TRI-AD

Data Scientist Challenge

Ankush Khullar
May 7, 2020

Part 2: Model Selection

The model to predict whether a given student will pass the test was created using Jupyter Notebooks with Python. Jupyter Notebooks were used because they provide a free, open-source web tool that can combine software code, computational output, explanatory text and multimedia resources in a single document that is easy to share. Python was selected because of its extensive set of intuitive libraries for data analysis (numpy, pandas, scipy), data visualization (matplotlib, seaborn), and machine learning (scikit-learn).

1. Data Cleaning

Based on the results from Part 1, it was determined that sex, age, and dojo_class had a significant relationship to whether a student passes the test. In order to use them in the model stage, they were first mapped to integers (i.e., 1 for Male/True, 0 for Female/False).

1.1. Data Imputation

Hours studied is likely to be related to whether a student passes the test. However, 40% of the observations are missing values for the hours_studied variable. One possible solution is to remove these observations. However, this would result in the loss of a significant amount of data. Therefore, the missing values were imputed.

One possible imputation method is to replace all missing values with the mean of hours studied. However, this would reduce the variance of hours studied, which would result in a loss of information. Instead, hours studied was imputed using the k-nearest neighbors algorithm (KNN) with k = 5. The variable pass was excluded in the KNN analysis to prevent target leakage.

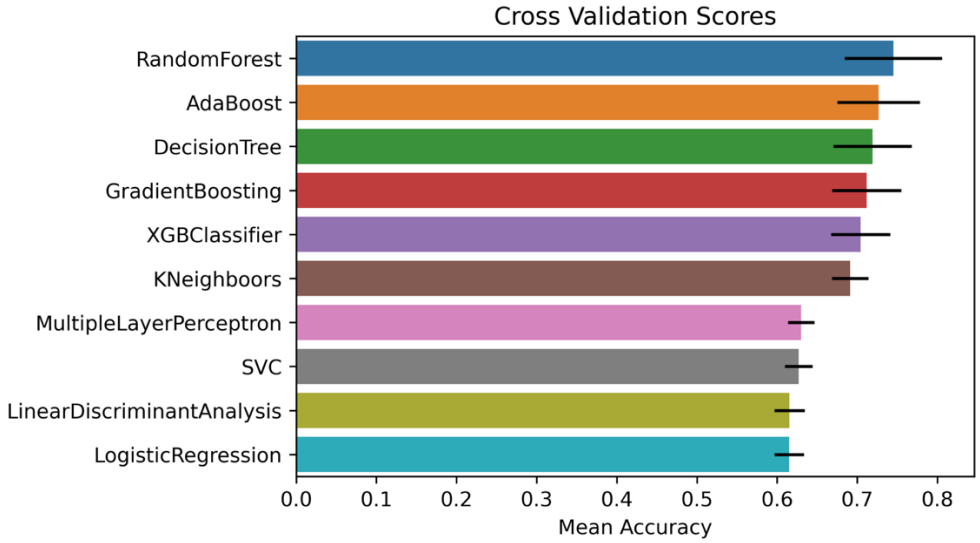
2. Model Selection

The data was split into training and test sets using 80% and 20% of the data, respectively. Approximately 78% of students in the training set passed the test, while 22% failed. Since the training set was imbalanced, the SMOTE algorithm was used to create synthetic observations of the minority class, which in this case was students that failed the test.

Several machine learning models for classification were tested, including logistic regression, support vector machine, decision tree, random forest, gradient boosted trees, multiple layer perceptron (MLP), KNN, linear discriminant analysis, AdaBoost, and XGBoost. A mix of linear and non-linear and parametric and non-parametric methods were selected to ensure that that

proper relationships between the target variable (pass) and the predictors (age, sex, hours_studied, and dojo_class) were identified. The best model was selected using 10-fold cross-validation on the training set using mean accuracy as the model evaluation criteria.

The model with the highest mean accuracy on the training set was the random forest model (*see figure below*). It is interesting to note that the models that performed the worst all assume a linear decision boundary. This indicates that the relationship between the predictors and whether a student passes the exam is not linear.



Since the random forest model has the highest mean accuracy on the training set, it was the best model. The training set accuracy for the random forest was 92.0%, and the test set accuracy was 90.7%. The fact that the training accuracy is only slightly higher than the test accuracy means that there was not significant overfitting of the training data.

The confusion matrix resulting from the random forest model can be found below. The model has precision of 75% and recall of 88% for students that fail the test and precision of 96% and recall of 92% for students that pass the test.