Student Performance Prediction

Data Science Course

Author: Seyedeh Kimia Arfaie Oghani

1. Problem Description

1.1. Introduction

In a university, a significant challenge is maintaining student retention and ensuring successful academic outcomes for all students. With the diverse backgrounds and learning paths that each student has, it is important to prevent strategies to enhance student performance. Early identification of dropping out or facing difficulties regarding this is crucial.

The primary challenge in this project is to develop a predictive model that can accurately identify students who are at risk of dropping out vs those who will graduate based on a range of variables available. These variables include academic trajectory, demographics, social factor and engagement in university activities. The importance of this project is that identifying at-risk students early can enable the university to provide targeted support to improve their retention and academic success.

1.2. Objectives

- **Predictive Modeling:** To develop a robust predictive model that can forecast a student’s likelihood of dropping out or succeeding, utilizing historical data from over 3,500 students, each characterized by 36 attributes. This model will help in understanding the key factors influencing student outcomes at the university.
- **Evaluation and Analysis:** To assess various techniques—including K-Nearest Neighbors, Linear Discriminant Analysis, Logistic Regression, Decision Trees, Random Forests, Neural Networks, Support Vector Machines, and Clustering—and determine which provides the most accurate and practical solution for the university's needs.

1.3. Dataset

The dataset includes records for 3,500 students, with each record comprising 36 attributes that capture a wide array of information such as:

- Demographic details like age, gender, and nationality.
- Academic data including previous qualifications, grades, and course enrollments.
- Socio-economic indicators such as scholarship status and parental qualifications.
- Behavioral aspects including participation in university events and payment of tuition fees.

These attributes provide a comprehensive profile that contributes to the predictive modeling task.

2. Preprocessing

2.1. Initial Data Review and Preprocessing

Preprocessing is a crucial step in any data science task, as it is necessary to prepare the data for developing the predictive model. One of the primary reasons why data preprocessing is necessary is to ensure data quality. Data collected from various sources may contain errors, missing values, inconsistencies, or even variables that wouldn’t serve as an informative feature for our model. In the following paragraphs the preprocessing procedures that were done will be explained:

1. First, looking at the variables the column “Nacionality” was renamed to “Nationality” as it is correct to maintain consistency and avoid potential errors in the future coding steps.
2. Then, the dataset was checked for null values in each column. Fortunately, no missing values were found. Handling missing data is crucial as it ensures the model trains on complete and accurate information.

The data was then inspected for its basic structure and initial statistics to understand the nature of the variable and if they are important for us or not. This step is crucial to ensure that the subsequent data manipulation and analysis steps are built on a clean and reliable dataset.

3. The target variable, initially holding the categories 'Dropout', 'Enrolled', and 'Graduate', was transformed to a numerical format; this is due to the fact that later in the project, due to the correlation analysis done on the data, categorical variables need to convert to numerical formats that enable the computation of correlation coefficients with other numeric features, providing insights into which variables have the most significant relationships with the target.

2.2. Correlation Analysis

Before going further into the next parts, first, it was decided to study how the variables are correlated with the target. First the pearson correlation between each variable and the target were calculated. This provides insights into the linear relationships between the target and each feature. Pearson's Correlation measures the linear relationship between two continuous variables. A Pearson correlation coefficient closer to +1 or -1 indicates a strong positive or negative linear relationship, respectively, while a coefficient around 0 suggests no linear correlation. However, it's important to note that correlation does not imply causation, and non-linear relationships are not captured by Pearson's correlation.

Therefore, Spearman's Correlation was also calculated.

Spearman Correlation: Unlike Pearson, Spearman correlation assesses monotonic relationships by using the rank values of the variables rather than their actual data points. This is particularly useful for dealing with non-linear relationships where the variables increase or decrease together but not necessarily at a constant rate.

The values are evident in the R markdown file which are sorted from highest to lowest values.

The table below shows both the correlations which are between the values of 0.04 and 0.04, indicating a low correlation on both Spearman's and Pearson's measures.

Table 1. Pearson's and Spearman's Correlations between the values of 0.04 and 0..04

Feature	Pearson's Correlation	Spearman's Correlation
Curricular units 1st sem (Credited)	0.03868	0.01441
Educational Special Needs	-0.00251	-0.00355
Father's qualification	-0.00205	0.02353
Inflation Rate	-0.02039	-0.01789
International	-0.00859	-0.00972
Mother's Occupation	-0.00017	0.03977
Mother's qualification	-0.03721	-0.01231
Nationality	-0.02447	-0.00999
Unemployment Rate	0.02265	0.02993

Looking at the correlation values, we see that some of the variables have significantly low correlations in both of the Pearson and Spearman correlations. For instance, *Educational Special Needs*, *Mother's qualification*, *Father's qualification*, *Nationality*, *Unemployment Rate*, *Inflation Rate* and *International*, which were decided to be removed from the variables of the dataset due to their low correlation with the target.

Low correlations indicate that changes in these variables are not associated with changes in the target variable, making them less useful for predicting student outcomes like dropout or graduation. Some variables exhibited differences in their Pearson and Spearman correlation values, suggesting that while they may not have a strong linear relationship with the target, they could still possess a monotonic relationship. This discrepancy was the reason for retaining certain variables for further analysis; such as Curricular unit 1st sem (Credited) which theoretically may have relevance or importance regarding the target. The variable shows potential predictive power in exploratory data analysis or preliminary modeling that suggests a more complex relationship than what linear correlation alone can reveal.

Removing variables with low correlations in both Pearson and Spearman analyses simplifies the model, potentially improving performance and interpretability, and in the end focusing more on relevant predictors

.

2.3. Distribution Analysis

In order to better to underestand the data and how it is distributed, the targets were counted and it was shown that 1106 students were in “Dropout”, 645 in “Enrolled”, and 1749 in “Graduate” class. Figure 1 shows the distribution of the dataset.

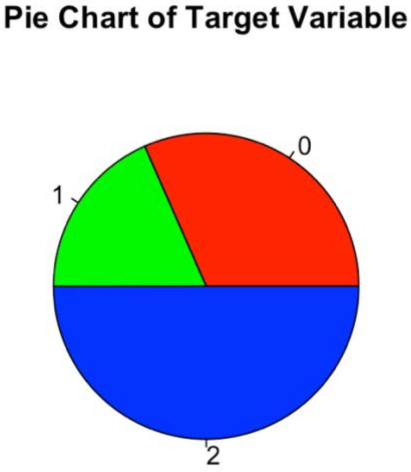

Figure 1 - Pie chart of the Target distribution, 0 indicating “Dropout”, 1: “Enrolled”, and 2: “Graduate”

2.4. Decision to drop “Enrolled” class

The category 'Enrolled' was removed from the analysis. This decision was based on the rationale that the status of 'Enrolled' students is still unresolved (i.e., they could either eventually graduate or drop out), which does not contribute to the predictive modeling of definitive outcomes like 'Dropout' or 'Graduate'. This step ensures that the models focus on outcomes that are certain, enhancing the clarity and effectiveness of the predictive analysis. These students still hae the opportunity to dropout from their program or continue and graduate, therefore their variables wouldn’t give us much informative insight, and their situation is unclear regarding their academic performance. By removing this class now we have two classes of “Dropout”, making up 38% of the dataset, and “Graduate”, making up 62% of the dataset.

2.5. Visualizations

Next, various visualizations were done to understand the distribution and relationship of features with respect to the target variable. For example, distributions of the target variable by gender, age, and marital status were visualized to gain insights into their potential impact on student outcomes.

According to the data there are many more female graduates than male graduates. There appears to be a much larger sample of female students than male students in the data; In female student only 551 out of 1878 were dropedout, while in male student 555 out of 977 were dropedout. From this visualization it is easy to see that males are much more likely to drop out than females, making Gender an important variable in our models.

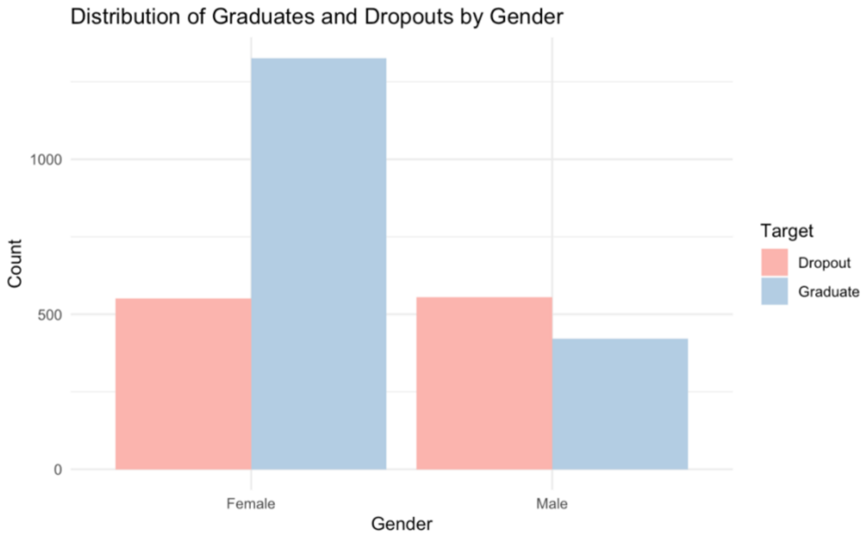

Figure 2 – Distribution of Graduates and Dropouts by Gender

Next, we have age. We see that the majority of students in the sample population are between the ages of 18-21, which is consistent with what we would expect with undergraduate enrollment.

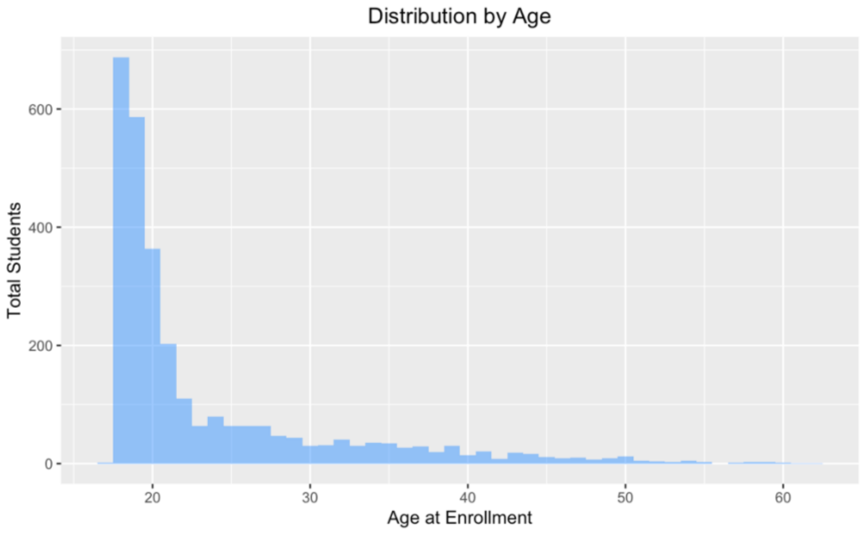

Figure 3 – Distribution of Graduates and Dropouts by Age

Next, we looked at distribution of students by their martial status. We see that there is a significance imbalance regarding the martial status as most of the students are single. We can say that marital status is not likely to be a significant influencial factor on overall student success and not an informative variable for our future models that want to predict student performance. However, as it has relative correlation with the target variable as shown before, it wasn't eliminated.

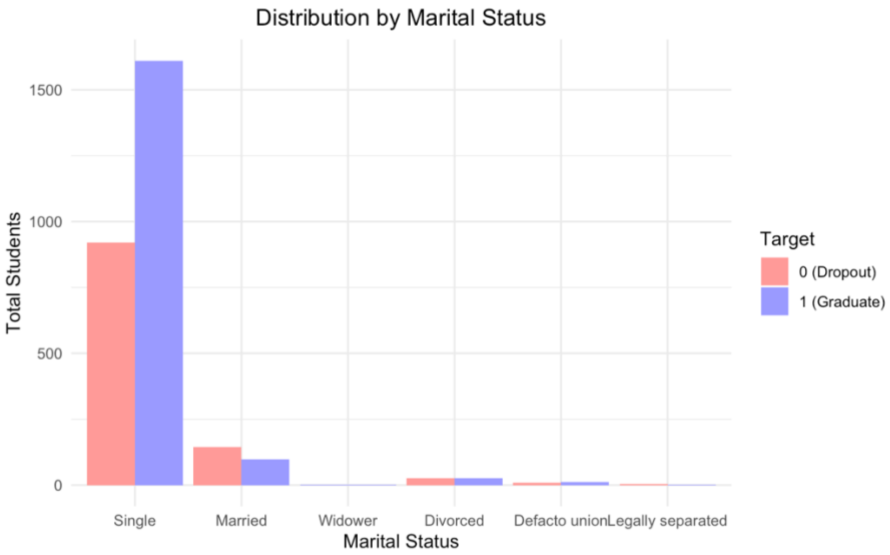

Figure 4 – Distribution of Graduates and Dropouts by Martial Status

3. KNN

The K-Nearest Neighbors (KNN) algorithm is employed to predict student outcomes, specifically distinguishing between 'Dropout' and 'Graduate'. KNN makes predictions based on how close a new data point is to known data points. It calculates the distance between the data point in question and all other points in the training set. The distance can be measured in several ways, the most common being the Euclidean distance. After calculating the distances, the algorithm sorts these values and picks the top 'k' nearest data points. Here, 'k' is a predefined number of neighbors that the algorithm will consider. In classification, KNN assigns a class to the new point based on the most common class among its 'k' nearest neighbors.

This method is particularly sensitive to the scale of the data because it relies on distance calculations to identify the 'closeness' of instances. Given the varied scales of features in our dataset, scaling is a crucial step before applying KNN to ensure that no single attribute disproportionately influences the distance calculations.

1. The data was scaled to normalize the range of independent variables, ensuring that attributes like age and number of curricular activities, which operate on different scales, contribute equally to the distance calculations.
2. The dataset was split into training (75%) and testing (25%) sets using a random seed to ensure reproducibility.
3. Parameter tuning: The KNN model involved tuning the number of neighbors (k) to determine the optimal value for our prediction task. We used cross-validation (10 folds) to estimate the accuracy of the model at different values of k. This method helps in assessing the model's performance more robustly by averaging out the variability in the training data's partitioning. In order to implement this in R, trainControl and train functions from caret package were used.

3.1. Results and Analysis

During the 10 fold cross-vlidation the training accuracy was calculated and k = 9 was found to be the optimal number for the neighbors. The accuracy peaked with k=9 and generally decreased with higher values of k, indicating that smaller neighborhoods provided a better balance of bias and variance. The model's optimal point at k=9 suggests that considering the seven closest neighbors provides the best generalization from training to unseen data. As k increases, the model's accuracy declines, likely due to the inclusion of more distant and potentially less relevant neighbors, thereby increasing the model's bias.

The model then was again created with k=9; by calculating the confusion matrix, the accuracy was calculated to be 85.28%. Below you can see the confusion matrix.

Table 2. Confusion Matrix for KNN method

	Actual	
Predicted	Dropout	Graduate
Dropout	204	23
Graduate	78	408

4. LDA

Linear Discriminant Analysis (LDA) is a classification and dimensionality reduction technique used in statistics and machine learning. It aims to project features onto a lower-dimensional space while maximizing the separation between multiple classes. LDA assumes that different classes generate data based on different Gaussian distributions. LDA is not sensitive to different ranges of variables in a dataset, therefore there is no need to scale the data beforehand.

This method is particularly sensitive to the scale of the data because it relies on distance calculations to identify the 'closeness' of instances. Given the varied scales of features in our dataset, scaling is a crucial step before applying KNN to ensure that no single attribute disproportionately influences the distance calculations.

1. The dataset was split into training (75%) and testing (25%) sets using a random seed to ensure reproducibility.
2. The LDA model is applied to the training data using the lda funciton from the MASS library. This function performs LDA considering all the predictors in the dataset.

4.1. Results and Analysis

The table shows the confusion matrix resulted from testing the LDA model on the test set. The accuracy was calculated to be 90.46% which is higher than the KNN method.

Table 3. Confusion Matrix for LDA method

	Actual	
Predicted	Dropout	Graduate
Dropout	220	12
Graduate	56	425

5. Logistic Regression

Logistic Regression is a predictive analysis used primarily for binary classification problems. It uses a logistic function to estimate the probability that a given input belongs to the default class, based on the logistic function's ability to model a binary outcome as a function of the predictor variables. In our context, the logistic regression model aims to predict whether a student will graduate or dropout, based on various explanatory variables.

The logistic function outputs a value between 0 and 1, which can be interpreted as the probability of the dependent variable being a 'success' or 'positive' instance A threshold (here 0.5) is used to decide the class assignment. If the predicted probability is greater than 0.5, the instance is predicted to be in the positive class; otherwise, it is placed in the negative class.

1. Data Splitting: The dataset was split into training (80%) and testing (20%) sets using a random seed to ensure reproducibility.
2. Model Fitting: Then, we used glm() with a binomial family to fit a logistic regression model. This method models the probability of graduation versus dropping out as a function of the predictors in our dataset.
3. Stepwise selection: Employing the step() function with both directions (Forward selection and Backward Elimination), you performed a stepwise selection to refine the model by including only

statistically significant variables. This step helps in reducing the complexity of the model, potentially enhancing generalizability and avoiding overfitting. In each step, a variable is considered for addition to or subtraction from the set of variables based on some predefined criterion, here is Akaike's Information Criterion (AIC).

4. Model Evaluation: After the model fitting, we evaluated its performance using manually implemented cross-validation, with the train function from caret library, feeding to it the outcome of our stepwise selection, and then calculated the accuracy using the test data.

5.1. Results and Analysis

The result shows that the final model include variables such as "Application Mode", "Daytime/Evening Attendance", "Debtor", "Tuition Fees up to date", "Scholarship Holder", "Curricular units 1st sem (credited)", "Curricular units 1st sem (approved)", "Curricular units 2nd sem (enrolled)", "Curricular units 2nd sem (approved)" and etc, as they have a low P value, indicating their significance in the model.

The test accuracy obtained (~92%) indicates a high level of model performance, meaning the model is effective at predicting the target based on the input variables. Below the confusion matrix is shown.

Table 4. Confusion Matrix for Logistic Regression method

	Actual	
Predicted	Dropout	Graduate
Dropout	237	19
Graduate	39	418

6. Decision Trees

Decision Trees are a type of supervised learning algorithm that are predominantly used in classification problems and operate by partitioning the dataset into subsets based on different conditions. This process essentially splits the data into branches, leading to a tree-like model of decisions

1. The dataset was split into training (75%) and testing (25%) sets using a random seed to ensure reproducibility.
2. The tree was built with the rpart library in R. Then predictions were made on the test set.
3. The model vas evaluated by creating the cofusion matrix and calculating the accuracy which was 89.1%.

6.1. Results and Analysis

Figure 5 shows the decision tree. We can see that "Curricular units 2nd sem (Approved)" and "Tuition Fees up to date" are the two variables chosen to build the tree indicating that they are significant. We see in the details of the tree which states if the "Curricular units 2nd sem (Approved)" is less than 3.5 then the tree will lead to Dropout, if not, it will lead to new branches. As we can see the tree is quite simple itself, and isn't complex, therefore it was decided not to go into pruning as a means to improve the performance. Table 5 also shows the confusion matrix for the tree which shows 89.1 % accuracy.

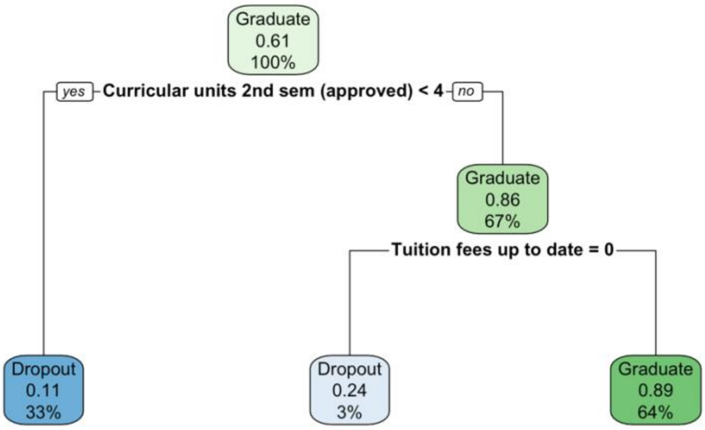

Figure 5 – Decision tree

Table 5. Confusion Matrix for Decision tree method

	Actual	
Predicted	Dropout	Graduate
Dropout	221	23
Graduate	55	414

7. Random Forests

Random Forest is a method used for classification and regression. It operates by constructing a multitude of decision trees during training and outputs the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees. Random Forests enhance the perfomance of decision trees by reducing the variance.

1. Cross-validation was used to assess the generalizability of the model. By dividing the data into 'k' folds and iteratively training on 'k-1' folds while testing on the remaining fold, we mitigate the risk of overfitting and get a more reliable estimate of the model's performance on unseen data.
2. TrainControl and train from caret library were used as we wanted to impleent cross validation.
3. mtry is a hyperparameter that needs tuning in order to get the optimal result from the method. “mtry” refers to the number of variables randomly sampled as candidates at each split when building the trees and we have to determine this value to build the model.
4. Cross validation results show that mtry = 8, gives the best accuracy.

7.1. Results and Analysis

The table shows the confusion matrix resulted from testing the Random forest with the optimal mtry = 8, on the test set. The accuracy was calculated to be 91.44% which is an improvement compared to the Decision tree model.

Table 6. Confusion Matrix for Random Forest method

	Actual	
Predicted	Dropout	Graduate
Dropout	230	15
Graduate	46	422

8. ANN

Artificial Neural Networks (ANN) are computational models inspired by the human brain, consisting of interconnected units called neurons. In classification tasks, ANNs aim to learn patterns from the input data to classify instances into predefined categories.

1. The dataset was split into training (75%) and testing (25%) sets using a random seed to ensure reproducibility.
2. We have the hyperparameter of size in ANN, as it is the number of hidden layers in the neural network. We also have decay which is the regularization parameter to prevent overfitting, which was chosen as 0.1. There is also max iteration, maximum number of iterations during training, which we chose it as 500.
3. The size hyperparameter was tunes, as we trained and tested multiple ANNs with various values for size.

8.1. Results and Analysis

The results display the accuracy for each number of neurons tested. The size with the highest accuracy indicates the optimal number of neurons for the hidden layer in the ANN model. We see that the size = 6 gives the best test accuracy which is 95%.

9. SVM

Support Vector Machine (SVM) is an algorithm used for classification tasks. It works by finding the hyperplane that best divides a dataset into classes. The hyperplane is chosen to maximize the margin, which is the distance between the hyperplane and the nearest data points from each class. These nearest points are known as support vectors. The SVM can handle linear and non-linear classification by using different kernel functions.

1. The dataset was split into training (75%) and testing (25%) sets using a random seed to ensure reproducibility.
2. Tune.svm function was used in order to tune the cost parameter. The results show that cost = 10 was the optimal cost for the dataset.

9.1. Results and Analysis

The table shows the confusion matrix resulted from testing the SVM model (optimal with cost = 10) on the test set. The accuracy was calculated to be 91.03%. Other kernels such as polynomial was also teste, however they didn't show satisfactory results, which indicate that the data is linearly seperable and there is no need to use other kernels.

Table 7. Confusion Matrix for SVM method

	Actual	
Predicted	Dropout	Graduate
Dropout	238	20
Graduate	44	412

10. Clustering Techniques - Kmeans

K-means clustering is an unsupervised learning algorithm used for partitioning a dataset into k distinct, non-overlapping clusters. The main idea is to define k centroids, one for each cluster, and assign each data point to the nearest centroid, thereby forming clusters. The algorithm iterates to optimize the positions of the centroids by minimizing the distance between data points and their respective centroids.

10.1. Results and Analysis

The accuracy of the model was calculated to be 79.41%.

11. Conclusion

In this project, multiple machine learning techniques were applied to predict student outcomes, specifically distinguishing between students who drop out and those who graduate. The following methods were evaluated: K-Nearest Neighbors (KNN), Linear Discriminant Analysis (LDA), Logistic Regression, Decision Trees, Random Forests, Artificial Neural Networks (ANN), Support Vector Machines (SVM), and K-Means Clustering. Below is a summary and comparison of their performances:

	Accuracy	**Analysis**
KNN	85.28%	KNN provided a good baseline accuracy. However, its performance tends to decrease as the number of neighbors increases, indicating that KNN is sensitive to the choice of 'k'. The optimal 'k' was found to be 9. KNN is simple and intuitive but can be computationally expensive for large datasets.
LDA	90.46%	LDA performed better than KNN, likely due to its ability to handle linear separability effectively. It maximizes the separation between the classes, making it a robust choice for this problem. It is also computationally efficient.
Logistic Regression	92%	Logistic Regression showed strong performance, with a clear identification of significant predictors such as “Application Mode”, “Debtor”, “Tuition Fees up to date”, and various curricular unit variables. The stepwise selection helped in refining the model, making it both accurate and interpretable.
Decision Trees	89%	Decision Trees provided a clear visual representation of the decision-making process and identified key variables like “Curricular units 2nd sem (Approved)” and “Tuition Fees up to date”. While the accuracy was good, it was slightly lower than LDA and Logistic Regression.
Random Forests	91.44%	Random Forests improved upon the Decision Trees by reducing variance through the ensemble method. It effectively identified “mtry = 8” as the optimal number of variables

		at each split. This method provided a balance between interpretability and performance.
ANN	95%	ANN achieved the highest accuracy, indicating its strong ability to model complex relationships in the data. The tuning of the 'size' parameter (number of hidden neurons) was crucial, with the best performance observed at size = 6. However, ANNs are less interpretable compared to simpler models like Logistic Regression.
SVM	91%	SVM performed well with the linear kernel, indicating that the data is likely linearly separable. The optimal cost parameter was identified as 10. Despite its high accuracy, SVMs can be computationally intensive, especially with large datasets.
Kmeans Clustering	79.5%	K-Means, being an unsupervised method, was less accurate than the supervised learning methods.

Based on the accuracy and the overall performance, the Artificial **Neural Networks (ANN) model** is identified as the best performing model with an **accuracy of 95%.** The high accuracy suggests that ANN effectively captures the complex relationships between the features and the target variable. ANNs are flexible and can model non-linear relationships, which is beneficial given the diverse and potentially complex relationships in the dataset. However, it is important to note that ANN models can be more challenging to interpret and require more computational resources compared to simpler models.

While ANN provides the best accuracy, it's crucial to balance performance with interpretability and computational efficiency. For practical deployment, a model like Logistic Regression or Random Forests might be preferred due to their high accuracy, interpretability, and lower computational requirements. However, now in this project since the highest accuracy is the primary goal and interpretability is less critical, ANN is the recommended choice and our final desicion.