Analysis of Final Grades for Math and Portuguese Students

Jing Meng, Kieran Simenson, Linda Mansour

STAT 385
05.05.2025

Table of Contents

Questions Addressed **2**

Preliminary Exploratory Data Analysis **2**

Method 1: Classification Trees **4**

Method 2: Logistic Regression **7**

Method 3: Lasso Regression **12**

Comparison of Results **16**

Future Steps and Importance of Findings **16**

Questions Addressed

When presented with the student performance dataset of Math and Portuguese students from UCI Machine Learning Repository, our group sat down and discussed variables that piqued our interest. We knew we wanted to focus on academic performance and ultimately decided on the following:

- Which variables help to classify whether students may pass or fail their third trimester?
- Which is the most accurate predictive method between classification trees, logistic regression, and lasso regression for passing the third trimester?

Preliminary Exploratory Data Analysis

Linda first merged two separate datasets, `math` and `portuguese` from the UCI Machine Learning Repository, and joined by shared features between both datasets such as `school`, `sex`, `age`, `address`, and `famsize`. The initial size of the merged dataset was 382 rows and 53 variables. In order to reduce the dataset size, we checked check for null or NA values, duplicate rows, and merged variables that are presumed to have high similarity (90% or higher) between both original datasets, like `guardian`, `Walc`, `Dalc`, and `romantic` by manually selecting one column from either dataset to represent both `math` and `portuguese`. This reduced the dataset from 53 variables to 38 variables after our initial cleaning. To visualize the correlation between all of our variables, we created a correlation heatmap featuring all numerical variables as shown in **Figure 1.** Analyzing the heatmap revealed the remaining highly correlated variables, such as `G1`, `G2`, and `G3` that may contribute to multicollinearity in our classification models if they remain untrimmed or cleaned further. However, seeing these highly correlated variables helps support hypotheses of consistent performance in school being a good predictor of a student's grade.

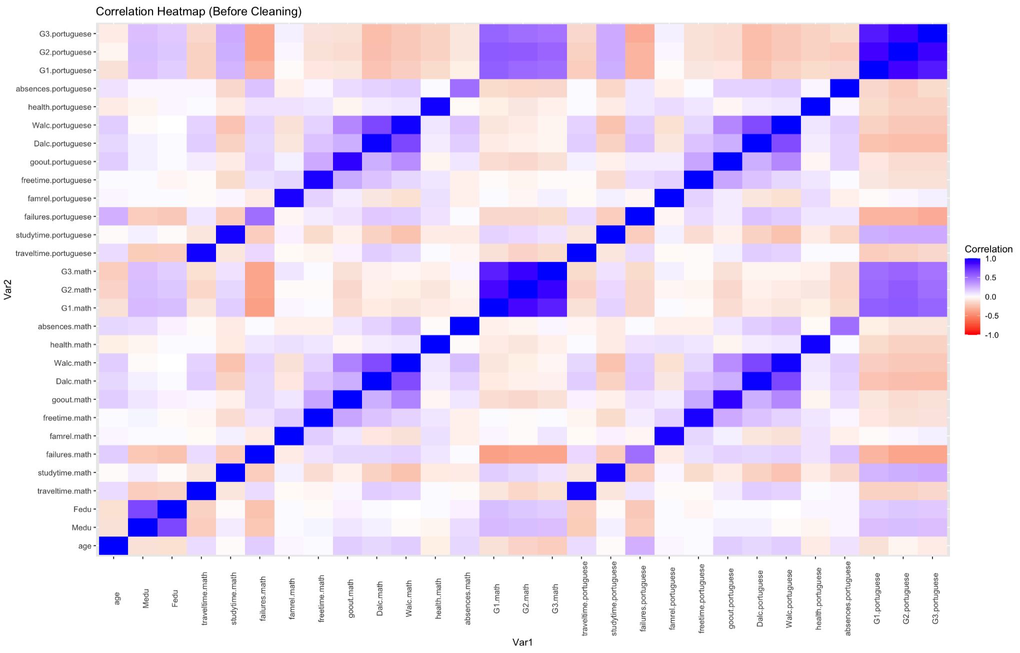


Figure 1: *All 29 numerical variables in a correlation heatmap*

after initial data cleaning.

Once the data was cleaned, new factors were created. Kieran created three new factors: `pass_fail_1`, `pass_fail_2`, and `pass_fail_3`. These factors corresponded with `G1`, `G2`, and `G3` respectively. For reference, `G1` is a numerical variable ranging from 0-20 that exists for both math and portuguese. The factor variable `pass_fail_1` takes the value of both math and portuguese `G1` (`G1.math` and `G2.portuguese`) and assigns `pass_fail_1` with one of four categories. They are as follows:

- Pass both: `G1.math` ≥ 10 & `G1.portuguese` ≥ 10
- Pass portuguese: `G1.math` < 10 & `G1. portuguese` ≥ 10
- Pass math: `G1.math` ≥ 10 & `G1.portugues` < 10
- Fail both: `G1.math` < 10 & `G1.portuguese` < 10

This was then repeated for `G2` and `G3`, and the original G1-3 values for both math and portuguese were removed from the dataframe. While we ultimately focused on the third trimester, all three trimesters were made into factors before we reached that decision.

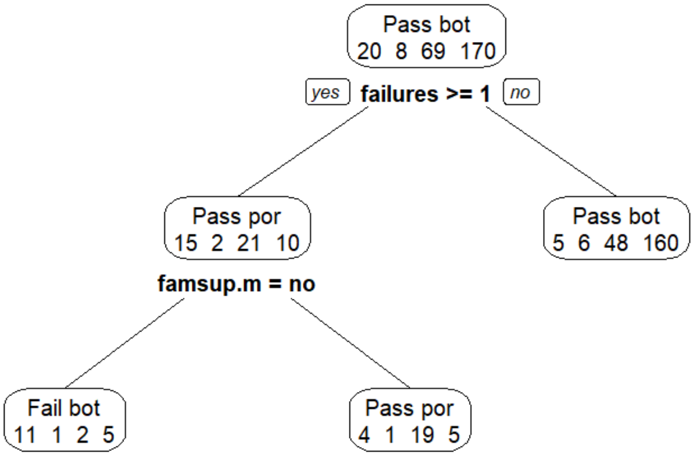
To simplify the prediction task, we created a binary outcome variable based on the third trimester's performance (`pass_fail_3`). This new variable, called `pass_binary`, assigns a value of `1` if a student passed both Portuguese and Math, and `0` otherwise. By focusing only on the "Pass both" outcome, we reduced a multi-class problem to a binary classification problem, which made model training and interpretation more straightforward.

To evaluate model performance fairly, Jing split the dataset into training and test sets. We tested two common ratios: a 70/30 split and an 80/20 split. The 70/30 split gives us more test data, which helps provide a more reliable estimate of the model's performance on unseen data — making it the preferred choice for evaluation. Meanwhile, the 80/20 split offers more training data, which can sometimes help the model learn better patterns. After comparing the results, we chose to proceed with the 70/30 split for our models.

Method 1: Classification Trees

Kieran led the coding and analysis of the third trimester using classification trees. We decided to use classification trees because they are good for categorical data, are highly comprehensible, and tend to utilize a low amount of processing power. To best cover all fit possibilities, five trees were created: a full tree, pruned tree, bagged tree, boosted tree, and random forest tree. Initially, the trees were meant to be created using the factor variable pass_fail_3. A copy of the training and test set was created that removed pass_fail_1, pass_fail_2, as well as the pass_binary variables, as they were highly correlated to pass_fail_3 and would detract from other variables of interest. A full tree was created with the rpart, rpart.plot, and mlbench libraries using the training data and subsequently pruned with the best cp (0.04124). As depicted in **Figure 2** , the pruned tree using pass_fail_3 resulted in no outcome for "Pass math".

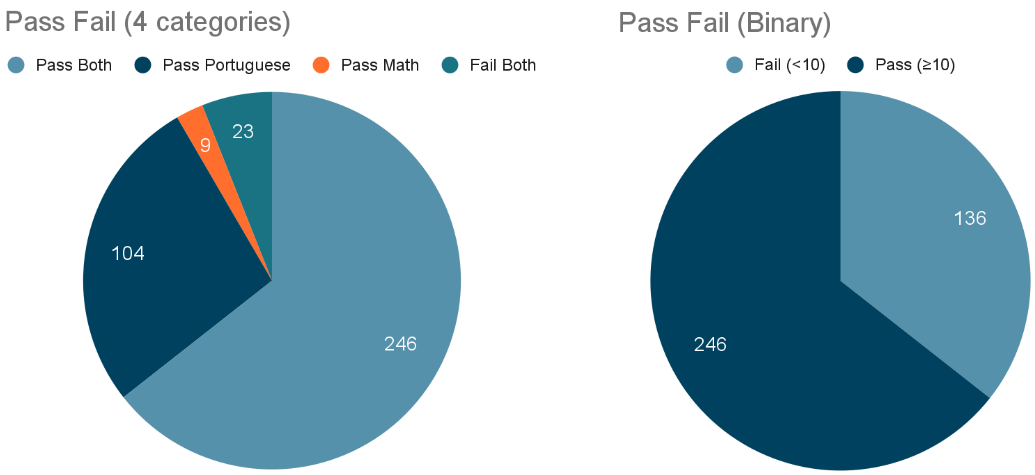
Figure 2: *Pruned tree using 4 categories*



This is most likely due to pass_fail_3 only including 9 values of "Pass math", a distinctly smaller value than the other three categories **(See Figure 3)**. With the train/test split that we used on our chosen seed, this resulted in only one value of "Pass math" within our test set. It was deemed unlikely that we would get accurate predictions for all four categories using this method, so analysis was then switched to use the pass_binary variable instead, dropping pass_fail_1, pass_fail_2, and pass_fail_3. The pass_binary split was much more even, as shown in **Figure 3** and therefore seemed like a better variable for prediction.

A new full tree was created and pruned, using the best cp of 0.0206 based on the minimum cross-validation error of the full tree's CP table. Error rates for both trees were noted for future comparison: 0.4087 for full and 0.3130 for pruned. Next, a tree was built using bagging through the ipred, tree, and adabag libraries, with an error rate of 0.2957. For the boosted tree, a loop was created that would run boosting through iterations 10-50 with depths 3-10, storing subsequent prediction errors in a matrix. From that loop, the minimum error was found to be 0.2783 at 18 iterations and a depth of 6. The boosted tree using these values predicted the test set with 0.4087 error. This high error, matching the full tree's error rate, is likely due to overfitting of the training set within the loop, making it a poor model for prediction.

Figure 3: *A comparison of values found in* `pass_fail_3` *(left) and* `pass_binary`



The final tree created was a random forest using the `randomForest` library. First, three random forests were created using `ntree = 300` and mtry = √p, p/2, and p, where p=32=the number of predictor variables used. When plotted (see **Figure 4**), the error rate seemed to stabilize around 150 trees, with m/p trending the lowest error. Tuning was then run with `nTry = 150` to find the best m value, which was found to be 10 (see **Figure 5**). This resulted in a random forest with an error of 0.3217.

Figure 4: *OOB error rate for 300 Random Forests with m=√p, p/2, and p*

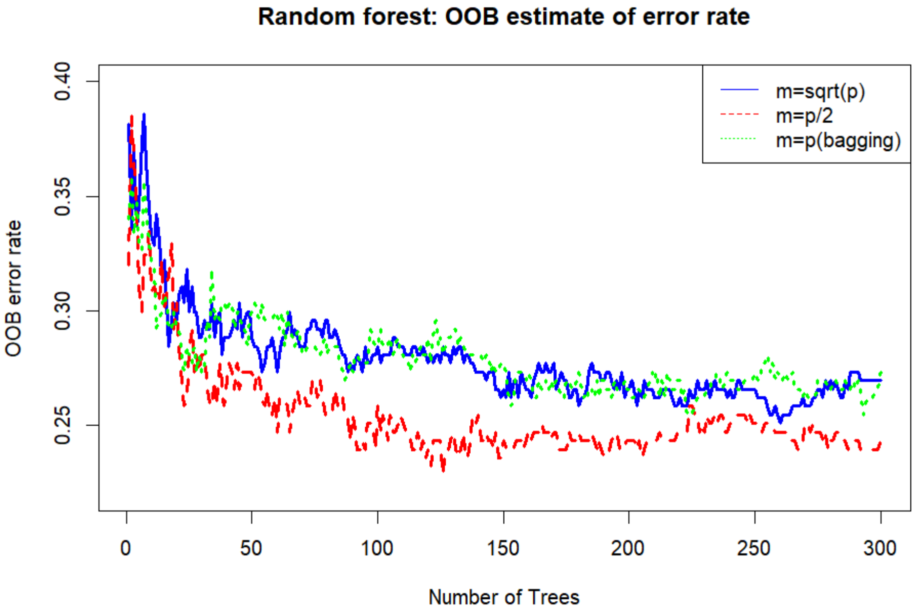
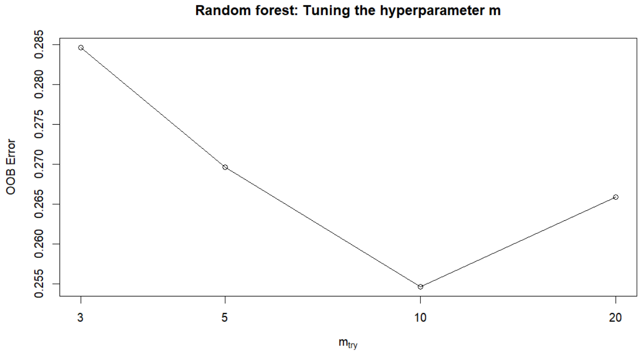


Figure 5: *Plot of tuning Random Forest at* `ntreeTry=150`



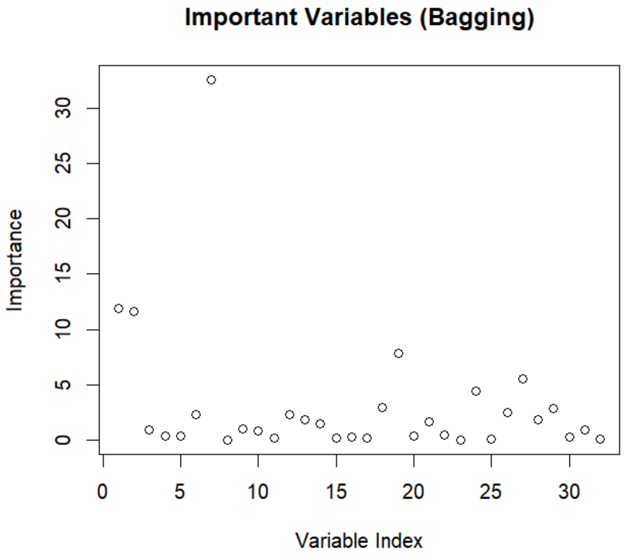
Finally, all trees were compared against each other by error rate, concluding in the following:

Full Tree	Pruned Tree	Bagging	Boosting	Random Forest
0.4087	0.3130	0.2957	0.2783	0.3217

Determining bagging to result in the lowest error rate and highest accuracy, the most important variables were then determined by observing the highest valued variables in the importance vector of the bagging results via plot (see **Figure 6**) and through outputting the top five sorted values. The top five variables were found to be:

failures.math	absences.math	absences.portuguese	Mjob	schoolsup.portuguese
32.585	11.944	11.591	7.831	5.550

Figure 6: *Important variables of Bagging by index*



Observing the plot, the top three, potentially four, variables seem to be the most critical, as once you reach the fifth variable, importance is much more condensed and it is harder to determine a hierarchy of importance.

In conclusion, of the trees, bagging was the most accurate with an accuracy of 70.43%. The accuracy isn't awful, but could certainly be improved. This may point to a less distinct cause of passing both classes and it could be beneficial to re-analyse with a larger dataset. Also, with a larger dataset, a four-category pass/fail factor such as `pass_fail_3` could be a much more viable option that could find more concrete distinctions.

Method 2: Logistic Regression

Linda focused on performing a logistic regression analysis on the data, since many of the variables were numerical. This means that we can use the logistic regression method to solve the classification problem of determining whether students passed or failed both courses in the third trimester paired with a stepwise selection to narrow down the model. Created earlier in the factor creation step, we use the target binary variable `pass_binary` which is represented by `pass_binary = 1` if students pass both courses, and `pass_binary = 0` if a student failed one or both courses.

Before performing the regression analysis, Linda had to clean the dataset further, as logistic regression is sensitive to multicollinearity, and 29 variables is still a lot. After installing and importing the `caret package,` Linda used `findCorrelation()` with a cutoff at 80% threshold in order to identify and remove four highly correlated variables out of only 11 numerical predictors. She also removed variables that were presumed to not have a major impact on the outcome of the question such as `activities,` and `higher` eight more variables manually. So, After starting with 29 predictors, Linda removed 4 highly correlated numerical variables using `findCorrelation()`, and manually dropped 8 more variables deemed less relevant to the question. This reduced the dataset to 26 columns. However, since several columns were categorical, R automatically converted them into multiple dummy variables when fitting the logistic regression model, resulting in 33 total predictors after the full model regression. Removing these variables initially helped to reduce multicollinearity and potential of messing with actual predictors by preventing flooding the model with redundant variables.

To explore some of the data Linda used specifically, she generated a scatterplot of each student's final math grade vs. their final Portuguese grades to visualize students that passed both courses, and those that failed either or both. This is visualized in **Figure 7**. This clear difference gave her the idea to continue on to using the logistic regression model since there appears to be a clear separation between students with lower grades and those with higher grades. There also appears to be more clustering above the line, implying that the majority of students scored higher in Portuguese compared to math.

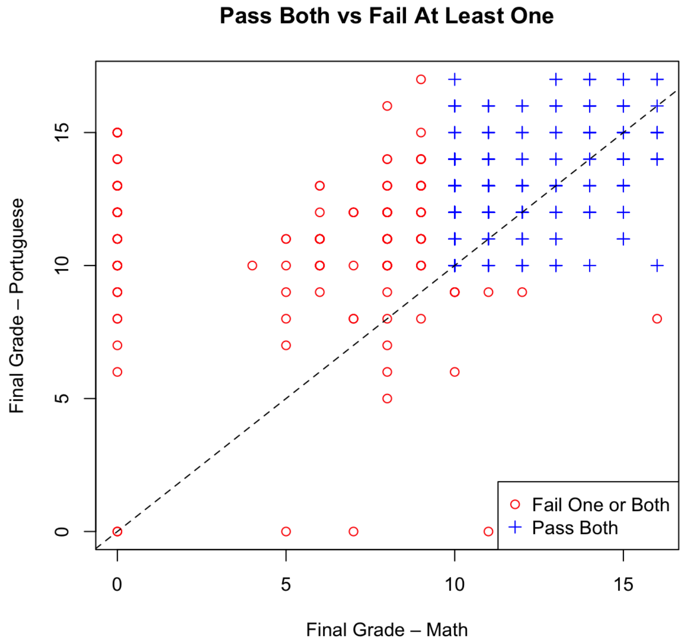


Figure 7: Scatterplot of students passing both vs. fail at least one of their courses.

Linda fit the full logistic regression model on the cleaned dataset using a 70/30 split. She then modeled pass_binary with a binomial distribution `glm()`, keeping the cutoff at 0.5 (due to the grade cutoff at 50%), and 50 iterations to ensure that the model converges, we managed to achieve an AIC value of 325.15, and model accuracy of 60% (95% CI: 50.45% - 69.02%) supported by the confusion matrix shown in **Figure 8.** The confusion matrix revealed that the full model correctly predicted 43% (19/44 students) as failing, and correctly predicted 70% (50/71 students) of passing students as passing.

Figure 8: Confusion Matrix for the full logistic regression model.

	Reference	
Prediction	Predicted Fail(0)	Predicted Pass(1)
0	19	21
1	25	50

To dive deeper into why the model had a low accuracy of 60%, Linda plotted the test set predicted probabilities in a histogram that visualizes the cases of students passing both or failing either/both courses. In **Figure 9**, left skewed distribution of scores is observed, and there are significant overlaps between predicted probabilities of failure and of passing. The red represents students who failed at least one course, and blue presents students who passed both courses. The purple signifies overlapping predicted probabilities, which helps to visualize the inaccuracies also represented in the confusion matrix.

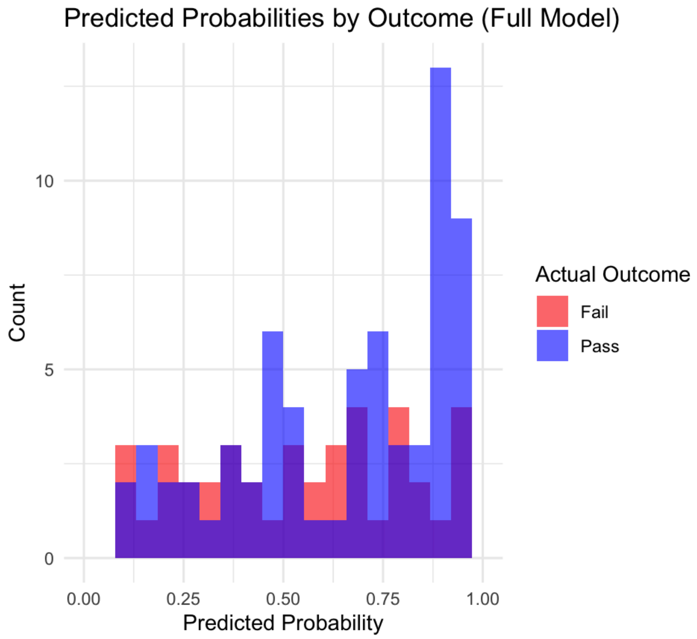


Figure 9: Predicted probabilities by student success outcome for the full logistic model. Significant overlap of predicted probabilities is evident among the entire probability range (0 to 1), with no particular grouping anywhere.

Linda proceeded with modeling by using the AIC stepwise selection method going both forwards and backwards using `step()`. This resulted in a more accurate model, with 72.17% accuracy, and a confusion matrix with better predictions than the full model. This ensured that the model would select only the most significant variables for prediction. In the end, the final logistic model selected six significant predictors: `age`, `famsizeLE3`, `failures.math`, `failures.portuguese`, `schoolsup.portugueseyes`, and `absences.portuguese`. The stepwise selection model yielded an AIC of 295.75 and 72.17% accuracy (95% CI: 63.05% - 80.13%). This is a 12% improvement from the full model, which also is supported by the confusion matrix in **Figure 10** that shows 43.2% (19 students) of students correctly predicted to fail at least one course, and 90.1% (64 students) correctly predicted to pass both. Though there is no significant change in predicting students who failed, there is significant improvement in predictions for passing students.

	Reference	
Prediction	Predicted Fail(0)	Predicted Pass(1)
0	19	7
1	25	64

Figure 10: Confusion Matrix for the stepwise logistic regression model.

To visualize further, Linda plotted the predicted probabilities of the stepwise logistic model in **Figure 11**, which produced a much clearer picture of the predicted probabilities. There is significantly less overlap between predictions, especially towards the left side of the graph where there were more students predicted to fail than correctly classified.

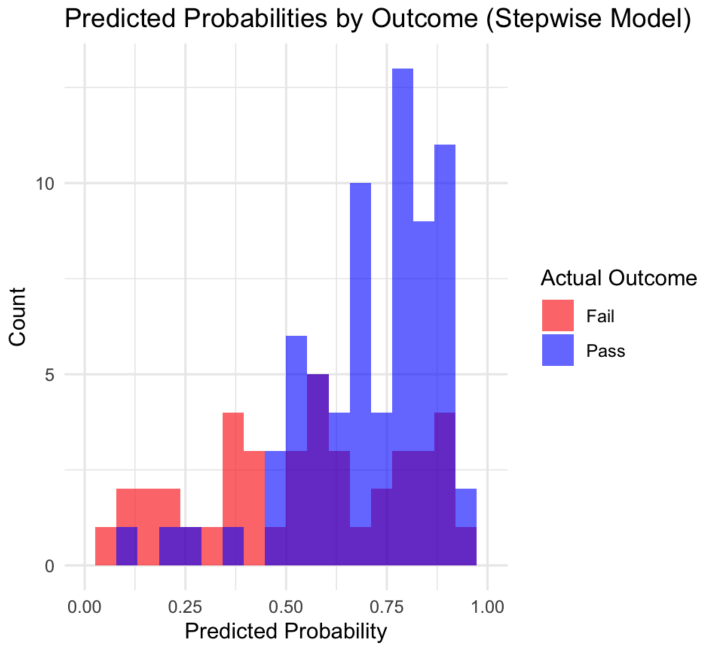
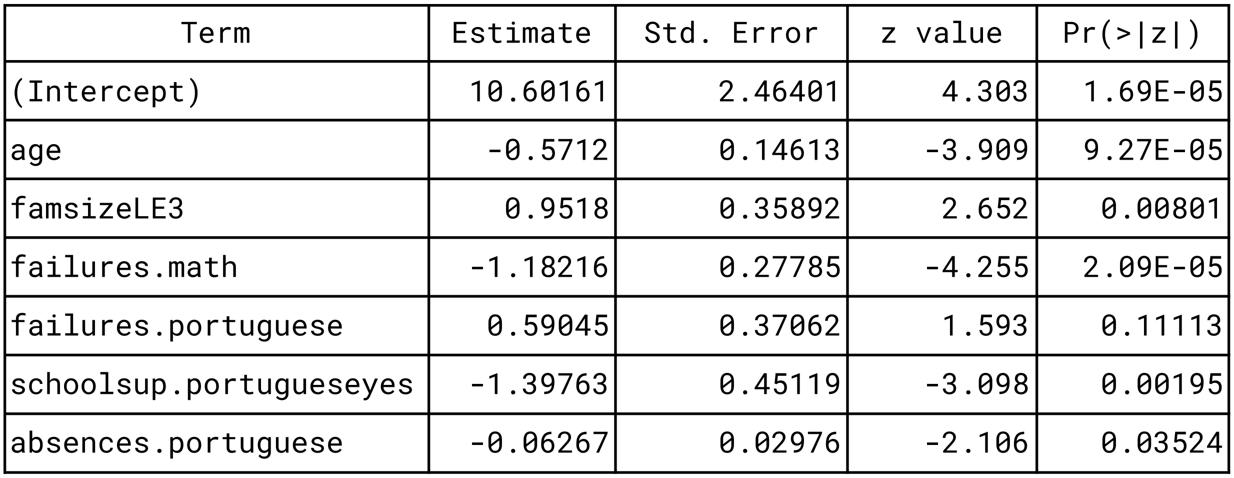


Figure 11: Predicted probabilities by student success outcome for the stepwise logistic model. Overlap of predicted probabilities is evident among a smaller range, mostly between 0.5 and 0.9. We also notice that there is a much higher number of correct predictions for students passing both courses compared to **Figure 9.**

The stepwise logistic model's six significant predictors are visualized with their p-values evaluated at a 0.05 level in **Figure 12**, with the most significant predictor being failures.math, which tells us that a student who may have failed math in the past has a higher prediction probability of failing the final trimester of math. The least significant of these six predictors is `failures.portuguese`, which does not make the 0.05 cutoff.



Term	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	10.60161	2.46401	4.303	1.69E-05
age	-0.5712	0.14613	-3.909	9.27E-05
famsizeLE3	0.9518	0.35892	2.652	0.00801
failures.math	-1.18216	0.27785	-4.255	2.09E-05
failures.portuguese	0.59045	0.37062	1.593	0.11113
schoolsup.portugueseyes	-1.39763	0.45119	-3.098	0.00195
absences.portuguese	-0.06267	0.02976	-2.106	0.03524

Figure 12: Stepwise selection on logistic regression results, includes standard error and p-values for the final significant predictors.

To take it a step further, Linda visualized these significant predictors in the stepwise histogram **Figure 13**, excluding those that do not meet the 0.05 significance level. Five predictors remain, including `age`, `famsizeLE3`, `failures.math`, `schoolsup.portugueseyes`, and `absences.portuguese.`

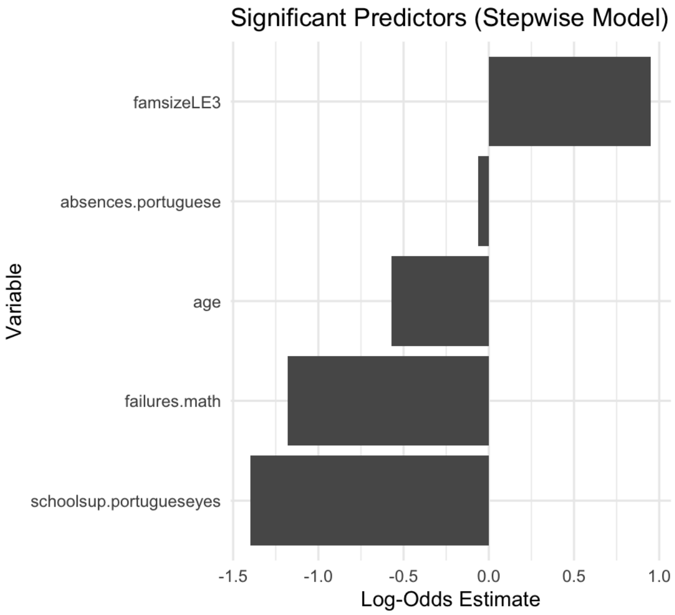


Figure 13: Histogram based on the significant predictor results of the stepwise selection logistic regression model.

The findings from the logistic regression models conclude that the initial full model was too large to obtain accurate predictions, yielding a 60% test set accuracy, and a high AIC at 325.15. After performing variable selection using stepwise selection on the same logistic model, we were able to increase the model accuracy to 72.17% and reduce the AIC to 295.75 which tells us that the stepwise model fits the data slightly better than the full model. The most influential predictors were past math failures, family size, age, Portuguese support, and Portuguese absences, which are mostly accurate in predicting whether a student passes both classes or not at the 0.5 (50%) grade cutoff.

To increase model accuracy in future endeavors, we could divide the data up in a different way, use a different split, and do some more EDA on predictors that may not be as linear as expected. For example, dividing failures up into three segments instead of “fail one, pass both” could help us paint a more accurate and specific picture of the answer to our problem. Additionally, we could focus on certain factors, whether it be financial, familial, social, or purely academic. By narrowing our question down, and further dividing up the cleaned data into more applicable categories, then the full models and tuned models might yield a more accurate reading.

Method 3: Lasso Regression

Jing applied Lasso Regression to predict student academic success, defined by whether a student passed both Portuguese and Math in the third trimester. The binary outcome variable pass_binary was created, where 1 represents students who passed both subjects and 0 indicates otherwise. Our dataset initially contained 38 predictors after merging and cleaning the original Portuguese and Math datasets.

We chose Lasso because of its ability to perform automatic variable selection. This is important in our case because including too many irrelevant features can lead to overfitting and poor generalization on unseen data. Lasso handles this by shrinking less important coefficients toward zero, effectively removing them from the model.

Before modeling, we excluded variables like pass_fail_1, pass_fail_2, and pass_fail_3, as they were derived from the same grades used to define the outcome and would have caused data leakage. We used the ‘glmnet’ package in R and split the data into training 70% and testing 30% sets using set.seed(05052025) for reproducibility.

To tune the model, we used **10-fold cross-validation** to select the optimal penalty parameter lambda. As shown in the **Figure 14** below, the cross-validation procedure identified λ = 0.038 as the value that minimized binomial deviance:

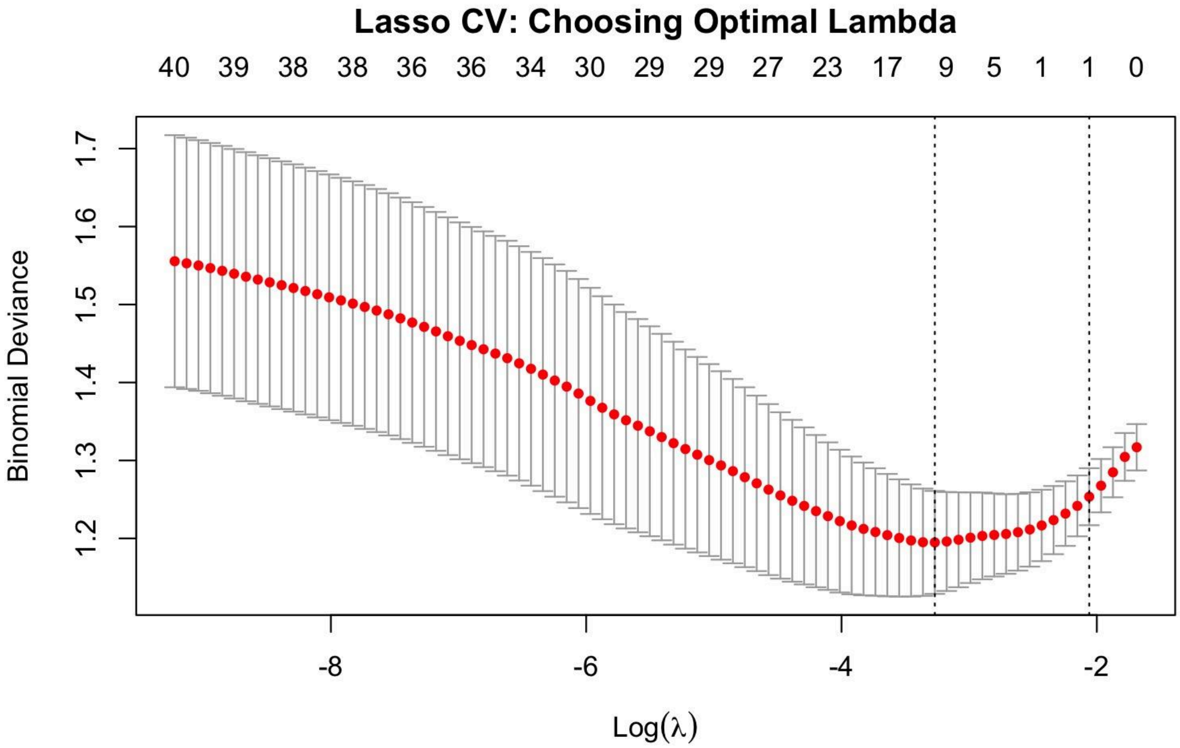


Figure 14: Lasso CV: Choosing the optimal Lambda

Next, using the best lambda, Lasso selected **11** predictors out of **38** total. Many of which reflected **demographic, academic, and behavioral factors**. These included:

- failures.math: History of math failure was the strongest negative predictor
- absences.portuguese: Frequent absences correlated with lower success
- schoolsup.portugueseyes: Receiving school support surprisingly linked to lower outcomes, possibly signaling at-risk status
- Mjobservices: Mother's employment in services had a positive effect
- higher.portugueseyes: Students planning to pursue higher education tended to succeed
- guardianother, famsup.portugueseyes, PstatusT, age, and parental education (Medu, Fedu)

The coefficient plot **Figure 15** below visualizes the selected predictors. Variables on the right (**positive coefficients**) increase the odds of success, while those on the left (**negative coefficients**) decrease them.

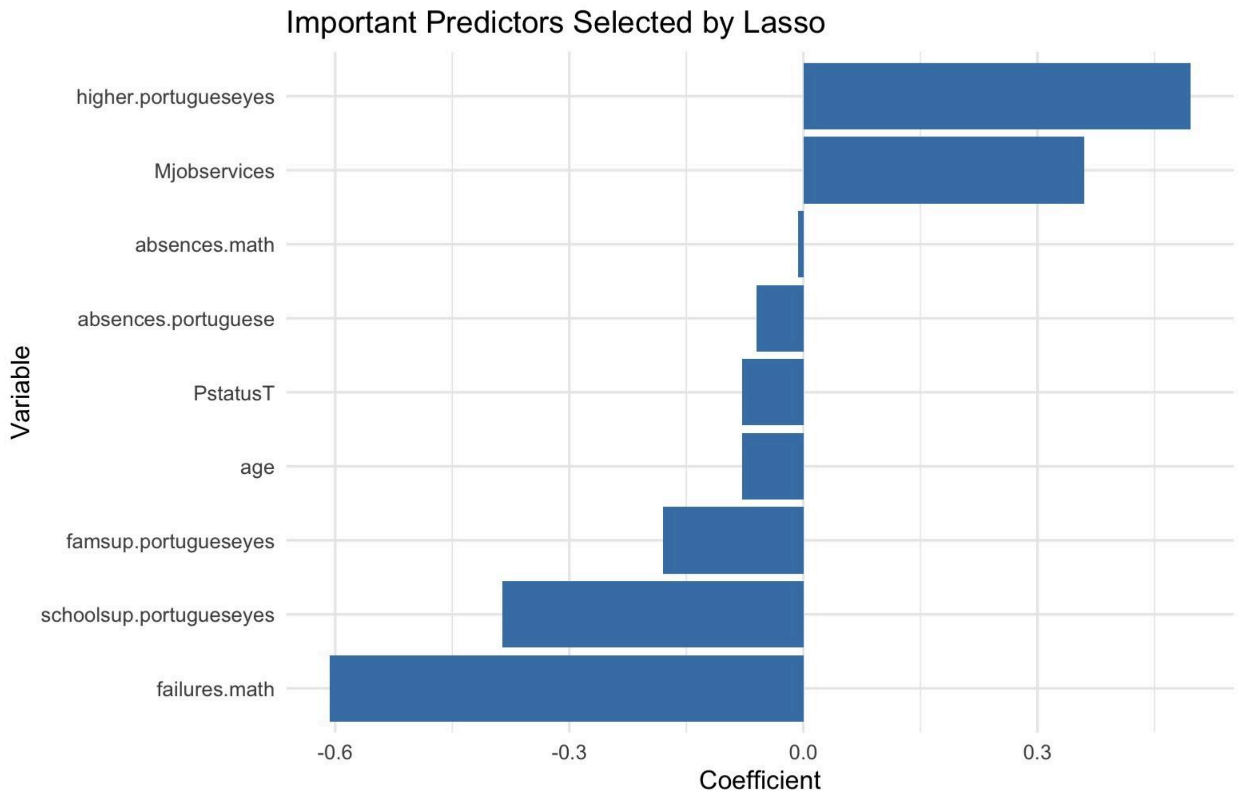


Figure 15: **Important Predictors Selected by Lasso**

Among them, `failures.math` had the strongest **negative effect**, indicating that students who failed math previously are much more likely to struggle again. Other negative predictors include `schoolsup.portugueseyes`, `famsup.portugueseyes`, and `absences.portuguese`, suggesting that frequent absences and reliance on school/family support may be signals of academic difficulty.

On the positive side, `higher.portugueseyes` and `Mjobservices` stood out. The desire to pursue higher education and having a mother employed in services may reflect motivation and socioeconomic stability, which help improve academic outcomes.

This **Figure 15** highlights key academic risk factors, such as frequent absences and previous course failures. It also shows the importance of motivation and family background—for instance, students who plan to pursue higher education or have parents in service jobs tend to perform better. The plot makes the results of the Lasso model easy to understand and provides clear direction for identifying which students might benefit most from additional support.

On the test set, the Lasso model achieved an accuracy of 70.43%, The confusion matrix **Figure 16** is shown below:

	Actual Fail	Actual Pass
Predicted 0	10	3
Predicted 1	29	73

Figure 16: Lasso Confusion Matrix

The model performed well in identifying students who passed both subjects (true positives = 73), but it also generated some false positives, predicting pass for 29 students who did not pass both. Still, the accuracy is reasonably strong for an educational dataset with some inherent noise.

In conclusion, Lasso helped us narrow down the most meaningful predictors and avoid overfitting by excluding redundant variables. It also provided interpretability, as the non-zero coefficients gave us insights into the key factors linked to academic success. One potential improvement would be to explore interaction terms or to combine Lasso with other models like logistic regression or random forest for comparison and model ensembling.

Comparison of Results

To answer our first question of what variables help classify if a student will pass or fail, we wanted to compare the important predictors between our models and highlight any overlap. Two variables stood out as important in all three models: `failures.math` and `schoolsup.portuguese`. Other predictors of interest included `Mjob`, `reason`, `schoolsup.math`, and `higher.math`, though those did not have as stand-out significance in all models. Regarding our second question of interest, we found our model accuracies **Figure 17** to be the following:

Classification Trees (Bagging)	Logistic Regression (Stepwise)	Lasso Regression
70.43%	72.17%	70.43%

Figure 17: Model Accuracy Comparison

It was interesting to note that our tree model and Lasso model had the same accuracy, and that logistic regression had a <2% positive difference. Ultimately, stepwise logistic regression was found to be the best model in terms of accuracy for predicting if a student will pass or fail their third trimester.

Future Steps and Importance of Findings

If we were to continue analyzing this data, there are a few models and questions that we would be interested in. Firstly, we would be interested in predicting success in previous trimesters and evaluating any differences. Secondly, we would be interested in fitting models for K-nearest neighbors as well as K-means clustering. It would be interesting to determine the best k value, but it would also be interesting to visualize a k=4 split to see if there is any correlation with our initial 4 category factor `pass_fail_3`.

As for our findings, by utilizing the same data for the same question over three different models, we were able to create a better understanding of the key differences between said models. Data preparation had to be altered slightly for each model type: logistic regression required variables to be purposefully selected to build the best fit whereas lasso utilizes shrinkage to minimize irrelevant variables and classification trees look at all predictors to create decisional splits based on their importance. We observed that models perceive different variables as important, which makes any overlap stand out. Based on these overlapping important predictors, we would be able to reach out and encourage the schools to put focus on preventing math failures, monitoring attendance, and helping students set long-term academic goals, backing up our findings with multiple models of significance.