جامعـــة خليفـــة
Khalifa University

MATH 318 Statistical Learning

Midterm Project

Predicting Letter Grades of Khalifa University Students Taking Calculus 1.

Students names and ID’s:

Dima Al Dakkak - 100061205

Joel Joseph Jaison -100061270

Mohammad Shobaki - 100059723

Instructor:

Idika E. Okorie

Abstract

In this research paper, we focus on predicting letter grades for Calculus 1 students at Khalifa University, utilizing predictive analytics. Khalifa University's commitment to academic excellence and innovative teaching methods sets the backdrop for our study. With a survey conducted in 2023 and 317 responses , we utilize Linear Discriminant Analysis, Support Vector Machines, Naive Bayes, Multinomial Logistics Regression, and Random Forest using R software. We additionally mention about the few methods we had conducted to conduct a very defined analysis on our dataset.

Results highlight that Multinomial Logistics Regression, Random Forest, and Linear Discriminant Analysis show the highest accuracy at 47.17%. After a thorough evaluation using the other metrics ,we see that no single model emerged decisively superior. Therefore, we leave the choice of the most suitable model to the discretion of the reader, considering their specific analytical needs and preferences. Our study recognizes the need for a broader feature selection and takes in account of possible data biases that could occur. Despite these limitations, the research offers valuable insights into predicting student grades in Calculus 1 at Khalifa University.

1. Introduction

With higher education constantly evolving, institutions strive to enhance their academic support services and educational processes with the primary goal of ensuring student success. Assessment and prediction of academic performance is an essential part of this endeavor, which can provide valuable insights into areas that need more attention and resources. In this research paper, we examine the detailed process of predicting letter grades for Calculus 1 students at Khalifa University, illuminating an area of educational analytics that can help us provide support and implement effective academic interventions.

Nestled in international academic acclaim, Khalifa University stands as a distinguished institution in the United Arab Emirates. Renowned for its commitment to academic excellence and innovative teaching methodologies, the university draws students from diverse backgrounds, fostering a vibrant academic community. Internationally top-ranked, it uniquely offers research and academic programs tailored to address the strategic, scientific, and industrial challenges pivotal to the UAE's knowledge economy transformation.

Calculus 1 is a foundational course in mathematics, it covers various topics that are crucial for colleges of sciences and engineering. However, the challenge lies in ensuring that all students irrespective of their prior mathematical background and major, receive the necessary support and guidance to excel in this critical subject.

With predictive analysis, it is a process that involves extracting insights from data to anticipate future trends and behaviors. In essence, it leverages various techniques, such as statistical modeling and machine learning algorithms, to identify patterns within datasets that helps identify where students might struggle, and professors can then swoop in with targeted support. Therefore, in this research, our main aim is to predict the student's grade letter based on many variables. Further, our goal is to predict letter grades and understand which predictor would help and guide students on how to approach Calculus 1 to ensure they get the best outcome possible. By exploring factors like gender, study habits, previous knowledge, etc. we hope to provide practical advice to both professors and students.

The topics that will be covered in this research start with the literature review where we discuss other research linked or related to our topic. After that, we go deeply into the methodology including data collection, material used, and the methodological procedure used in this study. Moreover, in results and discussions, we discuss all the models we used to predict the grades and explain the theory behind the result and the numbers such as the Accuracy, P-Value, Kappa, etc.. We then conclude by summarizing all the key points in our research.

1.2 Literature review

In this literature review, we examine some research articles that shed light on the use of machine learning to predict students' academic performance, with a particular focus on how these insights can relate to our research on predicting students' grades in Calculus 1.

The intersection of data mining and education, commonly referred to as Educational Data Mining (EDM), has witnessed a surge in interest in recent years, reflecting a growing recognition of its potential to unravel hidden patterns within educational datasets (Yağcı, 2022). Data Mining (DM), defined as the exploration of data for the discovery of new and valuable information, has found applications in various domains, with the field of education becoming a focal point for leveraging its capabilities (Yağcı, 2022). EDM, an extension of traditional DM methods tailored for educational contexts, utilizes classification algorithms to extract meaningful insights from diverse educational data sources, including student information, academic records, exam results, class participation, and attendance frequency (Yağcı, 2022). One prominent application of EDM is predicting students' academic performance, a task that has gained particular significance in higher education. The study by Yağcı (2022) contributes to this evolving landscape by proposing a novel model based on machine learning algorithms for forecasting the final exam grades of undergraduate students. Grounded in a dataset comprising academic achievement grades of students enrolled in a Turkish Language-I course, the model hinges on key parameters such as midterm exam grades, department data, and faculty data. The evaluation of machine learning algorithms, including Random Forests, Nearest Neighbor, Support Vector Machines, Logistic Regression, Naive Bayes, and K-Nearest Neighbor, showcases the model's ability to achieve a classification accuracy ranging from 70% to 75%. The significance of this study lies in its practical implications for higher education institutions. By demonstrating the efficacy of a data-driven approach in predicting academic performance, particularly in identifying students at high risk of failure, the research contributes to the establishment of learning analysis frameworks. Such frameworks are crucial for informing decision-making processes and enhancing educational outcomes (Yağcı, 2022). Moreover, the focus on a specific language course within the Turkish higher education context adds a nuanced perspective to the broader discussions on EDM applications. In conclusion, the literature reviewed here underscores the increasing importance of EDM in education and the potential of machine learning algorithms to predict students' academic performance. The study by Yağcı (2022) exemplifies this trend, emphasizing the practical implications of such models for learning analysis frameworks and early identification of at-risk students. As the field continues to evolve, further research in this domain promises to enhance our understanding of the intricate relationships within educational data, ultimately contributing to more informed decision-making in higher education contexts.

Educational data mining (EDM) has emerged as a valuable tool for extracting meaningful insights from educational datasets, aiding institutions in enhancing student outcomes. Al-Barrak and Al-Razgan (2016) contribute to this growing field through their case study titled "Predicting Students' Final GPA Using Decision Trees: A Case Study." This study explores the application of the J48 decision tree algorithm to predict students' final GPA based on their academic performance in previous courses. The use of decision trees in educational contexts has gained attention due to their ability to reveal classification rules in a comprehensible, tree-like structure. Decision trees provide a transparent framework for understanding the relationships between various academic factors and the ultimate academic achievement, such as final GPA. Al-Barrak and Al-Razgan's choice of the J48 algorithm aligns with the need for interpretable models in educational data mining. The research methodology involved collecting students' transcript data, encompassing final GPA and grades in all courses. Through pre-processing steps, the data was prepared for analysis, emphasizing the importance of data quality and integrity in the EDM process. The application of the J48 decision tree algorithm aimed to uncover classification rules that could contribute to predicting students' final GPA accurately. One notable aspect of the study is its focus on extracting valuable knowledge for predicting final GPA and identifying pivotal courses in students' study plans. This emphasis on specific courses aligns with the broader educational goal of optimizing study plans to improve overall academic performance. The findings of the study, including the identification of the most important courses, offer practical insights that can inform educational strategies and interventions. In conclusion, Al-Barrak and Al-Razgan's case study represents a significant contribution to the educational data mining literature. By applying the J48 decision tree algorithm to predict students' final GPA and identifying influential courses, the study provides a nuanced understanding of the factors shaping academic success. This research underscores the potential of data mining techniques in enhancing educational practices and decision-making processes, laying the groundwork for future studies in the dynamic intersection of education and data science.

Predictive analytics in the realm of higher education has gained considerable significance, with a focus on utilizing advanced analytics and machine learning techniques to extract meaningful insights for academic performance monitoring. One key performance indicator that has garnered attention is student grades, reflecting academic achievements. Over the past decade, researchers have proposed various machine learning techniques for predicting student grades, yet a critical research gap exists in identifying effective predictive models, particularly in the context of imbalanced multi-class classification. In addressing this gap, Bujang et al. (2021) present a comprehensive analysis of machine learning techniques for predicting final student grades in first-semester courses. Their study compares the accuracy performance of five well-known machine learning techniques: Decision Tree (J48), Support Vector Machine (SVM), Naive Bayes (NB), K-Nearest Neighbor (KNN), and Logistic Regression (LR). The objective is to provide a nuanced understanding of the effectiveness of these techniques using real-world datasets. The research unfolds in two modules. Firstly, the accuracy performance of the aforementioned machine learning techniques is rigorously evaluated. Secondly, a novel multi-class prediction model is proposed for addressing imbalanced multiclass datasets. The model integrates the Synthetic Minority Oversampling Technique (SMOTE) and wrapper-based Feature Selection (FS), demonstrating significant improvement in prediction accuracy. The results of the study are particularly noteworthy. The proposed model, incorporating SMOTE and wrapper-based Feature Selection with KNN, achieves an impressive accuracy of 99.6%. Furthermore, when applied independently with SVM, all feature selection algorithms demonstrate robust performance with an accuracy of 99.4%. These findings underscore the efficacy of the proposed model in discovering the optimal combination of feature selection algorithms and SMOTE for tackling imbalanced multi-classification challenges in student grade prediction. In conclusion, Bujang et al. (2021) research makes a substantial contribution to the field by not only evaluating the accuracy of machine learning techniques in predicting student grades but also by offering a practical and effective multi-class prediction model. Their work provides valuable insights for researchers and educators seeking to enhance predictive analytics in higher education.

Educational data mining has emerged as a vital field, with a particular emphasis on course performance prediction to enhance personalized teaching and alleviate student stress (Chen et al., 2023). Researchers have explored various predictive features and algorithms to develop accurate models for anticipating students' final performance in basic university courses. Chen et al. (2023) contribute to this growing body of knowledge by focusing on the prediction of basic course performance in universities. Their study incorporates a combination of objective and subjective features for enhanced predictive accuracy. Objective features such as GPA, grades of prerequisite courses, assignment scores, and inquiry count are considered alongside the subjective feature of individual interest. In their exploration of classification algorithms, the researchers employ Gaussian Support Vector Machine, Polynomial Support Vector Machine, BP Neural Network, Random Forest, and Logistic Regression. The comparative analysis reveals that the Gaussian Support Vector Machine, when coupled with the selected features, achieves optimal accuracy and Average Precision (AP), reaching an impressive 99%. This study adds to the existing literature by proposing a novel course performance prediction model for basic university courses. The emphasis on the Gaussian Support Vector Machine as a key classifier, combined with a thoughtful selection of features, presents a promising avenue for future research in the realm of educational data mining. As educators and institutions seek innovative methods to improve teaching strategies and reduce student pressure, the findings of this study offer valuable insights and contribute to the ongoing discourse on effective approaches to course performance prediction.

In addressing the pervasive challenge of student retention in higher education, scholars have explored various methodologies to enhance early and continuous feedback, a critical factor in improving educational outcomes. One notable contribution to this discourse is the work by Nabizadeh et al. (2022), who propose a method for predicting students' final grades within the context of a gamified course. The authors highlight the significance of early feedback and its potential impact on student retention. They argue that a method focusing on utilizing performance data exclusively from the current semester can offer valuable insights into predicting final grades. Central to their approach is the clustering of students based on experience points (XP), a metric commonly associated with gamified courses. This initial clustering aims to provide a foundation for further analysis. To enhance the robustness of their model, Nabizadeh et al. (2022) introduce a cluster size adjustment mechanism. This involves estimating cluster sizes and strategically introducing virtual students to smaller clusters, thereby promoting a more balanced representation. Additionally, the authors employ a feature selection technique to streamline the predictive model, discarding unimportant student attributes. The study conducted by Nabizadeh et al. (2022) relies on extensive data collected over nine years from a diverse pool of 679 students enrolled in the gamified course. The comparative analysis of their proposed method against alternative approaches reveals its superior performance, achieving an average accuracy of 78.02% only four weeks into the course. These findings underscore the efficacy of the proposed model in predicting final grades, showcasing its potential to significantly impact students' learning outcomes. In summary, the research by Nabizadeh et al. (2022) contributes valuable insights into the realm of student retention and academic performance prediction. The integration of gamification elements, coupled with innovative clustering and predictive modeling techniques, positions their method as a promising avenue for educators and institutions seeking to proactively address challenges associated with student success.

In conclusion, the reviewed literature underscores the transformative potential of educational data mining (EDM) and machine learning in predicting students' academic performance, laying a robust foundation for our specific focus on creating a predictive model for Calculus 1 grades. The insights gained from studies by Yağcı (2022), Al-Barrak and Al-Razgan (2016), Bujang et al. (2021), Chen et al. (2023), and Nabizadeh et al. (2022) collectively emphasize the efficacy of diverse machine learning techniques in forecasting academic outcomes. By leveraging algorithms, decision trees, and innovative approaches like gamification and clustering, these studies contribute valuable lessons to our pursuit of developing a tailored model for predicting the performance of Calculus 1 students. The practical implications, ranging from early identification of at-risk students to optimizing study plans, offer a roadmap for implementing our predictive analytics framework. As we delve into the realm of EDM to enhance educational outcomes, the synthesis of these studies guides us in harnessing the power of machine learning to address the unique challenges and dynamics of predicting success in Calculus 1, ultimately aiming to empower educators and institutions with proactive tools for supporting student achievement.

2. Methodology

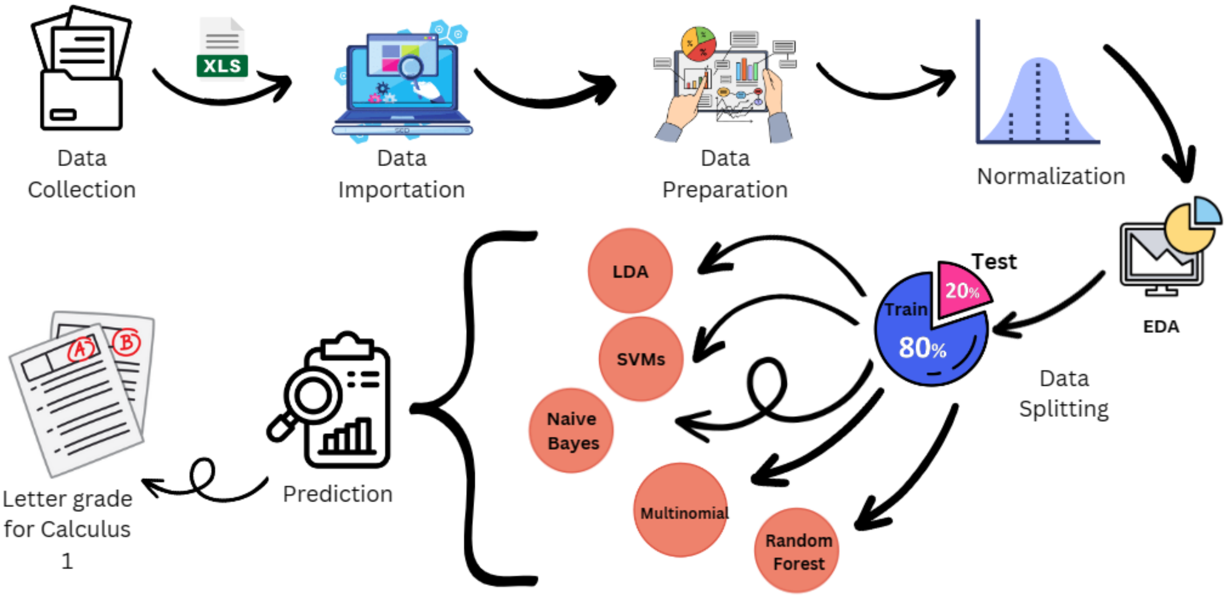


Figure 1. *Methodology Mindmap*

This section sheds light on the methodological procedure used in this study. First, we start by providing a brief description of how the data was prepared (2.1). Then, we will mention the materials used (2.2) and the procedures taken to conduct this analysis (2.3).

2.1 Data Collection

The data gathered for the study was obtained through a survey conducted in 2023 at Khalifa University. The survey questionnaire was designed using Google Forms and contains 12 questions on demographics and study methodologies (see Appendix A for the full questionnaire). The questionnaire garnered a total of 317 anonymous responses, which were later exported into Microsoft Excel for analysis. We have made all of the questionnaire's questions essential to respond in order to prevent the problem of missing data; therefore, only completed questionnaires were accepted. To ensure the questionnaire was collected consensually, we followed research ethics before publishing and sharing it.

2.2 Materials Used

We conducted the analysis using R software. It is a well-known statistical software developed by Ross Ihaka and Robert Gentleman, two statisticians from the University of Auckland, New Zealand. The software is universal and provides an extensible platform for statistical analysis and data visualization (Chambers, 2008).

2.3 Procedures

2.3.1 Data Preparation

In our research paper, we initiated the data analysis process by loading all necessary libraries and importing the dataset into the R environment (Chambers, 2008). We utilized the select tool to eliminate two columns from our dataset: "Did you take the Calculus 1 course in KU" and the Timestamp. These columns were deemed redundant due to several reasons. Firstly, the timestamp did not serve any value in our research goal. Secondly, the "Did you take the Calculus 1 course in KU" column was initially cleaned from the "No" variable as the rows were empty. This left us with only the "Yes" rows, making the column insignificant for our analysis.

Following the data-cleaning phase, we calculated the average study hours for each observation. This was done by taking the first and last values of the study hour range and dividing the result by the total number within that range. Afterward, we standardized all columns, converting character variables into factor representations and numerical variables into numeric formats. For improved clarity and convenience, we renamed the columns from A to L as shown in Table 1.

Table 1: *Detailed Column Attributes*

No.	Names of Attribute	Details of Attributes
1	A	Level
2	B	Age
3	C	Gender
4	D	What was your grade in Calculus 1
5	E	Did you take any prior calculus or math-related coursework in High School before enrolling in Calculus 1?
6	F	How many hours per week did you typically spend studying for Calculus 1?
7	G	Did you form study groups with classmates for Calculus 1?
8	H	How frequently did you use online resources to supplement your Calculus 1 studies?
9	I	Do you ever get anxious when doing the test(including quizzes,midterms and finals)?
10	J	Did you use to solve all recommended problems?
11	K	Did you follow up with the material or do you usually study for the test last

		minute?
12	L	Did you usually attend office hour?

During the analysis, we filtered out unusual responses to balance out the data. Subsequently, we noticed that the grades C-, D, and F were rarely given. Since it is generally understood that grades below a C are considered low, we removed them from the list of letter grades.

After completing the data preparation steps mentioned earlier, we proceeded with data visualization, followed by model development. At the model development stage, we divided our dataset into two parts: training and testing sets. The training set, which comprised 80% of the data, was used to train our model. On the other hand, the remaining 20% was for the testing set, which we used for predictions.

2.3.2 Data Visualization

After analyzing the distribution of grade letters in our data, it can be observed from Figure 2 that A and B have the highest percentage of occurrences with 22% and 20.8%, respectively. Following these, we have B+ with 16.7%, A- with 13.6%, C with 12.5%, and finally, C+ and B- with the lowest two percentages, 8% and 6.4%, respectively.

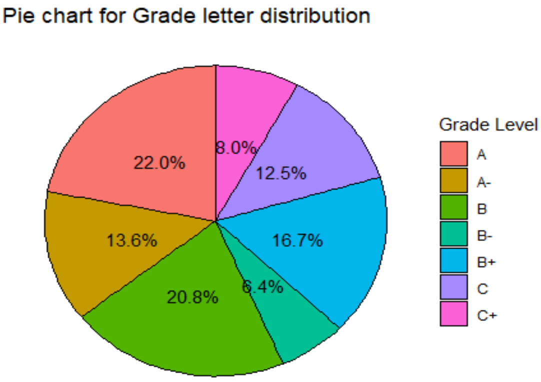


Figure 2. *Grade Letter Distribution*

After examining the level distribution of our data, we can observe from Figure 3 that juniors account for the highest percentage at 33.7%, followed by sophomores at 30.3%, freshmen at 18.6%, and seniors at the lowest percentage at 17.4%.

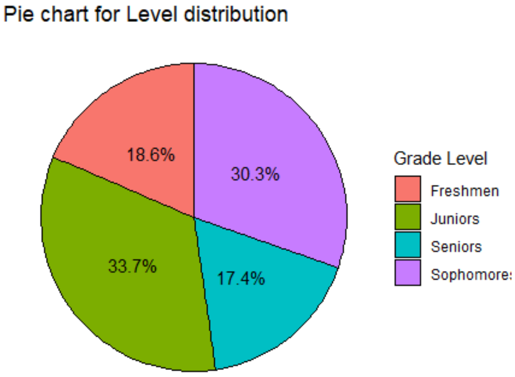


Figure 3. *Student Level Distribution*

The gender distribution in Figure 4 indicates that females make up 61% and males make up 39% of the gender distribution. Based on the pie chart in Figure 5, 87% of people have taken prior math coursework, while 13% have not. In Figure 6, it can be observed that among those who formed study groups, 75% of people had a percentage of not forming study groups, while the remaining 25% formed study groups.

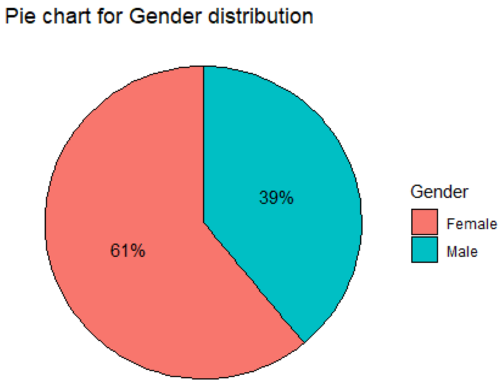


Figure 4. *Gender Distribution*

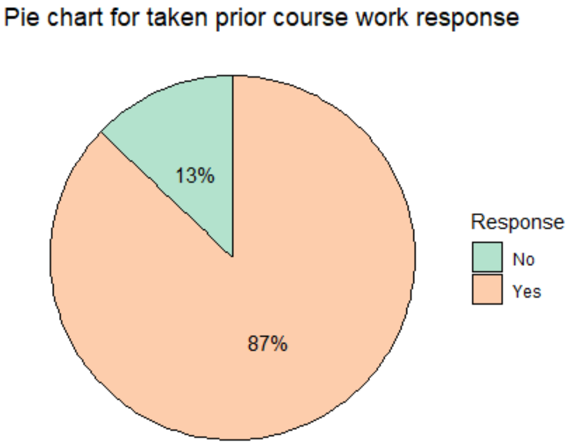


Figure 5. *Response Distribution For Prior Math Courses Taken*

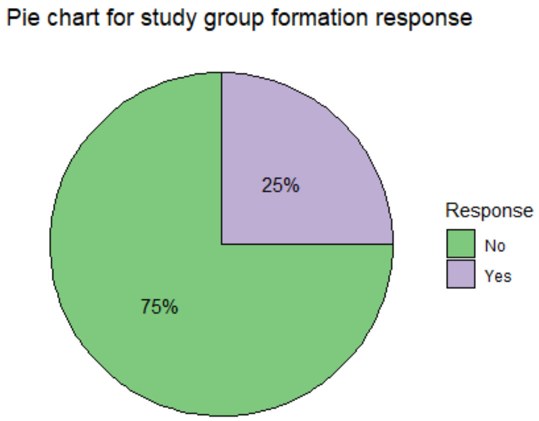


Figure 6. *Response Distribution For Study Groups Formation*

In Figure 7, individuals who often used online resources accounted for 56%, while those who rarely used such resources represented 39%. The lowest percentage was found among individuals who never used online resources, with only 6%. Figure 8 illustrates that people who experienced test anxiety had a distribution of 67%, while those who rarely experienced it had a distribution of 22%, and those who have never had test anxiety had 11%.

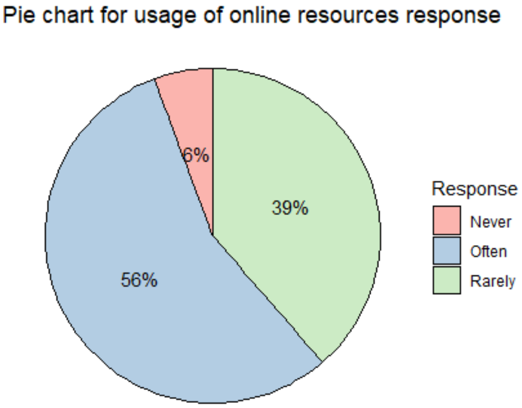


Figure 7. *Response Distribution For Online Resources Usage*

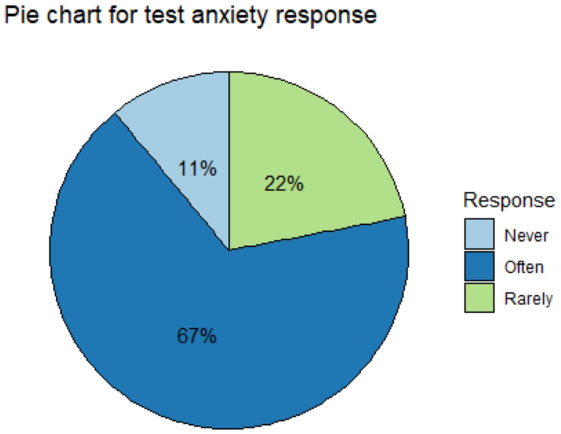


Figure 8. *Response Distribution For Test Anxiety*

The data presented in Figure 9 indicates that individuals who solved recommended problems while taking Calculus 1 had a distribution of 57%, whereas those who did not solve recommended problems had a distribution of 43%. Figure 10 shows that people who studied regularly had a distribution of 59%, while those who studied last minute had a distribution of 41%. Additionally, Figure 11 shows a distribution of 80% for people who did not attend office hours, while those who did attend had a distribution of 20%.

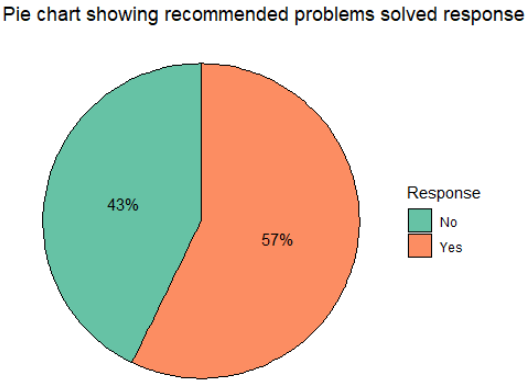


Figure 9. *Response Distribution For Solving Recommended Problems*

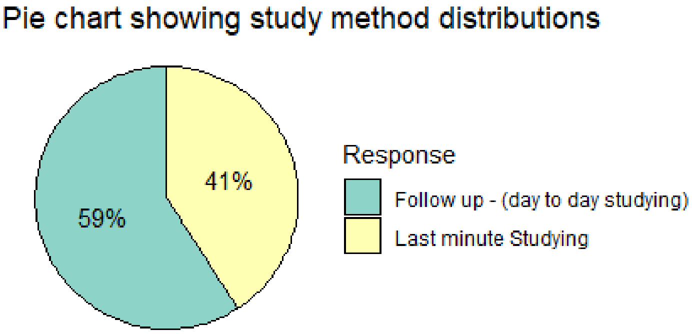


Figure 10. *Study Method Distribution*

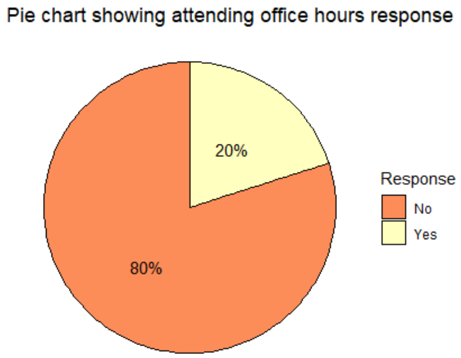


Figure 11. *Response Distribution For Attending Office Hours*

Based on Figure 12, it can be observed that a majority of participants, 79 each, belong to the age group of 19 and 20. The age group of 21 has a count of 44, followed by age 18 with a count of 24, age 22 with a count of 20, age 23 with a count of 9, age 17 with a count of 7, and age 24 with a count of 2.

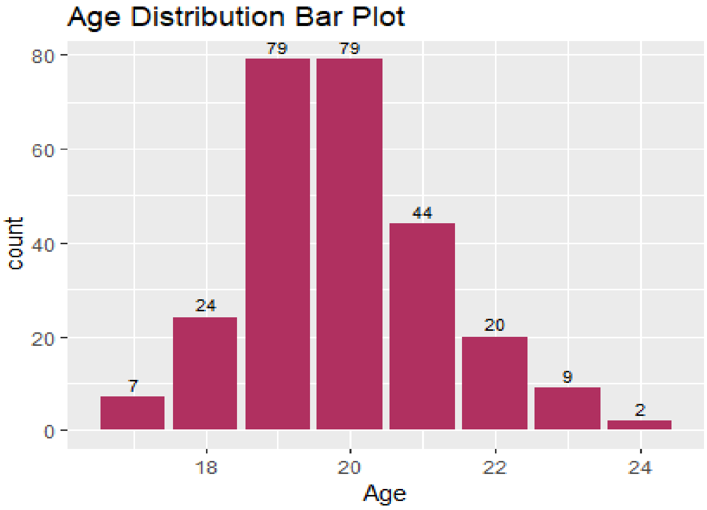


Figure 12. *Age Distribution*

According to Figure 13, the average study time for the participants was four hours, with a count of 118. 87 participants studied for an average of six hours, while 31 individuals studied for an average of eight hours. Additionally, 22 individuals studied for zero hours. The lowest count was for those who studied for ten hours, which was only six individuals.

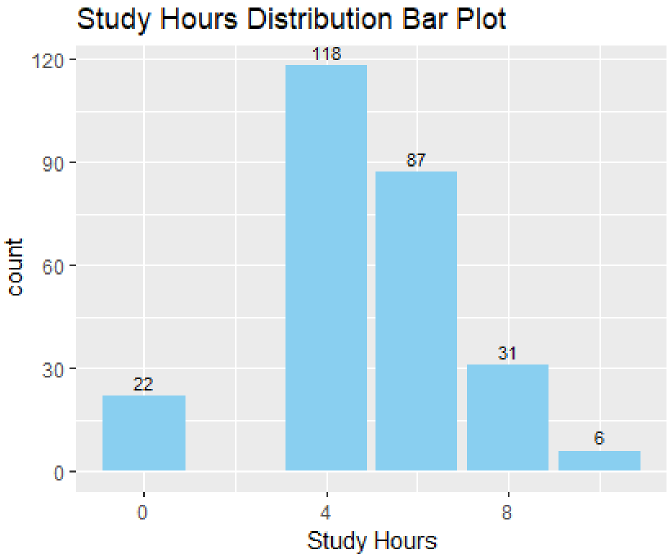


Figure 13. *Study Hour Distribution*

2.3.3 Classification

In our research, we aim to predict students' letter grades in Calculus 1 based on various factors related to their performance in this course. This will give us insight on how different factors affect students' grades. To get a deeper understanding we will use different classification techniques to figure out the important factors that determine how students' final grades are categorized.

2.3.3.1 Linear Discriminant Analysis

Linear Discriminant Analysis (LDA) is a technique that simplifies classification problems in supervised machine learning. LDA makes certain assumptions about the variables, namely that they are normally distributed with equal covariance matrices. By assuming normality, LDA can make predictions with minimal errors, which are only due to inaccuracies in the estimation of the mean and variance of the sample (Perme et al., 2004). The LDA is calculated by the following formula:

$$\delta_k(x) = x^T \Sigma^{-1} \mu_k - \frac{1}{2} \mu_k^T \Sigma^{-1} \mu_k + log\pi_k$$

In the given equation, $\delta_k(x)$ represents the linear discriminant function. The prior probability that an observation belongs to the kth class is denoted by π_k, while μ_k represents the mean parameter for the kth class and x^T is just the transpose of the feature vector x . The covariance matrix of all classes is represented by Σ (James et al., 2013). Further, to model the LDA in R we used the function \`lda\` from the \`MASS\` package (Chambers, 2008).

2.3.3.2 Support Vector Machines

Support Vector Machines (SVMs) are a powerful classification tool used in supervised machine learning. They can separate and classify data points by finding the optimal hyperplane between different categories, making them highly effective for accurate predictions and handling complex, non-linear data relationships. In simpler terms, SVMs can identify patterns in data and classify them accurately and efficiently (Meyer, 2001). The SVMs model can be calculated using the following formula:

$$K(x_i, x_{i'}) = \Sigma^p_{j=1} x_{ij} x_{i'j}$$

In the equation above, K represents the kernel function which measures the similarity between two observations. $K(x_i, x_{i'})$ is a generalization of the inner product of the observations x_i and $x_{i'}$. The equation calculates the kernel function as the sum of the products of corresponding components of the two vectors. This type of kernel is known as a linear kernel (James et al., 2013). In addition, to model the SVMs in R we used the function `svm` with the kernel specified to be 'linear' from the `e1071` package (Chambers, 2008; Meyer, 2001).

2.3.3.3 Naive Bayes

The Naive Bayes is a supervised classification model that assumes that each feature is independent. Despite its assumption, it has a strong theoretical foundation and is capable of handling real-world scenarios where the independence assumption is not entirely met. Naive Bayes is well-suited for high-dimensional data due to its computational efficiency and simplicity. In practice, it estimates the most likely class for a given test observation by calculating posterior probabilities based on the observed features and class prior probabilities (Taheri & Mammadov, 2013). The Naive Bayes can be calculated as follows:

$$P(Y = k|X = x) = \frac{\pi_k f_k(x)}{\sum_{j=1}^{K} \pi_j f_j(x)}$$

Where $\pi_1,\ldots, \pi_k$ are the prior probabilities and $f_k(x)$ is the joint distribution and can be represented by the product of marginal distributions. The $\sum_{j=1}^{K} \pi_j f_j(x)$ is just the summation over all possible classes j from 1 to K, of the prior probability and the joint distribution for each class from 1 to K, where K is the total number of classes (James et al., 2013). Furthermore, to model the Naive Bayes in R we used the function `naiveBayes` from the `e1071` package (Chambers, 2008).

2.3.3.4 Multinomial Logistic Regression

Multinomial logistic regression is a technique that handles multiclass classification tasks that extend the logistic regression technique. Its utility becomes most evident when the outcome involves more than two classes. The model accounts for multiple classes and enables accurate classification of data points by considering the relationship between the dependent and independent variables (El-Habil, 2012). The Multinomial is calculated as follows:

$$log(\frac{P(Y=k|X)}{P(Y=K|X)}) = \beta_{k0} + \Sigma^{p}_{j=1} \beta_{kj} x_{j}$$

The equation consists of several components. P(Y=k|X) indicates the probability of the dependent variable taking on the value k, given the independent variables X. Similarly, P(Y=K|X) is the probability of the dependent variable taking on the value K, given the independent variables X. β_{k0} stands for the intercept term for class k, while $\Sigma^{p}_{j=1} \beta_{kj} x_{j}$ represent the linear combination of the independent variables $x_{j}\, for\, j = 1,.., p$, with coefficients $\beta_{kj}\, for\, j = 1,.., p$ for class k (James et al., 2013). Moreover, to model the Multinomial logistic regression in R we used the function `multinom` from the `nnet` package (Chambers, 2008).

2.3.3.5 Random Forest

Random forest is a type of supervised classification technique that is based on decision trees. In this technique, each tree depends on the value of an independent variable and has the same distribution over all trees in the forest. The error rate in this model primarily depends on the strength of the individual trees in the forest and the correlation between the trees. As the number of trees in the forest increases, it results in a more accurate model. To split each node, predictors are randomly selected. Furthermore, the model can estimate the importance of each variable, which is a valuable property for determining the best predictors (Aljahdali & Hussain, 2013). To model the Random forest in R we used the function `randomForest` from the `randomForest` package (Chambers, 2008).

3. Results And Discussion

This section sheds light on the results obtained from our data modeling efforts and followed by a detailed discussion on these results. First, we present our conclusive dataset employed for data modeling (3.1). Subsequently, we delve into the components within the confusion matrix (3.2) , proceed to analyze the results, determine our optimal model (3.3), and conclude with an examination of the limitations associated with our findings (3.4).

3.1 Modeling and Data Transformation

3.1.1 Overview of Final Data

The primary objective of this study was to predict the grades of undergraduate students in Khalifa University based on data from students who had previously taken Calculus 1. We employed various algorithms such as Random Forest, Linear Discriminant Analysis (LDA), Support Vector Machines (SVMs), Naive Bayes and Multinomial logistic regression, which were calculated and compared to predict the letter grade of the students. This study focused on key parameters, namely C (Gender), F (Average Study Hours), D (Letter Grade) and K (Studying day-to-day or last minute). This decision was made using a tool in Random Forest which showed the variable importance. This in turn helped us to choose the best features to focus on. We then proceeded with the data transformation process, incorporating one-hot encoding to convert binary values into 0s and 1s. We assigned the value 0 for "Female" and 1 for "Male" in the gender classification. Similarly, for the variable "Studying day to day or last minute", we chose "Last minute" as 0 and the other option as 1. While developing the model we found that hours of studying, if the student followed-up or not and gender are independent variables and the Grades are the dependent variable. In assessing the model's performance, our emphasis lies in obtaining the confusion matrix for each of the five models. Primary considerations for the evaluation encompass accuracy, with potential scrutiny of additional metrics from the confusion matrix, including Sensitivity, Specificity, and Prevalence, were deemed necessary for a more comprehensive assessment.

3.2 Assessing Model Performance

3.2.1 Confusion Matrix

The confusion matrix serves as a reflection of the dataset's current state providing a breakdown of the model's accurate and erroneous predictions (Visa et al., 2011). The model's performance is assessed through its ability to correctly classify instances as well as those instances that it misclassified. Table 2 displays the confusion matrix, where the rows correspond to the actual sample quantities in the test set, and the columns represent the model's predictions. The values in Table 2 are as follows:

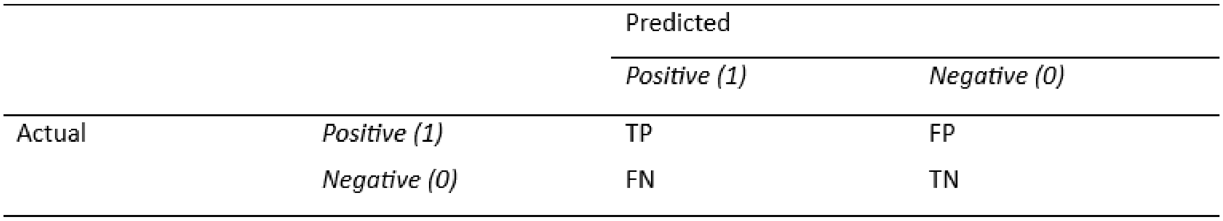
True Positive(TP): Instances where the positive class was correctly classified as positive

True Negative (TN): Instances where the negative class was correctly classified as negative

False Positive(FP): Instances where a negative class was incorrectly classified as positive

False Negative (FN): Instances where a positive class was incorrectly classified as negative

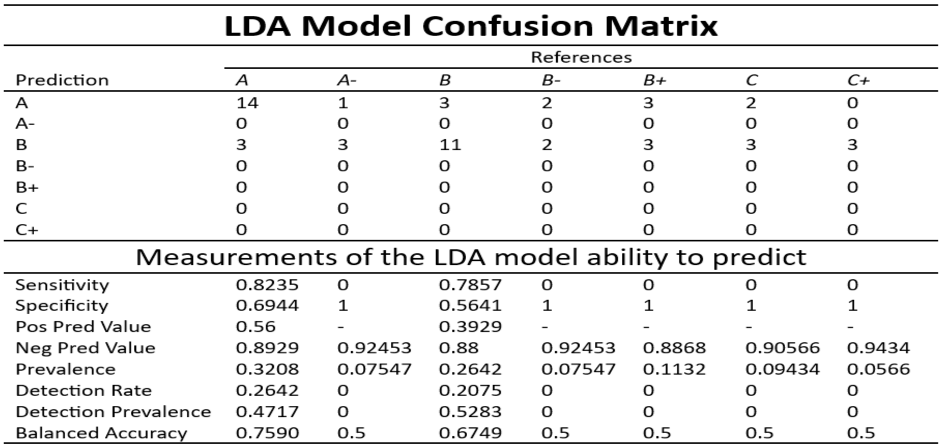
Table 2: *The Confusion Matrix*



		Predicted	
		Positive (1)	*Negative (0)*
Actual	*Positive (1)*	TP	FP
	Negative (0)	FN	TN

Table 3,4,5,6 and 7 are the confusion matrices of all the five models used in our study. The first half of the confusion matrices, which is of 7x7 dimensions, displays accurately predicted instances on the main diagonal, while the off-diagonal elements indicate the incorrectly predicted instances. The second half of the confusion matrix explains the key measurements which can help in assessing the model's predictive ability. These are Sensitivity, Specificity, Positive Predictive Value, Negative Predictive Value, Prevalence, Detection Rate, Detection Prevalence and Balanced Accuracy.

Table 3: *Confusion Matrix of LDA model and Some of its Metrics*



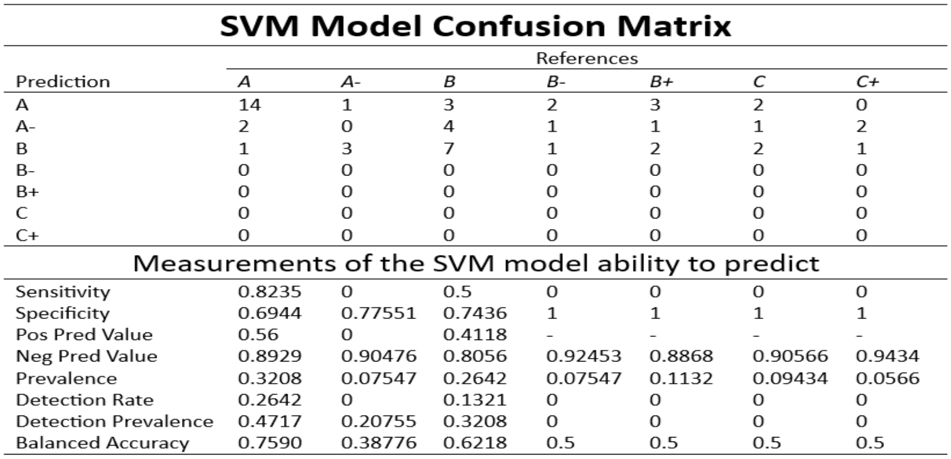
LDA Model Confusion Matrix

	References						
Prediction	*A*	*A-*	*B*	*B-*	*B+*	*C*	*C+*
A	14	1	3	2	3	2	0
A-	0	0	0	0	0	0	0
B	3	3	11	2	3	3	3
B-	0	0	0	0	0	0	0
B+	0	0	0	0	0	0	0
C	0	0	0	0	0	0	0
C+	0	0	0	0	0	0	0

Measurements of the LDA model ability to predict

Sensitivity	0.8235	0	0.7857	0	0	0	0
Specificity	0.6944	1	0.5641	1	1	1	1
Pos Pred Value	0.56	-	0.3929	-	-	-	-
Neg Pred Value	0.8929	0.92453	0.88	0.92453	0.8868	0.90566	0.9434
Prevalence	0.3208	0.07547	0.2642	0.07547	0.1132	0.09434	0.0566
Detection Rate	0.2642	0	0.2075	0	0	0	0
Detection Prevalence	0.4717	0	0.5283	0	0	0	0
Balanced Accuracy	0.7590	0.5	0.6749	0.5	0.5	0.5	0.5

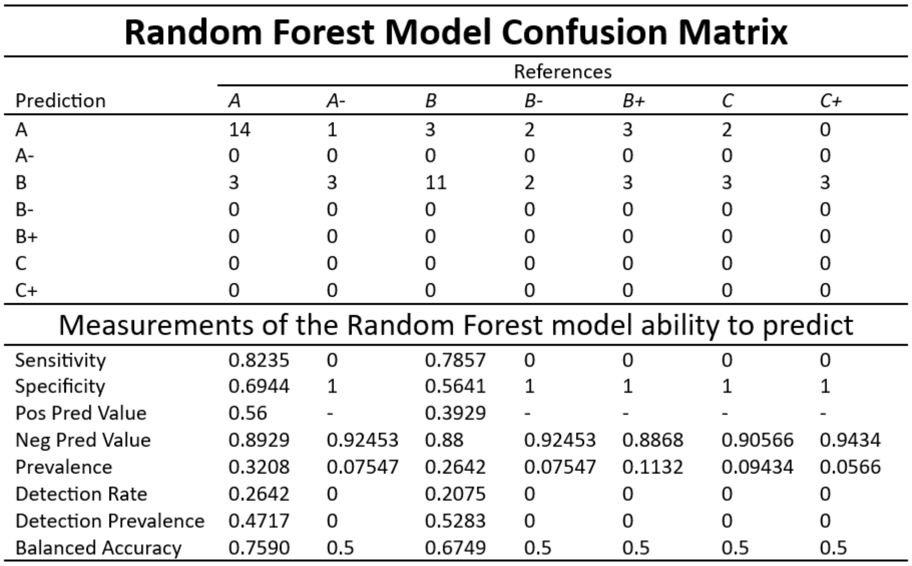
Table 4: *Confusion Matrix of SVM model and Some of its Metrics*



SVM Model Confusion Matrix

	References						
Prediction	*A*	*A-*	*B*	*B-*	*B+*	*C*	*C+*
A	14	1	3	2	3	2	0
A-	2	0	4	1	1	1	2
B	1	3	7	1	2	2	1
B-	0	0	0	0	0	0	0
B+	0	0	0	0	0	0	0
C	0	0	0	0	0	0	0
C+	0	0	0	0	0	0	0
Measurements of the SVM model ability to predict							
Sensitivity	0.8235	0	0.5	0	0	0	0
Specificity	0.6944	0.77551	0.7436	1	1	1	1
Pos Pred Value	0.56	0	0.4118	-	-	-	-
Neg Pred Value	0.8929	0.90476	0.8056	0.92453	0.8868	0.90566	0.9434
Prevalence	0.3208	0.07547	0.2642	0.07547	0.1132	0.09434	0.0566
Detection Rate	0.2642	0	0.1321	0	0	0	0
Detection Prevalence	0.4717	0.20755	0.3208	0	0	0	0
Balanced Accuracy	0.7590	0.38776	0.6218	0.5	0.5	0.5	0.5

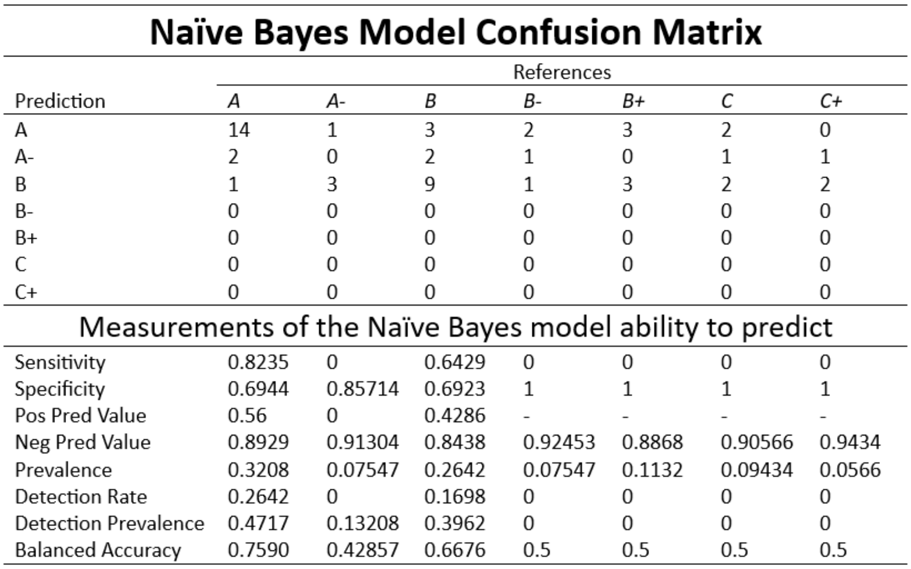
Table 5: *Confusion Matrix of Random Forest model and Some of its Metrics*



Random Forest Model Confusion Matrix

	References						
Prediction	*A*	*A-*	*B*	*B-*	*B+*	*C*	*C+*
A	14	1	3	2	3	2	0
A-	0	0	0	0	0	0	0
B	3	3	11	2	3	3	3
B-	0	0	0	0	0	0	0
B+	0	0	0	0	0	0	0
C	0	0	0	0	0	0	0
C+	0	0	0	0	0	0	0
Measurements of the Random Forest model ability to predict							
Sensitivity	0.8235	0	0.7857	0	0	0	0
Specificity	0.6944	1	0.5641	1	1	1	1
Pos Pred Value	0.56	-	0.3929	-	-	-	-
Neg Pred Value	0.8929	0.92453	0.88	0.92453	0.8868	0.90566	0.9434
Prevalence	0.3208	0.07547	0.2642	0.07547	0.1132	0.09434	0.0566
Detection Rate	0.2642	0	0.2075	0	0	0	0
Detection Prevalence	0.4717	0	0.5283	0	0	0	0
Balanced Accuracy	0.7590	0.5	0.6749	0.5	0.5	0.5	0.5

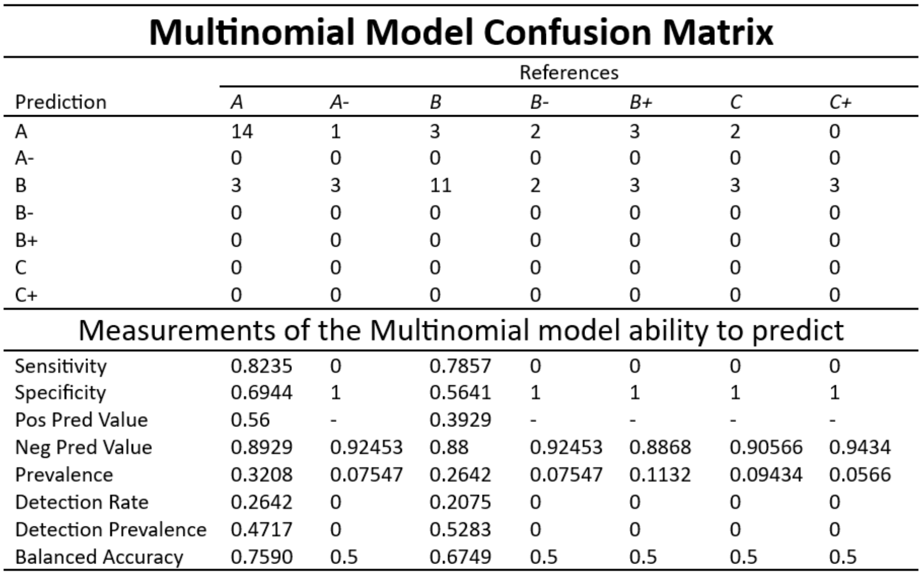
Table 6: *Confusion Matrix of Naive Bayes model and Some of its Metrics*



Naïve Bayes Model Confusion Matrix

	References						
Prediction	*A*	*A-*	*B*	*B-*	*B+*	*C*	*C+*
A	14	1	3	2	3	2	0
A-	2	0	2	1	0	1	1
B	1	3	9	1	3	2	2
B-	0	0	0	0	0	0	0
B+	0	0	0	0	0	0	0
C	0	0	0	0	0	0	0
C+	0	0	0	0	0	0	0
Measurements of the Naïve Bayes model ability to predict							
Sensitivity	0.8235	0	0.6429	0	0	0	0
Specificity	0.6944	0.85714	0.6923	1	1	1	1
Pos Pred Value	0.56	0	0.4286	-	-	-	-
Neg Pred Value	0.8929	0.91304	0.8438	0.92453	0.8868	0.90566	0.9434
Prevalence	0.3208	0.07547	0.2642	0.07547	0.1132	0.09434	0.0566
Detection Rate	0.2642	0	0.1698	0	0	0	0
Detection Prevalence	0.4717	0.13208	0.3962	0	0	0	0
Balanced Accuracy	0.7590	0.42857	0.6676	0.5	0.5	0.5	0.5

Table 7: *Confusion Matrix of Multinomial model and Some of its Metrics*



Multinomial Model Confusion Matrix

	References						
Prediction	*A*	*A-*	*B*	*B-*	*B+*	*C*	*C+*
A	14	1	3	2	3	2	0
A-	0	0	0	0	0	0	0
B	3	3	11	2	3	3	3
B-	0	0	0	0	0	0	0
B+	0	0	0	0	0	0	0
C	0	0	0	0	0	0	0
C+	0	0	0	0	0	0	0
Measurements of the Multinomial model ability to predict							
Sensitivity	0.8235	0	0.7857	0	0	0	0
Specificity	0.6944	1	0.5641	1	1	1	1
Pos Pred Value	0.56	-	0.3929	-	-	-	-
Neg Pred Value	0.8929	0.92453	0.88	0.92453	0.8868	0.90566	0.9434
Prevalence	0.3208	0.07547	0.2642	0.07547	0.1132	0.09434	0.0566
Detection Rate	0.2642	0	0.2075	0	0	0	0
Detection Prevalence	0.4717	0	0.5283	0	0	0	0
Balanced Accuracy	0.7590	0.5	0.6749	0.5	0.5	0.5	0.5

Sensitivity :

Sensitivity measures the ability of the model to correctly identify the instances of each class (Vapnik, 2010). Sensitivity is calculated by :

$$\frac{TP}{TP+FN}$$

Specificity:

Specificity measures the model's ability to correctly identify negative instances (Vapnik, 2010). Specificity is calculated by:

$$\frac{TN}{TN + FP}$$

Positive Predictive Value (Pos Pred Value):

Positive Predictive Value is a measure of how many of the predicted positive instances are actually true positives (Vapnik, 2010). This is calculated by :

$$\frac{TP}{TP + FP}$$

Negative Predictive Value (Neg Pred Value):

Negative Predictive Value measures the probability that a predicted negative instance is truly negative (Vapnik, 2010). This is calculated by :

$$\frac{TN}{FN + TN}$$

Prevalence:

Prevalence indicates how many true cases actually occurred out of all the observations (Vapnik, 2010). This is calculated by :

$$\frac{FN + TP}{TN + FP + FN + TP}$$

Detection Rate:

Detection Rate represents how many actual positive instances are correctly identified by the model (Vapnik, 2010). This is calculated by :

$$\frac{TP}{TP + FN}$$

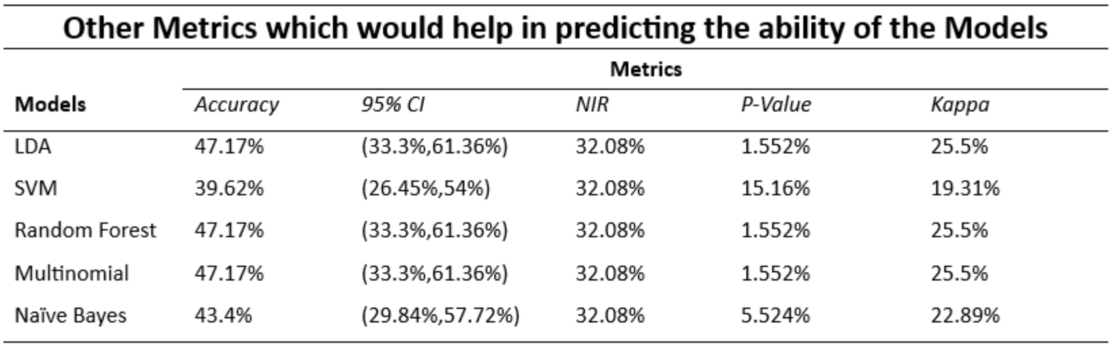
Detection Prevalence:

Detection Prevalence represents the proportion of positive predictions made by the model (Vapnik, 2010). This is calculated by :

$$\frac{TP + FP}{TP + TN + FP + FN}$$

These seven key measurements are assessed individually for each specific grade category within the dataset to evaluate the model's performance with respect to each grade. Table 8 showcases the key measurements such as Accuracy, 95% Confidence Interval, No Information Rate, P-Value and Kappa values of all the five models. These measurements evaluate the overall model.

Table 8: *Other Metrics Used for Predictions*



Other Metrics which would help in predicting the ability of the Models

	Metrics				
Models	*Accuracy*	*95% CI*	*NIR*	*P-Value*	*Kappa*
LDA	47.17%	(33.3%,61.36%)	32.08%	1.552%	25.5%
SVM	39.62%	(26.45%,54%)	32.08%	15.16%	19.31%
Random Forest	47.17%	(33.3%,61.36%)	32.08%	1.552%	25.5%
Multinomial	47.17%	(33.3%,61.36%)	32.08%	1.552%	25.5%
Naïve Bayes	43.4%	(29.84%,57.72%)	32.08%	5.524%	22.89%

Accuracy :

The proportion of correctly classified instances out of the total number of instances (James et al., 2013). It is measured by :

$$\frac{TP + TN}{TP + FP + TN + FN}$$

95 % CI (Confidence Interval):

A range of values that provides a level of confidence for the accuracy estimate (Moore, 2009). It is measured by:

$$\left(\frac{PN}{n} - Z.\sqrt{\frac{\frac{PN}{n}(1-\frac{PN}{n})}{n}}, \frac{PN}{n} + Z.\sqrt{\frac{\frac{PN}{n}(1-\frac{PN}{n})}{n}}\right)$$

Here the n value represents the total number of instances and Z value represents the critical value from the normal table corresponding to the confidence level (Moore, 2009).

No Information Rate (NIR):

NIR represents the accuracy that would be achieved by a model if it simply predicted the majority class for every instance without considering any features or patterns in the data (Bicego & Mensi, 2023) . It is measured by:

$$\frac{Number\ of\ Instances\ in\ the\ Majority\ Class}{Total\ number\ of\ instances}$$

P - Value [ACC > NIR]:

P-Value tests whether the model's accuracy is slightly different from the accuracy achieved by the No Information Rate model. Normally, a low P-Value suggests that the model's accuracy is statistically significant (James et al., 2013)

Kappa:

Kappa measures the agreement between the model's predictions and the actual outcomes,while considering the agreement that would be expected by chance (Wan et al., 2015). It is measured by:

$$\frac{P_o - P_e}{1-P_e}$$

Here the P_o is called the observed agreement which is the proportion of instances for which the two raters or models agreed (Wan et al., 2015). It is measured by :

$$\frac{TP + TN}{TP + TN + FN + FP}$$

And the P_e is called the expected agreement which represents the agreement that would be expected by chance. It is based on the marginal probabilities of agreement for each category (Wan et al., 2015). It is measured by:

$$\frac{((TP + FN)\ x\ (TP + FP)) + ((TN + FP)\ x\ (TN+ FN))}{(TP + TN + FN + FP)^2}$$

3.3 Evaluating Model Results

3.3.1 Mean Accuracy Comparison

Next, we compare the values of accuracy in the five models as accuracy is a valuable indicator of how well our model performs (Da Silva and Skinner, 2013). To do this, we created a bar chart which compares the values of the accuracies which is depicted in Figure 14**.** In Figure 14**,** we observe that the highest accuracy is 47.17% which is found in three models which are Multinomial, Random Forest and Linear Discriminant Analysis (LDA) and lower accuracies of 43.4% on the Naive Bayes and 39.62% on the Support Vector Machines model. Therefore, it appears that the best models are Multinomial, Random Forest and LDA. However, when we look further into other metrics in Table 8, such as 95% CI, NIR, P-Value, and Kappa, we find that they are identical for the three models. Hence, we cannot definitively determine which model is the best among the three. Therefore, we suggest that the reader can select the model according to their future analysis needs.

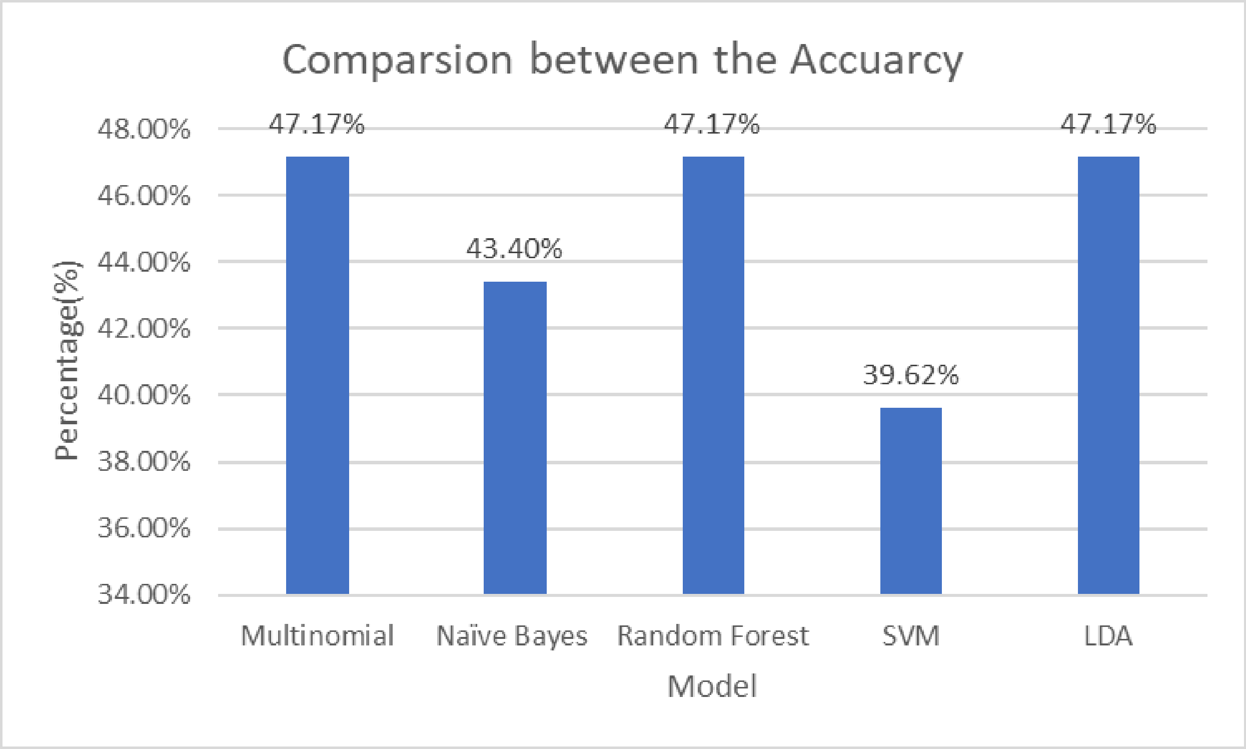


Figure 14 : *Mean Accuracy of the five models*

3.4 Limitations

We see that our model does not have very high accuracy and thereby in this section we aim to highlight the limitations of our study as we believe recognizing and addressing these limitations is essential to provide a comprehensive understanding of our research context:

A) Feature Selection:

We acknowledge that our study concentrated on Gender, Hours Studied, and Follow-up Material variables. While these factors were theoretically linked to predicting student grades, we recommend considering the inclusion of additional predictors, which we did not take into account, in the dataset for future research. This broader feature selection may provide a more comprehensive understanding of the factors influencing student performance.

B) Data Bias:

One notable limitation in our study pertains to the potential for data bias. Given that our analysis relied on self-reported data collected from participants, there is a reasonable concern regarding the accuracy of the information provided. Specifically, there exists the possibility that individuals who achieved lower grades may have been inclined to misrepresent their academic performance. This discrepancy could result from a variety of factors, such as social desirability bias, where participants tend to present themselves in a more favorable light, or the stigma associated with lower grades, which might encourage respondents to overstate their achievements. The presence of such data bias underscores the challenge of obtaining a completely accurate and unbiased representation of participants' academic experiences.

Note that the full code can be found in Appendix B.

4. Conclusion

In this research paper, we delved into the challenging realm of predicting letter grades for Calculus 1 students at Khalifa University. The intent of our research is to provide actionable insights for educators and administrators to enhance academic support services and interventions. Throughout our study, we employed five different classification models: Random Forest, Linear Discriminant Analysis (LDA), Support Vector Machine (SVM), Naive Bayes and Multinomial logistics regression to predict students' grades based on data related to gender, study hours and follow-up practices which was considered based on the Variable Importance tool in the Random Forest model.

The results of our study provide valuable insights into the predictive power of these models. Notably, the models, Multinomial logistic regression, Random Forest and LDA , demonstrated the highest accuracy, all scoring around 47.17%. However, to understand which among the three models is the most ideal we considered other metrics such as 95% Confidence Interval, No Information Rate, P-Value and Kappa values. These metrics yielded very similar results in all three models. Hence, the choice among them depends on the specific analytical needs and preferences of readers.

While this study provides valuable insights into predicting students' grades in Calculus 1, it is crucial to acknowledge its limitations. The potential for data bias exists due to self-reported data collection is one of the main limitations. Additionally, the modest accuracy rate of 47.17% suggests room for improvement, indicating the need for further research and the exploration of additional predictors to enhance predictive models.

As higher education continues to evolve, the pursuit of academic excellence remains paramount. Our research aligns with this overarching goal by providing a foundation for future studies to refine predictive models and advance the understanding of factors influencing student performance. Recognizing these limitations, we invite researchers and educators to build upon our work, fostering continuous improvement in academic support services for the benefit of students and institutions alike.

References

Al-Barrak, M. A., & Al-Razgan, M. (2016). Predicting Students Final GPA Using Decision Trees: A Case Study. *International Journal of Information and Education Technology*, 6(7), 528–533. https://doi.org/10.7763/ijiet.2016.v6.745

Aljahdali, S., & Hussain, S. N. (2013). Comparative prediction performance with support vector machine and random forest classification techniques. *International Journal of Computer Applications*, 69(11), 12-16. https://doi.org/10.5120/11885-7922

Bicego, M., & Mensi, A. (2023). Null/No Information Rate (NIR): a statistical test to assess if a classification accuracy is significant for a given problem. *arXiv (Cornell University).* https://doi.org/10.48550/arxiv.2306.06140

Bujang, S. D. A., Selamat, A., Ibrahim, R., Krejcar, O., Herrera-Viedma, E., Fujita, H., & Ghani, N. A. M. (2021). Multiclass Prediction Model for Student Grade Prediction Using Machine Learning. *IEEE Access, 9, 95608–95621*. https://doi.org/10.1109/access.2021.3093563

Chambers, J. (2008). Software for Data Analysis: Programming with R. *Springer* New York. https://doi.org/10.1007/978-0-387-75936-4

Chen, J., Luo, L., & Song, J. (2019, February). Course performance prediction for basic courses of universities based on support vector machines. *Journal of Physics: Conference Series*, 1168, 032066. https://doi.org/10.1088/1742-6596/1168/3/032066

Da Silva, D. N., & Skinner, C. (2013). The use of accuracy indicators to correct for survey measurement error. *Journal of the Royal Statistical Society Series C: Applied Statistics*, *63*(2), 303–319. https://doi.org/10.1111/rssc.12022

El-Habil, A. M. (2012). An Application on Multinomial Logistic Regression Model. *Pakistan Journal of Statistics and Operation Research*, 8(2), 271-291. https://doi.org/10.18187/pjsor.v8i2.234

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). An Introduction to Statistical Learning: with Applications in R (1st ed.). *New York: Springer.* https://doi.org/10.1007/978-1-4614-7138-7

Meyer, D. (2001). Support Vector Machines: The Interface to libsvm in package e1071. *R News*, 1(1), 41-45. https://www.researchgate.net/publication/242323440_Support_Vector_Machines_The_Interface_to_libsvm_in_package_e1071

Moore, D. S. (2009). Introduction to the Practice of Statistics. *WH Freeman and company.*

Nabizadeh, A. H., Goncalves, D., Gama, S., & Jorge, J. (2022, June 1). Early Prediction of Students' Final Grades in a Gamified Course. *IEEE Transactions on Learning Technologies*, 15(3), 311–325. https://doi.org/10.1109/tlt.2022.3170494

Perme, M., Blas, M., & Turk, S. (2004). Comparison of Logistic Regression and Linear Discriminant Analysis: A Simulation Study. *Metodološki Zvezki*, 1(1), 143-161. https://doi.org/10.51936/ayrt6204

Taheri, S., & Mammadov, M. (2013). Learning the naive Bayes classifier with optimization models. *International Journal of Applied Mathematics and Computer Science*, 23(4), 787-795. https://doi.org/10.2478/amcs-2013-0059

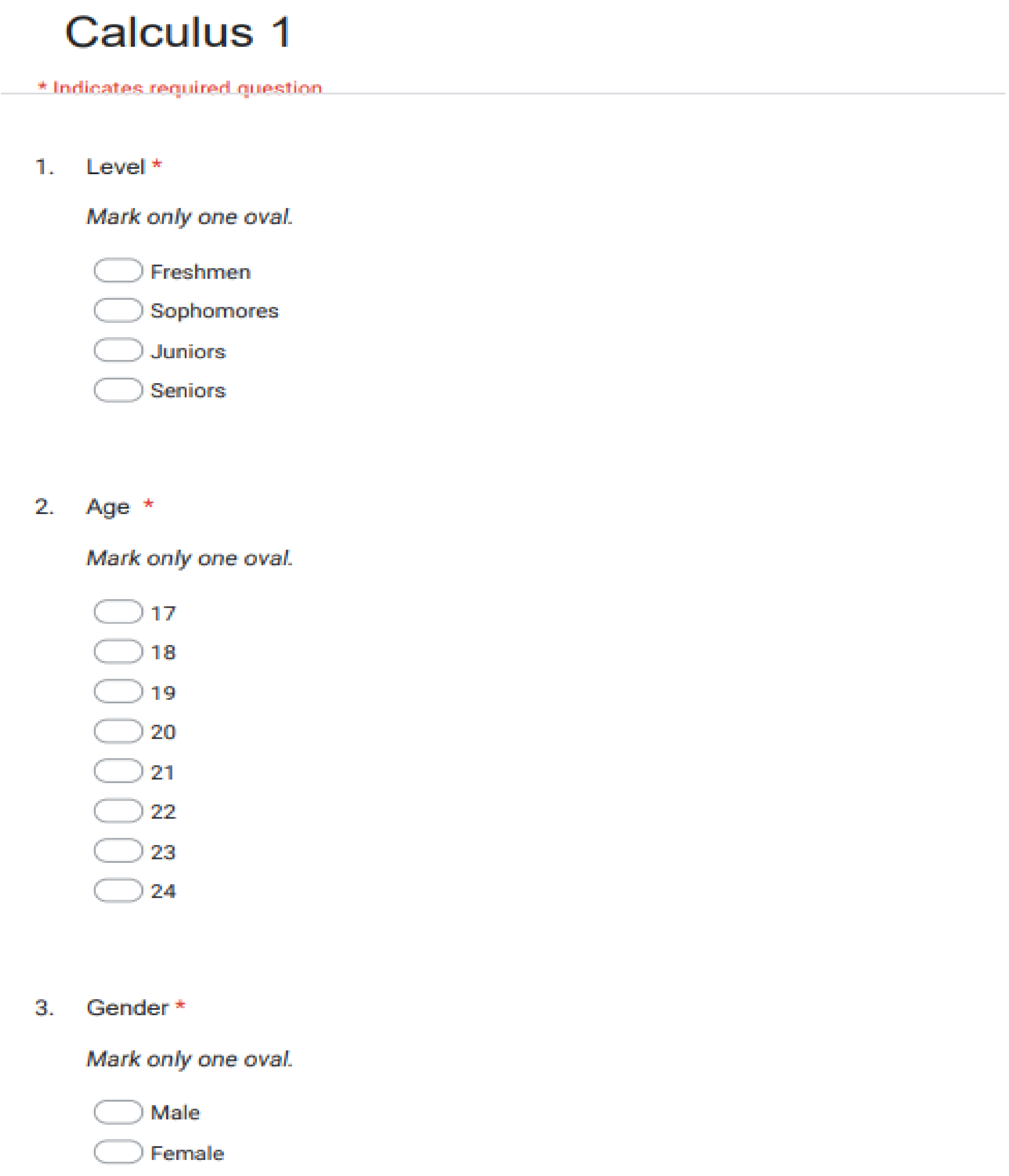
Vapnik, V. N. (2010). The nature of statistical learning theory. *Springer.*

Visa, S., Ramsay, B., Ralescu, A. L., & Van Der Knaap, E. (2011). *Confusion matrix-based feature selection. Maics*, 710(1), 120-127.

Wan, T. A. N. G., Jun, H. U., Zhang, H., Pan, W. U., & Hua, H. E. (2015). Kappa coefficient: a popular measure of rater agreement*. Shanghai archives of psychiatry*, *27*(1), 62.

Yağcı, M. (2022). Educational data mining: prediction of students' academic performance using machine learning algorithms. *Smart Learn. Environ.* 9, 11. https://doi.org/10.1186/s40561-022-00192-z

Appendix A: Questionnaire



Calculus 1

* Indicates required question

1. Level *

Mark only one oval.

○ Freshmen
○ Sophomores
○ Juniors
○ Seniors

2. Age *

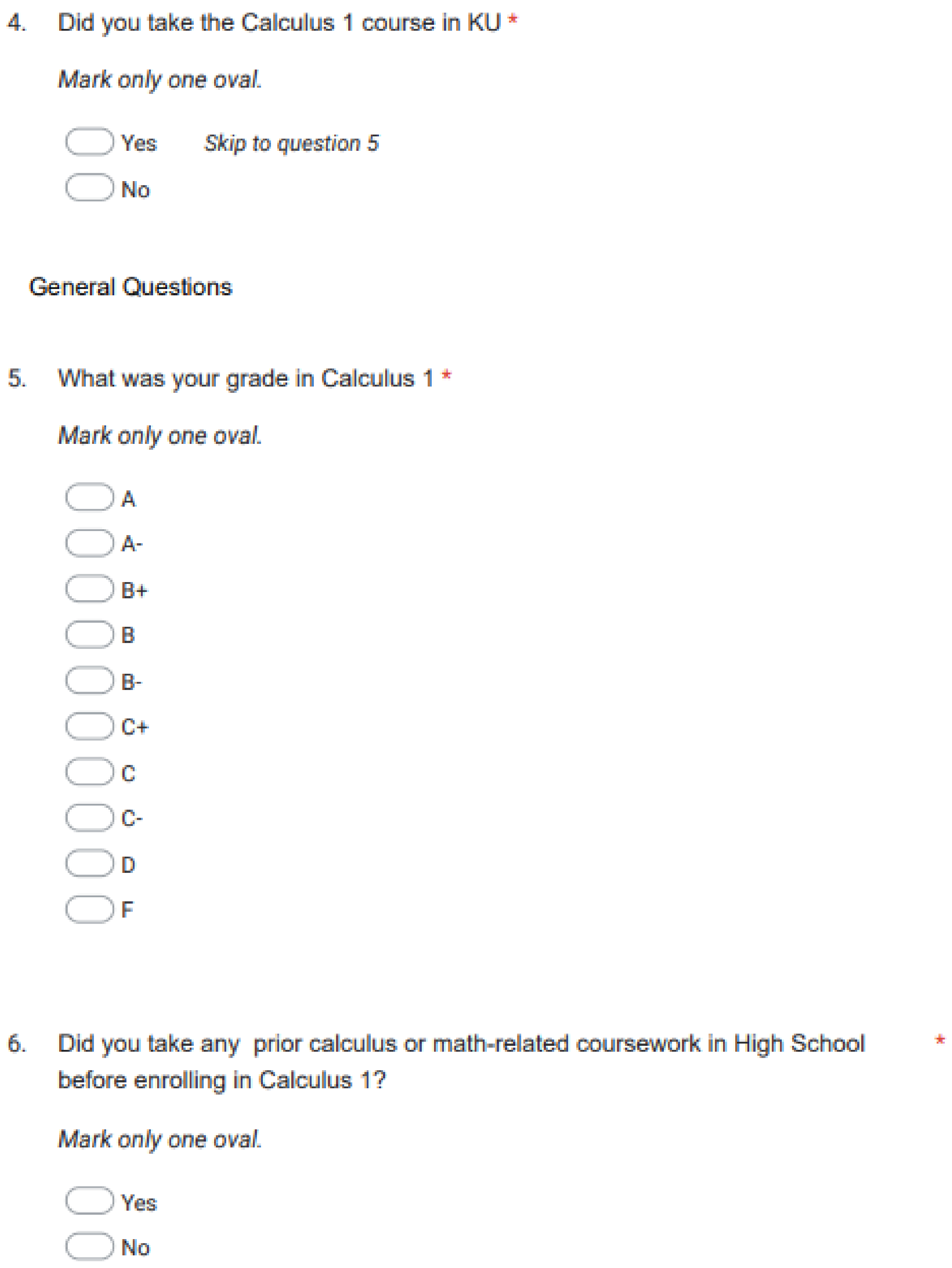
Mark only one oval.

○ 17
○ 18
○ 19
○ 20
○ 21
○ 22
○ 23
○ 24

3. Gender *

Mark only one oval.

○ Male
○ Female



4. Did you take the Calculus 1 course in KU *

Mark only one oval.

○ Yes *Skip to question 5*
○ No

General Questions

5. What was your grade in Calculus 1 *

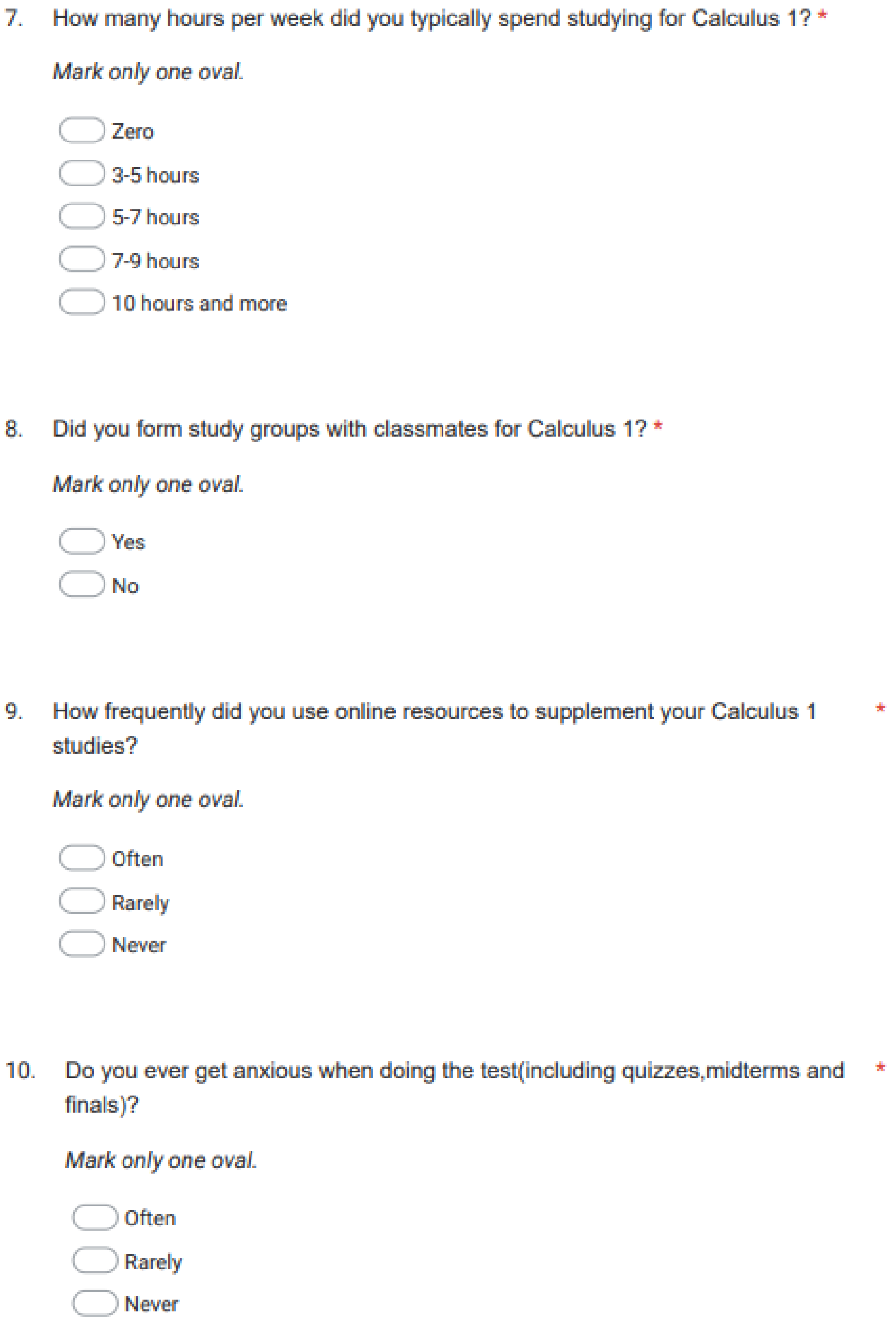
Mark only one oval.

○ A
○ A-
○ B+
○ B
○ B-
○ C+
○ C
○ C-
○ D
○ F

6. Did you take any prior calculus or math-related coursework in High School before enrolling in Calculus 1? *

Mark only one oval.

○ Yes
○ No



7. How many hours per week did you typically spend studying for Calculus 1? *

Mark only one oval.

○ Zero
○ 3-5 hours
○ 5-7 hours
○ 7-9 hours
○ 10 hours and more

8. Did you form study groups with classmates for Calculus 1? *

Mark only one oval.

○ Yes
○ No

9. How frequently did you use online resources to supplement your Calculus 1 studies? *

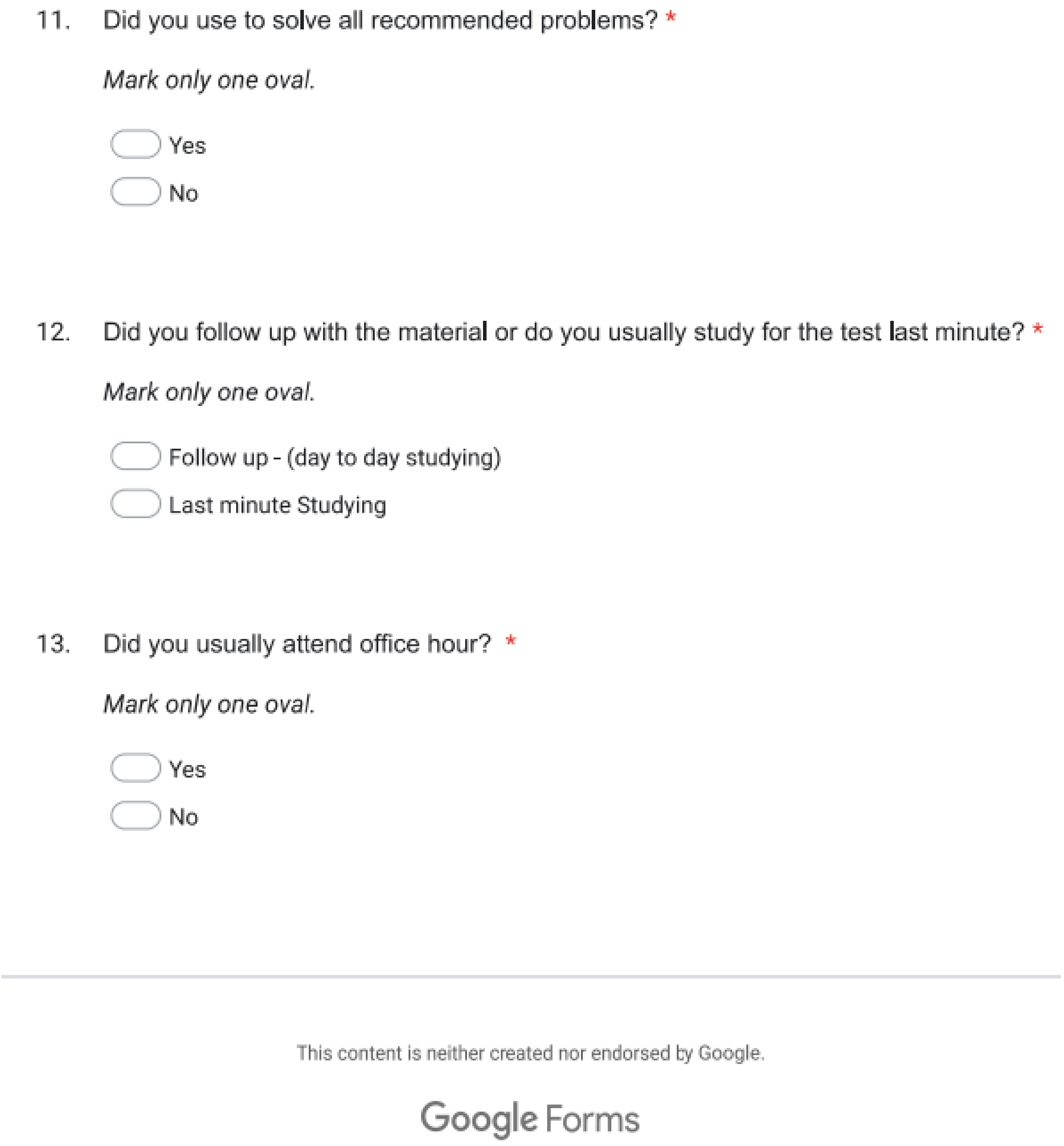
Mark only one oval.

○ Often
○ Rarely
○ Never

10. Do you ever get anxious when doing the test(including quizzes,midterms and finals)? *

Mark only one oval.

○ Often
○ Rarely
○ Never



11. Did you use to solve all recommended problems? *

Mark only one oval.

○ Yes

○ No

12. Did you follow up with the material or do you usually study for the test last minute? *

Mark only one oval.

○ Follow up - (day to day studying)

○ Last minute Studying

13. Did you usually attend office hour? *

Mark only one oval.

○ Yes

○ No

This content is neither created nor endorsed by Google.

Google Forms

Appendix B: Full Code

```
library(tidyverse)

library(caret)

data <- read.csv("C:/Users/dimad/OneDrive/Desktop/Multivariate Statistics/Calculus 1a.csv")

data

data <- filter(data,data$Did.you.take.the.Calculus.1.course.in.KU == "Yes")

data<-subset(data,select = -Did.you.take.the.Calculus.1.course.in.KU )

data <- subset(data, select = -Timestamp)

data

# Find the average of hours spent studying

data$How.many.hours.per.week.did.you.typically.spend.studying.for.Calculus.1.[data$How.many.hours.per.week.did.you.typically.spend.studying.for.Calculus.1. == "Zero"] <- 0

data$How.many.hours.per.week.did.you.typically.spend.studying.for.Calculus.1.[data$How.many.hours.per.week.did.you.typically.spend.studying.for.Calculus.1. == "3-5 hours"] <- 4

data$How.many.hours.per.week.did.you.typically.spend.studying.for.Calculus.1.[data$How.many.hours.per.week.did.you.typically.spend.studying.for.Calculus.1. == "5-7 hours"] <- 6

data$How.many.hours.per.week.did.you.typically.spend.studying.for.Calculus.1.[data$How.many.hours.per.week.did.you.typically.spend.studying.for.Calculus.1. == "7-9 hours"] <- 8

data$How.many.hours.per.week.did.you.typically.spend.studying.for.Calculus.1.[data$How.many.hours.per.week.did.you.typically.spend.studying.for.Calculus.1. == "10 hours and more"] <- 10

str(data$How.many.hours.per.week.did.you.typically.spend.studying.for.Calculus.1.)
```

```
data$How.many.hours.per.week.did.you.typically.spend.studying.for.Calculus.1.<-as.numeric(data$How.many.hours.per.week.did.you.typically.spend.studying.for.Calculus.1.)

str(data)

data$Level<-as.factor(data$Level)

data$Age<-as.numeric(data$Age)

data$Gender<-as.factor(data$Gender)

data$What.was.your.grade.in.Calculus.1<-as.factor(data$What.was.your.grade.in.Calculus.1)

data$Did.you.take.any..prior.calculus.or.math.related.coursework.in.High.School.before.enrolling.in.Calculus.1.<-as.factor(data$Did.you.take.any..prior.calculus.or.math.related.coursework.in.High.School.before.enrolling.in.Calculus.1.)

data$Did.you.form.study.groups.with.classmates.for.Calculus.1.<-as.factor(data$Did.you.form.study.groups.with.classmates.for.Calculus.1.)

data$How.frequently.did.you.use.online.resources.to.supplement.your.Calculus.1.studies.<-as.factor(data$How.frequently.did.you.use.online.resources.to.supplement.your.Calculus.1.studies.)

data$Do.you.ever.get.anxious.when.doing.the.test.including.quizzes.midterms.and.finals...<-as.factor(data$Do.you.ever.get.anxious.when.doing.the.test.including.quizzes.midterms.and.finals...)

data$Did.you.use.to.solve.all.recommended.problems.<-as.factor(data$Did.you.use.to.solve.all.recommended.problems.)

data$Did.you.follow.up.with.the.material.or.do.you.usually.study.for.the.test.last.minute.<-as.factor(data$Did.you.follow.up.with.the.material.or.do.you.usually.study.for.the.test.last.minute.)

data$Did.you.usually.attend.office.hour.<-as.factor(data$Did.you.usually.attend.office.hour.)

str(data)

data1<-data
```

```
colnames(data1) <- c("A","B","C","D","E","F","G","H","I","J","K","L")

data2<-data1

# Define the condition where the unusual responses exists

condition <- (data2$D == "A") & (data2$F < 6) & (data2$I == "Often") & (data2$J == "No") & (data2$K == "Last minute Studying")

condition2 <- (data2$D == "A") & (data2$F < 6) & (data2$J == "No") & (data2$K == "Last minute Studying")

condition3 <- (data2$D == "A") & (data2$F < 6) & (data2$K == "Last minute Studying")

# Subset the data frame to remove rows that meet the conditions

data5 <- data2[!condition, ]

data5 <- data2[!condition2, ]

data5 <- data2[!condition3, ]

conditions_to_remove<- c("C-", "D", "F")

data6 <- data5[!(data5$D %in% conditions_to_remove), ]

data6$D <- factor(data6$D)

#EDA

#Plot 1: Pie chart for Grade letter distribution

df <- data6 %>%

  group_by(D) %>%

  count() %>%

  ungroup() %>%

  mutate(perc = `n` / sum(`n`)) %>%
  arrange(perc) %>%
  mutate(labels = scales::percent(perc))
ggplot(df, aes(x = "", y = perc, fill = D)) +
  geom_col(color = "black") +
  coord_polar(theta = "y")+
   geom_text(aes(label = labels),
                                   position = position_stack(vjust =0.5))+
  guides(fill = guide_legend(title = "Grade Level"))+
  labs(title = "Pie chart for Grade letter distribution") +
  theme_void()
#Plot 2: Pie chart for level distribution
df1 <- data6 %>%
  group_by(A) %>%
  count() %>%
  ungroup() %>%
  mutate(perc1 = `n` / sum(`n`)) %>%
  arrange(perc1) %>%
  mutate(labels = scales::percent(perc1))
ggplot(df1, aes(x = "", y = perc1, fill = A)) +
  geom_col(color = "black") +
  coord_polar(theta = "y")+
  geom_text(aes(label = labels),
            position = position_stack(vjust =0.5))+
  guides(fill = guide_legend(title = "Student Level"))+
  labs(title = "Pie chart for Level distribution") +
  theme_void()
#Plot 3: Pie chart for gender distribution
df2 <- data6 %>%
  group_by(C) %>%
  count() %>%
  ungroup() %>%
  mutate(perc2 = `n` / sum(`n`)) %>%
  arrange(perc2) %>%
  mutate(labels = scales::percent(perc2))
ggplot(df2, aes(x = "", y = perc2, fill = C)) +
  geom_col(color = "black") +
  coord_polar(theta = "y")+
  geom_text(aes(label = labels),
            position = position_stack(vjust =0.5))+
  guides(fill = guide_legend(title = "Gender"))+
  labs(title = "Pie chart for Gender distribution") +
  theme_void()
```

```
#Plot 4: Pie chart for prior course work distribution

df3 <- data6 %>%

  group_by(E) %>%

  count() %>%

  ungroup() %>%

  mutate(perc3 = `n` / sum(`n`)) %>%

  arrange(perc3) %>%

  mutate(labels = scales::percent(perc3))

ggplot(df3, aes(x = "", y = perc3, fill = E)) +

  geom_col(color = "black") +

  coord_polar(theta = "y")+

  geom_text(aes(label = labels),

            position = position_stack(vjust =0.5))+

  guides(fill = guide_legend(title = "Response"))+

  labs(title = "Pie chart for taken prior course work response") +

  theme_void()+

  scale_fill_brewer(palette = "Pastel2")

#Plot 5: Pie chart for Study Group variable

df4 <- data6 %>%

  group_by(G) %>%

  count() %>%
  ungroup() %>%

  mutate(perc4 = `n` / sum(`n`)) %>%

  arrange(perc4) %>%

  mutate(labels = scales::percent(perc4))

ggplot(df4, aes(x = "", y = perc4, fill = G)) +

  geom_col(color = "black") +

  coord_polar(theta = "y")+

  geom_text(aes(label = labels),

        position = position_stack(vjust =0.5))+

  guides(fill = guide_legend(title = "Response"))+

  labs(title = "Pie chart for study group formation response") +

  theme_void()+

  scale_fill_brewer(palette = "Accent")

#Plot 6: Pie chart for online resource usage variable

df5 <- data6 %>%

  group_by(H) %>%

  count() %>%

  ungroup() %>%

  mutate(perc5 = `n` / sum(`n`)) %>%

  arrange(perc5) %>%

  mutate(labels = scales::percent(perc5))
```

```
ggplot(df5, aes(x = "", y = perc5, fill = H)) +
  geom_col(color = "black") +
  coord_polar(theta = "y")+
  geom_text(aes(label = labels),
            position = position_stack(vjust =0.5))+
  guides(fill = guide_legend(title = "Response"))+
  labs(title = "Pie chart for usage of online resources response") +
  theme_void()+
  scale_fill_brewer(palette = "Pastel1")
#Plot 7: Pie chart for test anxiety variable
df6 <- data6 %>%
  group_by(I) %>%
  count() %>%
  ungroup() %>%
  mutate(perc6 = `n` / sum(`n`)) %>%
  arrange(perc6) %>%
  mutate(labels = scales::percent(perc6))
ggplot(df6, aes(x = "", y = perc6, fill = I)) +
  geom_col(color = "black") +
  coord_polar(theta = "y")+
  geom_text(aes(label = labels),
            position = position_stack(vjust =0.5))+
  guides(fill = guide_legend(title = "Response"))+
  labs(title = "Pie chart for test anxiety response") +
  theme_void()+
  scale_fill_brewer(palette = "Paired")
#Plot 8: Pie chart for solve recommended problems variable
df7 <- data6 %>%
  group_by(J) %>%
  count() %>%
  ungroup() %>%
  mutate(perc7 = `n` / sum(`n`)) %>%
  arrange(perc7) %>%
  mutate(labels = scales::percent(perc7))
ggplot(df7, aes(x = "", y = perc7, fill = J)) +
  geom_col(color = "black") +
  coord_polar(theta = "y")+
  geom_text(aes(label = labels),
            position = position_stack(vjust =0.5))+
  guides(fill = guide_legend(title = "Response"))+
  labs(title = "Pie chart showing recommended problems solved response") +
  theme_void()+
  scale_fill_brewer(palette = "Set2")
#Plot 9: Pie chart for study method
df8 <- data6 %>%
  group_by(K) %>%
  count() %>%
  ungroup() %>%
  mutate(perc8 = `n` / sum(`n`)) %>%
  arrange(perc8) %>%
  mutate(labels = scales::percent(perc8))
ggplot(df8, aes(x = "", y = perc8, fill = K)) +
  geom_col(color = "black") +
  coord_polar(theta = "y")+
  geom_text(aes(label = labels),
            position = position_stack(vjust =0.5))+
  guides(fill = guide_legend(title = "Response"))+
  labs(title = "Pie chart showing study method distributions") +
  theme_void()+
  scale_fill_brewer(palette = "Set3")
#Plot 10: Pie chart for office hours variable
df9 <- data6 %>%
  group_by(L) %>%
  count() %>%
  ungroup() %>%
  mutate(perc9 = `n` / sum(`n`)) %>%
  arrange(perc9) %>%
  mutate(labels = scales::percent(perc9))
ggplot(df9, aes(x = "", y = perc9, fill = L)) +
  geom_col(color = "black") +
  coord_polar(theta = "y")+
  geom_text(aes(label = labels),
            position = position_stack(vjust =0.5))+
  guides(fill = guide_legend(title = "Response"))+
  labs(title = "Pie chart showing attending office hours response") +
  theme_void()+
  scale_fill_brewer(palette = "Spectral")
#Plot 11: Age Distribution Bar Plot
ggplot(data6, aes(x=B)) +
  geom_bar(fill = "maroon",position= "dodge")+
  labs(title = "Age Distribution Bar Plot")+
  xlab("Age")+
  geom_text(stat='count',aes(label=after_stat(count)),position = position_dodge(width = 1),vjust = -0.5, size = 3)
#Plot 12: Study hours Bar Plot
ggplot(data6, aes(x=F)) +
  geom_bar(fill = "#89CFF0",position= "dodge")+
  labs(title = "Study Hours Distribution Bar Plot")+
  xlab("Study Hours")+
  geom_text(stat='count',aes(label=after_stat(count)),position = position_dodge(width = 1),vjust = -0.5, size = 3)
#one hot encoding
data6<-subset(data6,select = -c(1,2,5,7,8,9,10,12) ) #Remove variables that make the accuracy higher
data6$C <- ifelse(data6$C=="Female",0,1)
data6$K <- ifelse(data6$K=="Follow up - (day to day studying)",1,0)
data6 <- data6[, order(names(data6))]
# Training and testing the data
set.seed(123)
data_obs <- nrow(data6)
training.sample <- sample(data_obs,size=trunc(0.8*data_obs))
training.sample
train<-data6[training.sample,]
test<-data6[-training.sample,]
```

```
# Random forest model
library(randomForest)
classifier_RF = randomForest(x = train[-2],
                             y = train$D,
                             ntree = 50)
y_pred<- predict(classifier_RF, newdata = test[-2])
confusion_mtx <-  confusionMatrix(y_pred, test[, 2])
print(confusion_mtx)
# variable importance
importance(classifier_RF)


# LDA model
library(MASS)
model<- MASS::lda(D ~ ., data = train)
summary(model)
model
predict(model,test)$class
table(test$D)
model$prior
pred_lda<-predict(model,test)$class
confusionMatrix(pred_lda,test$D)
```

```
# SVM model

library(e1071)

svm_model <- svm(D ~ ., data = train, kernel = "linear")

svm_model

summary(svm_model)

svm_model

predict(svm_model,test)

table(test$D)

pred_svm<-predict(svm_model,test)

confusionMatrix(pred_svm,test$D)

# Naive Bayes model

library(e1071)

nb_model <- naiveBayes(D ~ ., data = train)

pred_nb<-predict(nb_model, test)

confusionMatrix(pred_nb, test$D)

# Multinomial model

library(nnet)

multi <- nnet::multinom(D ~., data = train)

pred_mt <- predict(multi,test)

confusionMatrix(pred_mt,test$D)
```