Predicting Student Pass/Fail Rates

Benjamin Mauldin
Northwood University
Aldie, VA, USA
mauldinbm42@northwood.edu

I. Abstract

The COVID-19 pandemic forced many students online. In our post-pandemic world, remote learning has continued to retain a significant portion of students today, almost necessitating the world of learning analytics and the benefits they can bring to the remote classroom. This project was a small exercise in this world, and sought to predict student pass/fail rates of a machine learning dataset provided by Kaggle using machine learning methods in R. Logistic regression and random forest methods were used to predict the classification of students into either passing or failing categories, to high predictive success. Random forest was the more powerful method, with an accuracy metric of 100%. Logistic regression garnered 86.8% accuracy. If learning analytics like these were to be harnessed and perfected with high degrees of accuracy, this information could be used to help teaching staff identify and help struggling students, or allow students to themselves identify when they may need assistance.

Predicting Student Pass/Fail Rates

Benjamin Mauldin
Northwood University
Aldie, VA, USA
mauldinbm42@northwood.edu

II. Introduction

The near synchronous onset of rapid technological advancements like AI and social media, and the physical isolation of the COVID-19 pandemic has left many facets of our society irreversibly changed. Maybe one of the most critical of those facets changed is education.

In the face of the pandemic, some 77% of US public schools and 73% of US private schools reported moving some or all classes to distance learning formats in 2020 according to the National Center of Education Statistics [1]. Even after the pandemic subsided, distance learning has remained a significant portion of total learning, especially in postsecondary institutions. According to an IPEDS report for 2022 data, 26.6% of postsecondary learners in the US were learning exclusively online, while 27.7% reporting only some distance learning, and 45.7% reported zero online learning [2]. These facts make it apparent that distance learning is here to stay, with both its advantages and disadvantages.

For instance, in-person teachers can use in-person judgment to determine which students may need more assistance, but in the online classroom, the distance can obfuscate these judgments on student engagement. On the other hand, some students (such as those with certain types of disabilities) may find an online classroom more flexible, as they can pace themselves and not have to worry about disrupting other students' learning experiences. To help supplement the things that distance learning is missing, a series of statistical and machine learning techniques can be conducted on data collected in online classrooms in a field called learning analytics.

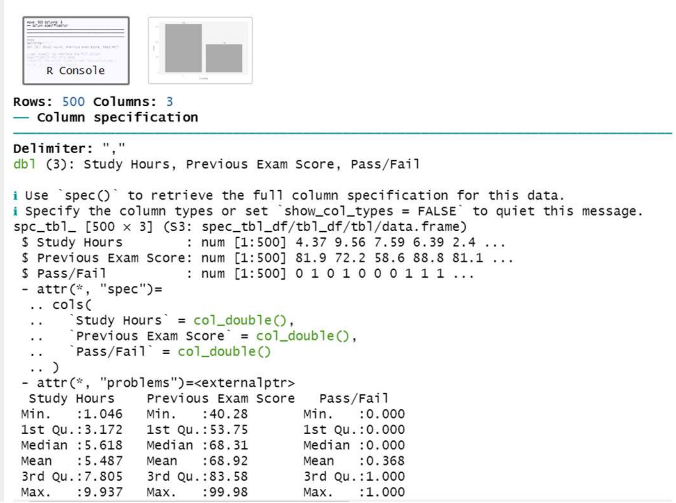
III. Learning Analytics

Learning analytics seeks to provide a litany of benefits to online classrooms to help students and educators alike. At its crux, learning analytics should allow educators to more deeply personalize instruction in an environment that is inherently isolated. Students could see data regarding teaching styles and weaknesses of prospective teachers whilst choosing classes, and teachers could see similar data regarding their students. This allows both sides to learn about each other and themselves and where to focus their collective energies. Applied on a wider scale, this could also allow institutions better information on where to allocate their resources. Aggregate student performance data could be used to predict future student performance, which could help identify struggling students, students at risk of drop-out, and even students who may be ahead. However, learning analytics is not without its drawbacks.

As with many machine learning endeavors, maybe the biggest concern is with privacy. To create these analytics, large amounts of sensitive student data has to be collected by learning institutions, which could be misused or compromised – transparency is a must. In addition, machine learning algorithms could perpetuate and even magnify biases present in the data, as is the case with other machine learning practices such as predictive policing. Analytics could also further widen the disparity between institutions with limited resources and institutions with larger resources. Lastly, analytics can be difficult to interpret, leading to misguided conclusions. Ethical concerns aside, this project is focused on predictive analytics.

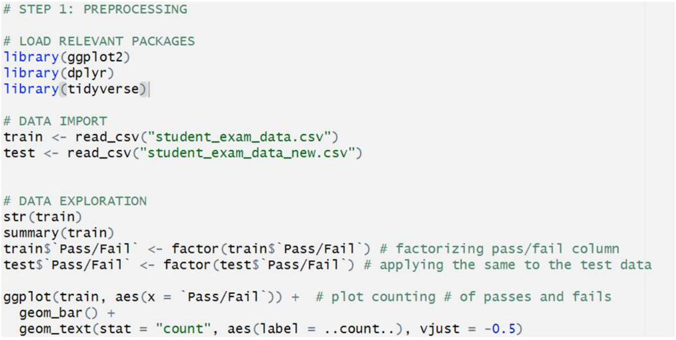
IV. Methods: Pre-Processing

Two datasets (one for training, one for testing), each containing the study hours, previous exam performance, and pass/fail categories of 500 students were downloaded from Kaggle. First the data must be pre-processed and analyzed. In R, these two datasets were imported and then converted to two data-frames to be explored. The structure and summary statistics of the data are below.

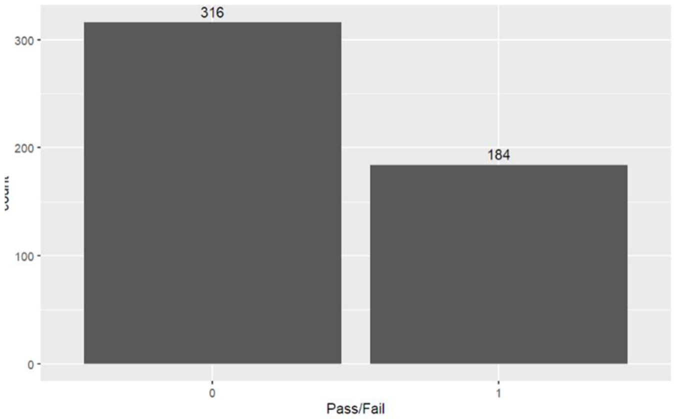


```
Rows: 500 Columns: 3
── Column specification ────────────────────────────────────────────────
Delimiter: ","
dbl (3): Study Hours, Previous Exam Score, Pass/Fail

i Use `spec()` to retrieve the full column specification for this data.
i Specify the column types or set `show_col_types = FALSE` to quiet this message.
spc_tbl_ [500 × 3] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
 $ Study Hours        : num [1:500] 4.37 9.56 7.59 6.39 2.4 ...
 $ Previous Exam Score: num [1:500] 81.9 72.2 58.6 88.8 81.1 ...
 $ Pass/Fail          : num [1:500] 0 1 0 1 0 0 0 1 1 1 ...
 - attr(*, "spec")=
  .. cols(
  ..   `Study Hours` = col_double(),
  ..   `Previous Exam Score` = col_double(),
  ..   `Pass/Fail` = col_double()
  .. )
 - attr(*, "problems")=<externalptr>
  Study Hours     Previous Exam Score   Pass/Fail
 Min.   :1.046   Min.   :40.28       Min.   :0.000
 1st Qu.:3.172   1st Qu.:53.75       1st Qu.:0.000
 Median :5.618   Median :68.31       Median :0.000
 Mean   :5.487   Mean   :68.92       Mean   :0.368
 3rd Qu.:7.805   3rd Qu.:83.58       3rd Qu.:1.000
 Max.   :9.937   Max.   :99.98       Max.   :1.000
```

These metrics reveal the Pass/Fail rate is actually a categorical variable, and thus must be factorized, and also making this a classification problem. The presence of labels also makes this a supervised problem. Accordingly, logistic regression and random forest methods were chosen as our two models for prediction. Additionally, a count plot for how many students passed and failed was also rendered. The pre-processing code and countplot are shown below.



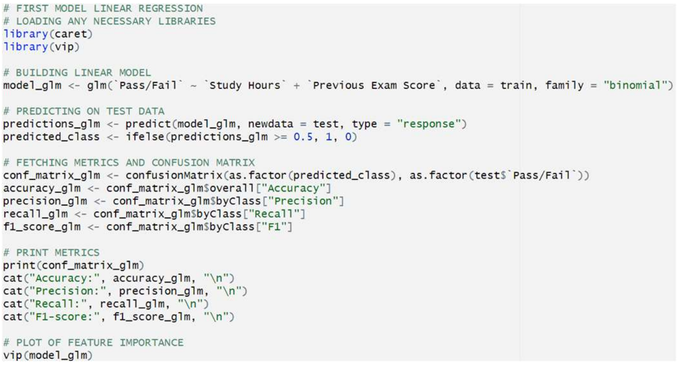
```
# STEP 1: PREPROCESSING

# LOAD RELEVANT PACKAGES
library(ggplot2)
library(dplyr)
library(tidyverse)

# DATA IMPORT
train <- read_csv("student_exam_data.csv")
test <- read_csv("student_exam_data_new.csv")


# DATA EXPLORATION
str(train)
summary(train)
train$`Pass/Fail` <- factor(train$`Pass/Fail`) # factorizing pass/fail column
test$`Pass/Fail` <- factor(test$`Pass/Fail`) # applying the same to the test data

ggplot(train, aes(x = `Pass/Fail`)) +  # plot counting # of passes and fails
  geom_bar() +
  geom_text(stat = "count", aes(label = ..count..), vjust = -0.5)
```



As you can see, 316 students failed, and 184 passed in the training data.

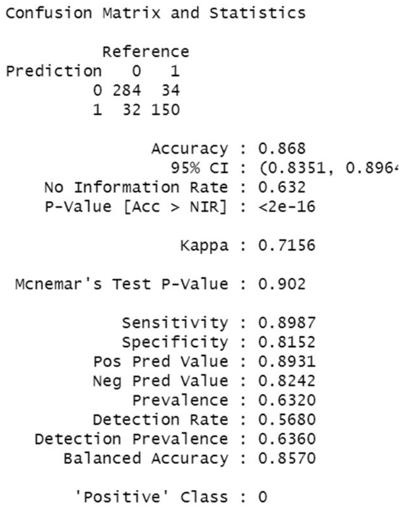
V. Model 1: logistic Regression

Logistic regression is the first and more simple method used. It uses the parameters of a logistic (s-shaped_curve) to model a simple relationship in the data, convert it to probabilities to classify the data. The code is shown below:

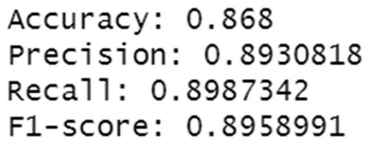


```
# FIRST MODEL LINEAR REGRESSION
# LOADING ANY NECESSARY LIBRARIES
library(caret)
library(vip)

# BUILDING LINEAR MODEL
model_glm <- glm(`Pass/Fail` ~ `Study Hours` + `Previous Exam Score`, data = train, family = "binomial")

# PREDICTING ON TEST DATA
predictions_glm <- predict(model_glm, newdata = test, type = "response")
predicted_class <- ifelse(predictions_glm >= 0.5, 1, 0)

# FETCHING METRICS AND CONFUSION MATRIX
conf_matrix_glm <- confusionMatrix(as.factor(predicted_class), as.factor(test$`Pass/Fail`))
accuracy_glm <- conf_matrix_glm$overall["Accuracy"]
precision_glm <- conf_matrix_glm$byClass["Precision"]
recall_glm <- conf_matrix_glm$byClass["Recall"]
f1_score_glm <- conf_matrix_glm$byClass["F1"]

# PRINT METRICS
print(conf_matrix_glm)
cat("Accuracy:", accuracy_glm, "\n")
cat("Precision:", precision_glm, "\n")
cat("Recall:", recall_glm, "\n")
cat("F1-score:", f1_score_glm, "\n")

# PLOT OF FEATURE IMPORTANCE
vip(model_glm)
```

Libraries were loaded, and the model was initialized. It was then used to predict on the testing data and compared to the actual testing data outcomes (after a threshold for classification was created). To calculate accuracy metrics and compare predictions to actual values, a confusion matrix was created. The output for accuracy metrics and the confusion matrix is below:

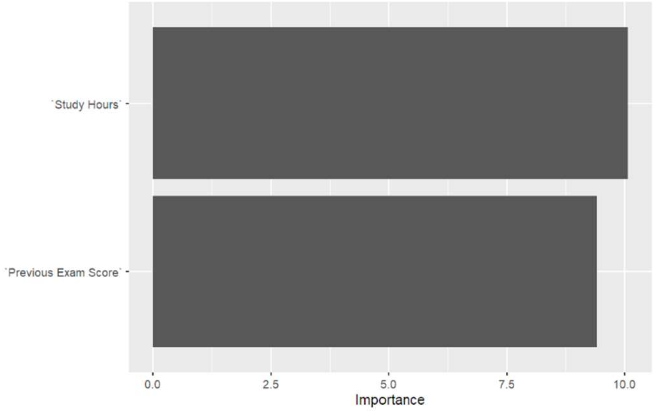


```
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 284  34
         1  32 150

               Accuracy : 0.868
                 95% CI : (0.8351, 0.896
    No Information Rate : 0.632
    P-Value [Acc > NIR] : <2e-16

                  Kappa : 0.7156

 Mcnemar's Test P-Value : 0.902

            Sensitivity : 0.8987
            Specificity : 0.8152
         Pos Pred Value : 0.8931
         Neg Pred Value : 0.8242
             Prevalence : 0.6320
         Detection Rate : 0.5680
   Detection Prevalence : 0.6360
      Balanced Accuracy : 0.8570

       'Positive' Class : 0
```



```
Accuracy: 0.868
Precision: 0.8930818
Recall: 0.8987342
F1-score: 0.8958991
```

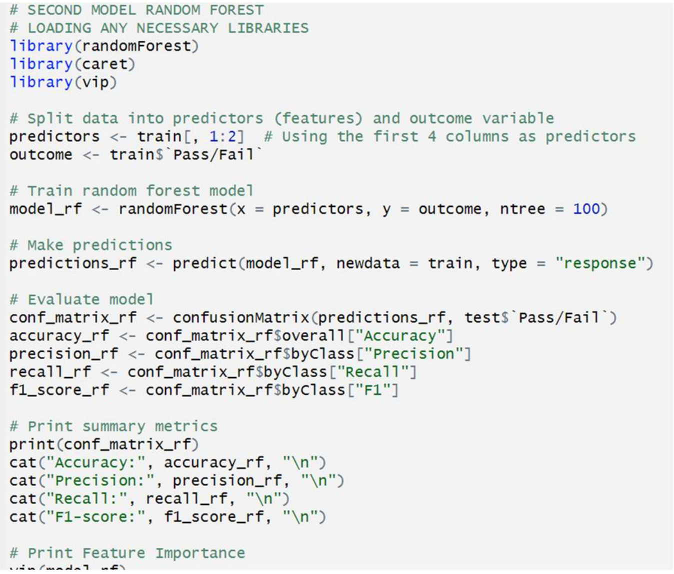
Additionally, a feature importance plot (shown below) was created to indicate how important each variable is for predictive success in this model.



Considering our data set is balanced, our accuracy of 86.8% is a good indicator of model performance. Our other metrics were: 89.3% precision, 89.9% recall, 89.6% F1-Score, and 81.5% specificity. Additionally, it seems study hours are marginally more important than previous exam scores in this model. Overall, these are good numbers.

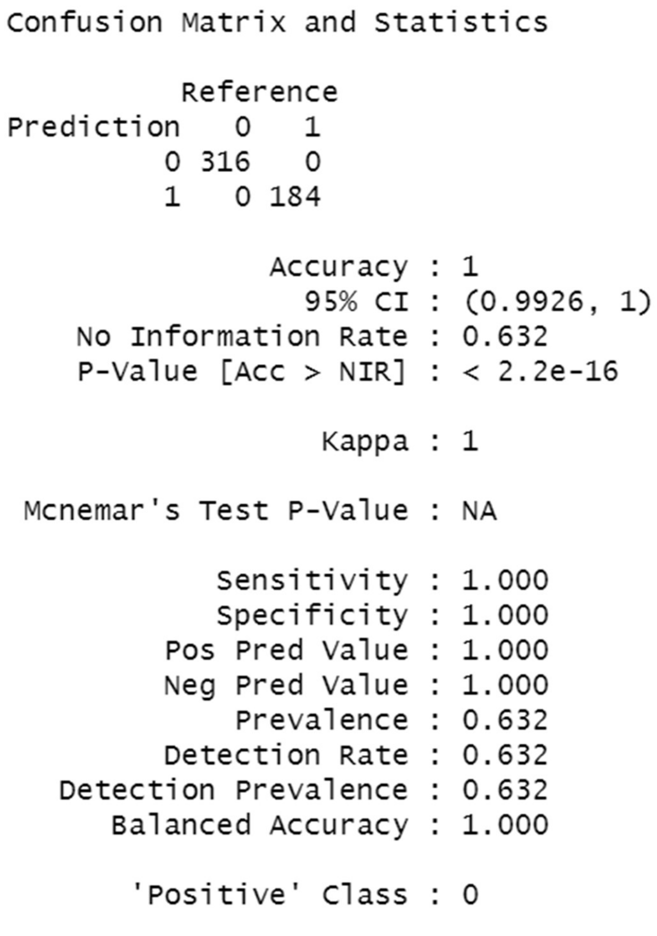
VI. Model 2: Random Forest

Random Forest was the other method used. It is an ensemble method that combines the outputs of many decision trees to create a more robust and accurate output. The code is shown below:

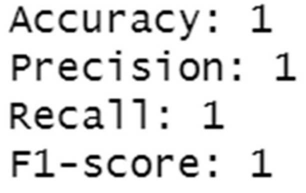


```
# SECOND MODEL RANDOM FOREST
# LOADING ANY NECESSARY LIBRARIES
library(randomForest)
library(caret)
library(vip)

# Split data into predictors (features) and outcome variable
predictors <- train[, 1:2]  # Using the first 4 columns as predictors
outcome <- train$`Pass/Fail`

# Train random forest model
model_rf <- randomForest(x = predictors, y = outcome, ntree = 100)

# Make predictions
predictions_rf <- predict(model_rf, newdata = train, type = "response")

# Evaluate model
conf_matrix_rf <- confusionMatrix(predictions_rf, test$`Pass/Fail`)
accuracy_rf <- conf_matrix_rf$overall["Accuracy"]
precision_rf <- conf_matrix_rf$byClass["Precision"]
recall_rf <- conf_matrix_rf$byClass["Recall"]
f1_score_rf <- conf_matrix_rf$byClass["F1"]

# Print summary metrics
print(conf_matrix_rf)
cat("Accuracy:", accuracy_rf, "\n")
cat("Precision:", precision_rf, "\n")
cat("Recall:", recall_rf, "\n")
cat("F1-score:", f1_score_rf, "\n")

# Print Feature Importance
vip(model_rf)
```

The same confusion matrix and accuracy metric outputs as logistic regression were calculated, listed below:

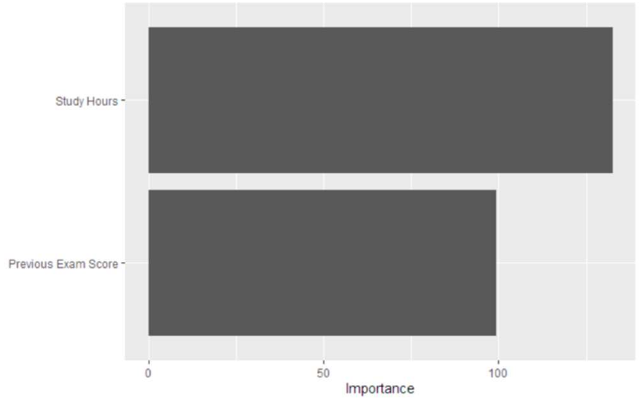


```
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 316   0
         1   0 184

               Accuracy : 1
                 95% CI : (0.9926, 1)
    No Information Rate : 0.632
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 1

 Mcnemar's Test P-Value : NA

            Sensitivity : 1.000
            Specificity : 1.000
         Pos Pred Value : 1.000
         Neg Pred Value : 1.000
             Prevalence : 0.632
         Detection Rate : 0.632
   Detection Prevalence : 0.632
      Balanced Accuracy : 1.000

       'Positive' Class : 0
```



```
Accuracy: 1
Precision: 1
Recall: 1
F1-score: 1
```

Our outputted summary metrics are all 100%, as is shown by the calculations above. This is much better than our previous model. A similar feature importance chart was also created (shown below).



This shows study hours to be significantly more predictive than previous exam scores.

VII. Conclusions

Learning analytics are becoming increasingly important as more classrooms become remote. Despite some ethical and privacy considerations, analytics have great potential to help personalize online education and provide helpful and actionable information to students and faculty alike. This project sought to predict passing or failing grades in a class based on a student's study hours put in and previous exam scores, using both random forest and logistic regression models. Study hours were found to be more important in both models. Random forest was found to be highly successful with 100% in all outputted confusion matrix metrics (accuracy, specificity, etc.), and logistic regression also found a high degree of success: considering 86.8% accuracy, 89.3% precision, 89.9% recall, 89.6% F1-Score, and 81.5% specificity. Mastering learning analytics could prove to be a large benefit to society, and almost necessary going forward.

VIII. References

[1] NCES. (2022, August). *US Education in the time of COVID*. National Center for Education Statistics. https://nces.ed.gov/surveys/annualreports/topical-studies/covid/

[2] Hill, P. (2024, January 21). *Fall 2022 IPEDS Report*. On EdTech Newsletter. https://onedtech.philhillaa.com/p/fall-2022-profile-us-higher-ed-online-education

[3] Lantz, B., Carchedi, N., & Solomon, N. (n.d.). *Supervised Learning in R: Classification*. DataCamp. https://app.datacamp.com/learn/courses/supervised-learning-in-r-classification