Playoffs Round 2: Predicting NFL Spreads, Totals, and Results

Tim Lynch, Grant McGrew, Nicole Bi, Conner Heyn, and Jalen McClain

Data Information

The process to prepare the datasets for model creation involved removing unnecessary variables, creating new variables, and joining several different datasets with the original Game Results dataset. First, the team names of teams who had moved locations, such as the Los Angeles Rams, since 2000 were changed to the team's current version of the name so that the team names were consistent. Then, the abbreviations of each team were added to the Game Results dataset to allow it to be joined with other datasets that only reported teams' abbreviations rather than their full names. We also used the score_home and score_away variables to create the Spread, Total, and Result variables that will be predicted on throughout this paper. Once the original Game Results file was cleaned, our focus shifted towards cleaning and merging each year's offensive and defensive stats to prepare them to be joined to the Game Results dataset. Since we are predicting results for 2020 games based on only about 6 or 7 games played for each team so far, all of the non-rank or non-per-attempt statistics were converted to per game averages, so that the per game averages from this year could easily be input into a model. All of the offensive statistics were given the prefix "Off_" and all of the defensive statistics were given the prefix "Def_" before being joined to the Game Results dataset by team and year for both the home team and the away team. Each statistic in the joined dataset has a "_home" or "_away" suffix indicating whether the statistic was a statistic for the home team or the away team. The final step before adding in new variables and outside data involved removing variables that would not be useful to our predictions of Spread, Total, or Result, such as the stadium or the weather, since it is extremely difficult to predict the weather weeks in advance of a game.

Once the Game Results dataset was cleaned, which contained all of the relevant offensive and defensive statistics for each team, we considered various new variables that may improve on the ability of existing statistics to predict the Spread, Total, or Result. Turnover Ratio was one of these new variables that was included in our final dataset, computed by the sum of a team's interceptions and fumbles while on offense, and divided by the sum of a team's interceptions and fumbles while on defense. This variable was important to include in our model creation process, because turnovers have a tremendous impact on a game's outcome and the Turnover Ratio variable reflects the ability of teams to maintain the ball and prevent the other team from gaining favorable field position and the ability of teams to force turnovers and provide their offense with favorable field position. Since it incorporates both offense and defense, the Turnover Ratio variable also has a better direct correlation with Spread and Result than just looking at the number of turnovers committed by the offense or the number of turnovers forced by the defense.

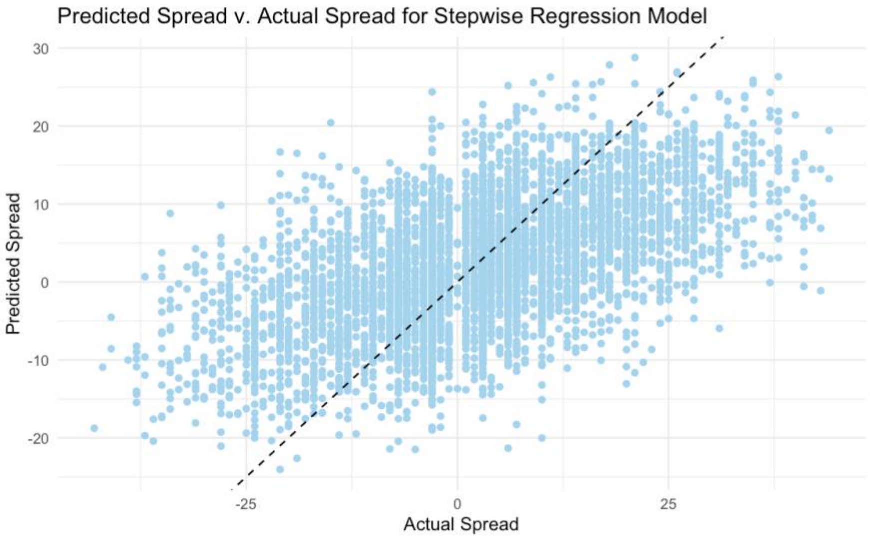
In addition to creating Turnover Ratio, we also sought several new variables from a variety of sources that would improve our ability to predict game outcomes. While the offensive and defensive statistics obtained from the original repository provide a comprehensive overview of each team's offensive and defensive capabilities, many of these statistics do not provide insight into how a team performs in high leverage situations, such as on third down or inside the red zone. A team's ability to convert on third downs is crucial, because converting allows them to extend drives to get better chances at scoring, but also keep the ball out of the other team's possession. Similarly, it is important that a team has consistent success within the red zone to maximize their win probability. Subsequently, we added values from Pro Football Reference for each team's conversion percentages on 3rd downs and each team's red zone percentage, defined by the percentage of times a team has possession inside the 20 yard line and scores a touchdown on the same drive. Additionally, we also added several statistics summarizing a team's offensive line performance, such as Run Block Rank, Pass Block Rank, and Adjusted Sack Rate, since the offensive line is also important to a team's success and many of the statistics in the original repository focused more on the rushing or passing accomplishments of a team. Finally, we also introduced the Defense-Adjusted Value over Average (DVOA) metric from FootballOutsiders.com, an advanced metric which aggregates how each team performs on each play compared to the league baseline average for the situation (down, distance, etc.). The DVOA variable was particularly helpful in developing a model for predicting spread.

Methodology for Spread

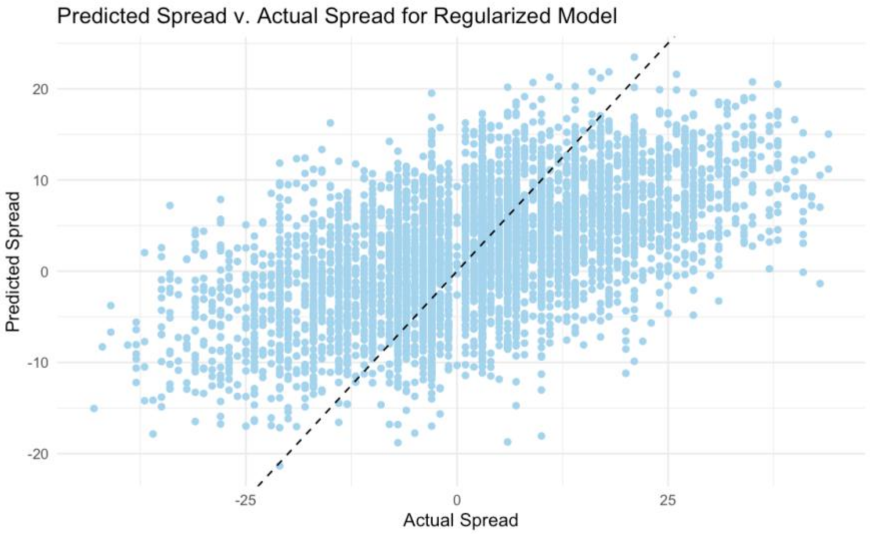
In the process of determining the best model for predicting the spread of a particular NFL game, defined by the difference between the home team's points and the away team's points, a variety of model selection methods were considered after removing any outliers in which the absolute value of the spread was 45 points or more. Since the histogram of spread for NFL games from 2000 to 2019 is roughly normally distributed, our model creation process considered a variety of both stepwise regression and regularization techniques performed via cross validation on a range of subsets of the original data. First, backwards and forward selection produced models via 10-fold cross validation on the subset of nearly all the offensive and defensive statistics in addition to the new variables previously introduced into the data. However, within the data, there was a strong prevalence of linear dependence between predictors, such as between a team's passing yards and a team's total yards, which was affecting the fit of the model. Predictors that were linearly dependent on other predictors were removed from the set of possible predictors, and backwards, forward, and stepwise selection procedures via 10-fold cross validation were redone with this new subset of predictors. Performing these procedures improved the fit of each model to the test data, resulting in reduced RMSE values. Of the three methods, backwards selection and stepwise selection each produced very similar models with RMSE values lowering than those produced via forwards selection.

Before further considering potential adjustments to the models produced via stepwise regression methods, we also considered whether regularization methods, such as ridge regression, lasso, and elastic net methods, would improve the model's predictive ability. To determine the best regularization model using the cv.glmnet function in R, 5 models were fitted with a range of alpha values from 0 to 1 to determine which model would produce a lambda value corresponding to the minimum RMSE. This process was repeated not only on a subset with the majority of the possible predictors in our data, but also on a subset with only about 15 of the possible predictors in our data that had strong correlations with spread or had been routinely been selected as significant predictors in our stepwise regression models. To further decrease the subset of predictors, new predictors were created by the difference of the home and away team's values for a given statistic. This smaller subset of predictors was useful, because the dimension of the matrix considering all possible interactions between these 15 predictors was so small that the cv.glmnet function could effectively produce a model considering all possible interaction effects between these 15 predictors. The lasso model produced using the smaller subset of predictors had the lowest RMSE of the regularization models considered.

Of all the models considered, the model produced via stepwise regression and 10-fold cross validation had the lowest RMSE with an RMSE value of 11.92, consisting of the Vegas Predicted Spread, Defensive Expected Points for the home team, both the offensive and defensive points scored for both the home team and the away team, and the DVOA for both the home team and the away team. The fit of the model to the test data was improved by adding the interaction between the home team's offense and the away team's defense as well as the interaction between the away team's offense and the home team's defense. The addition of these interaction terms made this model the best model for predicting spread of all the models we considered. Not only does this model minimize the error on predicting the test data via k-fold cross validation, the residuals are also normally distributed, and there is a positive relationship between the predicted spread and actual spread of a particular game, as seen in the below graph.



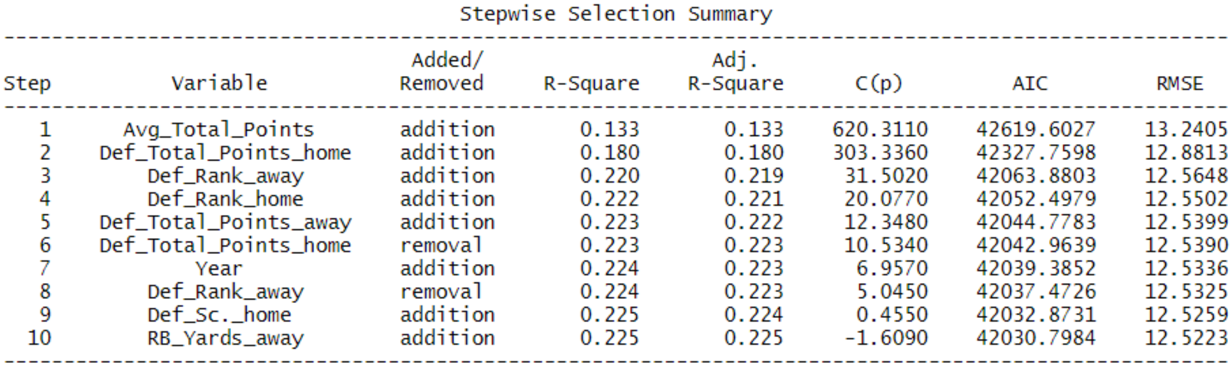
This model, however, can only be applied to make predictions when the Vegas Predicted Spread is known, which means that a different model will be used to predict the last two weeks of games. For models produced without the Vegas Predicted Spread in the set of possible predictors, both the model produced via stepwise regression and the regularized model produced with the smaller subset of predictors that considered all possible interactions had similarly low RMSE values. For the regularized model, a range of alpha values were considered to find the optimal lambda value at each alpha value. The cross validation error is minimized when alpha is 1, resulting in a model predicting Spread consisting of each team's DVOA values as well as the interaction between DVOA values with a number of other predictors, including each team's total points, offensive ranks, 3rd down conversion rates, turnover ratio, scoring percentage, and yards per attempt. Even though this model and the one produced via stepwise regression had similar RMSE values, we ultimately chose the model produced via regularization to be a better predictor of the spread when the Vegas Predicted Spread is unknown, since the penalization criteria of regularization methods generally leads to less variable prediction error. Below is the relationship between the predicted spread and the actual spread for the regularized model, revealing similar performance to the model used when the Vegas Predicted Spread is known.



Methodology for Total:

When trying to create the best model for predicting the Total Score of an NFL game, also known as the ‘Over/Under,’ we considered a number of different models throughout our selection process. To begin, we started by looking at the distribution of the total scores over the last 20 or so NFL seasons. When looking at a histogram, the distribution of total scores displayed a shape similar to that of a bell curve indicating that our data was relatively normal. We then proceeded to create subsets of the data to only include observations that would likely be useful for predicting scores of games for this 2020 season. Subsequently, we decided to remove all games prior to the 2007 NFL season, due to these seasons having much lower average team point totals compared to succeeding seasons. The average team point totals from 2000-2006 were around only 334 points, whereas average team point totals in the years following climbed and averaged 360 points per season. This illustrates how NFL games have become increasingly higher scoring over the years due to higher powered offenses. Additionally, very few players who played in 2006 or earlier are still in the NFL. Next, we decided to remove outliers, or games that had extremely low scores or extremely high scores. Therefore, we removed games that had a total score of less than 13 points and those that had a score of 86 points or higher, which previously accounted for only 1 percent of our total dataset (33 out of 3741 observations). We also decided to remove a number of variables that we found were linearly dependent on one another using the function ‘detect.lindep()’ in the ‘plm’ package. As previously mentioned, many of the total offensive and defensive statistics were correlated with the passing and rushing statistics that composed the totals and affected the fit of our models. This cut down our total number of variables from 145 to approximately 87.

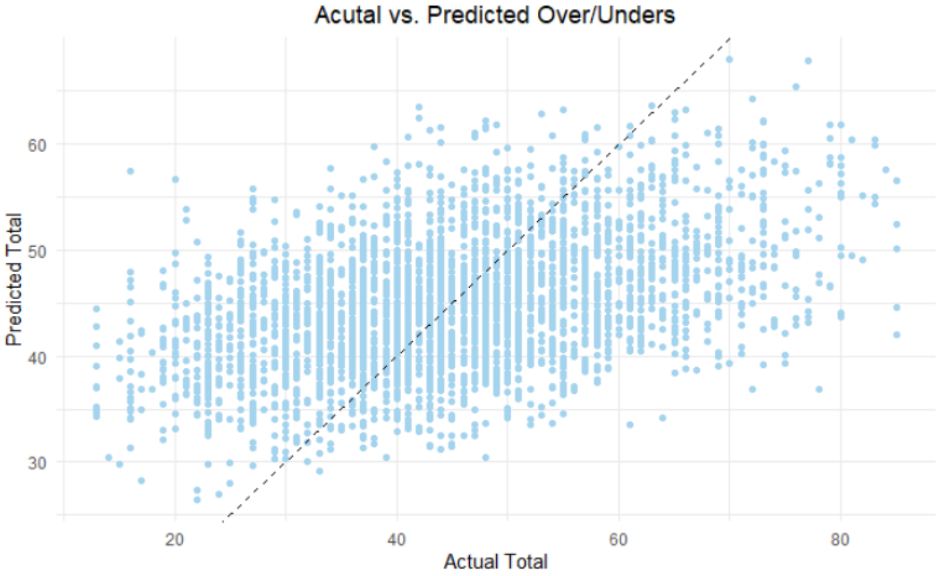
Following this, we performed forward, backward, and stepwise variable selection methods to select the variables that would be the most useful for predicting the total scores of NFL games. Both our forward and stepwise selection methods came up with very similar variables and we ultimately decided to utilize the variables chosen in the stepwise method. The variables that we chose to use can be seen in the table below.



```
                                  Stepwise Selection Summary
-----------------------------------------------------------------------------------------------
                                 Added/                  Adj.
Step          Variable           Removed     R-Square   R-Square      C(p)         AIC        RMSE
-----------------------------------------------------------------------------------------------
   1      Avg_Total_Points       addition      0.133      0.133     620.3110    42619.6027    13.2405
   2    Def_Total_Points_home    addition      0.180      0.180     303.3360    42327.7598    12.8813
   3        Def_Rank_away        addition      0.220      0.219      31.5020    42063.8803    12.5648
   4        Def_Rank_home        addition      0.222      0.221      20.0770    42052.4979    12.5502
   5    Def_Total_Points_away    addition      0.223      0.222      12.3480    42044.7783    12.5399
   6    Def_Total_Points_home    removal       0.223      0.223      10.5340    42042.9639    12.5390
   7            Year             addition      0.224      0.223       6.9570    42039.3852    12.5336
   8        Def_Rank_away        removal       0.224      0.223       5.0450    42037.4726    12.5325
   9        Def_Sc._home         addition      0.225      0.224       0.4550    42032.8731    12.5259
  10        RB_Yards_away        addition      0.225      0.225      -1.6090    42030.7984    12.5223
-----------------------------------------------------------------------------------------------
```

To reiterate, the final variables used in our model were ‘Avg_Total_Points,’ which were the combined points averages of the two teams playing against one another, ‘Def_Rank_home,’ ‘Def_Total_Points_away,’ ‘Year,’ ‘Def_Sc._home,’ and ‘RB_Yards_away’. ‘RB_Yards_away’ stood for “Run Blocking Yards away” and was one of the variables we brought in from an outside source. This variable could be important, because it is crucial to establish a good run game in order to truly have a balanced and high scoring offense.

After going through the process of variable selection, we created a number of different models utilizing various different regression methods. The first method we tried was a simple least squares regression model using 10 fold cross validation. This model produced very good results and had a mean absolute error of only 9.61 on our test data. We also tried to implement a number of interaction terms on this model, as well as polynomial terms. However, none of the interaction/polynomial terms we attempted to use seemed to improve the model and none of them was statistically significant. We then considered a number of regularization methods such as ridge, lasso, and elastic net regression models that would hopefully improve the predictive power of our model. For these models, we used cross validation and we included the 87 variables that were not linearly dependent on one another to allow the regression models to operate on more than 6 variables. Lasso regression models especially are able to get rid of or exclude unnecessary parameters in their models. To create these models, we utilized the 'cv.glmnet()' function, and all three methods produced similar results to that of our linear regression model. These models resulted in a slightly higher mean T absolute error with the lasso model performing the best. Although, the lasso model resulted in slightly worse mean absolute error (9.64) we ultimately decided to use this model as it incorporates a penalization factor that allows it to predict better in the long run due to a small amount of bias. The predicted total scores were plotted against the actual total scores of our data set and can be seen in the graph below.



We also created a random forest regression model, but it did not predict as well as the regularization models we had already created. Ultimately, we chose to use the lasso regression model to predict the total score of upcoming games, because it had what we believed to be a relatively low MAE and did not over fit like most least squares regression models do.

Methodology for Result

To begin the model creation process for Result, we executed forwards, backwards, and stepwise selection methods on our given set of predictors. For this set of predictors, we used the offensive and defensive statistics, conversion percentages, offensive line statistics, and our 'TurnoverRatio' metric. A little extra cleaning was done to the data before building these models. We updated the Expected Points metric given in the Offensive and Defensive datasets to a per game average, and we also converted all percentages to be decimals. A generalized linear model was used as the baseline for these initial selection methods given the binary outcome of Result. For each prospective model, we used the train function in the caret package to do 10 fold cross validation and compute a prediction accuracy. This function automatically takes care of computing the train and test sets and can be used with a variety of different models. Forwards selection gave us a model with 18 predictors and a 71.2% prediction accuracy across the test sets. The backwards selection model had 24 predictors with a 71.5% prediction accuracy and the stepwise model had 14 predictors with a 71.1% prediction accuracy. Due to the very similar accuracies, but the added simplicity of the stepwise model only having 14 predictors, we decided to do a little more exploring on this set of predictors. In this next stage, we used different types of models on this predictor subset. A K-Nearest Neighbors model gave us a 67.3% prediction accuracy. We also tried a glmnet model, which resulted in a 71.4% prediction accuracy, a linear discriminant analysis with a 71.2% accuracy, and a random forest model with a 67.1% accuracy. Lastly, we ran a neural net classification model. This attempt gave us a 71.4% accuracy in the cross-validation evaluation. Based on these examples, the other types of models did not do much to improve upon the basic generalized linear model and were worse at predicting the result in some cases.

For comparison, we took a step back from the stepwise predictor set to experiment with some other ideas. We figured that team rankings in certain categories would serve as a good comparison between teams and be effective at determining the winners and losers of games. For this model, we took the offensive and defensive ranks, as well as some of the offensive line rankings we found online, and used them to assign each team a quadrant ranking in each category. Since the ranking variables assigned a number 1-32 to each team in a given year, we converted these to values 1-4 in order to get more observations in each category. Teams 1-8 were given a value of 1, teams 9-16 were assigned a value of 2, teams 17-24 were placed in the 3 category, and teams 25-32 were given a 4. These rankings were then factored and used as the predictors for Result. We then tried a generalized linear model, which resulted in a 69.4% accuracy, and a random forest model with a 67.3% accuracy. Adding some interactions between offensive and defensive ranks and pass and run blocking rankings to the generalized linear model gave a 69% accuracy. Changing the quadrant ranking system to a 1-8 ranking system had a 69.8% accuracy with the generalized linear model. Based on our group's knowledge of football, we also theorized that a model based on third down conversion and red zone percentages could be a good predictor of winning teams. However, a generalized linear model based on these predictors only achieved a 63.3% accuracy in the cross-validation analysis. Adding our created 'TurnoverRatio' metric to this model improved the accuracy to 67.9%, but a random forest model with the same predictors resulted in a 65.2% accuracy. After this, we added sacks to the group of predictors in the generalized linear model, but the accuracy stayed at 67.9%. Some experimental graphing of predictors and Result led us to try a simple model with our 'Avg_Total_Points' metric (sum of the home and away teams' offensive total points) and the defensive rank of the home team as our only two predictors. There seemed to be a pretty evident split in Result based on plotting these predictors, but the generalized linear model created from these predictors ended with a 62.8% accuracy through cross-validation.

At this point in the process, we were yet to find a model that improved much upon the initial models created from our selection methods, so we went back to experimenting with the smaller 14 predictor model from stepwise selection. To start again, we began by adjusting this predictor set to make it more consistent. The initial stepwise set included many variables for only one of the teams. For instance, it included 'Off_Passing_Y/A_away', but not 'Off_Passing_Y/A_home'. We thought it made logical sense to include predictors for both teams, so we added several variables to this model. We also changed 'Sacks_home' to 'Adj_Sack_Rt_home' in order to match it with 'Adj_Sack_Rt_away', and we took out 'Off_Passing_Yds_away' because it seemed redundant with 'Off_Passing_Y/A_away'. The generalized linear model we constructed based off of this adjusted predictor set scored a 71.4% accuracy with cross-validation. This accuracy was just about the same as before. Through data examination, we noticed that some statistics seemed to change over the years, and we had seen some changes on other models when limiting the range of years in the data set. We tried another generalized linear model on this predictor set and only included the years 2010-2019, but the model did not improve any with a 71.3% accuracy. Going back to including all years, we tried some more on expanding the adjusted stepwise predictor set. In this next step, we ran several models on a new predictor set that added two interaction terms. We included an interaction term between ‘Off_Total_Points_home’ and ‘Def_Total_Points_away’ and another one between ‘Off_Total_Points_away’ and ‘Def_Total_Points_home’. Each of these predictors was already included individually in the predictor set. We tested a generalized linear model with a 71.5% accuracy, a random forest model with a 69.2% accuracy, and K-Nearest Neighbors and glmnet models with 69.1% and 71.4% accuracies, respectively. Once again, we failed to see any improvement upon our top accuracies achieved through cross-validation analysis. Earlier in our Result model creation process, we saw a 4.6% accuracy improvement from adding our ‘TurnoverRatio’ metric to a model. We tried adding this to our predictor set with the interaction terms and experimented with the same models, but the top performing glm and glmnet models still only scored a 71.4% accuracy. Due to not seeing much improvement in accuracy, we fitted a generalized linear model with the Vegas spread (converted to Home score - Away score) as the only predictor of result to use as a baseline for comparison. This model only achieved a 66.1% accuracy, which made us feel slightly better that the Vegas prediction was actually worse than our current model.

As a final step, we decided to test a couple more potential interaction terms. On top of the adjusted stepwise model with ‘Total_Points’ interactions and the ‘TurnoverRatio’, we added two more interaction terms between `Off_Passing_Y/A_away’ and ‘Def_Passing_Y/A_home’ and between ‘Off_Passing_Y/A_home’ and ‘Def_Passing_Y/A_away’. We only considered a generalized linear model and a glmnet model with these predictors, due to their general outperformance of other model types in our previous attempts. During cross-validation analysis, both models produced a 71.4% accuracy. After this attempt, we decided to take out this second group of interaction terms to create our final model. Due to the ever-so-slight improvement of the first interaction terms and the importance of the ‘TurnoverRatio’ in an earlier model, we decided to leave these variables in our final predictor set. Over the course of the model making process, generalized linear models produced the highest accuracies through cross-validation, so we decided to use a glm for our final predictive model. After fitting a generalized linear model on our predictor set consisting of the adjusted stepwise variables (adjusted for consistency of variables included for each team), ‘Total_Points’ interactions, and the ‘TurnoverRatio’ metrics for each team, we predicted the Result of each game and added them to our final data set.