Machine Learning Based Predictions and Predictors for Student Success

By: Allison A., Nat A., David H., Matthew S.

Abstract

Student success is a vital factor in many facets of life; education impacts an individual's potential income, long term well-being, and their contribution to society. It is therefore paramount to understand the factors which contribute to student success (or the lack thereof). Our study seeks to implement Machine Learning techniques to predict the performance of students based on a number of social and numerical parameters, and identify those factors which most contribute to student success. We will implement four machine learning algorithms: Support Vector Machines (SVM), K-Nearest Neighbors (KNN), Linear Discriminant Analysis (LDA), and Random Forests (RF) on a dataset of Portuguese students whose grades are recorded for their Math and Portuguese classes. The previous study our work was based on guided us to consider student success as a function of earlier grades and social factors, then separately as a function of exclusively social factors (Cortez and Silva). We will discover that the Random Forest algorithm outperforms our other models in most cases, and also presents interesting information regarding the relative importance of our variables. Further, all of our models were able to outperform a baseline dummy predictor, indicating it is possible to predict student performance based on the collected variables. We conclude that while earlier grades are a significant indicator of future grades, there are other social factors with meaningful impact on student success.

Introduction

The goal of education is to provide the resources and motivation for students to develop and learn. The question that we wanted to ask was if it was possible to use machine learning to

predict a students' final grade and additionally how well the models could perform, having either only social and demographic features versus having all the features available.The data used in this paper contains records from two Portuguese schools, datasets for two core subjects: Mathematics and a Portuguese language course. For each subject, the dataset contains largely categorical features which are social or demographic in nature. Some of the features include the students' school, sex, address type (urban vs rural), parental status, etc. Additionally, there are attributes that are numerical in nature such as the number of absences and grades for each subject. Grades for the students were recorded across three periods and the final period grade was used to determine whether the student passed or failed the course. We extracted the target values and converted them to a binary designation of "Pass" and "Fail" on the basis that a grade greater than or equal to 10 was passing. The two datasets then contained only the features which would be used for training and testing. The features primarily consisted of string-based values that required preprocessing to convert them into numerical values. Since our dataset consisted primarily of categorical data along with numerical data, we applied standard score normalization to ensure values were consistent. Finally, the remaining datasets were divided into four sets; two datasets contained math scores and two contained scores for Portuguese. Of the two containing math scores, one featured all parameters, including G1 and G2 (grades from earlier in the year) while the second included only social factors (everything except G1 and G2). The two datasets for Portuguese were distinguished in the same fashion.

Methodology

Baseline

For any machine learning model it is vitally important to include a baseline system for comparison. We chose a baseline classifier that makes uniformly random predictions. Since we

are looking to perform a binary classification, the uniform process mimics if we were to make predictions based on flipping a coin, which is not likely to be an effective classification method.

Support Vector Machines

A Support Vector Machine seeks to find a hyperplane in n-dimensional space (where n is the number of features) that separates two (or sometimes more) classes. While many hyperplanes exist to separate the classes, SVM seeks to maximize the space between the classes (the Maximal Margin Classifier), allowing for some leeway for the classification of future data points. The regularization parameter C affects the flexibility of the margin, allowing for more or less space between the classes and the separating hyperplane. The simplest example of a SVM is a separating line (where the boundary can be represented as two parallel lines some distance away from the line). SVM offers a lot of flexibility for non-linear data. For example, using a polynomial or radial kernel allows for a non-linear boundary. SVM is capable of working in higher dimensions, but the separating hyperplane is difficult if not impossible to imagine.

K-Nearest Neighbors

K-Nearest Neighbors assumes that like things are close together. KNN is a supervised machine learning algorithm that takes a labeled set of data and seeks to classify new data points based on their proximity to preexisting points (whose class is already known). Usually, this proximity is determined by Euclidean distance, which works well in n-dimensional space. K-Nearest Neighbors is so named because the algorithm seeks a variable (k) number of neighbors from which to classify new data. Generally, the appropriate k is unknown and cross-validation methods are implemented to experimentally determine the best k for a particular dataset. KNN is particularly apt when data is nonlinear and is relatively easy to understand, but is sensitive to noisy data.

Linear Discriminant Analysis

Linear Discriminant Analysis (LDA) is a classification method used in statistical learning. LDA aims to maximize the separability between classes so the most accurate predictions can be made. It does this by projecting the data points onto an axis in a way that maximizes the distance between the means of the classes while minimizing the variability of the classes. Doing this reduces the dimensionality of the data while still taking the information from all the predictors into account. LDA assumes that each class has a common variance and covariance, and that each predictor has normally distributed density. It is a popular model to use when there are more than two classes, and it works when there is not enough data to accurately estimate variances.

Random Forest

Random Forest classification is an ensemble classification model that generates multiple decision trees where each tree is generated on a subset of the dataset and predictors, a process known as "bagging". A classification prediction is made by applying the sample data to each tree in the random forest and aggregating the results in a majority vote system. The random subset selection ensures the models generated have an increased variability, covering a wider range of the predictors. The results from all the trees in the model are averaged together to produce an accurate prediction. The random forest classifier is controlled primarily through various hyper-parameters which shape the way the forest is constructed. The most noteworthy parameters are the number of estimators, the max depth and max features. The number of estimators determines the number of trees in the forest. Too many estimators could cause it to take a long time to train the model with either no increase in performance or possibly a decrease in accuracy. The max depth determines how large each tree will be. Trees that are too large could

result in a model that fits the training set too much and doesn't perform well on the test set. The number of predictors chosen to build the tree is determined by the max features parameter. Typically this parameter is based on the square root of the number of predictors. This helps to ensure the number of predictors is a subset of the total number of predictors.

Data Analysis

Our goal is to predict a student's school performance (G3: pass or fail) based on a number of categorical (eg. parents' education, internet access) and quantitative features (eg. number of absences, prior grades). To that end, our study will implement four classification models: Support Vector Machines (SVM), Linear Discriminant Analysis (LDA), K-Nearest Neighbors (KNN), and Random Forest Classifiers (RF). The results of these models will be compared with each other and against a uniform dummy predictor, which will predict that roughly half of the students will pass and the other half will fail. Since G1 (grades from early in the school year) and G2 (grades from the middle of the school year) vastly outperformed all other predictors, we created two datasets, one including G1 and G2, and one excluding them, to see if we could predict G3 based exclusively on social factors.

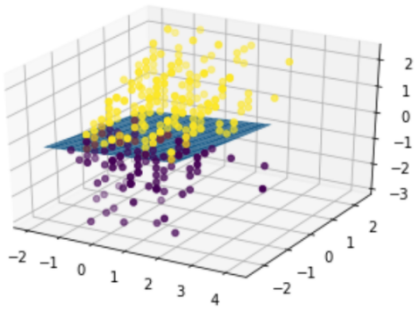
All models are summarized by a classification report and a confusion matrix (found in the Tables and Figures section), and their success will largely be determined based on their weighted average f1-score (where a score closer to 1 indicates a better model). We performed cross-validation on all models that included G1/G2 (including the dummy predictor). The results of cross-validation (which can be seen in Figure 11) suggest our classification reports were well within their expected range, though KNN and LDA-Math display relatively higher variation, indicating these models were more sensitive to the randomness inherent in the train/test split. Every model was limited by the dataset's relatively few cases of failing grades in Portuguese. Of

649 Portuguese students, 100 received a failing grade, amounting to 15.41% (compared to 130 out of 395 math students receiving a failing grade, 32.91%). This imbalance made the models highly sensitive to the number of instances with “Portuguese, G3: fail” in the training set.

To further improve on our models, we suggest future studies endeavor to find more balanced datasets, experiment with stratifying their train/test splits, and implement other models such as regression or multi-level classification.

Support Vector Machines

Since Support Vector Machines depend largely on their hyperparameters, we tested a range of C values, gamma values, as well as three kernels. For the dataset that included G1 and G2, GridSearchCV identified {'C': 100, 'gamma': 0.001, 'kernel': 'rbf'} for the math portion of the data, and {'C': 10, 'gamma': 0.01, 'kernel': 'rbf'} for the Portuguese portion. For the social data (with predictors G1 and G2 removed), GridSearchCV identified {'C': 0.1, 'gamma': 1, 'kernel': 'linear'} for the math portion and {'C': 100, 'gamma': 0.001, 'kernel': 'rbf'} for the Portuguese portion. All these models were evaluated based on their f1-scores, whose weighted averages ranged from .72 to .89. A simpler model (kernel = 'linear', C = 5), whose dimensions were reduced to G1, G2, and age resulted in a f1-score of .91 (see Graph 1). Compared with the dummy predictor (with f1-scores of .49 for math and .52 for Portuguese with all predictors present; scores of .46 for math and .5 for Portuguese with only social predictors), the SVC models performed much better. The models with all predictors saw an improvement of .40 for math and .35 for Portuguese, and the models with only social predictors saw improvements of .23 for math and .25 for Portuguese. The SVC universally performed better than or equal to the KNN model, less than or as well as the RF model, and was comparable to LDA (sometimes better, sometimes worse).



Separating plane for dimensionally-reduced model.

Classification Report for Linear SVC (age/G1/G2)

	precision	recall	f1-score	support
Fail	0.87	0.83	0.85	24
Pass	0.93	0.95	0.94	55
accuracy			0.91	79
macro avg	0.90	0.89	0.89	79
weighted avg	0.91	0.91	0.91	79

Graph 1. 3D plot and Classification Report for the dimensionally-reduced model.

Again, we find a noteworthy difference between the models including G1 and G2 and those that don't, suggesting much of the predictive power of our models rely on these predictors. The error in the social model is largely attributable to this discrepancy; further error can be explained by the randomness inherent in train_test_split (indeed, changing the value of random_state caused the f1-scores to vary by an absolute difference of as much as 0.05).

K-Nearest Neighbors

We evaluated K-Nearest Neighbors' ability to predict whether a student will pass or fail in a math or Portuguese class, using a data set with both previous grades and social data, and a data set with only social data. We used confusion matrices, F1 scores, and cross-validation scores to evaluate its performance. Compared to the dummy classification, there was an average of a 27.75% increase in mean cross-validation scores; total average of 80.25% versus the dummy's 52.5%.

Other systems evaluated were Support Vector Machine (SVM), Linear Discriminant Analysis (LDA), and Random Forest Classifier. Compared to SVM, KNN saw a 1.50% average decrease in mean cross-validation scores. Compared to LDA, the decrease was 1.25%, and

compared to Random Forest it was 3%. KNN had the lowest performance of the four systems evaluated, but still performed better than random guessing.

KNN's highest errors for this dataset were in correctly predicting a failing grade. Errors increased when previous grades were removed from the dataset and the model was made to use only social parameters to make its predictions.

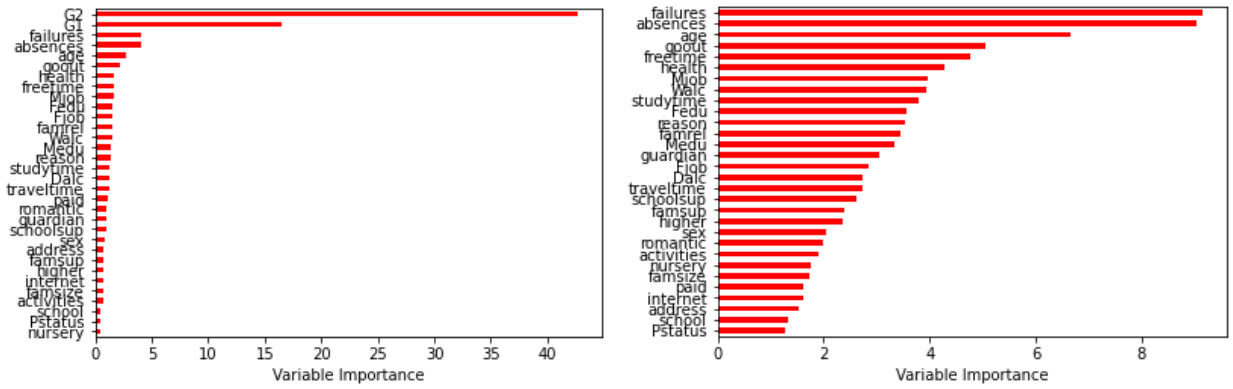
Linear Discriminant Analysis

LDA outperformed the baseline dummy classifier by a substantial margin on all data sets, including and excluding previous grades (see Figures 7 and 8). However, it was evident that the LDA model had more predictive accuracy when previous grades were included. For the all-inclusive data sets, the LDA weighted f1-score was .90 for Math and .86 for Portuguese (Table 3); an effective contrast to the dummy classifier's weighted f1-scores of .49 for Math and .52 for Porgutuese (Table 5). As for the social data sets, LDA produced a weighted f1-score of .68 for Math and a score of 0.82 for Portuguese (Table 3). The weighted f1-scores of 0.46 and 0.50 were given by the social dummy classifier for Math and Portuguese, respectively (Table 5). Because our datasets had such a large number of predictors, we attempted to implement feature selection using Forward Stepwise Selection on the LDA models. Unfortunately, the weighted f1-scores came out worse after applying the feature selection than they were originally–a low of 0.59 and a high of 0.74–so we opted to leave it out of the final LDA models (Table 6). We hypothesized that because LDA already performs dimensionality reduction automatically, reducing dimensions even further caused the LDA model to degrade. At the end, Cross Validation K-Fold was performed on all the models (except the models with Forward Stepwise Selection) with a k value of 10. For each LDA model, the mean cross validation score supported the corresponding results when LDA was applied to that data set.

Random Forest

For the random forest classifier, the hyper-parameters were tested on a range of values for each dataset. The first parameter tested was n_estimators, which determines the number of trees constructed for the forest. For the Math and Portuguese dataset, we found the best performing value was 100 estimators with a weighted f1-score ~.91 for math and ~.90 for Portuguese. The next parameter tested was max_depth, which determines the maximum number of levels each tree can contain. This helps prevent the trees from overfitting to the training data. We tested this parameter on a range of values from 1 through 15 and found that both datasets perform best with a max depth of 9 with an average weighted f1 score of .91 and .89.

Once the parameters were finalized, we found that the random forest classifier performed well on both the Math and Portuguese data set containing all of the predictors. For the Math data set we were able to achieve a weighted f-1 score of .91 and .89 for the Portuguese (Table 4). This was nearly a +.40 weighted f1 score difference when compared to the baseline classifier. The model was then fit and tested on the data sets with only the social/demographic features and the performance decreased to .75 for Math and .78 for Portuguese (Table 4). This was still an improvement over the baseline of +.25 for the weighted f1 score. After analyzing the results the variable importance provided a nice visual representation for which predictors had the most impact on the model. Data summarizing variable importance is summarized in Graph 2.



Graph 2. Variable Importance including G1/G2 (left) and without (right).

The results from the variable importance further exemplify the impact of having previous student performance scores for making accurate predictions. In future studies, we would like to explore how the model would have performed by using subset selection on variable importance to see if we could improve our results.

Conclusion

We found we are able to predict the success of a student's future grade using the given predictors in the data sets with relative accuracy. While predictive accuracy is highly correlated with previous grade scores, it is still possible to get better-than-random predictions using only social and demographic information as well. The Random Forest model performed the best out of our four classifiers, likely due to its use of variable importance and its ability to handle a large amount of predictors. Two difficulties we encountered were the large number of predictors relative to the sample sizes, and an imbalance in the predicted classes in the Portuguese data set. To expand upon our data analysis, future studies could attempt to predict the final grade (G3) scores using regression instead of classification. Additionally, feature selection could be applied to other models as well to see if it improves their performance.

References

P. Cortez and A. Silva. “Using Data Mining to Predict Secondary School Student Performance.” *In A. Brito and J. Teixeira Eds., Proceedings of 5th FUture BUsiness TEChnology Conference (FUBUTEC 2008)* pp. 5-12, Porto, Portugal, April, 2008, EUROSIS, ISBN 978-9077381-39-7, https://archive.ics.uci.edu/ml/datasets/Student+Performance. Accessed 27 Apr. 2021.

Appendix

David H. was our team leader, and was in charge of editing the final report. His classification model was SVM, which he coded and analyzed in the report. He also wrote the report’s methodology.

Matt S. set up our powerpoint slides and written documents. He was also in charge of the preprocessing in the code, and wrote and analyzed the Dummy Classifier and Random Forest Classification. He also wrote the paper’s introduction.

Nat A. was in charge of editing the powerpoint. He also wrote and analyzed the KNN model, and he wrote the report’s appendix.

Allison A. was in charge of editing the Colab code. She also wrote and analyzed the LDA model, and she wrote the report’s abstract and conclusion.

Tables and Figures

Figure 1. Confusion Matrices for SVM, G1/G2 included.

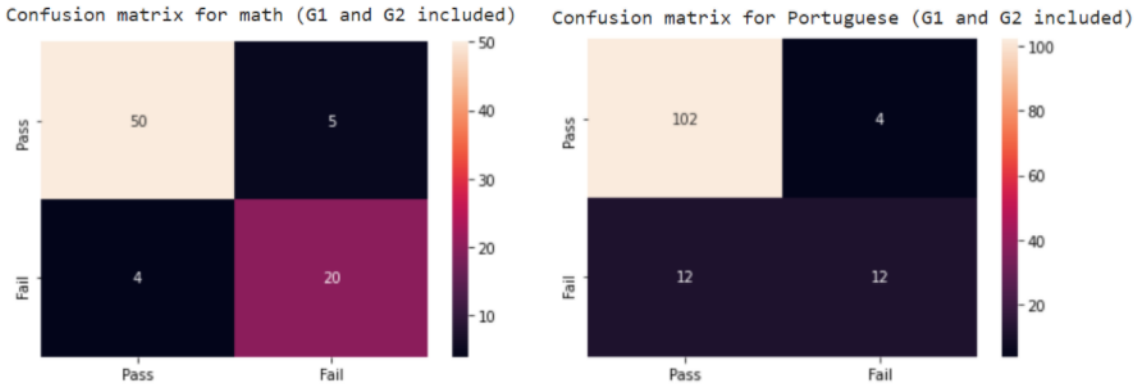


Figure 2. Confusion Matrices for SVM, G1/G2 excluded.

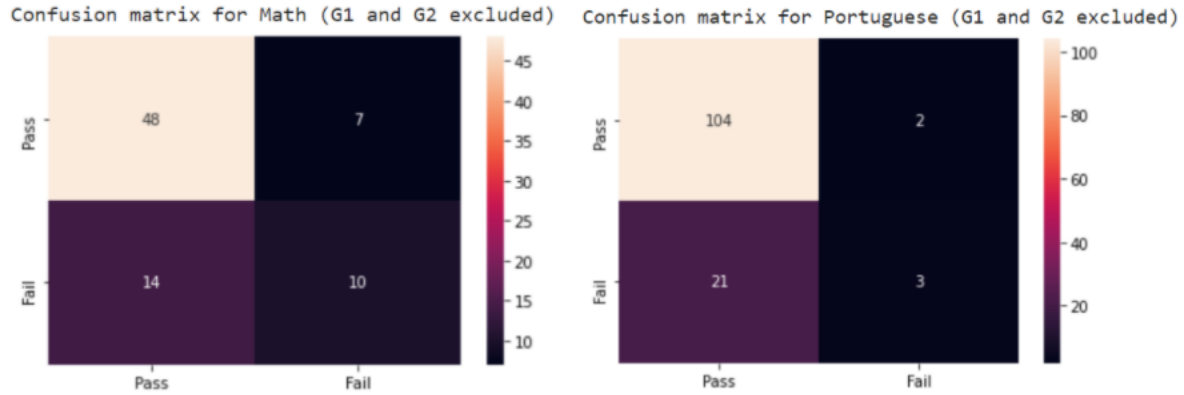


Table 1. Classification Reports for SVM.

Classification Report, Math Scores (G1, G2 included)

	precision	recall	f1-score	support
Fail	0.80	0.83	0.82	24
Pass	0.93	0.91	0.92	55
accuracy			0.89	79
macro avg	0.86	0.87	0.87	79
weighted avg	0.89	0.89	0.89	79

Classification Report, Math Scores (G1, G2 NOT included)

	precision	recall	f1-score	support
Fail	0.59	0.42	0.49	24
Pass	0.77	0.87	0.82	55
accuracy			0.73	79
macro avg	0.68	0.64	0.65	79
weighted avg	0.72	0.73	0.72	79

Classification Report, Portuguese Scores (G1, G2 included)

	precision	recall	f1-score	support
Fail	0.75	0.50	0.60	24
Pass	0.89	0.96	0.93	106
accuracy			0.88	130
macro avg	0.82	0.73	0.76	130
weighted avg	0.87	0.88	0.87	130

Classification Report, Portuguese Scores (G1, G2 NOT included)

	precision	recall	f1-score	support
Fail	0.60	0.12	0.21	24
Pass	0.83	0.98	0.90	106
accuracy			0.82	130
macro avg	0.72	0.55	0.55	130
weighted avg	0.79	0.82	0.77	130

Figure 3. Confusion Matrices for KNN, G1/G2 included.

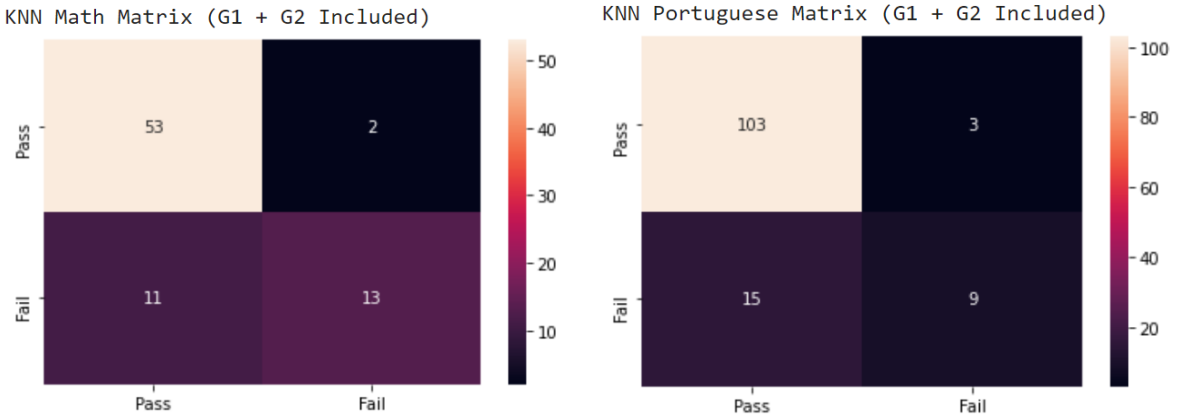


Figure 4. Confusion Matrices for KNN, G1/G2 excluded.

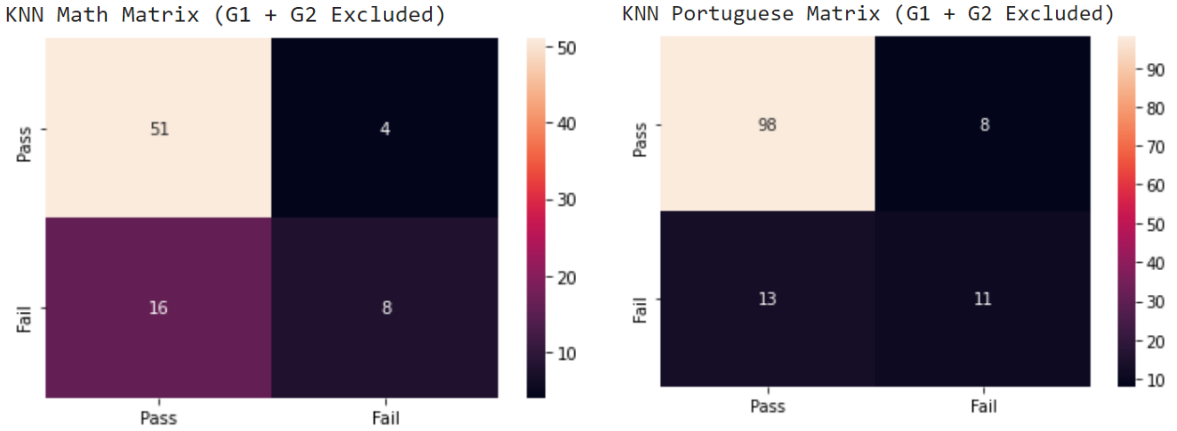


Table 2. Classification Reports for KNN.

KNN Math Classification Report (All)

	precision	recall	f1-score	support
Fail	0.87	0.54	0.67	24
Pass	0.83	0.96	0.89	55
accuracy			0.84	79
macro avg	0.85	0.75	0.78	79
weighted avg	0.84	0.84	0.82	79

KNN Math Classification Report (Social)

	precision	recall	f1-score	support
Fail	0.67	0.33	0.44	24
Pass	0.76	0.93	0.84	55
accuracy			0.75	79
macro avg	0.71	0.63	0.64	79
weighted avg	0.73	0.75	0.72	79

KNN Port. Classification Report (All)

	precision	recall	f1-score	support
Fail	0.75	0.38	0.50	24
Pass	0.87	0.97	0.92	106
accuracy			0.86	130
macro avg	0.81	0.67	0.71	130
weighted avg	0.85	0.86	0.84	130

KNN Port. Classification Report (Social)

	precision	recall	f1-score	support
Fail	0.58	0.46	0.51	24
Pass	0.88	0.92	0.90	106
accuracy			0.84	130
macro avg	0.73	0.69	0.71	130
weighted avg	0.83	0.84	0.83	130

Figure 5. ROC curves for KNN, G1/G2 included.

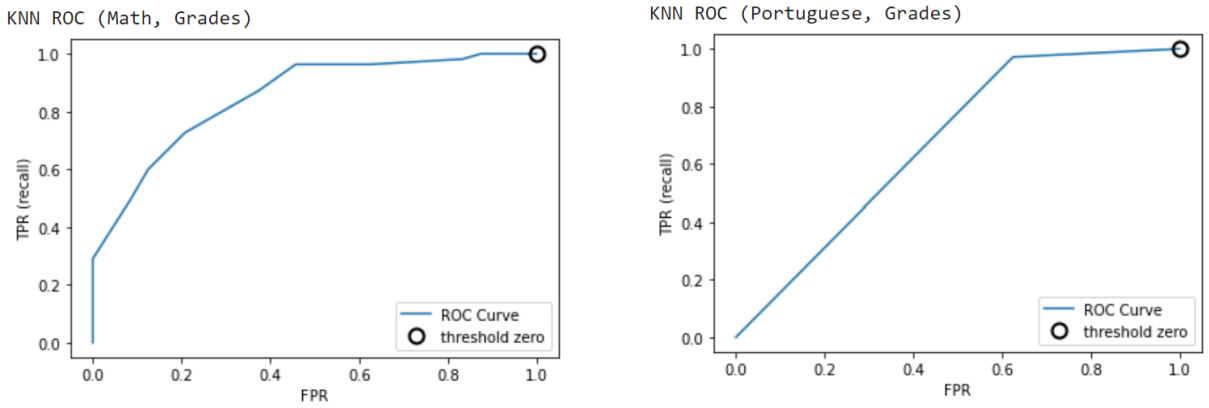


Figure 6. ROC curves for KNN, G1/G2 excluded.

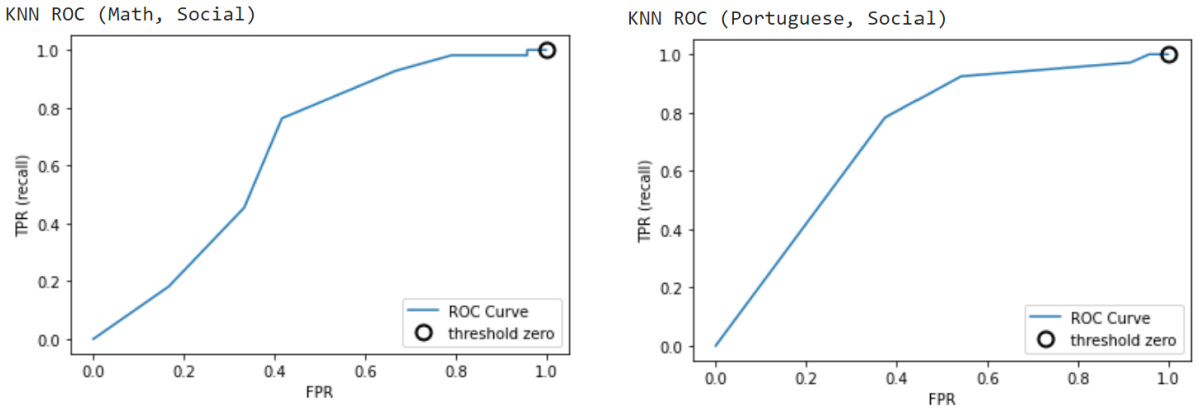


Figure 7. Confusion Matrices for LDA, G1/G2 included.

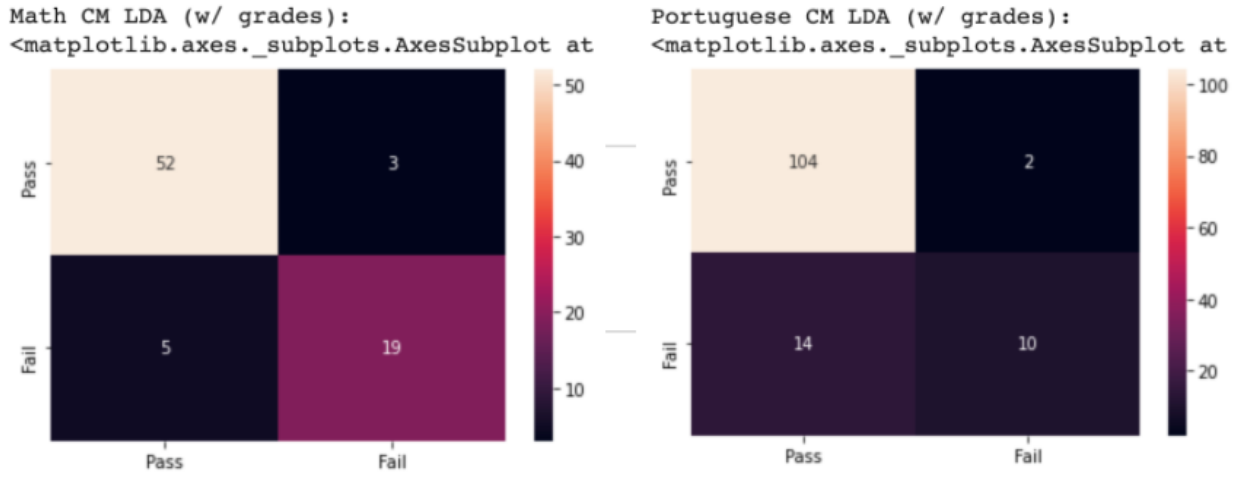


Figure 8. Confusion Matrices for LDA, G1/G2 excluded.

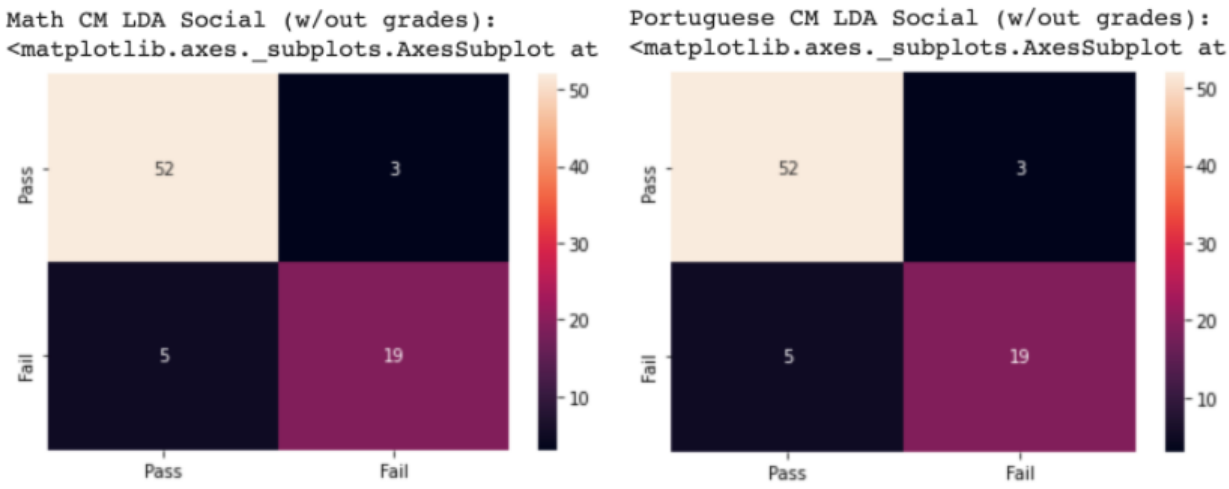


Table 3. Classification Reports for LDA.

LDA Classification Report, Math Scores (w/ grades)

	precision	recall	f1-score	support
Fail	0.86	0.79	0.83	24
Pass	0.91	0.95	0.93	55
accuracy			0.90	79
macro avg	0.89	0.87	0.88	79
weighted avg	0.90	0.90	0.90	79

LDA Classification Report, Math Scores Social (w/out grades)

	precision	recall	f1-score	support
Fail	0.48	0.46	0.47	24
Pass	0.77	0.78	0.77	55
accuracy			0.68	79
macro avg	0.62	0.62	0.62	79
weighted avg	0.68	0.68	0.68	79

LDA Classification Report, Portuguese Scores (w/ grades)

	precision	recall	f1-score	support
Fail	0.83	0.42	0.56	24
Pass	0.88	0.98	0.93	106
accuracy			0.88	130
macro avg	0.86	0.70	0.74	130
weighted avg	0.87	0.88	0.86	130

LDA Classification Report, Por Scores Social (w/out grades)

	precision	recall	f1-score	support
Fail	0.86	0.25	0.39	24
Pass	0.85	0.99	0.92	106
accuracy			0.85	130
macro avg	0.86	0.62	0.65	130
weighted avg	0.85	0.85	0.82	130

Figure 9. Confusion Matrices for RF, G1/G2 included.

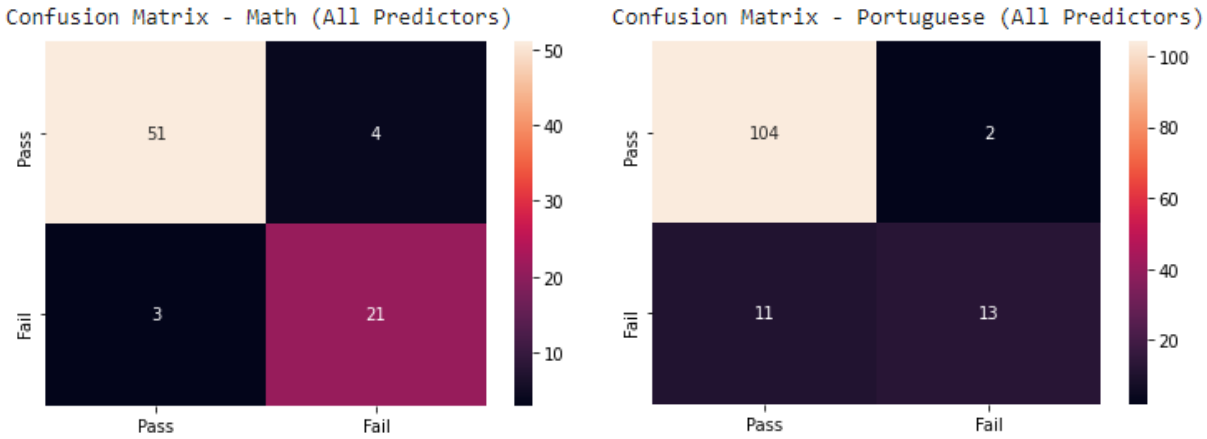


Figure 10. Confusion Matrices for RF, G1/G2 excluded.

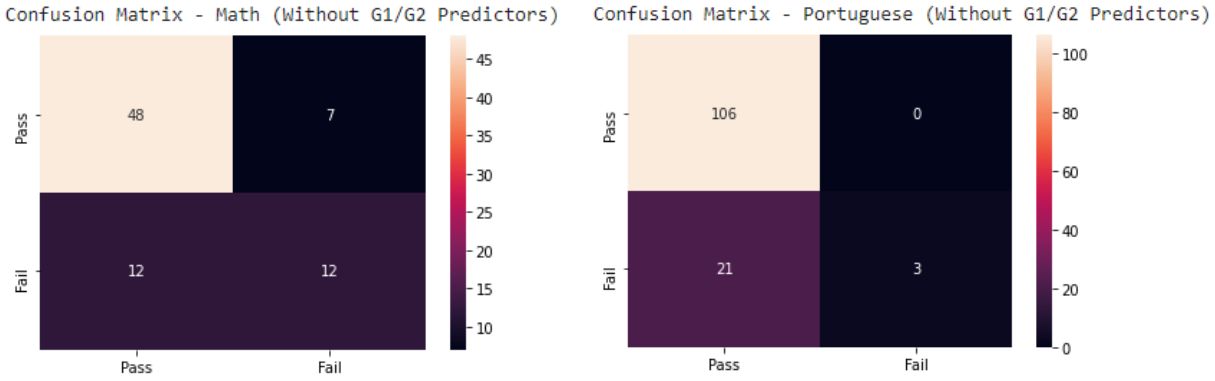


Table 4. Classification Reports for RF.

Math - Random Forest (All predictor) Classication Report

	precision	recall	f1-score	support
Pass	0.94	0.93	0.94	55
Fail	0.84	0.88	0.86	24
accuracy			0.91	79
macro avg	0.89	0.90	0.90	79
weighted avg	0.91	0.91	0.91	79

Math - Random Forest (Without G1/G2) Classication Report

	precision	recall	f1-score	support
Pass	0.80	0.87	0.83	55
Fail	0.63	0.50	0.56	24
accuracy			0.76	79
macro avg	0.72	0.69	0.70	79
weighted avg	0.75	0.76	0.75	79

Portuguese - Random Forest (All predictor) Classication Report

	precision	recall	f1-score	support
Pass	0.90	0.98	0.94	106
Fail	0.87	0.54	0.67	24
accuracy			0.90	130
macro avg	0.89	0.76	0.80	130
weighted avg	0.90	0.90	0.89	130

Portuguese - Random Forest (Without G1/G2) Classication Report

	precision	recall	f1-score	support
Pass	0.83	1.00	0.91	106
Fail	1.00	0.12	0.22	24
accuracy			0.84	130
macro avg	0.92	0.56	0.57	130
weighted avg	0.87	0.84	0.78	130

Table 5. Classification Reports for Dummy Classifiers.

Math - Dummy Classifer Classication Report

	precision	recall	f1-score	support
Pass	0.74	0.45	0.56	55
Fail	0.33	0.62	0.43	24
accuracy			0.51	79
macro avg	0.53	0.54	0.50	79
weighted avg	0.61	0.51	0.52	79

Math Social - Dummy Classifer Classication Report

	precision	recall	f1-score	support
Pass	0.70	0.60	0.65	55
Fail	0.31	0.42	0.36	24
accuracy			0.54	79
macro avg	0.51	0.51	0.50	79
weighted avg	0.58	0.54	0.56	79

Portuguese - Dummy Classifer Classication Report

	precision	recall	f1-score	support
Pass	0.79	0.42	0.54	106
Fail	0.16	0.50	0.24	24
accuracy			0.43	130
macro avg	0.47	0.46	0.39	130
weighted avg	0.67	0.43	0.49	130

Portuguese Social - Dummy Classifer Classication Report

	precision	recall	f1-score	support
Pass	0.84	0.46	0.60	106
Fail	0.21	0.62	0.31	24
accuracy			0.49	130
macro avg	0.53	0.54	0.46	130
weighted avg	0.73	0.49	0.54	130

Table 6. Classification Reports for LDA w/ Forward Stepwise Selection

LDA Classification Report, Math Scores (w/ grades)

	precision	recall	f1-score	support
Fail	0.41	0.38	0.39	24
Pass	0.74	0.76	0.75	55
accuracy			0.65	79
macro avg	0.57	0.57	0.57	79
weighted avg	0.64	0.65	0.64	79

LDA Classification Report, Math Scores Social (w/out grades)

	precision	recall	f1-score	support
Fail	0.29	0.21	0.24	24
Pass	0.69	0.78	0.74	55
accuracy			0.61	79
macro avg	0.49	0.50	0.49	79
weighted avg	0.57	0.61	0.59	79

LDA Classification Report, Portuguese Scores (w/ grades)

	precision	recall	f1-score	support
Fail	0.29	0.08	0.13	24
Pass	0.82	0.95	0.88	106
accuracy			0.79	130
macro avg	0.55	0.52	0.51	130
weighted avg	0.72	0.79	0.74	130

LDA Classification Report, Portuguese Scores Social (w/out grades)

	precision	recall	f1-score	support
Fail	0.20	0.08	0.12	24
Pass	0.82	0.92	0.87	106
accuracy			0.77	130
macro avg	0.51	0.50	0.49	130
weighted avg	0.70	0.77	0.73	130

Figure 11. Cross Validation Results, All Models, All Parameters.

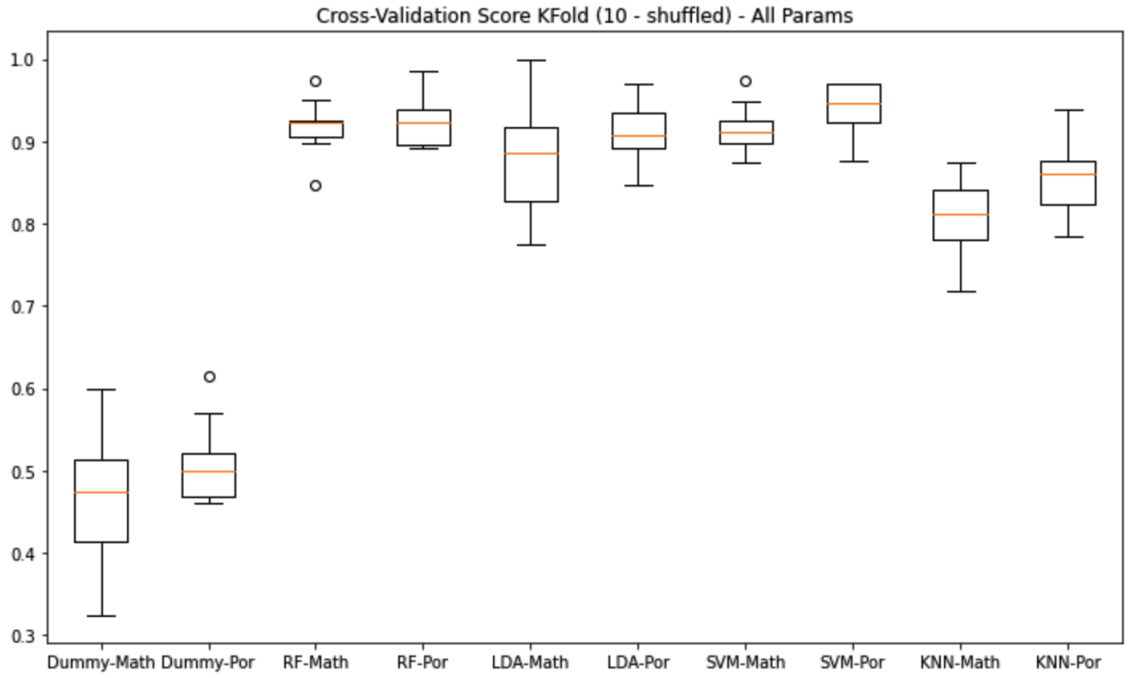


Table 7. Summary of Model Performance (bold and underlined indicates top performance)

Dataset\Classifier	Baseline	SVM	KNN	LDA	RF
Math	.46	.89	.82	.90	**.91**
Portuguese	.55	.89	.84	.86	**.90**
Math - Social	.56	.72	.72	.68	**.76**
Portuguese - Social	.53	.77	**.83**	.82	.76

Link to Google Colab

https://colab.research.google.com/drive/1opbjacOYtid8GL4Qv631b4RQEzP7Dyx-?usp=sharing

Note: On attempting to paste the code into the appendix, the document increased from 17 to 60 pages. We decided it would be best to link to the code instead.