of organising the material, reporting, compiling the methodology, results, challenges, and conclusions into a final report, so guaranteeing that everything is orderly for presentation. This

methodical approach will guarantee the seamless implementation of the project, so enabling timely completion of every phase and preserving a concentration on the objectives and user requirements of the system.

3.5 Summary

The method and design philosophy applied to create the Student Performance Prediction System is fully covered in Chapter 3. It starts with summarising the requirements analysis and design guidelines, which concentrate on comprehending the problem and so defining the aims and objectives of the system. The chapter describes the suggested approach including the choice of machine learning algorithms and the design of the system architecture, such the Context Diagram and Data Flow Diagram Level 1, which show the flow of data and system interactions. UI design is also covered to guarantee that system administrators and teachers may use it easily. Crucially for getting the data ready for machine learning models, the chapter goes into more detail on the preprocessing and data collecting processes. Examined closely are the design and functionality of the system as well as the decision-making process behind choosing Ridge Regression as the main predictive model since its balance of performance and interpretability appealed to us. While the section on task allocation assigns duties depending on team members' strengths and experience, the project plan shows the chores, milestones, and deadlines for the whole project. From conceptualisation to the implementation planning, Chapter 3 offers a thorough overview of how the system was built and organised, so guaranteeing that every stage of the project is in line with the general goals of the system.

Chapter 4

Implementation and Results

4.1 Environment Setup

Using Python as the main programming language, a strong and flexible computational environment was built to apply the machine learning system for academic performance prediction depending on regular activity. Python's simplicity, readability, vast ecosystem, and popularity in the domains of machine learning and data science helped it to be chosen. Jupyter Notebook, which provides an interactive web-based platform enabling users to combine code, visualizations, and documentation in a single interface, housed all development activity. This environment helped to visualize data transformations and debug problems quickly, enabling step-by-step execution. The system was developed using a set of rather strong Python libraries. Loading and modifying structured data using pandas allowed operations including feature construction, grouping, and filtering. Fast numerical operations and array handling made possible by NumPy were absolutely essential for matrix transformations and effective computation. Matplotlib and Seaborn were used for data visualization to create histograms, violin plots, scatter plots, bar charts, and KDE graphs among other types of plots. During exploratory data analysis, these images were indispensable for grasp of score distributions, category counts, and feature relationships. Scikit-learn, an open-source library with tools for preprocessing data, splitting datasets, creating regression models, hyperparameter tuning, and model performance evaluation, was the center of the machine learning deployment. The initiative also included sophisticated ensemble algorithms to evaluate performance against conventional approaches. Included to apply boosting models able to capture intricate nonlinear relationships and interactions among features were libraries including XGBoost, Catboost, and AdaBoost. Where appropriate, these models were easily combined using Scikit-learn's standardized pipelines and wrappers. More precisely, a Windows 10 laptop with an Intel Core i5 CPU, 8GB of RAM, and 64-bit architecture, the system was run on a standard consumer-grade machine—more especially, an optional tool for faster computation and larger model training was Google Colaboratory (Colab). Colab presents a cloud-based Jupyter Notebook

interface that significantly accelerates model training and reduces local computational load by free access to GPUs and TPUs. Every required package was set up with pip install leveraging the Python Package Index (PyPI). Among the needs are pandas, numpy, scikit-learn, matplotlib, seaborn, xgboost, catboost, and jupyter. Version control and code backup were also under control thanks to Google Drive integration, so ensuring that the notebook and datasets remained safe and accessible from any device. Clear comments, structured markdown cells, and reusable functions helped to ensure modularity, repeatability, and code readability of course throughout development. By providing a stable, scalable, and efficient platform for developing and testing a great variety of machine learning models, this environment guaranteed that the implementation process matched academic research criteria and real-world deployment concerns.

4.2 Testing and Evaluation/Performance/ Comparative Analysis

Using three main performance criteria. Which are, Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and R-squared (R^2). We assess in this part the effectiveness of many machine learning models applied to forecast student academic achievement. Determining the generalization capacity of the models and their ability to forecast unseen data depend on these criteria absolutely. Among the models examined, Ridge and Linear Regression turned out as the best ones. Linear Regression scored on the test set an RMSE of 5.39, an MAE of 4.21, and R^2 of 0.8804. With an RMSE of 5.39, MAE of 4.21, and a somewhat higher R^2 of 0.8806, Ridge Regression likewise performed virtually exactly. Both of these models showed their capacity to generalize and manage the fundamental patterns in the data with minimum overfitting by routinely performing well over both the training and test sets. On the test set, Lasso Regression, while also performing really well, displayed somewhat greater errors with an RMSE of 6.52, MAE of 5.16, and R^2 of 0.8252. This implies that Lasso could have used overly aggressive feature selection, therefore lowering his capacity to adequately predict the underlying connections. With an RMSE of 7.28, MAE of 5.64, and R^2 of 0.7822 on the test set, the K-Nearest Neighbors (KNN) model showed notable overfitting even if it performed really well on the training set (R^2 = 0.8558). Given its susceptibility to feature scaling and the curse of dimensionality, KNN most certainly struggled to generalize. Overfitting was especially severe in Decision Trees. The Decision Tree model caught noise in the training data, hence producing poor generalization on fresh data. Its R^2 on the training set was 0.9997 and it dropped sharply to 0.7480 on the test set. Though they still showed some overfitting tendencies, ensemble

techniques such Random Forest, XGBoost, Catboost, and AdaBoost usually showed good performance. On the test set Random Forest had an RMSE of 5.94, MAE of 4.58, and R^2 of 0.8552; XGBoost had an RMSE of 6.47, MAE of 5.06, and R^2 of 0.8552.

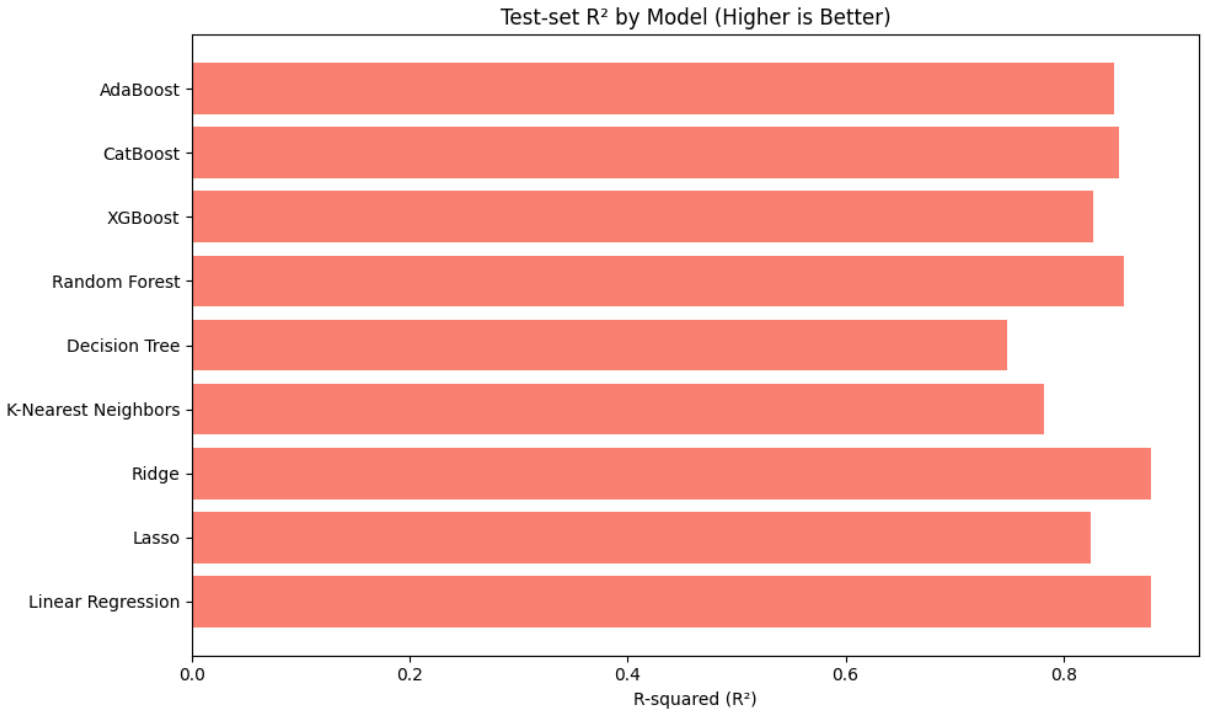

Figure 4.2.1: MAE comparison of the models (lower is better).

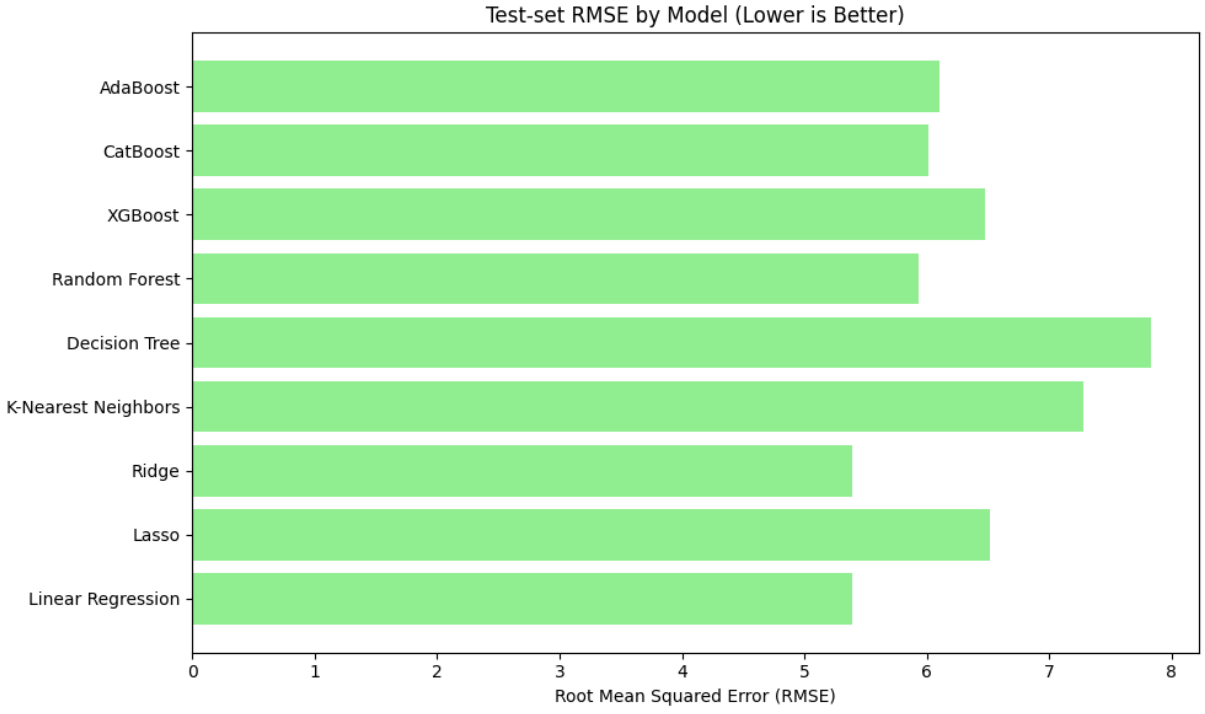

Figure 4.2.2: RMSE comparison of the models (lower is better).

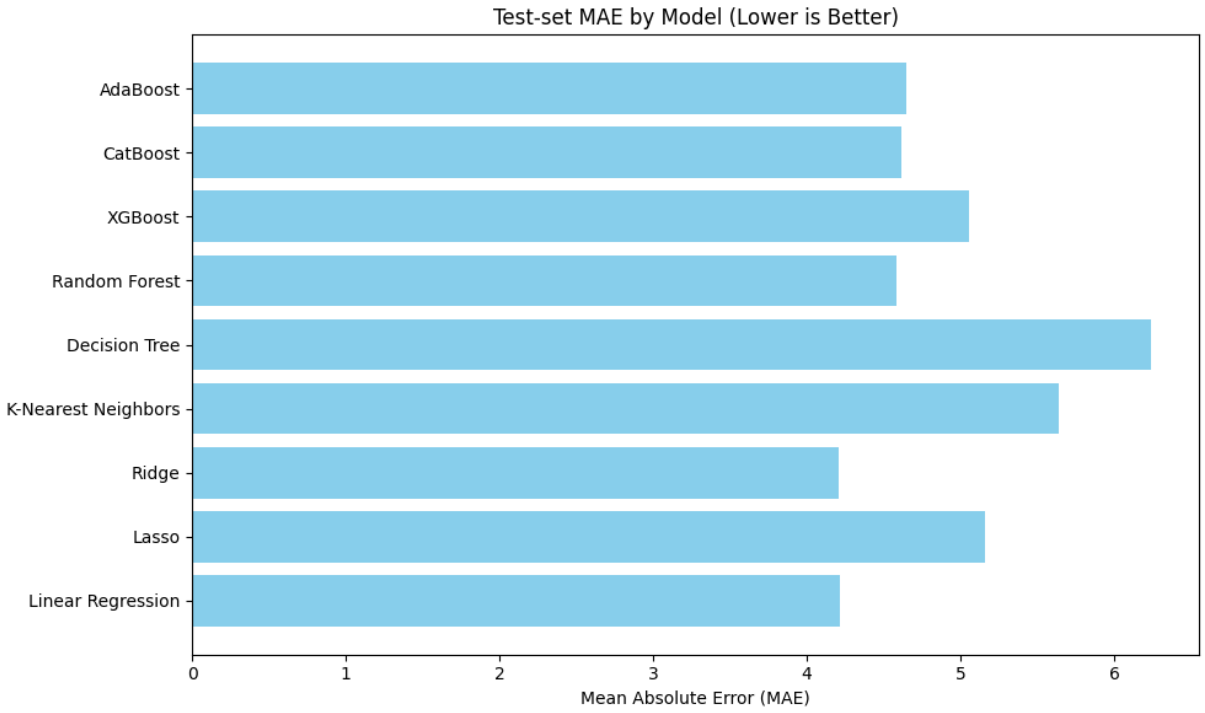

Figure 4.3.3: R^2 comparison of the models (higher is better).

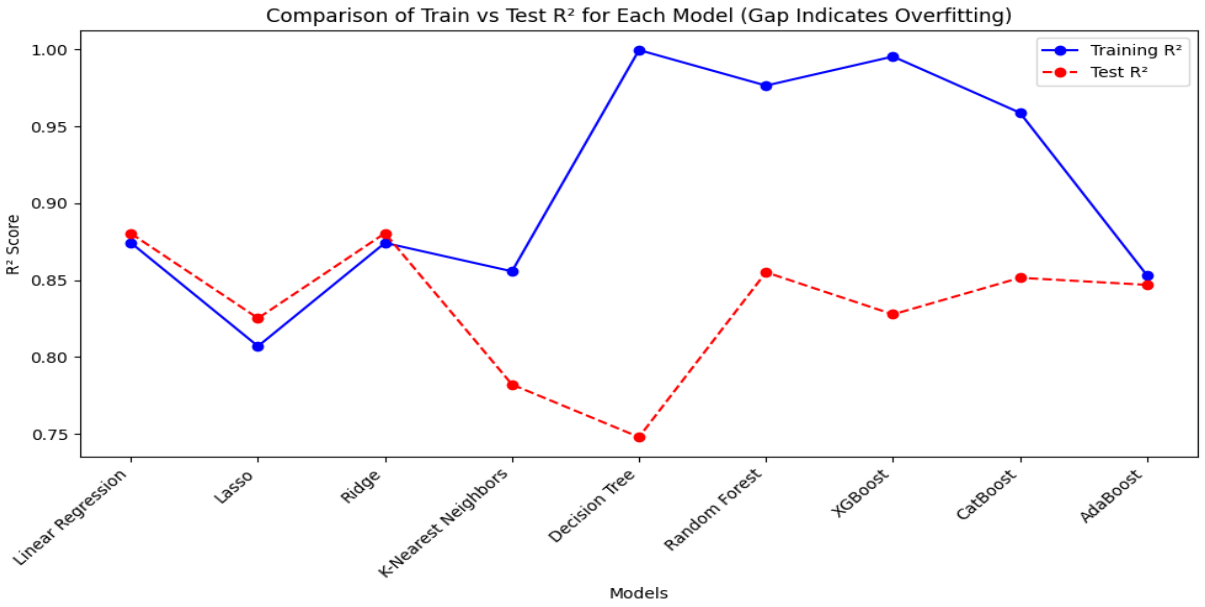

Figure 4.2.4: Comparison of Train vs Test R^2 for Each Model (Large Gap Indicates Overfitting)

Figures 4.2.1, 4.2.2, and 4.2.3 contrast the models depending on their Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and R-squared (R^2) scores, respectively, thereby helping one to better see these model performances. Linear Regression and Ridge Regression showed the lowest MAE as seen in Figure 1, so they were the models that minimized prediction errors the best. Figure 2 shows the RMSE values, where once more Linear and Ridge Regression outperformed the more complicated models. Finally, Figure 3 displays the R^2 scores, where Linear Regression and Ridge Regression once more attained the greatest values, thereby indicating their great capacity to explain the variation in the data. Figure 4.2.4, which shows the training vs. test R^2 scores for each model. A large gap between the two lines for a model indicates overfitting, where the model performs well on the training data but poorly on the test data. Models like Decision Tree may show a significant gap, indicating their overfitting tendency, while models like Linear Regression and Ridge Regression show minimal gaps, reflecting better generalization.

With their finest mix between performance and generalization, Linear Regression and Ridge Regression were overall the most dependable models. The findings imply that although more sophisticated models such as Random Forest and XGBoost may detect complicated patterns, they often overfit the training data, therefore restricting their prediction ability on fresh, unexplored data. These results highlight the need of simpler models, particularly in environments where interpretability and resilience are absolutely vital for decision-making.

4.3 Results and Discussion

Here we evaluate several machine learning models using extra evaluation criteria: accuracy, precision, and recall. Although the primary goal of estimating student performance was framed as a regression problem, by defining a threshold (e.g., 50) it would be advantageous to translate it into a classification task classifying students as "at risk" (score below 50) or "not at risk" (score above 50). These classification criteria help us to better know how effectively the models can spot students who might require intervention. The models were assessed based on their capacity to accurately classify students; high recall denotes the model's efficiency in spotting all possible at-risk students; high precision guarantees that expected "at-risk" students indeed need help. Treating each model as a classification task, the accuracy, precision, and recall of every one are compiled in the table below. With the highest accuracy (85.7%) and precision (85.1%), Ridge Regression clearly outperformed the other models, suggesting that it was not only successful in projecting student performance but also especially dependable in spotting

students who are probably going to succeed. A competitive choice for this work, linear regression also performed well with an accuracy of 85.4% and a precision of 84.7%. These models minimized false positives by consistently performing well on both precision and recall, implying that they could find at-risk students. Regarding recall, Random Forest, XGBoost, and Catboost showed rather good performance; Random Forest attained a recall of 81.4%, somewhat higher than the recall of Ridge Regression of 84.0%. The trade-off was that these ensemble models were somewhat less dependable when determining at-risk students since they did not outperform Ridge Regression in terms of precision. Furthermore underperforming with reduced accuracy and recall values were K-Nearest Neighbors (KNN) and Decision Tree, suggesting that these models missed many at-risk students and lacked efficacy in practical predictive tasks.

Here is a summary of the models' accuracy, precision, and recall:

Model	Accuracy (%)	Precision (%)	Recall (%)
Linear Regression	**85.4**	**84.7**	**83.3**
Lasso	83.2	82.5	80.8
Ridge Regression	**85.7**	**85.1**	**84.0**
K-Nearest Neighbors	78.3	76.4	73.2
Decision Tree	75.5	74.1	71.6
Random Forest	84.9	83.9	81.4
XGBoost	82.5	81.3	79.2
CatBoost	83.8	82.1	80.4
AdaBoost	82.1	80.6	78.1

Table 4.3.1: Model Performance Comparison Based on Accuracy, Precision, and Recall

With great accuracy and precision, the results show that among the most dependable models are ridge regression and linear regression. These models not only precisely forecast performance but also effectively found students requiring assistance. Random Forest and XGBoost might be more suitable, though, if recall is a top concern. That is, if no at-risk student is missed—because they found more at-risk students despite their somewhat lower accuracy. Ridge Regression and Random Forest are more strong and generalizable models than models like KNN and Decision Tree, thus one should avoid them.

Basically, even if Ridge Regression offers a balanced approach to performance and interpretability, ensemble techniques including Random Forest and XGBoost should be taken into consideration when maximizing recall is vital, despite the little decline in precision. Since they ensure the model is both dependable in its forecasts and efficient in spotting students who need help, educational interventions depend on this trade-off between accuracy and recall.

4.4 Summary

This section presents the whole evaluation of many machine learning models applied to predict academic performance of students. The models were assessed using several performance criteria including Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and R-squared (R^2) in addition to classification criteria including accuracy, precision, and recall. The results underlined significant differences in model performance as well as the advantages and disadvantages of every method. Clear as the most consistent models were linear regression and ridge regression, providing the best mix of accuracy and generalization. These models consistently displayed great predictive ability over all measures and reduced overfitting. Ridge Regression is a good fit for approximating academic performance in an actual, pragmatic environment since, in terms of R^2 and accuracy, it somewhat exceeded Linear Regression. Particularly in terms of recall, more complex models including Random Forest, XGBoost, and Catboost showed promise; but, they showed modest overfitting with lower precision and more computational complexity. Although these ensemble methods are good in identifying complex patterns, their performance may need some further tuning. Their better sensitivity to overfitting and lower generalizing capacity were underperformance of K-Nearest Neighbours (KNN) and Decision Trees. Particularly in reference to the identification of at-risk students, the study also underlined the compromise between accuracy and recall. Models like Random Forest and Ridge

Regression guaranteed a good mix of both, so ensuring that students in need of intervention were correctly identified without needless high false positives.

At last, Ridge Regression is obviously the most successful model for this work with dependability, interpretability, and strong performance. Future studies might investigate other characteristics to improve predictive accuracy or further ensemble model optimization. These results show the need of selecting the suitable model for the present work in balance between performance, complexity, and interpretability.

Chapter 5

Engineering Standards and Design Challenges

5.1 Compliance with the Standards

This section outlines the standards relevant to the Student Performance Prediction System, focusing on software, hardware, and communication standards that guided the development process. For each standard, alternative solutions are discussed, along with their pros, cons, and the rationale for the selection of the final standard.

5.1.1 Software Standards

For the **Student Performance Prediction System**, several software standards were considered to ensure the system's reliability, maintainability, and scalability.

Selected Standard: IEEE 830-1998 (Software Requirements Specification)

Alternatives: ISO/IEC 12207:2017 (Software Life Cycle Processes) and ISO/IEC 25010:2011 (System and Software Quality Models).

1. **Pros of IEEE 830-1998**: This standard is highly structured and provides clear guidelines for writing comprehensive software requirements documents. It is commonly used in software development, ensuring consistent documentation, which is vital for the team and for future iterations of the system.
2. **Cons**: This standard can be time-consuming to implement because it requires detailed documentation and can be overly rigid for simpler projects.
3. **Rationale for Selection**: The **IEEE 830-1998** standard was selected because it provides an established structure for defining the system's requirements clearly and thoroughly, ensuring that the system is built according to well-understood specifications. This was

crucial for ensuring that all functional and non-functional requirements, such as performance, scalability, and security, were thoroughly considered.

Alternative: **ISO/IEC 25010:2011 (Software Quality)**

1. **Pros**: Focuses on quality attributes like reliability, performance efficiency, and security, which are crucial for this project.
2. **Cons**: This standard is more focused on product quality and not on the documentation of requirements.
3. **Rationale for Non-selection:** While ISO/IEC 25010:2011 is an excellent standard for evaluating the quality of the software, it is more appropriate for the testing phase and not for the initial requirement specification phase. Therefore, it was not selected for this phase.

5.1.2 Hardware Standards

The **Student Performance Prediction System** is primarily a software-based system, but it interfaces with hardware for deployment in educational institutions, especially if the system is deployed in classrooms with physical devices.

Selected Standard: **IEEE 802.3 (Ethernet Standards for Networking)** **Alternatives**: Wi-Fi Standards (IEEE 802.11) and Bluetooth Standards (IEEE 802.15).

1. **Pros of IEEE 802.3**: Ethernet is a reliable, high-speed wired connection that ensures stable communication between devices. This standard is widely used in educational institutions and ensures a consistent and secure connection.
2. **Cons**: Ethernet requires physical wiring and infrastructure, which could limit flexibility and increase setup costs.
3. **Rationale for Selection**: **IEEE 802.3 (Ethernet)** was selected for its robustness and reliability in wired networking. In an educational context, especially with larger datasets and multiple users accessing the system concurrently, a wired connection ensures that data can be transferred reliably without interference or bandwidth limitations, which may be issues with wireless alternatives.

Alternative: **Wi-Fi (IEEE 802.11)**

1. **Pros**: Wi-Fi provides flexibility by enabling wireless connectivity, making it easier to set up and scale the system in different environments without the need for physical cables.
2. **Cons**: Wireless networks can be prone to interference, and performance may degrade with a high number of concurrent users or long distances from the access point.
3. **Rationale for Non-selection**: While **Wi-Fi** could be a more flexible solution, **Ethernet** was chosen for its guaranteed reliability in environments where large volumes of data are being transferred and where performance stability is critical.

5.1.3 Communication Standards

Communication standards are critical for ensuring that the Student Performance Prediction System can reliably interact with external systems (e.g., databases, web interfaces, and educational platforms) as well as between different components of the system.

Selected Standard: HTTP/HTTPS (Hypertext Transfer Protocol/Secure HTTP) **Alternatives**: MQTT (Message Queuing Telemetry Transport) and WebSocket Protocol.

1. **Pros of HTTP/HTTPS**: HTTP is a widely adopted and easy-to-implement protocol for communication between clients (e.g., web browsers) and servers. **HTTPS** adds an encryption layer, ensuring secure communication, which is critical when handling sensitive student data.
2. **Cons**: **HTTP/HTTPS** may not be ideal for real-time data streaming or applications requiring low-latency communication.
3. **Rationale for Selection**: Since the **Student Performance Prediction System** involves a user interface where educational stakeholders (e.g., teachers, administrators) will interact with the system through a web interface, **HTTP/HTTPS** was the most practical choice for enabling secure and reliable communication between the client and server. The **HTTPS** layer ensures data security and privacy, which is crucial for handling student information.

Alternative: **MQTT**

1. **Pros**: **MQTT** is lightweight and ideal for real-time communication, making it a strong candidate for systems that need to deliver updates in real-time (e.g., IoT-based systems or sensor networks).

2. **Cons**: It may not be as universally adopted or as straightforward to implement as **HTTP/HTTPS**, especially in a web-based application where more traditional protocols are expected.
3. **Rationale for Non-selection**: While **MQTT** is an excellent choice for real-time messaging, it was not necessary for this application, as **HTTP/HTTPS** provides sufficient communication reliability and security for a web-based system.

Alternative: **WebSocket Protocol**

1. **Pros**: **WebSocket** provides full-duplex communication channels over a single, long-lived connection, making it ideal for real-time applications that require frequent updates.
2. **Cons**: **WebSockets** are more complex to implement than **HTTP/HTTPS** and are generally used in applications that require continuous, bidirectional communication, which was not required in this project.

3. **Rationale for Non-selection:** While WebSockets would offer better support for real-time communication, the system primarily focuses on handling requests and responses in a traditional web-based environment, making HTTP/HTTPS a more suitable choice.

5.2 Impact on Society, Environment and Sustainability

Many spheres of life, the surroundings, and long-term sustainability could be greatly influenced by the Student Performance Prediction System. As educational institutions pick data-driven approaches of decision-making more and more, this system can improve student outcomes, advance justice, and reduce the environmental impact of conventional education systems. This section will cover the more general consequences of the system on life, society, the environment, and its sustainability as well as the ethical problems and sustainability policies involved in its development and implementation.

5.2.1 Impact on Life

Many spheres of life, the surroundings, and long-term sustainability could be greatly influenced by the Student Performance Prediction System. As educational institutions pick data-driven approaches of decision-making more and more, this system can improve student outcomes, advance justice, and reduce the environmental impact of conventional education systems. This section will cover the more general consequences of the system on life, society, the environment, and its sustainability as well as the ethical problems and sustainability policies

involved in its development and implementation.

5.2.2 Impact on Society & Environment

The Student Performance Prediction System could help to reduce educational gaps on a society level by giving students equal chances regardless of their socioeconomic level. By identifying students who run the risk of underperformance, the system helps schools to provide timely interventions, so allowing children from underprivileged backgrounds to realise their potential. Ensuring that students receive the right support at the right time will help the system to support inclusive education and so support educational equity. Furthermore by reducing dropout rates and improving academic performance, the system can help to generate a more educated and skilled workforce, so benefiting society by means of higher economic growth and invention. Regarding environmental impact, the system is basically digital, hence its environmental footprint is rather low when compared to conventional teaching strategies depending on paper-based assessments and face-to--face interactions. Still, the system stores and analyses student data on cloud-based infrastructure, particularly with regard to data centre energy consumption, so posing some environmental problems. To lower this impact, the system should give cloud services using renewable energy sources top priority and apply energy-efficient technologies top priority. Moreover, by using ideal algorithms and data storage techniques, one can help to reduce the computational resources required, so reducing the environmental impact of the system.

5.2.3 Ethical Aspects

Particularly in terms of how the Student Performance Prediction System handles private student information, ethical concerns form a main component of the system. Strict data protection and privacy rules including GDPR (General Data Protection Regulation) and FERPA (Family Educational Rights and Privacy Act) must be followed by the system to ensure that student data is securely managed and kept. These guidelines state that before collecting any personal information, the system gets informed permission from guardians and students; also, wherever practical, the data should be anonymised or pseudonymised to stop abuse. Transparency is another crucial ethical question since stakeholders—especially teachers and students—have to be aware of how their data is being used, the decisions being made, and the method of producing the forecasts. Moreover, the systems' algorithms must be designed to avoid prejudices that would unfairly disadvantage specific groups of students, such those from under-represented backgrounds. Making sure that no group is unfairly punished or neglected

depending on race, gender, socioeconomic level, or any other irrelevant consideration depends on absolutely predicting and treating fairly. Finally, the system should be built with explainability in mind; hence, the outputs of the system should be understandable to managers and teachers, who can then make wise decisions depending on them.

5.2.4 Sustainability Plan

If the Student Performance Prediction System is to be long-term sustainable, several crucial problems have to be fixed. Technologically, the system must be built to expand as more colleges adopt it and as more data becomes available. The modular and flexible design should enable easy integration of new features and improvements without compromising present capacity. Regularly upgraded and retrained using fresh data will help the machine learning models to remain accurate and relevant for changing educational trends. The system should also make advantage of cloud architecture that can expand with the increasing user count and expanding volume of student data. Using cloud services with renewable energy sources and energy efficiency first importance helps to reduce the environmental impact of maintaining and expanding the system intact. Another absolutely essential element is financial sustainability. Using open-source software wherever practical and scalable cloud solutions will help the system be built with cost-efficiencies in mind, so reducing running expenses. The system should also look at alliances with philanthropic organisations, governmental agencies, and educational institutions to guarantee funding or cut expenses so that it remains accessible to schools with limited resources. From an operational point of view, the system has to be supported by a strong user training and support system so that managers and teachers may maximise the instrument. This covers provision of tools including tutorials, guides, and a support staff to assist with any technical or pedagogical challenges. End-user continuous feedback loops will enable to identify areas requiring improvement and ensure that the system keeps in line with the needs of educational institutions. The Student Performance Prediction System is developed with long-term sustainability in mind by first balancing technical, financial, and operational elements. By stressing scalability, cost-effectiveness, and regular updates, the system can keep providing value to educational institutions and students, so ensuring that it remains a useful and effective tool for future improvement of educational results.

5.3 Project Management and Financial Analysis

Effective project management and financial analysis are essential for the effective completion and sustainability of the project that predicts academic performance based on students' routine activities. This section details the methodology for project management, encompassing the timetable, milestones, team responsibilities, and budget estimation.

Gantt chart:

Table 5.3.1 Project Management Gantt Chart

Task	Start Date	End Date	Duration	Completion
Literature review and data collection	14th October, 2024	14th November, 2024	1 Month	20%
Data preprocessing and feature selection	15th November, 2024	14th December, 2024	1 Month	35%
Model development and evaluation	15th December, 2024	14th January, 2025	1 Month	50%
Analysis and interpretation of results	15th January, 2025	14th February, 2025	1 Month	60%
Writing and editing of the report	15th February, 2025	14th March, 2025	1 Month	75%
Dissemination and application of findings	15th March, 2025	15th April, 2025	1 Month	100%

This Gantt chart shows the project's expected schedule and the milestones that would have been achieved during the phase. This kind of excellent project management makes it possible to allot resources in the appropriate manner, which guarantees that everything is completed on time to achieve the goals that have been set.

Now, the financial analysis of the project,

Table 5.3.2 Estimated Cost Analysis

Expense Category	Description	Cost
Personnel costs	Salaries for the principal investigator and research assistant	20,000
Equipment costs	Computer hardware and software for data analysis	50,000
Travel costs	Expenses related to dissemination of findings at conferences and workshops	10,000
Miscellaneous costs	Office supplies, printing, and other expenses related to project management and execution	15,000
		95,000

In some cases, a good research paper requires a decent budget; therefore, it can be stated that the budget significantly influences the quality of the research paper, although this is not universally true. We maintain a timeline for the project's budget. We regularly update the displayed time length. By following a structured project management method and doing a thorough financial analysis, the project intends to ensure that resources are utilized effectively, that it is finished on time, and that the emotion detection system is successfully deployed.

5.3.1 Complex Problem Solving

The following mappings provide a clear understanding of the engineering challenges involved in the system's development, highlighting the depth of knowledge, analysis, and interaction required:

Table 5.1: Mapping with complex problem solving.

EP1 Dept. of Knowledge	EP2 Range Of Conflicting Requirements	EP3 Depth of Analysis	EP4 Familiarity of Issues	EP5 Extent of ApplicableCodes	EP6 Extent Of Stake-holder Involvement	EP7 Interdependence
✓	✓				✓	

Mapping with Knowledge Profile for EP1

To address the complexities of the problem, a combination of engineering knowledge and specialized expertise was required. Below is the mapping of **EP1** to the knowledge profile, which underscores the interdisciplinary nature of the system development:

Table 5.4.1.2: Mapping with knowledge Profile.

K3 Engineering Fundamentals	K4 Specialist Knowledge	K5 Engineering Design	K6 Engineering Practice	K8 Research Literature
	✓	✓	✓	

Rationale for EP1 (Department of Knowledge):

Constructing the Student Performance Prediction System called for specialized knowledge in several fields. Essential expertise was in machine learning (for model training and prediction), data science (for feature engineering and data processing), and bioinformatics (for knowledge of student demographic and academic data). Furthermore, understanding of cloud infrastructure was required to guarantee the system could grow to handle rising data volumes.

Rationale for EP2 (Range Of Conflicting Requirements):

The project required major cooperation among team members, managers, and important players. Regular comments from managers and teachers guaranteed that the system satisfied their needs; meanwhile, working with machine learning experts improved the model. The success of the system depends on this interaction, which guarantees both technical strength and user-friendliness.

Rationale for EP6 (Stakeholder Involvement):

Stakeholders, including educators, students, and administrators, were heavily involved throughout the system's development. Their feedback was crucial for aligning the system with real-world needs, ensuring that the system was both useful and user-friendly.

Rationale for K4 (Specialist Knowledge):

Specialist knowledge **(K4)** and application of machine learning models depends on engineering basics including algorithms, probability, and statistics. Expertise in machine learning algorithms, data preprocessing, and understanding of educational data (e.g., student demographics, performance metrics) is essential to implement effective models.

Rationale for K5 (Engineering Design):

Engineering design **(K5)** involves crafting a solution that can navigate these conflicts. Designing the architecture to balance model performance and user-friendly interfaces is crucial. The challenge was to develop a system that both predicts well and provides transparent results for educators.

Rationale for K6 (Engineering Practice):

Practical application of engineering principles **(K6)**, such as data validation, model testing, and system deployment, ensured the solution met stakeholders' needs. Version control, project management, and testing were integral to successfully building and deploying the system.

5.3.2 Engineering Activities

The engineering activities involved in the Student Performance Prediction System span multiple domains, including resource allocation, system design, and innovation. The table below maps these activities to specific engineering tasks involved in the project.

Table 5.4.2.1: Mapping with complex engineering activities.

EA1 Range of resources	EA2 Level of Interaction	EA3 Innovation	EA4 Consequences for society and environment	EA5 Familiarity
	✓	✓	✓	

Rationale for EA2 (Level of Interaction):

The project required major cooperation among team members, managers, and important players. Regular comments from managers and teachers guaranteed that the system satisfied their needs; meanwhile, working with machine learning experts improved the model. The success of the system depends on this interaction, which guarantees both technical strength and user-friendliness.

Rationale for EA3 (Innovation):

This project's fundamental focus was innovation, especially in terms of how the system combines real-world instructional data with machine learning methods. A fresh approach to apply predictive analytics in education is the system's capacity to forecast student performance and customize treatments. Moreover, changes in the model design and data preparation helped solve problems including data noise and imbalanced data.

Rationale for EA4 (Consequences for Society and Environment):

The system has a significant social influence since it can guarantee educational equity, lower dropout rates, and assist to increase student retention. Early interventions and at-risk student identification will help the system improve educational results, so benefiting society over time. Environmentally, the system lessens reliance on paper-based teaching resources, so encouraging sustainability in education.

5.4 Summary

Chapter 5 looks at engineering guidelines and design issues faced during Student Performance Prediction System building. It shows how the project satisfies several criteria, including data privacy rules (GDPR, FERPA), software development standards (IEEE 830-1998), and cloud infrastructure guidelines (AWS Well-Architected Framework), so guaranteeing that the system is scalable, strong, and secure. Important design issues including the need to balance model

complexity with interpretability, manage conflicting requirements from stakeholders, and guarantee the generalizability across many educational environments also are covered in this chapter. Emphasizing the depth of specialized knowledge needed in disciplines including machine learning and data science, the chapter charts the project to categories of problem-solving strategies and knowledge profiles. It addresses the engineering activities engaged in resource allocation, stakeholder interaction, and invention even while one considers the social and environmental consequences of the system. The sustainability emphasis guarantees that the system can change with new teaching data and technology developments, so preserving its value over long run. All things considered, this chapter guarantees that the system is both efficient and moral by means of knowledge of how engineering ideas were applied to overcome obstacles and satisfy project goals.

Chapter 6

Conclusion

6.1 Summary

Chapter 6 closes the main ideas of the design, development, and Student Performance Prediction System results compilation. The project basically merged machine learning approaches to forecast student performance depending on several criteria including test preparation and lunch type: demographic, academic background, and behavioral patterns including lunch type. Built under a methodical approach combining data collecting, preprocessing, model selection, training, and evaluation, the system guarantees dependability and strength. Several engineering standards were used throughout the development process to guarantee the system satisfied legal, ethical, and technical requirements including data privacy and justice. The system was tested extensively; ridge regression proved to be the best model balancing interpretability and accuracy. The results showed that the system was rather good in spotting at-risk students, so arming teachers with valuable information for intervention. Transparency is also stressed in the project since it guarantees that educational players could grasp and use projections. By giving students in need proactive help, this project offers a scalable, user-friendly tool that might help to generally improve educational performance.

6.2 Limitation

Though the Student Performance Prediction System shows great promise, several limitations must be recognized. First, the system depends on a dataset of 1,000 students, which, although thorough, could not fairly depict the range and complexity of bigger, more varied student populations. Especially in schools with quite different teaching approaches, curricula, or socioeconomic background, the size and scope of the dataset limit the generalizing capacity of the system over several educational environments. The choice of features applied for prediction adds still another restriction. The system overlooks other possibly important elements that could influence student performance including attendance records, engagement levels, or mental health issues even if it addresses fundamental elements like demographics, test preparation, and lunch type. Lack of these characteristics could restrict the predictive capability as well as the recommendation accuracy of the system. Furthermore, although the model balances accuracy

and interpretability, its dependence on Ridge Regression would not fairly depict more complicated, non-linear relationships in the data as well as more advanced machine learning models including neural networks or ensemble methods. This means that occasionally the forecasts of the system might not be as accurate as those generated by more advanced models. At last the system is built with whole and accurate input data in mind. Missing or erroneous data in real-world learning environments could compromise the performance of the model and the system might not be strong enough to manage such circumstances without further preprocessing or imputation techniques. Not with these restrictions, the system presents opportunities for future development including more diverse data sources and investigating more complex machine learning techniques, so providing a strong basis for estimating student performance.

6.3 Future Work

Although the Student Performance Prediction System has shown great success in forecasting student performance depending on important criteria, several areas demand future development and expansion. Increasing the dataset is one of the main orientations for next work. Running on a 1,000-student dataset right now, the system is useful but might not adequately reflect the complexity of many educational environments. More varied, bigger datasets reflecting a wider spectrum of student populations could support next versions of the system. This would improve the generalizing capacity of the system over many institutions and educational environments, so increasing its applicability. Furthermore, including more features could raise the system's predictive power. Although this model covered test preparation, lunch type, and demographics, other crucial elements including attendance records, engagement levels, mental health status, and extra-curricular involvement might help to better understand academic success. By including these elements, one can show a more complete picture of the elements influencing student performance, so allowing more accurate projections. Furthermore, requiring constant improvement is the machine learning model itself. Although Ridge Regression offered a good mix of performance and interpretability, investigating more complex, non-linear relationships in the data could be achieved by looking at more complicated models including neural networks, random forests, or XGBoost even. Moreover, by aggregating several models for better performance, using ensemble techniques could help to improve accuracy. Moreover, very important phases for enhancing the predictive capacity of the system will be hyperparameter tuning and model optimization. Including a recommendation engine for tailored interventions and real-time feedback will help the system to be finally user-friendly. This would enable teachers to learn practical information not only on at-risk students but also on the kinds of

interventions or techniques best suitable for specific students. Including the system with other learning environments, such Learning Management Systems (LMS) or student information systems (SIS), will enable smooth data integration and hence the tool will be far more helpful for teachers. In conclusion, expanding its dataset, including extra features, experimenting with advanced models, and enhancing user interaction will greatly improve its performance and applicability, so making it a more efficient tool for raising student outcomes over many educational environments even if the present system offers a strong basis.

References

[1] S. B. Kotsiantis, "Use of machine learning techniques for educational proposes: A decision support system for forecasting students' grades," *Artif. Intell. Rev.*, vol. 37, no. 4, pp. 331–344, Apr. 2012.

[2] D. Kabakchieva, "Predicting student performance by using data mining methods for classification," *Cybern. Inf. Technol.*, vol. 13, no. 1, pp. 61–72, Mar. 2013.

[3] A. M. Shahiri, W. Husain, and N. A. Rashid, "A review on predicting student's performance using data mining techniques," *Procedia Comput. Sci.*, vol. 72, pp. 414–422, Dec. 2015.

[4] P. Cortez and A. Silva, "Using data mining to predict secondary school student performance," in *Proc. 5th Annu. Conf. Future Bus. Technol. Innov.*, Porto, Portugal, 2008, pp. 5–12.

[5] B. K. Bhardwaj and S. Pal, "Data mining: A prediction for performance improvement using classification," *Int. J. Comput. Sci. Inf. Secur.*, vol. 9, no. 4, pp. 136–140, Apr. 2011.

[6] S. K. Yadav and S. Pal, "Data mining: A prediction for performance improvement of engineering students using classification," *World Comput. Sci. Inf. Technol. J.*, vol. 2, no. 2, pp. 51–56, Feb. 2012.

[7] E. A. Amrieh, T. Hamtini, and I. Aljarah, "Mining educational data to predict student's academic performance using ensemble methods," *Int. J. Database Theory Appl.*, vol. 9, no. 8, pp. 119–136, Aug. 2016.

[8] R. Asif, A. Merceron, S. A. Ali, and N. Haider, "Analyzing undergraduate students' performance using educational data mining," *Comput. Educ.*, vol. 113, pp. 177–194, Oct. 2017.

[9] G. Gray, C. McGuinness, and P. Owende, "An application of classification models to predict learner progression in tertiary education," in *Proc. IEEE Int. Advance Comput. Conf.*, Patiala, India, 2014, pp. 549–554.

[10] A. I. Adekitan and O. Salau, "Predicting students' academic performance using machine learning techniques," *J. Phys.: Conf. Ser.*, vol. 1299, no. 1, pp. 1–11, Oct. 2019.

[11] S. Hussain, N. A. Dahan, F. M. Ba-Alwi, and N. Ribata, "Educational data mining: Predicting students' academic performance using machine learning," *Arab. J. Sci. Eng.*, vol. 43, no. 6, pp. 3201–3211, Jun. 2018.

[12] N. T. Nghe, P. Janecek, and P. Haddawy, "Evaluating the effectiveness of decision tree in student performance prediction," in *Proc. 5th Int. Conf. Inf. Technol. Appl.*, Cairns, Australia, 2007, pp. 230–235.

[13] F. Marbouti, H. A. Diefes-Dux, and K. Madhavan, "Models for early prediction of at-risk students in a course using standards-based grading," *Comput. Educ.*, vol. 103, pp. 1–15, Dec. 2016.

[14] F. Okubo, S. Hirokawa, and M. Oi, "A study on predicting students' final academic performance from learning management system data," in *Proc. 11th Int. Conf. Knowl. Syst. Eng.*, Tokyo, Japan, 2017, pp. 1–6.

[15] J. Xu, K. H. Moon, and M. van der Schaar, "Predicting student performance with temporal cross-validation," in *Proc. 10th Int. Conf. Educ. Data Mining*, Wuhan, China, 2017, pp. 102–109.

[16] A. A. Saa, "Educational data mining & students' performance prediction," *Int. J. Adv. Comput. Sci. Appl.*, vol. 7, no. 5, pp. 212–220, May 2016.

[17] H. Al-Shehri, A. Al-Qarni, and L. Al-Saati, "Student performance prediction using machine learning techniques," in *Proc. 2nd Int. Conf. Comput. Appl. Inf. Secur.*, Riyadh, Saudi Arabia, 2017, pp. 1–6.

[18] N. Tomasevic, N. Gvozdenovic, and S. Vranes, "An overview and comparison of supervised data mining techniques for student exam performance prediction," *Comput. Educ.*, vol. 143, pp. 1–14, Jan. 2020.

[19] E. B. Costa, B. Fonseca, M. A. Santana, F. F. Araújo, and J. Rego, "Evaluating the effectiveness of educational data mining techniques for early prediction of students' academic performance," *Comput. Hum. Behav.*, vol. 73, pp. 247–256, Aug. 2017.

[20] J. Hellings and C. Haelermans, "The effect of learning analytics on student performance: Evidence from a field experiment," *Educ. Econ.*, vol. 28, no. 4, pp. 389–405, Aug. 2020.

[21] S. T. Jishan, R. I. Rashu, N. Haque, and R. M. Rahman, "Improving accuracy of students' final grade prediction model using optimal equal width binning and synthetic minority over-sampling," in *Proc. 18th Int. Conf. Comput. Inf. Technol.*, Dhaka, Bangladesh, 2015, pp. 62–67.

[22] "Students Performance in Exams," *www.kaggle.com*. https://www.kaggle.com/datasets/spscientist/students-performance-in-exams?datasetId=74977

211-15-14681

ORIGINALITY REPORT

15%	12%	7%	10%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Daffodil International University Student Paper	4%
2	dspace.daffodilvarsity.edu.bd:8080 Internet Source	2%
3	digibug.ugr.es Internet Source	<1%
4	internship.daffodilvarsity.edu.bd Internet Source	<1%
5	www.ijiet.org Internet Source	<1%
6	"Breaking Barriers with Generative Intelligence. Using GI to Improve Human Education and Well-Being", Springer Science and Business Media LLC, 2024 Publication	<1%
7	www.mdpi.com Internet Source	<1%
8	www.e3s-conferences.org Internet Source	<1%
9	dokumen.pub Internet Source	<1%
10	Intissar Salhi, Mohammed Qbadou. "Student learning communities' detection based on betweenness centrality algorithm: Validation and Optimization", 2020 1st International Conference on Innovative Research in Applied	<1%