Team 1

March Madness

**Introduction**

March Madness is the biggest stage for college basketball. Where 68 collegiate teams from all over the country — both big and small — compete for seven rounds in a single elimination tournament to determine who is the best team in college basketball that year. One of the biggest appeals for the college basketball fanbase is the infamous bracket. The bracket is a grid of all the seeded teams that maps out their path to the championship. People participate in office pools or online groups to guess the winners of each game to either win bragging rights or a lump sum of money. The initial days of March Madness are the best days of sports because not only will it have you glued to your couch and TV to see buzzer beaters, Cinderella stories, breakout stars, and unforgettable moments, but in the matter of a weekend, your whole bracket could implode and leave you speechless.

Every year, millions and millions of brackets get filled out for the tournament. People win hundreds if not thousands of dollars by accurately predicting the winners of 67 games. These 67 games encompass some of the most exciting moments in sports and the odds of predicting the winner of every game correctly is 1 in 9.2 quintillion, yet every year millions of people still try to beat the odds. Due to the very low likelihood of achieving a perfect bracket, people have turned to data science to try and raise the odds.

While the odds are low in correctly predicting the outcome of the NCAA Tournament, it’s plausible to take a data-driven approach to predict the winners of each game and eventually the winner of the tournament. Every year, data scientists and statisticians compete in tournaments of their own via Kaggle to try and predict a winner using machine learning methods and tools. Our group decided to join in on the fun and try our hand at predicting the final champion. The major questions we wanted to explore were how do people analyze this data? What do the results mean? And what can be done with this data to predict a winner? The analysis and models below will explain how past performance of college team's data is helpful and crucial in predicting the winner of March Madness.

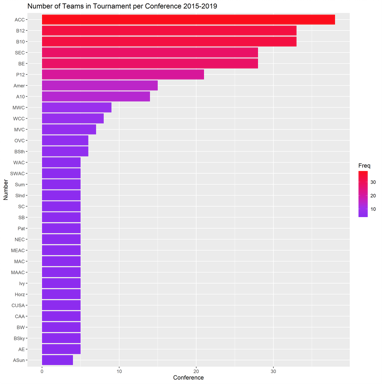
**Analysis and Models**

The data that was analyzed for this project was a dataset downloaded from Kaggle that had season statistics for NCAA men’s teams from the years of 2016-2020. Unfortunately, the 2020 data could not be used to predict the tournament outcomes because there was no tournament held last year due to Covid-19. The data consisted of several rows of team data and variables that are believed to be the factors in what contributes to a team's success. We preprocessed the data to manipulate the fields which allowed us to calculate different basketball statistics that we thought would determine a champion of the tournament.

**Exploratory Data Analysis**

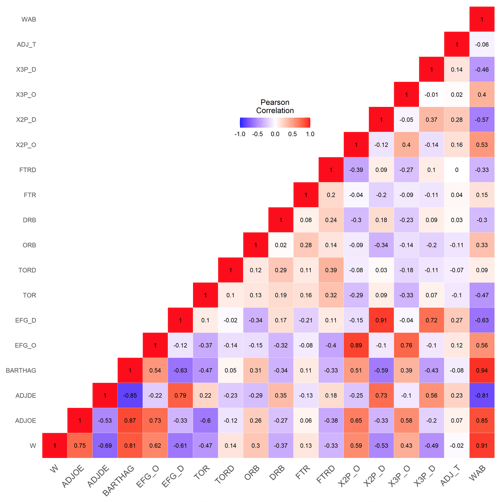
To determine which statistics would contribute to a team winning games, we began the Exploratory Data Analysis by looking at the dataset and determining which variables were most correlated with defeating another team. At first, we looked at the number of teams in the tournament from certain conferences, performed a Pearson correlation, and ran a correlation of variables for teams in the tournament, and a correlation of variables for all teams.

The illustration below shows the number of teams in the tournament per conference for the past five years. Based on this chart, it was clear that the majority of schools came from the ACC, Big 12, and the Big 10. We wanted to investigate these basic stats to see if coming from one of those conferences played more of a factor in winning the tournament because if there are more teams from a certain conference, the higher the chances a winner would be from that particular conference.

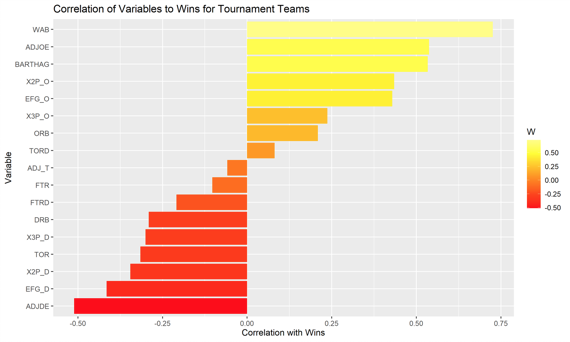


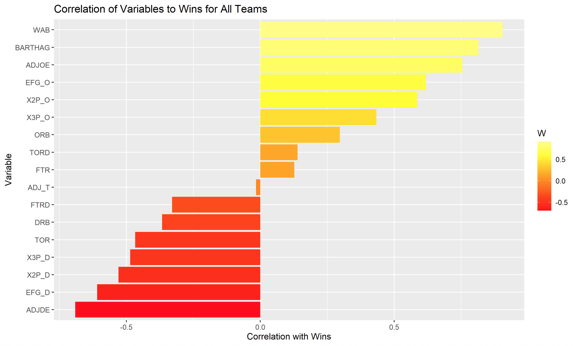
Pearson Correlation:

Pearson Correlation compares the coefficient values of variables and their relationship to one another. The closer to 1 or –1 indicates a stronger relationship between the variables. This was important because different variables needed to be selected for the models. According to the correlation, it’s evident that variables like Adjusted Offensive Efficiency (ADJOE) are strongly related to Wins Above Bubble (WAB). That said, we decided to select those specific variables to feed into our models.



Correlation of variables to wins for tournament teams and all teams: Lastly, we investigated the correlation of variables to wins for all teams. With these statistics, we saw that wins against bubble, adjusted offensive efficiency, power rating, 2 point percentage, and effective field goal percentage were the top variables that contributed to a team’s success. These correlations were in line with the Pearson chart.





After running this exploratory data analysis, we were able to determine which variables would be used in our models. Secondly, the data analysis showed us that regardless of how a team performed, these variables were constant and necessary to contribute to winning games, so we felt confident that these factors would play out well in our models. That said, we decided to explore the following models: Linear Regression, K-Means, Decision Tree Predictions, Linear Binary Predictions, and a Linear Target Confusions Matrix.

## Decision Tree

dt.1.cm$table

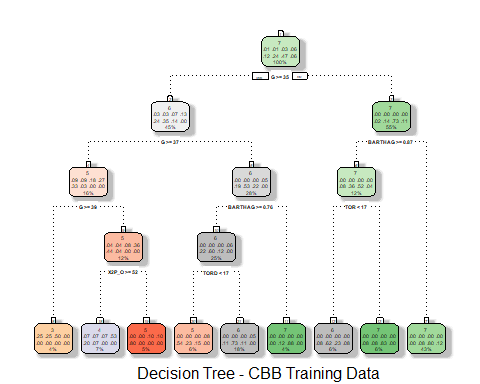
## Reference  
## Prediction 1 2 3 4 5 6 7 8  
## 1 0 0 0 0 0 0 0 0  
## 2 0 0 0 0 0 0 0 0  
## 3 2 2 1 0 0 0 0 0  
## 4 0 0 1 5 2 0 0 0  
## 5 0 0 1 0 7 6 3 0  
## 6 0 0 1 3 6 14 9 2  
## 7 0 0 0 0 1 12 52 6  
## 8 0 0 0 0 0 0 0 0

dt.1.acc

## Accuracy   
## 58.09

## Plot the Model

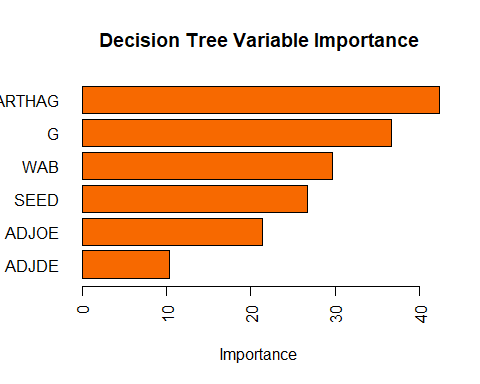
fancyRpartPlot(dt.1, sub = 'Decision Tree - CBB Training Data')



Decision Tree did not have leaf nodes for 1st or 2nd place.

## Variable Importance in Tree

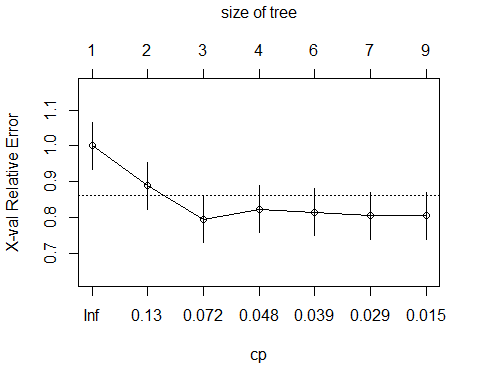
imp <- varImp(dt.1, scale = TRUE) # most important variables  
imp$Variable <- rownames(imp)  
imp <- imp[order(imp$Overall),]  
imp.df <- as.data.frame(tail(imp))  
barplot(imp.df$Overall,  
 main = "Decision Tree Variable Importance",  
 xlab = "Importance",  
 names.arg = imp.df$Variable,  
 las = 2,  
 col = "#F76900",  
 horiz = TRUE)



The most important variables are BARTHAG, G, WAB, SEED, ADJOE, ADJDE. These are the variables that are used to look at the prediction of what team has the best chance of winning the tournament.

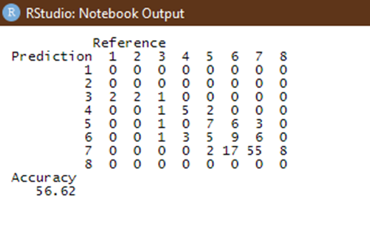
## Plot cp to determine how to prune the tree

plotcp(dt.1)



## Prune the Tree

dt.2 <- prune(dt.1, cp = 0.029)  
dt.2.pred<- predict(object=dt.2,test\_noTarget, type="class")  
dt.2.cm <- confusionMatrix(dt.2.pred, test$Target)  
dt.2.acc <- round(dt.2.cm$overall[1]\*100,2)  
dt.2.cm$table



The decision tree was not a good model for this project due to the lack of a first or second place team being identified. A first and second place team being identified would be crucial to picking the winners of the tournament. The accuracy was 56.62 which will be compared to the accuracy of the further models to identify the correct model.

**K Means**

## K - Nearest Neighbors

knn.1 <- knn(train, test, train$Target, k= (round(sqrt(nrow(train)),2)))  
knn.1.cm <- confusionMatrix(knn.1, test$Target)  
knn.1.acc <- round(knn.1.cm$overall[1]\*100,2)  
knn.1.cm$table

## Reference  
## Prediction 1 2 3 4 5 6 7 8  
## 1 0 0 0 0 0 0 0 0  
## 2 0 0 0 0 0 0 0 0  
## 3 0 0 0 0 1 0 0 0  
## 4 2 1 0 1 0 0 0 0  
## 5 0 0 0 5 3 10 2 0  
## 6 0 1 2 1 9 6 8 0  
## 7 0 0 2 1 3 16 54 8  
## 8 0 0 0 0 0 0 0 0

knn.1.acc

## Accuracy   
## 47.06

KNN did not work well because the data set was relatively small. There were only 4 years' worth of data to determine the winners of the 2021 tournament. Since there were only four years, it did not have many examples of teams that finished in the top 4. Using a larger data set would have potentially made the KNN a better model to use. Compared to the Decision tree model, the KNN has a lower accuracy rating so this is an inferior model.

**SVM - Linear Binary Predictions & Linear Target Confusion Matrix**

#SVM using Ryan's Target values and Linear Regressions 4 significant attributes  
#SVM1 poly  
svm.1.tune <- tune(svm, Target ~ .,   
 data=train,   
 kernel="polynomial",  
 ranges=list(cost=c(.01,.1,1,10,100,1000)))  
  
   
  
svm.1.tune.best.performance <- round(svm.1.tune$best.performance, 3)  
  
   
  
svm.1 <- svm(Target ~ W + ADJOE + ADJDE + BARTHAG, data=train, kernel="polynomial", cost = svm.1.tune.best.performance, scale=FALSE)  
svm.1.pred <- predict(svm.1, test\_noTarget, type="class")  
svm.1.Ptable <- table(svm.1.pred, test$Target)  
svm.1.cm <- confusionMatrix(svm.1.pred, test$Target)  
svm.1.acc <- round(svm.1.cm$overall[1]\*100,2)  
svm.1.cm$table

## Reference  
## Prediction 1 2 3 4 5 6 7 8  
## 1 2 0 0 0 0 0 0 0  
## 2 0 0 0 1 0 0 0 0  
## 3 0 0 0 2 3 1 1 0  
## 4 0 1 0 1 0 0 0 0  
## 5 0 0 2 2 5 5 4 0  
## 6 0 0 1 2 5 14 11 1  
## 7 0 1 1 0 3 12 46 7  
## 8 0 0 0 0 0 0 2 0

svm.1.acc

## Accuracy   
## 50

#SVM2 Linear  
svm.2.tune <- tune(svm, Target ~ .,   
 data=train,   
 kernel="linear",  
 ranges=list(cost=c(.01,.1,1,10,100,1000)))  
  
   
  
svm.2.tune.best.performance <- round(svm.2.tune$best.performance, 3)  
  
   
  
svm.2 <- svm(Target ~ W + ADJOE + ADJDE + BARTHAG, data=train, kernel="linear", cost = svm.1.tune.best.performance, scale=FALSE)  
svm.2.pred <- predict(svm.2, test\_noTarget, type="class")  
svm.2.Ptable <- table(svm.2.pred, test$Target)  
svm.2.cm <- confusionMatrix(svm.2.pred, test$Target)  
svm.2.acc <- round(svm.2.cm$overall[1]\*100,2)  
svm.2.cm$table

## Reference  
## Prediction 1 2 3 4 5 6 7 8  
## 1 2 0 0 0 0 0 0 0  
## 2 0 0 0 0 0 0 0 0  
## 3 0 1 0 0 0 0 0 0  
## 4 0 0 0 0 0 0 0 0  
## 5 0 0 0 0 0 0 1 0  
## 6 0 1 3 8 12 13 12 0  
## 7 0 0 1 0 4 19 51 8  
## 8 0 0 0 0 0 0 0 0

svm.2.acc

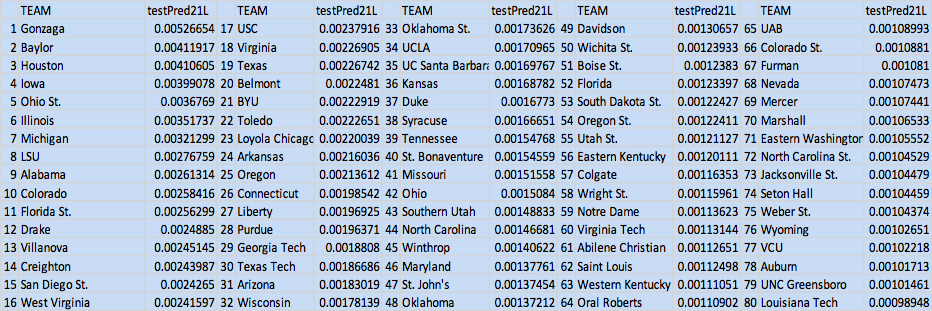
## Accuracy   
## 48.53

#SVM3 Sigmoid  
svm.3.tune <- tune(svm, Target ~ .,   
 data=train,   
 kernel="sigmoid",  
 ranges=list(cost=c(.01,.1,1,10,100,1000)))  
  
   
  
svm.3.tune.best.performance <- round(svm.3.tune$best.performance, 3)  
  
   
  
svm.3 <- svm(Target ~ W + ADJOE + ADJDE + BARTHAG, data=train, kernel="sigmoid", cost = svm.1.tune.best.performance, scale=FALSE)  
svm.3.pred <- predict(svm.3, test\_noTarget, type="class")  
svm.3.Ptable <- table(svm.3.pred, test$Target)  
svm.3.cm <- confusionMatrix(svm.3.pred, test$Target)  
svm.3.acc <- round(svm.1.cm$overall[1]\*100,2)  
svm.3.cm$table

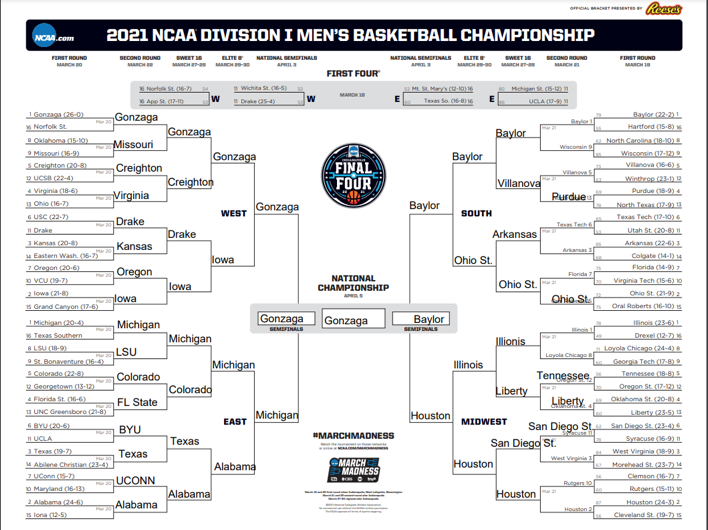
## Reference  
## Prediction 1 2 3 4 5 6 7 8  
## 1 0 0 0 0 0 0 0 0  
## 2 0 0 0 0 0 0 0 0  
## 3 0 0 0 0 0 0 0 0  
## 4 0 0 0 0 0 0 0 0  
## 5 0 0 0 0 0 0 0 0  
## 6 0 0 0 0 0 0 0 0  
## 7 2 2 4 8 16 32 64 8  
## 8 0 0 0 0 0 0 0 0

svm.3.acc

## Accuracy   
## 50



The binary linear regression was the best model output with 50 accuracy. This output was used to create the final bracket. By going down the bracket and looking up the prediction rate for the two teams competing in each game, we were able to fill out the bracket.



**Results**

Looking at the outputs of the models, the best one was the linear binary for making predictions. By using the matrix with each team having a prediction score, the scores were compared and the highest prediction rate was the winner of the game. This logic was used throughout the entirety of preidcting the winners of ecah of the 67 games. By doing this process, the winner would be Gonzaga which is the overall top pick in the rankings. When the project was finished, the first round of the tournament was played. By using the model, our predictions came out to be 64% accurate with 21 games predicted correctly. This success rate was higher than the accuracy of any of the models, so it worked well to use for the bracket.

**Conclusions**

While the process allowed us to use different methods to look at the accuracy and predict the winners of the games, the prediction could have been better if there was more data used. By only looking at the last four years of tournament games, the models did not have enough data to train against winners with only 4 examples of overall winners. The tournament lives through upsets which makes predicting the outcome of the games extremely difficult. A very accurate prediction could be made by using a weighting system based on certain variables. These variables could include performance at beginning of season compared to the end, tournament appearances in the past, home-court advantage, number of returning players, tenure of the coaching staff, and even the number of titles the program has had before. The fun part of March is the unpredictability of the games, but by using an analytical approach that is more fine-tuned to certain attributes like the beforementioned, a more accurate prediction can be made.