Machine Learning Report:

Predicting the Outcome of Men's College Basketball Games via the Utilization of Gradient Boosting

Conor McGrath -Tiffany Tseng -Tina Wang - Michael Mayor - Brock Gallagher

Machine Learning ITAO 70310

Professor Martin Barron

11/23/2020

Abstract

In this report we use gradient boosting to predict the outcome of men's and women's college basketball games. Our data set was immense with over 92,000 observations that stretch 17 years for men and over 50,000 observations that stretch 10 years for women. After we explain our data, tuning methods, selection of hyper parameters, and model selection we analyze how our model is able to successfully predict the score difference in a game with a test RMSE of 11.06815/12.48979 (M/W) and how it is able to successfully predict the binary outcome of a game with 73.08%/75.16% (M/W) accuracy. Our exploration into our models performance highlights the difference between the machine learning methods and the powers of gradient boosting for datasets similar to ours.

Introduction

The end goal of our machine learning project was to predict the outcome of an upcoming college basketball game. In order to do this we used previous performance data to build a model in which we could predict which team will ultimately win a particular game and by how much that team will win by. This is an interesting problem because many have tried to determine the outcome of a college basketball game in order to make money via gambling. However, solving this problem is also important to college coaches as it can allow them to determine what factors are most important in determining the outcome of a game. Understanding the determinants of a victorious basketball game can help college coaches make strategic adjustments based on data in preparation for a particular opponent.

Related Work

The odds of predicting a completely perfect bracket are nearly impossible “one in 9.2 quintillion” to be exact, but that has not stopped thousands of data scientists every year competing in a machine learning competition hosted by Google (Weininger). Data scientists compete to not only predict the outcome of the tournament but to also compete on their level of certainty for every possible tournament game outcome. This tournament has brought a lot of attention to machine learning and has shown many people the interesting relationship between sports and data science. Using a data set that is familiar to the world has made machine learning more approachable for others and has highlighted some of the key topics in machine learning making March Madness a perfect example of how machine learning can be utilized.

Data Description

Our dataset is *Men’s and Women’s Regular Season Detailed Results* provided by Google Cloud & NCAA March Madness Analytics. The Men’s dataset contains game by game team-level box scores for all regular seasons of historical data since the 2003 season. More specifically, we have chosen 92,832 observations and 50 variables with our target variable being the difference in the score (result). The women’s dataset has 56,793 observations of 48 variables from the historical data since the 2010 season. The variables and structure are identical to the Male Regular Season Detailed dataset except for the conference ranking. While the dataset only contained game by game variables we believed it would be

important for our model to contain important cumulative season averages for each team. Because we used this data set to create our own cumulative season variables we chose to exclude all observations from early on in the season so that our model would not have any NA's in it. There were also other observations with NA's that we chose to omit as they were a very small amount of our data. The derived variables that we created to include in our model can be found in Appendix #1.

Besides the cumulative average for each team we also believed that it would be important to include the conference ranking. We used this variable to help strengthen the comparison for out of conference games. These variables were not included in the original dataset so we sourced this information from another dataset and joined it to our master dataset. Unfortunately, we were only able to find this information for the men and were not able to locate this information for the women.

Before we began modeling we split our data set into training data and testing data. Our train data set had 80% of the observations with the test data set containing the remaining 20% of the observations. We ensured that the data was split evenly so that both datasets contained similar information to protect the accuracy of our model.

Methods

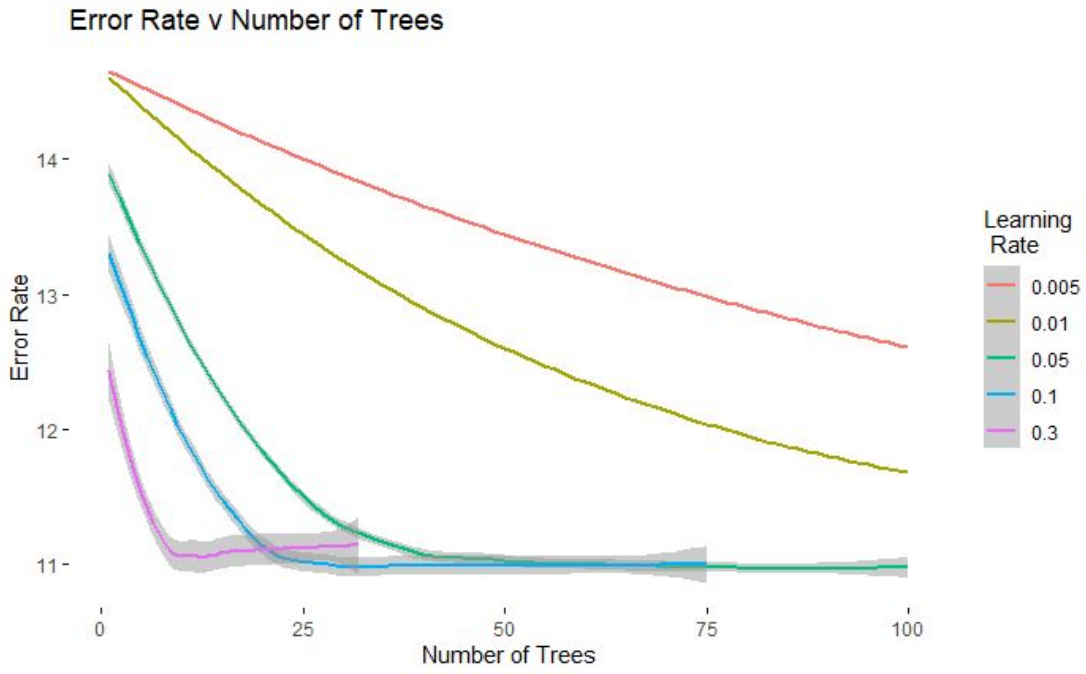
Our machine learning project required the implementation of a model that can handle regression prediction well, for that reason we chose to use gradient boosting. As Jerome H. Friedman, the data scientist from Stanford University who developed explicit regression gradient boosting algorithms, stated "gradient boosting constructs additive regression models by sequentially fitting a simple parameterized function (base learner) to current "pseudo"-residuals by least squares at each iteration." (Jerome) In more simple terms our team favored Dr. Shirin Elsinghors description of gradient boosting as "an ensemble learner". This means it will create a final model based on a collection of individual models. The predictive power of these individual models is weak and prone to overfitting but combining many such weak models in an ensemble will lead to an overall much improved result." (Elsinghorst) Essentially, what our model is

doing is building many models based off of small samples from our data set. The final model that we use for our prediction is a combination of all of these models.

Results

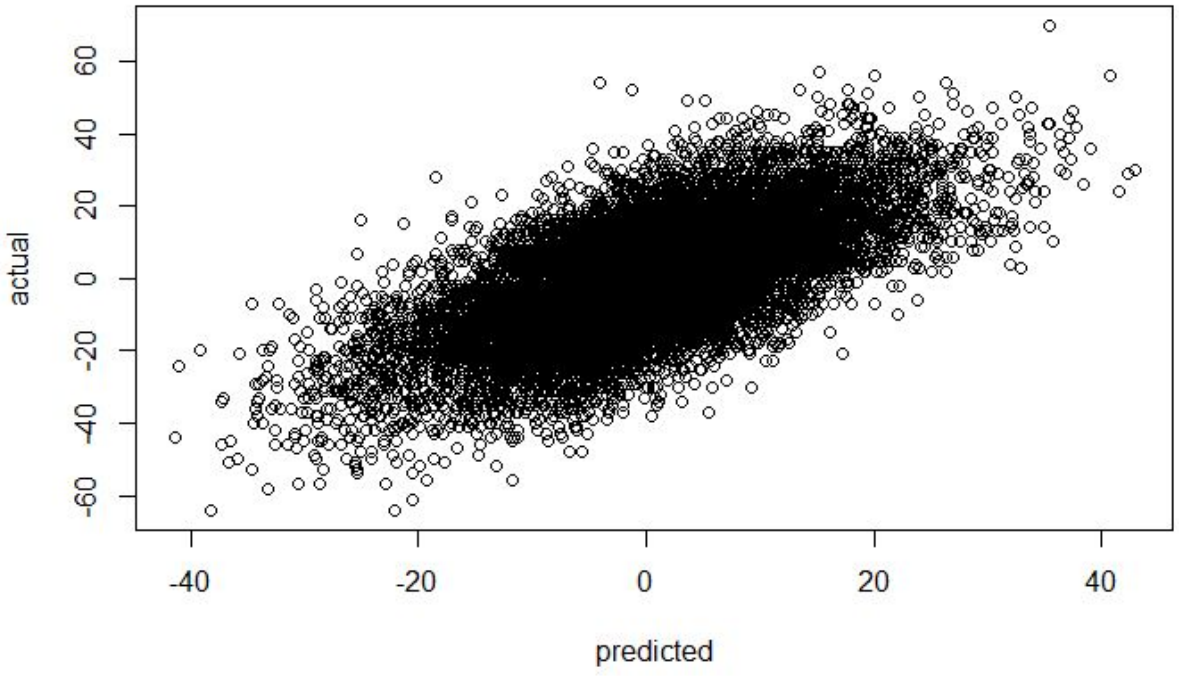
Identifying the hyper parameters that we used in our model was our first step, the hyper parameters we identified were the optimal number of iterations and ETA. In order to find the optimal number of iterations that we would use for our final model we used the function xgb.cv. We identified the optimal number of iterations as 14 and it is important to note that for our model to determine the optimal number of iterations when we ran the xgb.cv function we changed the default learning rate from 0.3 to 0.1.

We tuned our model by adjusting the ETA for five different models where the learning rates were (.005 - .01 - .05 - .1 - .3). As you can see in the visualization below our optimal model utilized a learning rate of 0.1.



When we analyzed our tuning models in more detail the model with a learning rate of 0.3 actually had the lowest error rate but because the line oscillates upward after reaching its minimum point we did not use that learning rate and instead used the learning rate of 0.1.

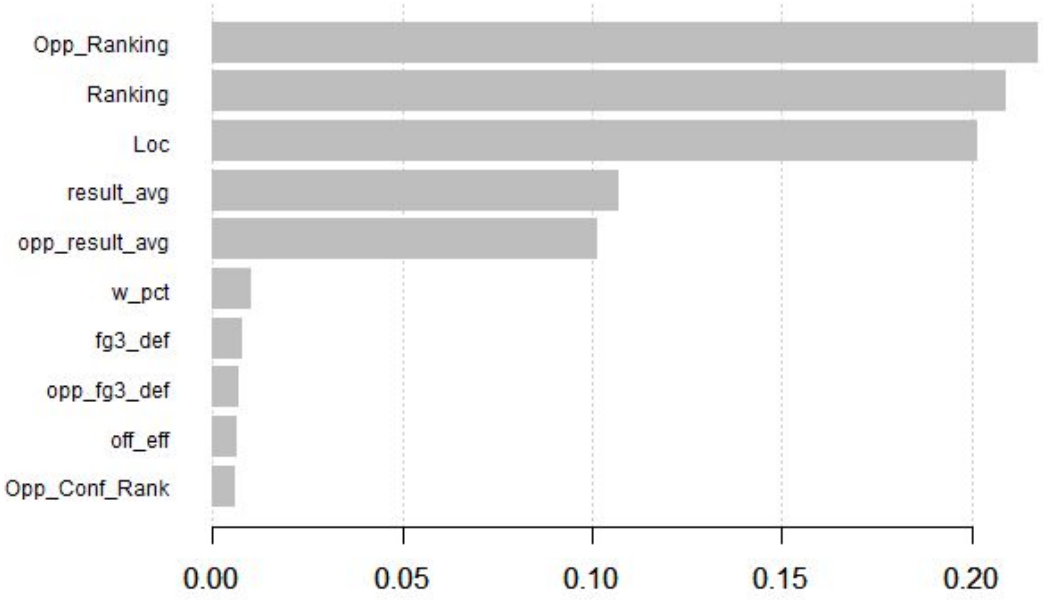
After we had determined our hyperparameters we were ready to run our final model on the test data that we would use for our final predictive modeling. Because we are building a regression model we chose reg:squarederror (which is the default) to be our objective function. We were very pleased with our models performance which had a RMSE Training of 10.11784 and a RMSE Testing of 11.06815. Below is a scatter plot that maps our model's predicted outcomes again the actual outcomes.



Our model shows a nice upward slope which tells us that our model has done an accurate job predicting the target variable. When our model predicted a large negative outcome (meaning that the result team lost by a large number of points) the actual event was in-fact a large negative outcome. Furthermore, when our model predicted a large positive outcome (meaning that the result team won by a large number of points) the actual event was in-fact a large positive outcome. This visualization also provides us with a better understanding of our RMSE testing statistic and the issues that it presents. When looking at the extremes of the observations there is clearly more distance between the predicted and actual results. If we were to train our model on outcomes that were only between -20 and 20 we believe that our RMSE would be less than our observed 11.06815.

We also wanted to test our models ability to predict an event as either win or loss. When we tested our model using a confusion matrix our model shows as good if not better predictive capability. The accuracy of our model was 73.2%.

Finally, we wanted to identify the most important variables for our model. A visualization for the top 10 variables can be seen below.



-- **WOMEN'S SECTION TO ADD**

We followed the same methodology when predicting the women teams' results. Unfortunately, we are not able to go in-depth on our process due to length constraints but we do describe the major findings later in the report.

Comparison of other models

In addition to the XGBoost model, we also built a linear regression and a random forest to compare the results and ensure that gradient boosting was the best method to use for this dataset.

Linear regression - Men's

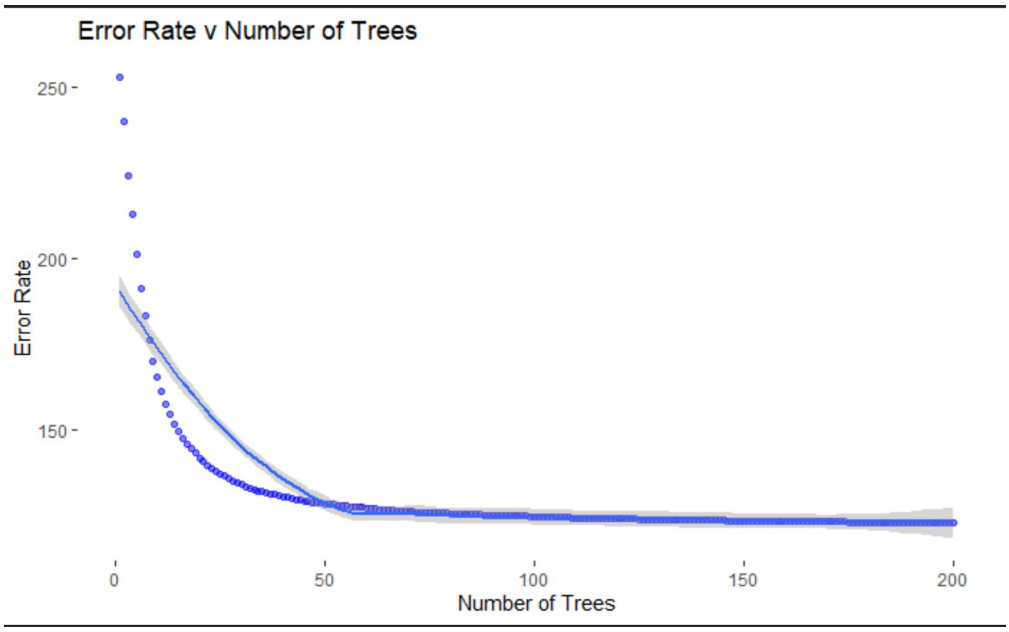
Linear regression model had a train RMSE of 10.99 and a test RMSE of 11.09 which is similar to our gradient boosting but still inferior which supports that gradient boosting was the best method to choose.

Linear regression - Womens

Linear regression model had a train RMSE of 13.29503 and a test RMSE of 13.40878 which is similar to our gradient boosting but still slightly inferior making gradient boosting the best method to choose.

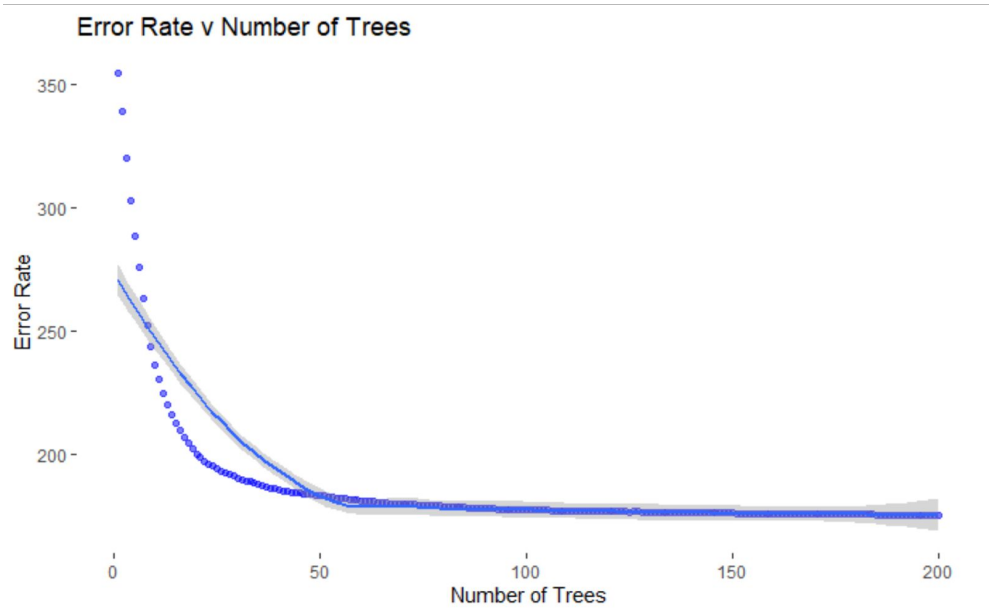
Random Forest - Men's

The random forest model for men's regular season score differential prediction has a RMSE Training of 4.476794 and a RMSE Testing of 11.15636. For the win or loss prediction, the confusion matrix shows an accuracy rate of 72.75%. In comparison, the XGBoost model has slightly lower RMSE, and slightly higher accuracy and sensitivity rates, which make it a better model for both numerical and classification prediction. As you can see in the graphic below the optimal number of trees for this model was 100.



Random Forest - Women's

The random forest model for women's regular season score differential prediction has a RMSE Training of 5.300194 and a RMSE Testing of 13.25185. For the win or loss prediction, the confusion matrix shows an accuracy rate of 72.72%. In comparison, the XGBoost model has slightly lower RMSE, and slightly higher accuracy and sensitivity rates, which make it a better model for both numerical and classification prediction. As you can see in the graphic below the optimal number of trees for this model was 100.



There is a visual representation for the comparison of all of our models for both men and for women that can be found in Appendix #2 .

Discussion

When looking at our model's variable importance there are interesting takeaways. Most notably are the team ranking and opponent ranking. This makes a lot of sense; if team A has a better ranking than team B, team A should win the majority of the time and that is because the committee that determines a team's rank is considering many of the variables that are included in our model.

The second most important variable is location. This is very interesting as our model proves that the “home team advantage” is in-fact a variable that contributes strongly to the outcome of a game.

The next most important variables are result average and opponent results average. These variables measure a team’s average score differential in a certain season, leading up to their next game. If a team has a large positive result average that means that they tend to win by a large margin and if a team has a large negative result average that means that they tend to lose by a large margin.

Conclusion and Future work

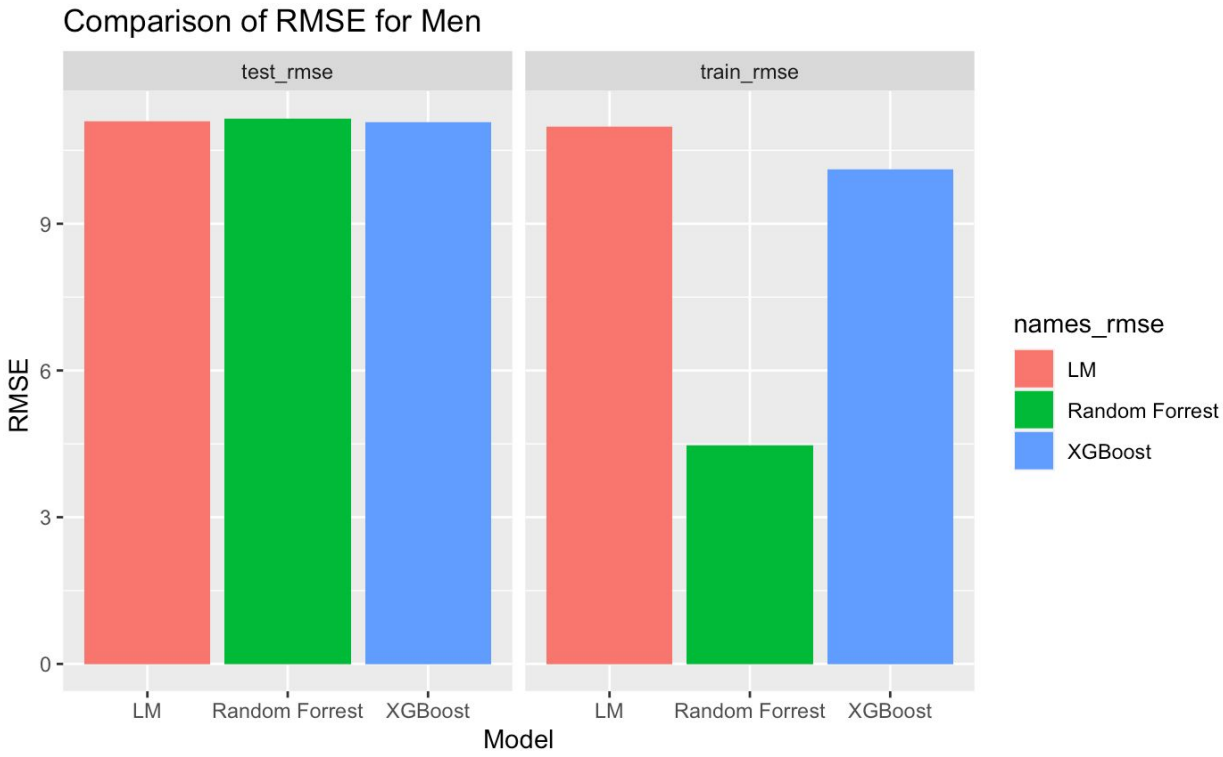
In conclusion our team was able to build a successful machine learning model that can predict the difference in a games ending score with a RMSE of 11.06815 for men and a RMSE of 12.49 for women while being able to predict the binary outcome (W/L) with an accuracy of 73.2% for men and 75.16% for women. . Throughout this project we learned that the most important variables are team ranking, the location (home or away), and the average results (by how much a team is winning or losing by on average).

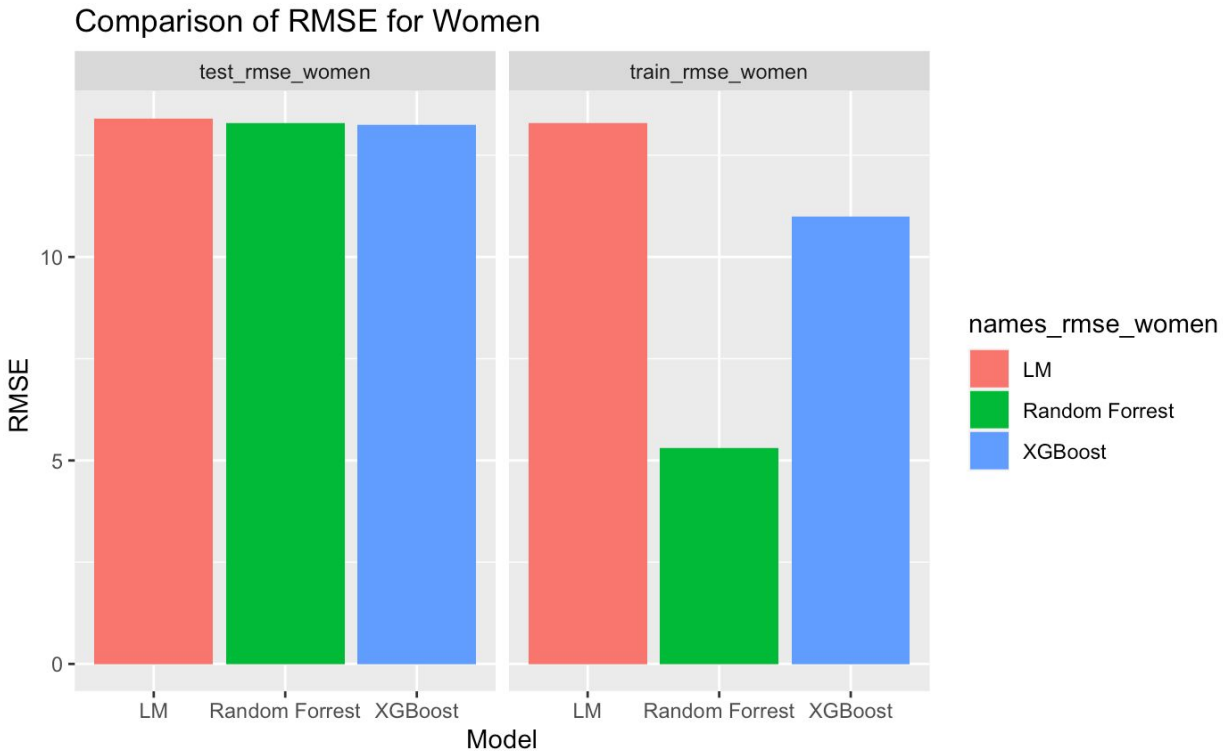
After analyzing the results of our models there are some changes we wish we could make if we had more time and resources. Firstly, we realized that location was a very important variable but we would have liked to add one more variable that measured the distance a team had to travel for an away game to see if that generated any more predictive power. Additionally, we would also be interested in seeing the importance of the team's “average player age” variable on the outcome. We believe that experienced teams (those with a majority of upperclassmen on their roster) may be more likely to be successful than younger teams with little to no experience. Another important variable we could include would be number of injured players. We believe teams that play without many of their players due to injury will be at a great disadvantage. Finally, we only created training and test datasets; with such a large number of observations we believe that we could have also created a validation dataset to confirm our accuracy and would have done so under different time constraints.

Appendix #1

- WFGAcc - Winning Team Field Goal Accuracy
- WFG3Acc - Winning Team 3 Point Field Goal Accuracy
- WFTAcc - Winning Team Free Throw Accuracy
- LFGAcc - Losing Team Field Goal Accuracy
- LFG3Acc - Losing Team 3 Point Field Goal Accuracy
- LFTAcc - Losing Team Free Throw Accuracy
- NumWPoss - Number of Possession (Winning Team)
- NumLPoss - Number of Possession (Losing Team)
- WOffEff - Winning Team Offensive Efficiency
- WDefEff - Winning Team Defensive Efficiency
- LOffEff - Losing Team Offensive Efficiency
- DefflEff - Losing Team Defensive Efficiency

Appendix #2


Comparison of RMSE for Men
test_rmse
train_rmse
RMSE
9
6
3
0
LM
Random Forrest
XGBoost
Model
names_rmse
LM
Random Forrest
XGBoost


Comparison of RMSE for Women
test_rmse_women
train_rmse_women
RMSE
10
5
0
LM
Random Forrest
XGBoost
Model
names_rmse_women
LM
Random Forrest
XGBoost

Bibliography

Best_Schools. “25 Fun Facts About March Madness.” *TheBestSchools.org*, Thebestschools.org, 13 Nov. 2019, thebestschools.org/magazine/25-fun-facts-about-march-madness/.

Dewey, Conor. “Machine Learning Madness: Predicting Every NCAA Tournament Matchup.” *Medium*, Towards Data Science, 23 Jan. 2020, towardsdatascience.com/machine-learning-madness-predicting-every-ncaa-tournament-matchup-7d9ce7d5fc6d.

Elsinghorst, Dr. Shirin. “Machine Learning Basics - Gradient Boosting & XGBoost.” *Shirin's PlaygRound*, 29 Nov. 2018, www.shirin-glander.de/2018/11/ml_basics_gbm/.

Friedman, Jerome H. “Stochastic Gradient Boosting.” *Computational Statistics & Data Analysis*, vol. 38, no. 4, 2002, pp. 367–378.

Weininger, Lotan. “How We Predicted March Madness Using Machine Learning.” *Medium*, Medium, 19 Oct. 2019, medium.com/@lotanweininger/march-madness-machine-learning-2dbacc948874.

Zifan Shi, Sruthi Moorthy, Albrecht Zimmermann. “Predicting NCAAB match outcomes using ML techniques – some results and lessons learned” https://arxiv.org/pdf/1310.3607.pdf