Predicting a March Madness Winner

MIST 5620 Group Project Report

Dr. Salge

April 29th, 2021

Daniel Saul	dps41296@uga.edu
Daniel Nguyen	dln38273@uga.edu
Emma Biancheri	evb71983@uga.edu
Erick Escanuela	eae56462@uga.edu
Kyle Brewer	kdb97363@uga.edu

Table of Contents

I. Executive Summary

Along with the other 16.2 million ESPN brackets created for the 2021 men's basketball NCAA March Madness Tournament, we all strive to create that perfect bracket to win bragging rights among friends, family, and colleagues. The purpose of this report is to do just that by determining significant variables and helping users predict the winner of each game in each round of the tournament. However, the all-embracing problems we are trying to solve is that there has never been a perfect bracket created in the previously recorded years of the tournament, sports bettors don't always seek out the correct determinants when making picks, and teams do not focus on the most important development factors for success in the postseason.

The "Context and Question" section provides a basis for what we are trying to analyze and predict in this report. It gives a description of what the March Madness tournament is and expands upon the questions we are trying to address: What are the best offensive statistics of tournament champions? Is defensive efficiency important to win games? Does belonging to a specific division better a team's chances at advancing? Next, the "Data and Variables" section provides a description of the dataset we used, in addition to a description for each variable given. The figures in the "Descriptive Visualizations" section visually summarize our basketball data and relevant trends among variables. Then, the "Method" and "Results" sections explain the methods we followed and the outputs that support our analysis. By combining a traditional OLS regression and supervised machine learning model, we were able to determine statistically significant variables and use them to create a bracket for the 2021 season. This resulting bracket ended up being 34% more accurate than the average bracket score. The "Limitations" section describes what additional data we could have collected to further enhance our analysis and reduce discrepancies related to creating a perfect bracket and winning tournament games. Lastly, based off our complete analysis, we provide recommendations for players, coaches, the sports betting industry, and bracket makers striving for success in all realms of the NCAA March Madness tournament.

Overall, the goal and value we bring with this report is to help bracket creators and bettors to make the correct picks with more accuracy as well as show coaches and players where they should utilize their training and practice. For reproducibility, all R code used for modelling and visualization can be found in Appendix A while all Tableau documentation for the figures can be found in Appendix B.

II. Context and Question

Context

March Madness is a long-awaited NCAA basketball tournament played every year for both Division I men and women's college teams; however, we will just be focusing on data from the men's tournament. The dataset we utilized for this project resulted from the transformation and combination of several datasets found on kaggle. Initial transformations included filtering the data to the past five years and only those teams who made it to the tournament, selecting which statistics and measures to analyze, and joining these datasets based on team ID and season.

Our dataset describes the NCAA Division I men's basketball teams invited to play in the March Madness tournament during the 2015, 2016, 2017, 2018, 2019, and 2021 seasons. Data from 2020 is omitted because the tournament was cancelled due to the global pandemic. Apart from tournament wins, the remaining variables reflect performance during the regular season before the tournament begins. Because we wrangled and transformed our data before the completion of this year's March Madness tournament, the tournament wins column is null for all teams playing in the 2021 tournament.

Question

The overarching goal is to predict the correct winner of the tournament before any games are played for that said season. Our project evaluates which variables in our dataset are the best predictors of a team's performance in the March Madness tournament, as measured by the number of tournament wins. For instance, the efficiency of offenses and defenses, belonging to a specific conference, or past championship titles may give certain teams an advantage over others and their significance must be gauged to help determine a winner. This is useful to examine because it provides insight to which statistics and measures have the largest impact on a team's success in the March Madness tournament and identifies patterns among tournament champions and runner ups over the past five years. We also aim to predict the ultimate March Madness champion and the winners of each tournament game preceding the final one. Given that 16.2 million March Madness brackets were created by fans through ESPN's men's tournament challenge alone, there are a large number of people who could benefit from such insights, and the topic was of interest to our group as well.

III. Data and Variables

Data

Each year, 68 teams are invited to compete in the March Madness tournament. Because we are analyzing this year's teams and the five previous years, there are 408 total observations in our dataset. Our target of interest is the number of tournament games won by a given team. This is a numeric variable; the tournament champion will have won six games, where a team eliminated in the first round will have won zero. Our dataset contains 20 other variables. Four of these variables - season, teamID, teamName, and conference - are descriptive, but the remaining will be used as predictors for the target. All the variables considered in our project are listed below.

Variables

Variable	Data Type	Mean	Median	Standard Deviation	Description
season	date	N/A	N/A	N/A	Year in which the team of interest played in the March Madness Tournament
teamID	numeric	N/A	N/A	N/A	Unique identifier assigned to each team in Kaggle
teamName	class	N/A	N/A	N/A	The Division I college basketball school
conference	class	N/A	N/A	N/A	The athletic conference the school participates in
seed	numeric	N/A	N/A	N/A	Seed in the NCAA March Madness Tournament
gamesPlayed	numeric	32.7	34	4.216	The number of games played in the regular season
gamesWon	numeric	23.7	23	4.7	The number of games won in the regular season
winPercentage	numeric	0.723	0.722	0.105	The percentage of games won in the regular season
tempo	numeric	68.1	68.2	3.2	The number of possessions per 40 minutes a team would have against an average Division I team
defensiveRebounds	numeric	596.4	591	122.6	Total defensive rebounds made by the team in the regular season

blocks	numeric	92.6	89	32.6	Blocks accomplished by the team in the regular season
defensiveEfficiency	numeric	96.7	96	5.1	The defensive efficiency (points allowed per 100 possessions) a team would have against the average Division I offense
steals	numeric	151.8	150	40.9	Steals accomplished by the team in the regular season
totalAssists	numeric	343.7	339.5	83.9	Assists made by the team in the regular season
threePointAccuracy	numeric	0.382	0.379	0.028	The percent accuracy of the three pointers attempted by the team during the regular season
freeThrowAccuracy	numeric	0.723	0.723	0.038	The percent accuracy of the free throws attempted by the team during the regular season
fieldGoalPercentage	numeric	0.524	0.523	0.027	Effective field goal percentage shot
turnoverPercentage	numeric	0.189	0.188	0.023	The turnover percentage allowed
offensiveEfficiency	numeric	111.5	111.3	6.4	The offensive efficiency (points scored per 100 possessions) a team would have against the average Division I defense
wins	numeric	N/A	N/A		The number of wins in the March Madness Tournament; the champion will have 6 and those eliminated in the first round will have 0

IV. Descriptive Visualizations

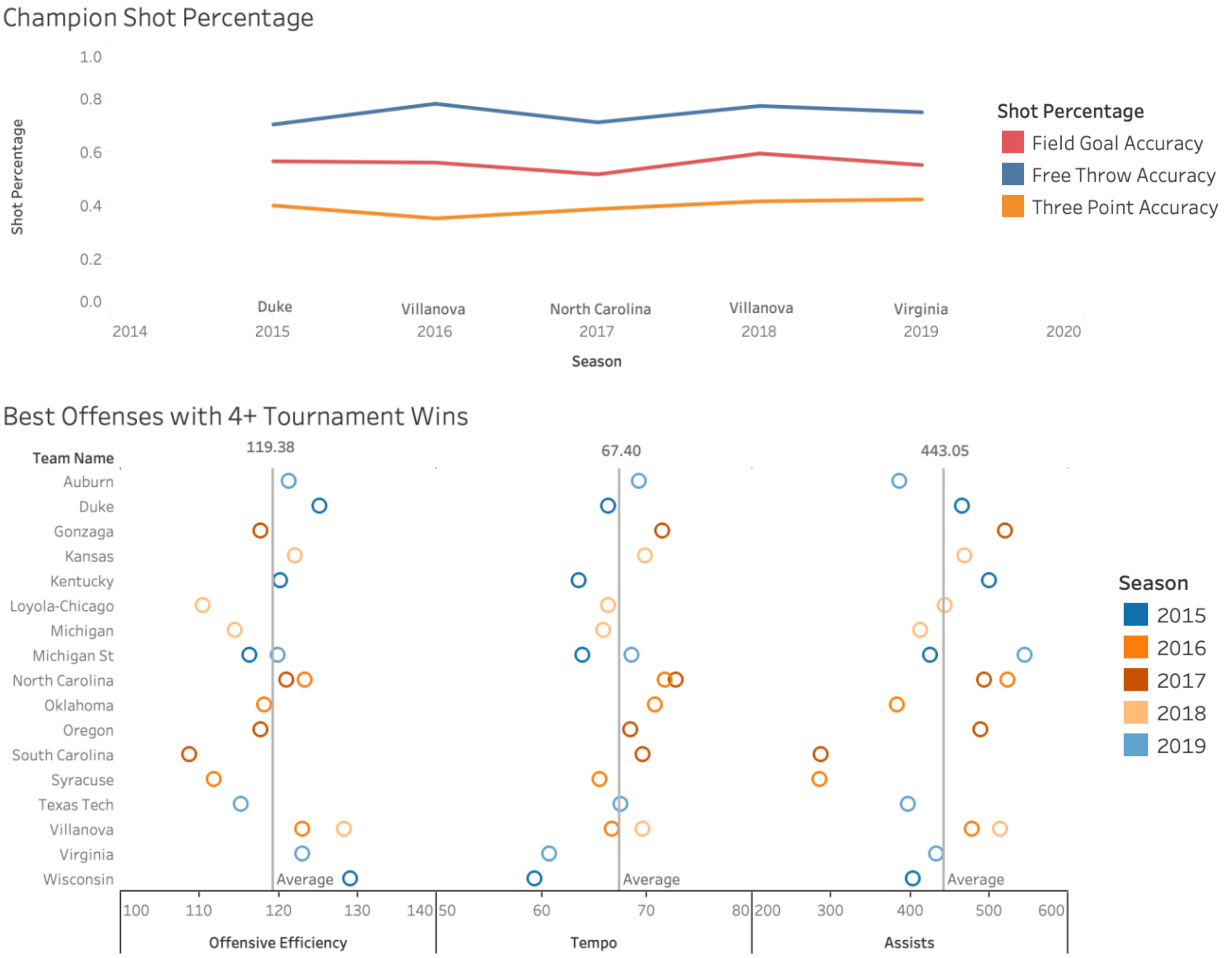


Figure 1

This visualization displays some of the most important variables such as shot percentage of past champions and offensive success, which consists of efficiency, tempo, and number of assists. We can see that winners of past March Madness tournaments are the most accurate at shooting free throws, then field goals, and lastly three pointers. In contrast, North Carolina won the 2017 championship with the lowest shooting percentages among the others. Regarding the best offenses with 4 or more wins, in other words making it past the Elite 8, teams like North Carolina are one of the top fastest paced teams with high efficiency. Averages are included to reveal the typical numbers these top teams put up in the regular season before the tournaments. The statistics in the visuals are important for determining highly influential variables. The documentation of these graphs can be found in Appendix B.

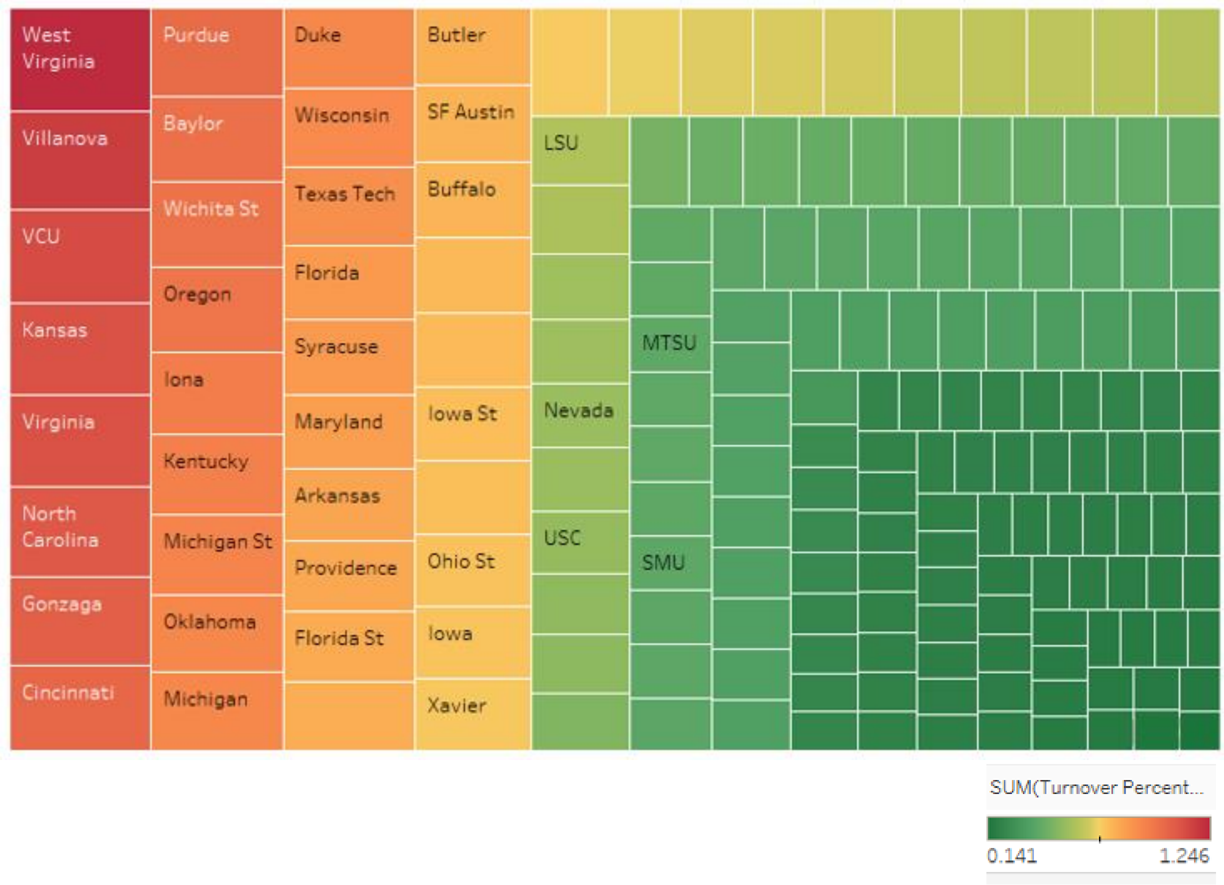


Figure 2

The distribution of the tournament wins is shown in heat map visualization in above. The ranking order begins from green (the low rank zone), yellow and amber color (medium ranking zone) and red (high rank zone). It is seen that the high rank zone shows higher proportions of tournament wins than the low rank zones. The interpretation is that the teams that have won the tournament ranked higher than those that have not won a championship in terms of winning for the next year, or being the top 5. It can also be concluded that the teams that have won the championships are the same that have won 6 times in the tournament wins column. Details on the reproducibility of this visualization are included in Appendix 2.

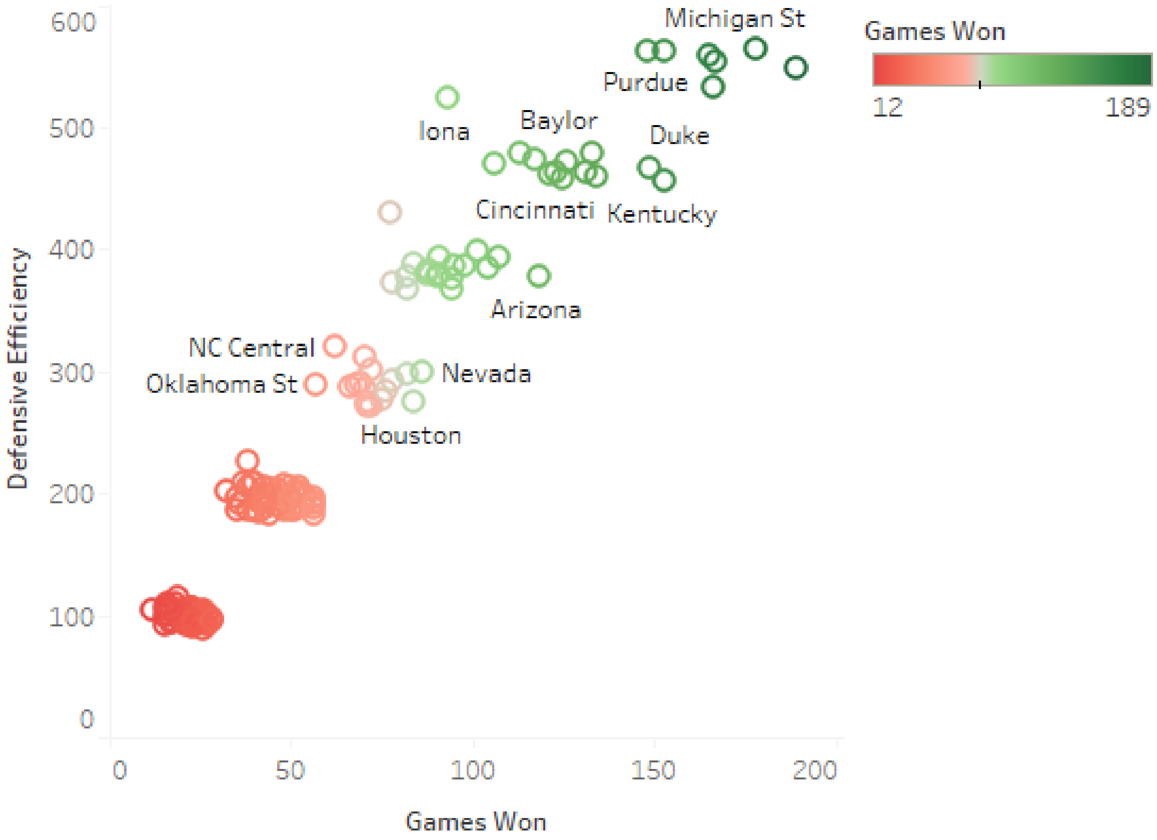


Figure 3

This visual reveals the correlation between a team's defensive efficiency and their number of regular season wins. When examining patterns between games won and defensive efficiency, it's notable that the top teams (teams that usually win the most games) have a higher defensive efficiency throughout the years 2015-2019. There are some outliers that are present such as Iona and Texas Southern. In these cases, defensive efficiency doesn't always relate to more wins, but a correlation still exists. There is no coincidence that the past tournament winners include Virginia, Baylor, North Carolina, and Villanova, all of which are at the top of the charts in terms of defensive efficiency. The documentation of this figure can be found in Appendix B.

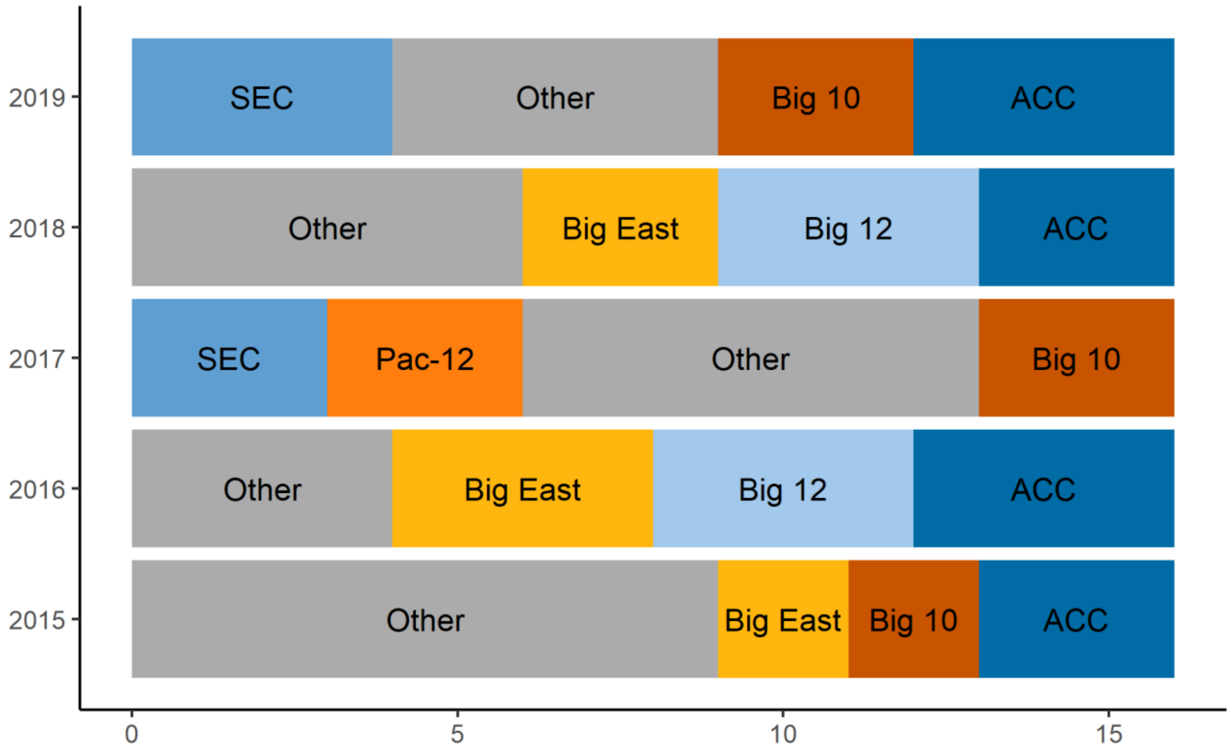


Figure 4

This visualization analyzes the conference membership of those teams advancing to the Sweet 16 round of the March Madness tournament in the past five years. The horizontal axis displays the number of teams playing in this round. From this figure, we can observe the consistent presence of certain conferences among the top 16 Division I teams. This helps us consider if conference membership contributes to the success of a team, as a given conference may have unique resources or sponsors that lead to a more competitive performance relative to others. For ease of understanding, the top 3 conferences from each season are displayed, with the rest being grouped as “other”. It appears that the ACC, Big East, and Big 10 display a degree of dominance compared to other conferences, as they have been one of the top conferences for at least three recent seasons. The SEC, Big 12, and Pac-12 also were top conferences for at least one year, indicating their strength as well. While causation cannot be inferred from this bar chart, there is a clear pattern in the same conferences consistently having top competitors and this trend hints at the impact of conference membership. The supporting code for this bar chart is in Appendix A.

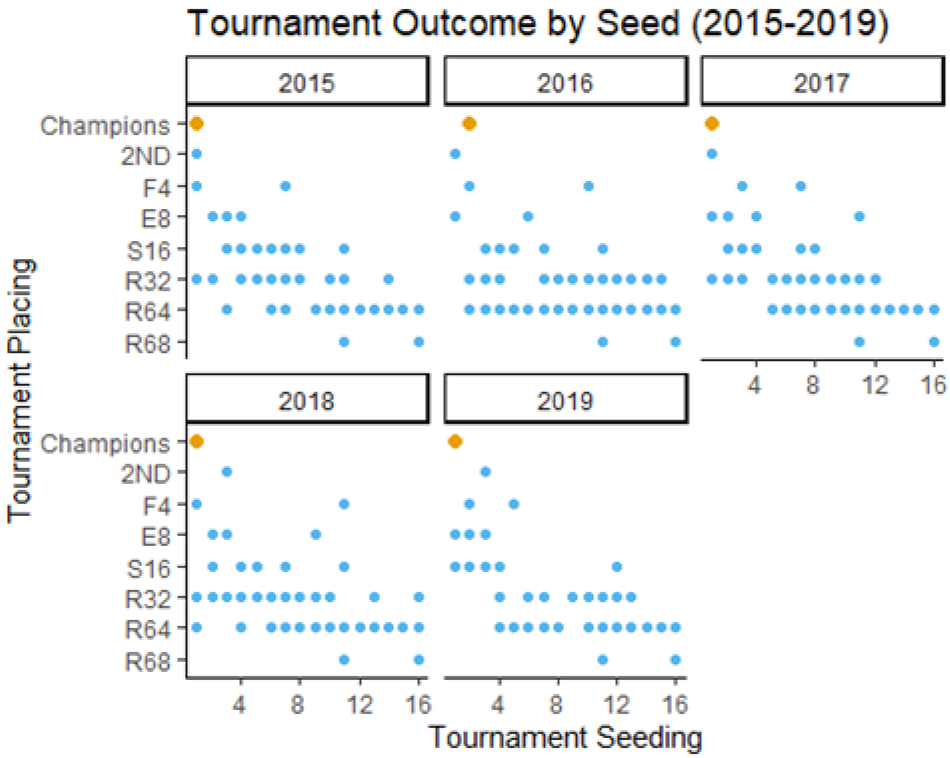


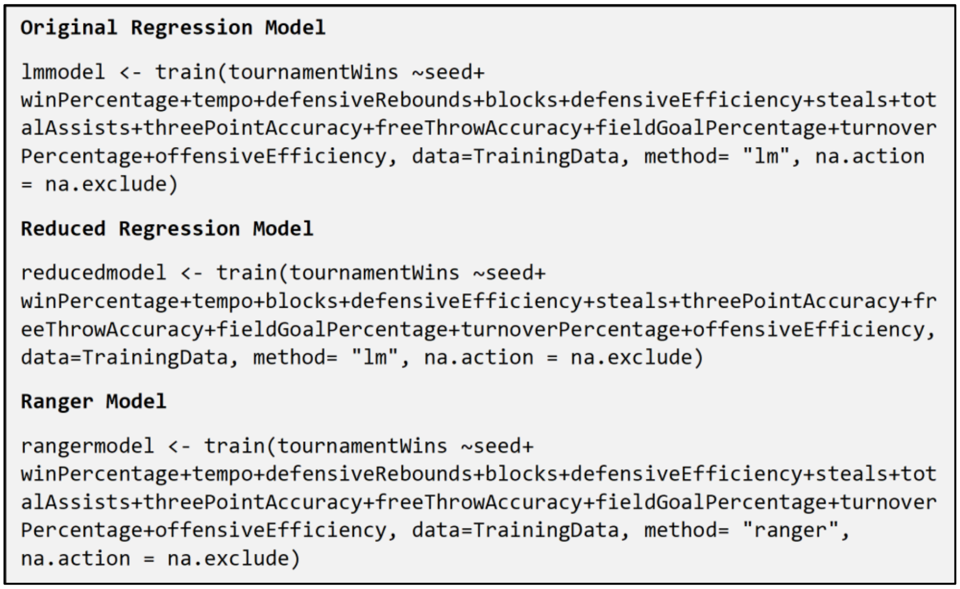
Figure 5

This visual displays the correlation between a team's final tournament placement and their pre-tournament seeding. Teams that were did not make it to the tournament were omitted from the visual. There seems to be a correlation between a team's seeding and their final placing, as shown by the similar data throughout the years. It is notable that from the years selected for our hypothesis, even though some highly seeded teams were eliminated early in the tournament, only teams seeded 1 or 2 won the tournament from the years 2015-2019. Ultimately, it is evident that the teams seeded 1 or 2 are much more likely to win the tournament than those seeded lower. This figure was created in an R Markdown document, which is part of Appendix A.

V. Method

A traditional OLS regression was the primary approach used to analyze the team characteristics with the greatest impact on team performance. By observing each predictor's t-value and variance in this model's output, we can strengthen our understanding of the statistical significance and amount of variance explained by each.

We utilized a supervised machine learning approach to predict a team's performance in the March Madness tournament. We measured performance in terms of tournament games won as our target variable and focused on creating a model to predict this variable. As can be inferred, a higher number of predicted tournament wins means the given team advances farther and performs better in the tournament. Using our model's output, a bracket can be built by declaring the team with the highest number of predicted tournament wins as the champion, naming the team with the second highest as their opponent in the championship, and continuing to work backwards in this fashion until all 127 tournament games have forecasted winners.



Original Regression Model

```
lmmodel <- train(tournamentWins ~seed+
winPercentage+tempo+defensiveRebounds+blocks+defensiveEfficiency+steals+tot
alAssists+threePointAccuracy+freeThrowAccuracy+fieldGoalPercentage+turnover
Percentage+offensiveEfficiency, data=TrainingData, method= "lm", na.action
= na.exclude)
```

Reduced Regression Model

```
reducedmodel <- train(tournamentWins ~seed+
winPercentage+tempo+blocks+defensiveEfficiency+steals+threePointAccuracy+fr
eeThrowAccuracy+fieldGoalPercentage+turnoverPercentage+offensiveEfficiency,
data=TrainingData, method= "lm", na.action = na.exclude)
```

Ranger Model

```
rangermodel <- train(tournamentWins ~seed+
winPercentage+tempo+defensiveRebounds+blocks+defensiveEfficiency+steals+tot
alAssists+threePointAccuracy+freeThrowAccuracy+fieldGoalPercentage+turnover
Percentage+offensiveEfficiency, data=TrainingData, method= "ranger",
na.action = na.exclude)
```

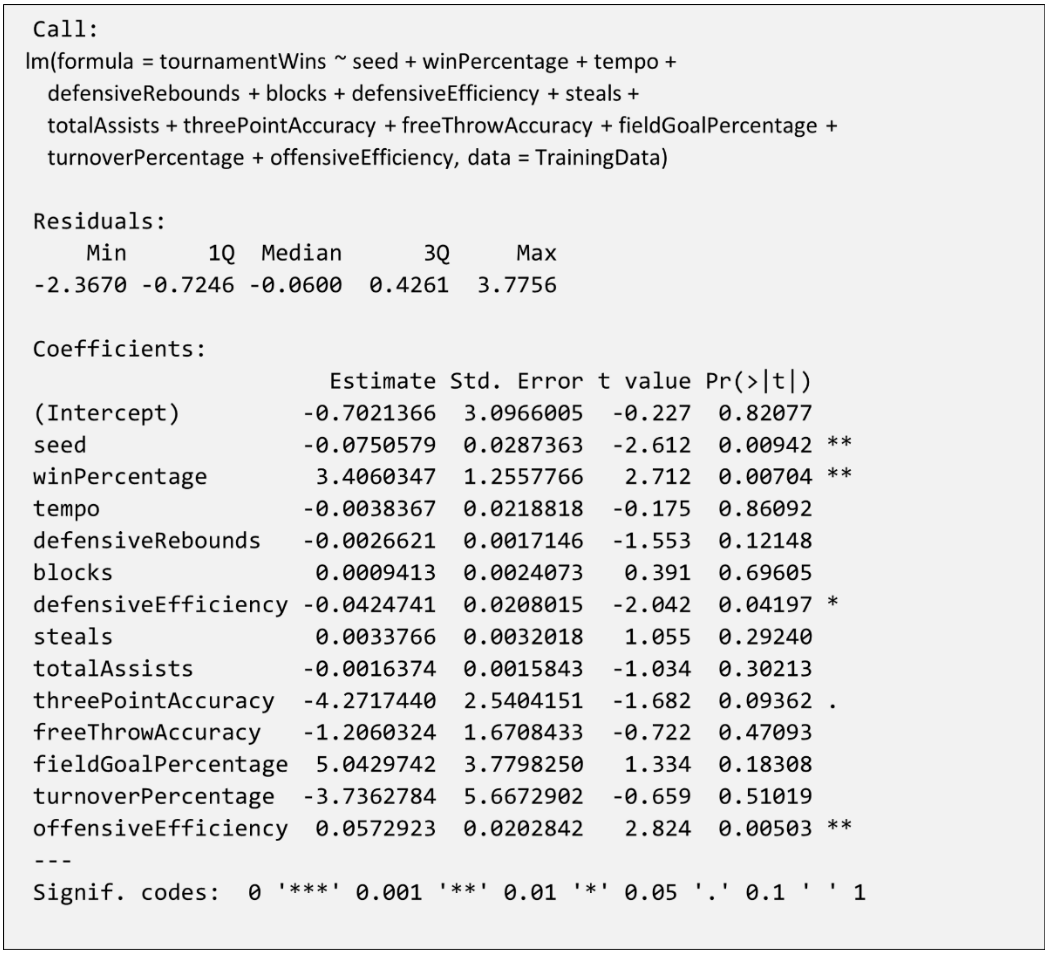
While we did not use unsupervised machine learning, we ran multiple supervised learning models. The first of these was a regression model built off all 16 original predictors. We then examined the correlation between variables to find which variables should be removed to avoid multicollinearity. Based on this analysis, the variables describing total assists and defensive rebounds were removed to yield a reduced regression model with only 14 variables and lower intercorrelations among predictors than before. To improve our model's prediction ability, we specified a different method, the ranger method, and compared this model's performance to the other regression models.

After the conclusion of the March Madness tournament on April 5th, we were able to gather data on the actual number of games won by each team. We could then compare our projections to their actual values. Two important metrics used to do this were the root mean square error (RSME) and R^2 value for each of our three models. RMSE reveals how much the predictions deviate from the actual values on average. R^2 represents the proportion of the variance for the number of tournament wins that is explained by the predictors in the model.

Because our model aims to predict the performance of Division I Men's basketball teams in the 2021 March Madness tournament, we will also evaluate our model based on how its accuracy compares to the average bracket submitted through the NCAA's official annual contest. We will score our bracket based on the methodology of this same contest and compare our performance to others using the data released about the average bracket score.

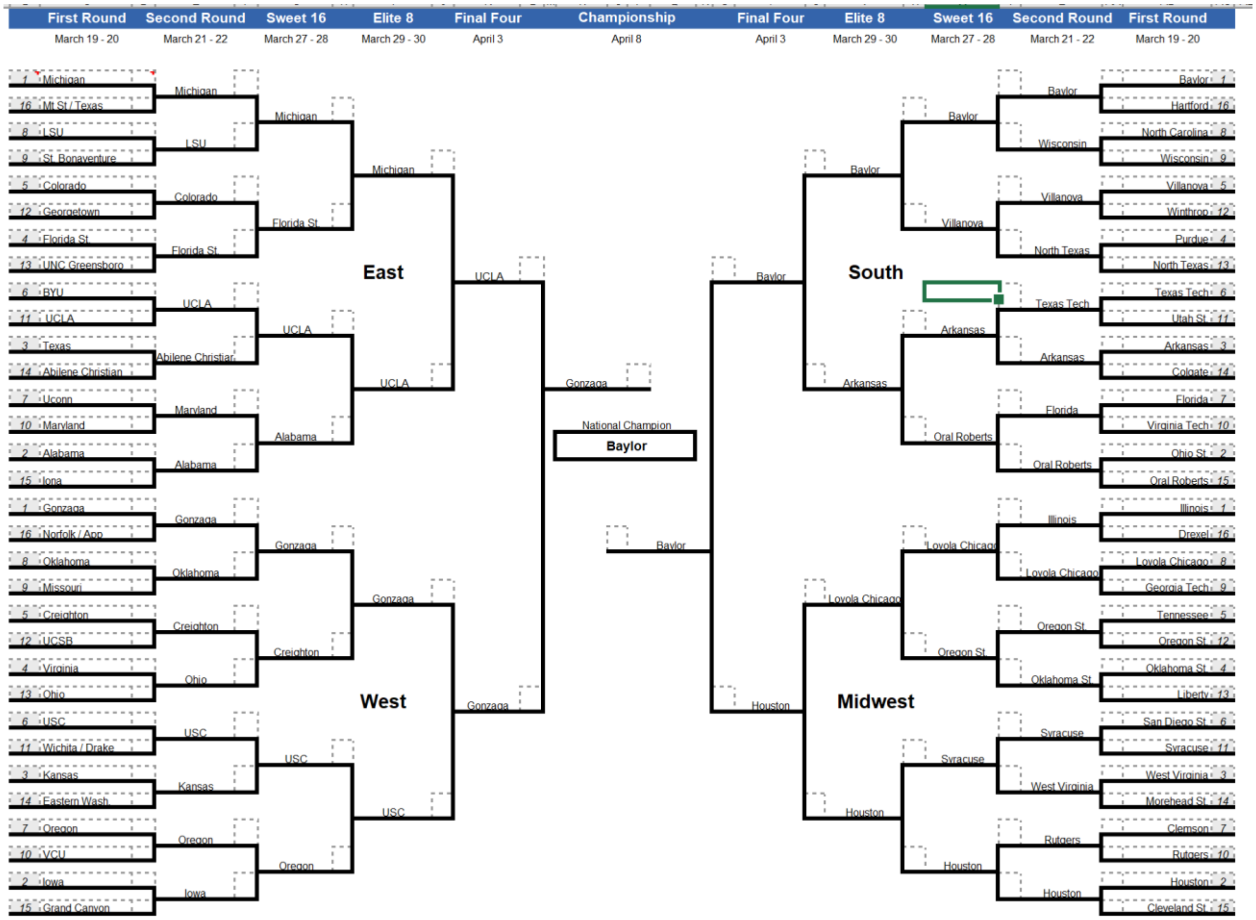
We split our data into testing and training sets based on season. We compiled the data on team statistics and tournament wins from the five most recent seasons (2015 - 2019) in our training data and built our models upon this. We then tested our model and predicted tournament wins for the testing dataset, which includes data from the 2021 regular season for the 68 teams invited to play in the tournament. Except for the tournament wins variable, all the other variables describe teams' regular season performance. Because of this, leakage was not of concern when constructing our model because all the data included would have been available to stakeholders at the time of use. In addition, because the training and testing datasets were distinguished by season, there is no risk of testing data leaking into the training data. To prevent overfitting and ensure a parsimonious model, we created the regression model with a reduced number of features as referenced earlier. We selected what features of this model to eliminate after examining the correlation among variables and excluding those with the highest correlation. This reduced model addresses multicollinearity and allows the model to be generalized to previously unseen data points or future seasons.

VI. Results

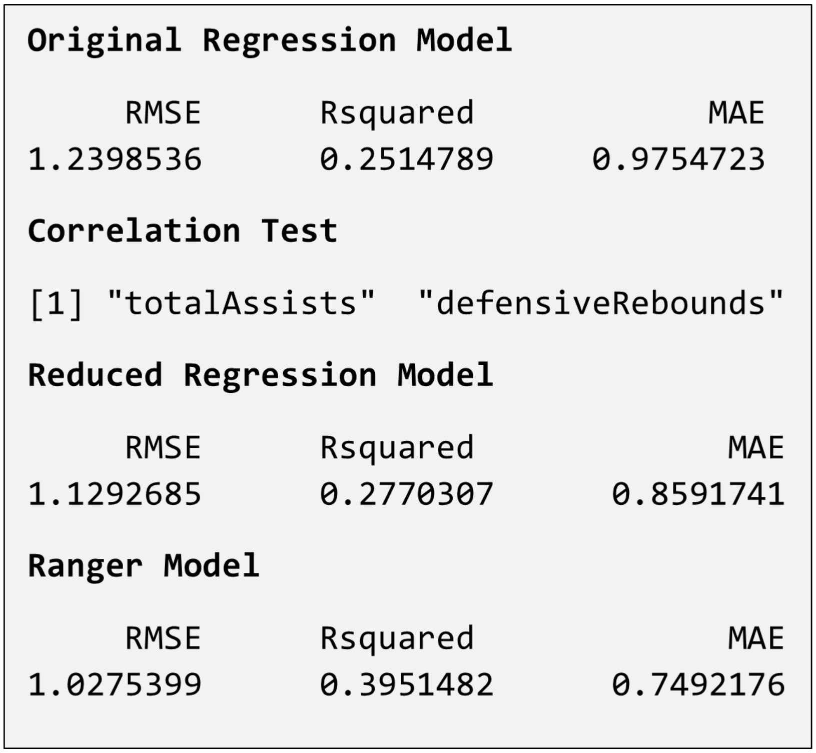


```
Call:
lm(formula = tournamentWins ~ seed + winPercentage + tempo +
   defensiveRebounds + blocks + defensiveEfficiency + steals +
   totalAssists + threePointAccuracy + freeThrowAccuracy + fieldGoalPercentage +
   turnoverPercentage + offensiveEfficiency, data = TrainingData)

Residuals:
    Min      1Q  Median      3Q     Max
-2.3670 -0.7246 -0.0600  0.4261  3.7756

Coefficients:
                      Estimate Std. Error t value Pr(>|t|)
(Intercept)         -0.7021366  3.0966005  -0.227  0.82077
seed                -0.0750579  0.0287363  -2.612  0.00942 **
winPercentage        3.4060347  1.2557766   2.712  0.00704 **
tempo               -0.0038367  0.0218818  -0.175  0.86092
defensiveRebounds   -0.0026621  0.0017146  -1.553  0.12148
blocks               0.0009413  0.0024073   0.391  0.69605
defensiveEfficiency -0.0424741  0.0208015  -2.042  0.04197 *
steals               0.0033766  0.0032018   1.055  0.29240
totalAssists        -0.0016374  0.0015843  -1.034  0.30213
threePointAccuracy  -4.2717440  2.5404151  -1.682  0.09362 .
freeThrowAccuracy   -1.2060324  1.6708433  -0.722  0.47093
fieldGoalPercentage  5.0429742  3.7798250   1.334  0.18308
turnoverPercentage  -3.7362784  5.6672902  -0.659  0.51019
offensiveEfficiency  0.0572923  0.0202842   2.824  0.00503 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We foresee our models being of value to two different audiences: those creating brackets for as well as betting on the March Madness champion and the coaches and players directly competing in the tournament. For this second group, our OLS regression provides insight to the strongest predictors of team performance and therefore the most important areas to develop and improve on as a team in order to increase the number of tournament wins. Our model revealed seed, regular season win percentage, and offensive efficiency to have the strongest relationships with tournament wins, with defensive efficiency and three-point accuracy also being statistically significant at the 0.1 level. Coaches and players could interpret this and work on their three-point accuracy, offensive efficiency, and defensive efficiency during practice. Improvements in these would have a positive impact on their tournament performance but also likely impact their seed and regular season win percentage, which this model also suggests are strong predictors of tournament strength.



From a fan perspective, these individuals would be interested in accurately predicting the winners of each game in the tournament. We can judge this accuracy using the score of our bracket compared to the average performance as well as the RMSE and R^2 values. As explained earlier, our supervised machine learning approach resulted in a predicted number of tournament wins for all 68 teams invited to the march madness tournament. Using these predicted values from our ranger model, the above bracket was created and scored according to the scoring guidelines provided by the NCAA official bracket competition. 124 points are possible under these scoring conventions, and our bracket received 85. The most recent data from the NCAA reveals the average user bracket score to be 63.9. This means our model yielded 34% more points than the average bracket and has a meaningful impact on bracket performance as a result.



```
Original Regression Model

     RMSE       Rsquared             MAE
1.2398536      0.2514789      0.9754723

Correlation Test

[1] "totalAssists"  "defensiveRebounds"

Reduced Regression Model

     RMSE       Rsquared              MAE
1.1292685      0.2770307       0.8591741

Ranger Model

     RMSE       Rsquared              MAE
1.0275399      0.3951482       0.7492176
```

All our models have a root mean squared error between 1 and 2 and R^2 greater than 0.25. While these values may suggest our models cannot relatively predict the data accurately, this reflects the uncertain nature and general lack of predictability of March Madness, which contributes to the thrill and popularity of betting on this tournament. However, we feel our OLS model still is valuable for coaches and players and our supervised machine learning approach still aids in creating a significantly more accurate bracket than the average submission.

VII. Limitations

With March Madness being such a large and fluid sports tournament, there are limitations that come along with the data collection process and this report. First, we would have liked to have more than five complete years of tournament data. As mentioned earlier, the COVID-19 pandemic cancelled the 2020 season March Madness tournament, leaving that year void for our analysis. Also, the timing of the data collection for this report fell before the ending of the 2021 season tournament. This limited us with the up-to-date data for our testing data because we were not able to distinguish a winner for the current season from the beginning. For the seasons prior to 2015, we wanted to collect data from the years 2011-2014, but it was quite difficult to find adequate data from those seasons with similar variables to those analyzed in this report. In addition, before 2011, there were 65 teams in the tournament instead of the 68 teams that participate today, completely changing the dynamic of how the tournament is played and the variables needed to win. We would have liked to collect the same variables mentioned above such as assists, rebounds, and offensive efficiency for the 2020 season and years prior to 2015. The data would have been useful to help understand how modern college basketball is played compared to older styles in addition to being able to analyze a larger number of observations.

Second, we were limited by how many factors were found to be significant in winning March Madness as well as the restraint of time to find and analyze each of these variables. With crowds in arenas limited due to the pandemic, the statistics of the 2021 season may include a discrepancy because certain teams traditionally run off of fan noise and hype to play better. Furthermore, we believed collecting more variables such as number of upperclassmen on the team, average winning streaks during the season, past success of a coach on a said team, playing games on the road, or popularity at the school would have been valuable. This data would be collected in percentage, whole number, and categorical formats. The new variables would have been useful for training and testing our model, as well as possibly making it more accurate to successfully predict a March Madness winner with higher R^2 values. Regarding causality, we do believe there is a correlation between the variables in our model and number of tournament wins, but we would proceed with caution before concluding causality. This is because we did not have enough time to test all the significant variables possible like the select few mentioned above, resulting in other variables potentially affecting the independent variable. Additionally, it was challenging to determine a control group since that is realistically impossible in the tournament and unbelievable upsets can happen such as the 15th seed Oral Roberts defeating the 2nd seed Ohio State in the most previous tournament--the 2021 season. Therefore, we cannot completely infer causality.

VIII. Recommendations

Our analysis of how to be successful in predicting a NCAA March Madness tournament winner can be used to the advantage of multiple different audiences. As mentioned earlier in the “Results” section, coaches/players, bettors, and tournament bracket creators can benefit from our statistical model and recommendations. Based off our model yielding 34% more points than an average NCAA March Madness bracket and questions answered by the summarizing visuals above, our recommendations include:

- For sports bettors and bracket creators, make your game winning predictions based on the two teams’ offensive efficiency, defensive efficiency, and win percentage from the season preceding the tournament, as well as the tournament seed given to the teams. You should choose the higher statistic among the teams. However, there is a disclaimer for this recommendation due to the “madness” of the tournament. Anything can happen in the tournament; 15 seeds can defeat 1 seeds and 1 seeds can remained undefeated to win the tournament. This inherent limitation in the broadness of variables and wildness that can affect collegiate games during March, is mentioned in greater detail in the “Limitations” section.
- Coaches should allocate the majority of practice time to improving their offensive efficiency and shooting percentages, especially from the three-point arc. While coaches are responsible for strategy, this recommendation should also be used for player development to improve their shooting skills. According to our visual analysis, the average offensive efficiency of teams who had 4 or more tournament wins in the past 5 years was 119.38, so picking teams with offensive numbers close to this average will benefit you as a bracket creator or bettor. Three-point accuracy is another significant predictor in determining tournament wins, and as we all know, the more points you score, the more likely you are to win the game. Since 2016, three-point accuracy has been increasing among championship-winning teams, proving the shot’s importance for success. This shot is the farthest and most difficult shot to make, so a team must sharpen their long-distance shooting skills to help score more points in close-game situations.
- For both coaches and bracket creators, do not focus on teams with a high tempo of play since it is not significant in our model with a high p-value of 0.86092. Speed of play and more possessions do not always help an offensive score and win. Players taking their time on the court may lead to better scoring opportunities and stamina retention.
- When building your bracket, an important variable is which conference your top teams are from. According to Figure 4, the Atlantic Coast Conference (ACC) has sent most of the teams in the Sweet 16 to the tournament in the previous years. If you needed a little

more influence on which team to pick in a key matchup for a bet or bracket, pick the team that belongs to the ACC. A notable mention are also teams from the Big 10 and Big 12; these being the next runner up to having the most teams in the Sweet 16.

The key takeaway from our model and analysis is that someone trying to predict a winner in the tournament based off the “defense wins championships” phrase may not be considering the most impactful variables on March Madness success. Yes, defensive efficiency was statistically significant in our model, and this is not a message to deter you from using it, but the offensive statistics seem to outweigh the importance of the defensive statistics based on our recommendations and analysis above. You must use the combination of seeding, win percentage, conference, and significant offensive variables within our model to build a more accurate bracket and to beat the odds of 1 in 9.2 quintillion to obtain the perfect bracket.

IX. Appendix A: R Code

Data Wrangling

Training Data

We pulled data on NCAA Division I basketball teams from kaggle. Several transformations were performed to prepare this data for analysis and OLS regression. The training data analyzes the teams who played in the five most recent March Madness tournaments. These tournaments took place in the 2015, 2016, 2017, 2018, and 2019 seasons, as this tournament did not take place in 2020.

```
library(tidyverse)

## -- Attaching packages --------------------------------------- tidyverse
1.3.0 --

## v ggplot2 3.3.3     v purrr   0.3.4
## v tibble  3.0.5     v dplyr   1.0.3
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.1

## -- Conflicts ------------------------------------------
tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()

RegSeasonData <- read_csv("RegSeasonData.csv")

##
## -- Column specification --------------------------------------------------
------
## cols(
##   .default = col_double(),
##   WLoc = col_character()
## )
## i Use `spec()` for the full column specifications.

TeamData <- read_csv("MNCAATourneyTeams.csv")

##
## -- Column specification --------------------------------------------------
------
## cols(
##   season = col_double(),
##   teamID = col_double(),
##   teamName = col_character(),
##   conference = col_character(),
##   seed = col_double(),
##   tournamentWins = col_double()
## )
```

```
OffenseDefense <- read_csv("OffenseDefense.csv")

##
## -- Column specification ---------------------------------------------------
------
## cols(
##   teamID = col_double(),
##   TEAM = col_character(),
##   gamesPlayed = col_double(),
##   gamesWon = col_double(),
##   offensiveEfficiency = col_double(),
##   defensiveEfficiency = col_double(),
##   fieldGoalPercentShot = col_double(),
##   turnoverPercentAllowed = col_double(),
##   tempo = col_double(),
##   season = col_double()
## )

GroupedData <- RegSeasonData %>%
  filter(season < 2020) %>%
  group_by(WTeamID, season) %>%
  summarize(total3made = sum(WFGM3), total3attempt = sum(WFGA3), totalftmade
= sum(WFTM), totalftattempt = sum(WFTA), defensiveRebounds = sum(WDR), blocks
= sum(WBlk), steals = sum(WStl), totalAssists = sum(WAst)) %>%
  mutate(threePointAccuracy = total3made/total3attempt) %>%
  mutate(freeThrowAccuracy = totalftmade/totalftattempt) %>%
  subset(select = -c(total3made,total3attempt,totalftmade,totalftattempt))%>%
  rename(teamID = WTeamID)

## `summarise()` has grouped output by 'WTeamID'. You can override using the
`.groups` argument.

Joined <- right_join(GroupedData, TeamData, by = c("season","teamID")) %>%
  arrange(season, tournamentWins)


Joined <- left_join(Joined, OffenseDefense, by = c("season", "teamID")) %>%
  subset(select = -c(TEAM)) %>%
  mutate(winPercentage = gamesWon/gamesPlayed) %>%
  select(season, teamID, teamName, conference, seed,gamesPlayed, gamesWon,
winPercentage, tempo,defensiveRebounds, blocks, defensiveEfficiency, steals,
totalAssists, threePointAccuracy, freeThrowAccuracy, fieldGoalPercentShot,
turnoverPercentAllowed, offensiveEfficiency, tournamentWins)


write_csv(Joined, "TrainingData.csv")
```

Testing Data

Using the same kaggle sources, the testing data analyzes the 68 teams who played in the 2021 NCAA March Madness Tournament. Though the 2021 tournament was still in progress when we wrangled this data, the same variables as the training dataset are included because they reflect regular season performance. The exception to this is wins however, as this is the target we are attempting to predict through our model.

```
library(tidyverse)

RegSeason21 <- read_csv("21RegSeasonData.csv")

##
## -- Column specification -----------------------------------------------------
------
## cols(
##   .default = col_double(),
##   WLoc = col_character()
## )
## i Use `spec()` for the full column specifications.

TeamData21 <- read_csv("21Teams.csv")

##
## -- Column specification -----------------------------------------------------
------
## cols(
##   season = col_double(),
##   teamID = col_double(),
##   teamName = col_character(),
##   conference = col_character(),
##   seed = col_double(),
##   tournamentWins = col_logical()
## )

OffenseDefense21 <- read_csv("OffenseDefense21.csv")

##
## -- Column specification -----------------------------------------------------
------
## cols(
##   teamID = col_double(),
##   TEAM = col_character(),
##   gamesPlayed = col_double(),
##   gamesWon = col_double(),
##   offensiveEfficiency = col_double(),
##   defensiveEfficiency = col_double(),
##   fieldGoalPercentShot = col_double(),
##   turnoverPercentAllowed = col_double(),
##   tempo = col_double(),
##   season = col_double()
## )

GroupedData21 <- RegSeason21 %>%
  group_by(WTeamID, season) %>%
  summarize(total3made = sum(WFGM3), total3attempt = sum(WFGA3), totalftmade
= sum(WFTM), totalftattempt = sum(WFTA), defensiveRebounds = sum(WDR), blocks
= sum(WBlk), steals = sum(WStl), totalAssists = sum(WAst)) %>%
  mutate(threePointAccuracy = total3made/total3attempt) %>%
  mutate(freeThrowAccuracy = totalftmade/totalftattempt) %>%
  subset(select = -c(total3made,total3attempt,totalftmade,totalftattempt))%>%
  rename(teamID = WTeamID)

## `summarise()` has grouped output by 'WTeamID'. You can override using the
`.groups` argument.

Joined21 <- right_join(GroupedData21, TeamData21, by = c("season","teamID"))
%>%
  arrange(season)

Joined21 <- left_join(Joined21, OffenseDefense21, by = c("season","teamID"))
%>%
  subset(select = -c(TEAM)) %>%
  mutate(winPercentage = gamesWon/gamesPlayed) %>%
  select(season, teamID, teamName, conference, seed,gamesPlayed, gamesWon,
winPercentage, tempo,defensiveRebounds, blocks, defensiveEfficiency, steals,
totalAssists, threePointAccuracy, freeThrowAccuracy, fieldGoalPercentShot,
turnoverPercentAllowed, offensiveEfficiency, tournamentWins)


write_csv(Joined21, "TestingData.csv")
```

Figure 4

Visualization 4

The following visualization will attempt to analyze if there is a pattern between conference membership and participation in the March Madness tournament.

```
library(ggplot2)
library(dplyr)

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

options(scipen=999)

Joined <- read.csv("TrainingData.csv")

Joined$season <- as.numeric(Joined$season)

ProjectA <- Joined %>%
  mutate(count = 1) %>%
  filter(tournamentWins >1)
```

```
ProjectB <- ProjectA %>%
  group_by(conference, season) %>%
  summarise(participatingTeams = sum(count)) %>%
  arrange(season, desc(participatingTeams))

## `summarise()` has grouped output by 'conference'. You can override using
the `.groups` argument.

 ProjectC <- ProjectB %>%
   mutate(conAb = ifelse(conference == "American Athletic Conference", "ACC",
                    ifelse(conference == "Big East Conference", "Big East",
                    ifelse(conference == "Big Ten Conference", "Big 10",
                    ifelse(conference == "Pacific-10 Conference", "Pac-12",
                    ifelse(conference == "Atlantic 10 Conference", "A10",
                    ifelse(conference == "Big 12 Conference", "Big 12",
                    ifelse(conference == "Southeastern Conference", "SEC",
conference))))))))

ProjectD <- ProjectC %>%
  mutate(other = ifelse(((season == 2015) & (conAb != "ACC") & (conAb != "Big
East") & (conAb != "Big 10")), "Other",
                  ifelse(((season == 2016) & (conAb != "ACC") & (conAb !=
"Big East") & (conAb != "Big 12")), "Other",
                  ifelse(((season == 2017) &  (conAb != "SEC") & (conAb !=
"Big 10") & (conAb != "Pac-12")), "Other",
                         ifelse(((season == 2018) & (conAb != "ACC") & (conAb
!= "Big East") & (conAb != "Big 12")), "Other",
                                ifelse(((season == 2019) & (conAb != "ACC") &
(conAb != "SEC") & (conAb != "Big 10")), "Other",
                                       ifelse(((season ==2021) & (conAb !=
"ACC") & (conAb != "SEC") & (conAb != "Big 10")), "Other", conAb)))))))

ProjectE <- ProjectD %>%
  group_by(other, season) %>%
  summarise(participatingTeams = sum(participatingTeams)) %>%
  arrange(season, participatingTeams)

## `summarise()` has grouped output by 'other'. You can override using the
`.groups` argument.

ggplot() +
  geom_bar(aes(y = participatingTeams, x = season, fill = factor(other),
label = other), data = ProjectE, stat="identity") +
  theme(panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        axis.line = element_line(colour = "black"),
        legend.position = "none",
        axis.title.y=element_blank(),
        axis.title.x=element_blank()) +
geom_text(aes(x = season, y = participatingTeams, label = other, group =
other), data=ProjectE,position = position_stack(vjust = .5)) +
coord_flip() +
scale_y_continuous(labels = function(participatingTeams)
paste0(participatingTeams))  +
  scale_fill_manual(values = c("Other" = "#ababab",
                               "ACC" = "#006ba4",
                               "SEC" = "#5f9ed1",
                               "Big 10" = "#c85300",
                               "Big 12" = "#a2c8ec",
                               "Big East" = "#ffb60e",
                               "Conference USA" = "#006ba4",
                               "A10" = "#ffb60e",
                               "Pac-12" = "#ff7e0e"))

## Warning: Ignoring unknown aesthetics: label

ggsave("ProjectVisual.png")

## Saving 5 x 4 in image
```

Figure 5

Visualization 5

The following visual attempts to present a pattern/correlation between a team's pre-tournament seedings and their overall performance in the tournament for the years 2015-2019. The steps taken to create this visual were to omit any data that had any missing information. This would eliminate teams that were unable to participate in the tournament. They were then filtered for the years 2015-2019 in order to show a pattern that has been seen in the tournament recently. Then, in order to actually graph the correlation, POSTSEASON and SEED variables were graphed together.

```
library(rvest)

## Loading required package: xml2

library(tidyverse)

## -- Attaching packages --------------------------------------- tidyverse
1.3.0 --

## v ggplot2 3.3.3     v purrr   0.3.4
## v tibble  3.0.6     v dplyr   1.0.3
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.1

## -- Conflicts ------------------------------------------
tidyverse_conflicts() --
## x dplyr::filter()          masks stats::filter()
## x readr::guess_encoding() masks rvest::guess_encoding()
## x dplyr::lag()             masks stats::lag()
## x purrr::pluck()           masks rvest::pluck()
```

```
library(tidyr)
library(dplyr)
library(ggplot2)
library(readr)

allmmData <- read_csv("cbb.csv")

##
## -- Column specification --------------------------------------------------
------
## cols(
##   .default = col_double(),
##   TEAM = col_character(),
##   CONF = col_character(),
##   POSTSEASON = col_character()
## )
## i Use `spec()` for the full column specifications.

mmData <- allmmData %>%
  na.omit(mmData) %>%
  filter(YEAR > 2014) %>%
  group_by(YEAR)
highlight_df <- mmData %>%
    filter(POSTSEASON == "Champions")
  ggplot(mmData, mapping = aes(x = SEED, y = POSTSEASON)) +
    geom_point(color = '#56B4E9') +
    geom_point(data = highlight_df, aes(x = SEED, y = POSTSEASON), color =
'#E69F00', size = 2) +
    facet_wrap(~ YEAR) +
    theme_classic() +
    scale_y_discrete(limits =
c("R68","R64","R32","S16","E8","F4","2ND","Champions")) +
    labs(x = "Tournament Seeding", y = "Tournament Placing",
         title = "Tournament Outcome by Seed (2015-2019)")
```

Modeling

Regression Model

This model evaluates which variables are significant in predicting the number of tournament wins.

```
library(tidyverse)

## -- Attaching packages --------------------------------------- tidyverse
1.3.0 --

## v ggplot2 3.3.3     v purrr   0.3.4
## v tibble  3.0.5     v dplyr   1.0.3
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.1
## -- Conflicts ------------------------------------------
tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
TrainingData <- read.csv("TrainingData.csv")

model <- lm(tournamentWins ~ seed+
winPercentage+tempo+defensiveRebounds+blocks+defensiveEfficiency+steals+total
Assists+threePointAccuracy+freeThrowAccuracy+fieldGoalPercentage+turnoverPerc
entage+offensiveEfficiency, data=TrainingData)

summary(model)
```

```
##
## Call:
## lm(formula = tournamentWins ~ seed + winPercentage + tempo +
##     defensiveRebounds + blocks + defensiveEfficiency + steals +
##     totalAssists + threePointAccuracy + freeThrowAccuracy +
fieldGoalPercentage +
##     turnoverPercentage + offensiveEfficiency, data = TrainingData)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.3670 -0.7246 -0.0600  0.4261  3.7756
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)         -0.7021366  3.0966005  -0.227  0.82077
## seed                -0.0750579  0.0287363  -2.612  0.00942 **
## winPercentage        3.4060347  1.2557766   2.712  0.00704 **
## tempo               -0.0038367  0.0218818  -0.175  0.86092
## defensiveRebounds   -0.0026621  0.0017146  -1.553  0.12148
## blocks               0.0009413  0.0024073   0.391  0.69605
## defensiveEfficiency -0.0424741  0.0208015  -2.042  0.04197 *
## steals               0.0033766  0.0032018   1.055  0.29240
## totalAssists        -0.0016374  0.0015843  -1.034  0.30213
## threePointAccuracy  -4.2717440  2.5404151  -1.682  0.09362 .
## freeThrowAccuracy   -1.2060324  1.6708433  -0.722  0.47093
## fieldGoalPercentage  5.0429742  3.7798250   1.334  0.18308
## turnoverPercentage  -3.7362784  5.6672902  -0.659  0.51019
## offensiveEfficiency  0.0572923  0.0202842   2.824  0.00503 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.013 on 325 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.4374, Adjusted R-squared:  0.4149
## F-statistic: 19.44 on 13 and 325 DF,  p-value: < 2.2e-16
```

Machine Learning approach

The following code builds the three machine learning models we created and analyzes their RSME and R-Sqaured values. All models aim to predict the number of wins each team will have in a given season's March Madness tournament as close to the actual number as possible.

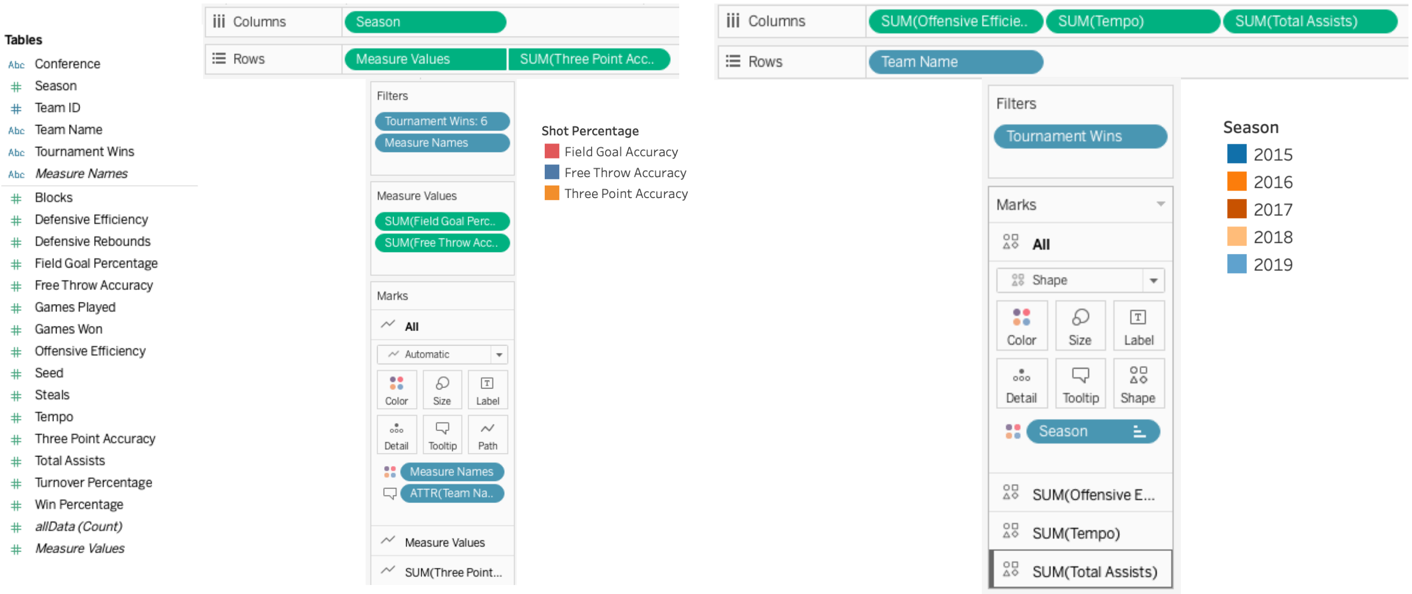
```
library(caret)

## Warning: package 'caret' was built under R version 4.0.5

## Loading required package: lattice

##
## Attaching package: 'caret'

## The following object is masked from 'package:purrr':
##
##     lift

library(tidyverse)
library(dbplyr)

##
## Attaching package: 'dbplyr'

## The following objects are masked from 'package:dplyr':
##
##     ident, sql

library(corrplot)

## Warning: package 'corrplot' was built under R version 4.0.5

## corrplot 0.84 loaded

TrainingData <- read.csv("TrainingData.csv")
TestingData <- read.csv("TestingData.csv")

lmmodel <- train(tournamentWins ~seed+
winPercentage+tempo+defensiveRebounds+blocks+defensiveEfficiency+steals+total
Assists+threePointAccuracy+freeThrowAccuracy+fieldGoalPercentage+turnoverPerc
entage+offensiveEfficiency, data=TrainingData, method= "lm", na.action =
na.exclude)

predictedlm <- predict(lmmodel, TestingData)

postResample(pred =predictedlm, obs = TestingData$tournamentWins)

##      RMSE  Rsquared       MAE
## 1.2398536 0.2514789 0.9754723
```

```
correlation <- cor(TrainingData[,8:20],use = "complete.obs")

findCorrelation(correlation, cutoff =  .7, names = TRUE)

## [1] "totalAssists"      "defensiveRebounds"

reducedmodel <- train(tournamentWins ~seed+
winPercentage+tempo+blocks+defensiveEfficiency+steals+threePointAccuracy+free
ThrowAccuracy+fieldGoalPercentage+turnoverPercentage+offensiveEfficiency,
data=TrainingData, method= "lm", na.action = na.exclude)

predictedreduced <- predict(reducedmodel, TestingData)

postResample(pred =predictedreduced, obs = TestingData$tournamentWins)

##      RMSE  Rsquared       MAE
## 1.1292685 0.2770307 0.8591741

rangermodel <- train(tournamentWins ~seed+
winPercentage+tempo+defensiveRebounds+blocks+defensiveEfficiency+steals+total
Assists+threePointAccuracy+freeThrowAccuracy+fieldGoalPercentage+turnoverPerc
entage+offensiveEfficiency, data=TrainingData, method= "ranger", na.action =
na.exclude)

predictedranger <- predict(rangermodel, TestingData)

postResample(pred =predictedranger, obs = TestingData$tournamentWins)

##      RMSE  Rsquared       MAE
## 1.0275399 0.3951482 0.7492176

modelPredictions <- cbind(TestingData, lmModel1 = predictedlm, lmModel2 =
predictedreduced, rangerModel= predictedranger) %>%
  arrange(predictedranger)

write_csv(modelPredictions, "modelPredictions.csv")
```

Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.

X. Appendix B: Tableau

Figure 1



Tables
Abc Conference
Season
Team ID
Abc Team Name
Abc Tournament Wins
Abc *Measure Names*
Blocks
Defensive Efficiency
Defensive Rebounds
Field Goal Percentage
Free Throw Accuracy
Games Played
Games Won
Offensive Efficiency
Seed
Steals
Tempo
Three Point Accuracy
Total Assists
Turnover Percentage
Win Percentage
allData (Count)
Measure Values

Columns: Season
Rows: Measure Values, SUM(Three Point Acc..
Filters: Tournament Wins: 6, Measure Names
Measure Values: SUM(Field Goal Perc.., SUM(Free Throw Acc..
Marks: All, Automatic
Measure Names
ATTR(Team Na..
Measure Values
SUM(Three Point...

Shot Percentage
🟥 Field Goal Accuracy
🟦 Free Throw Accuracy
🟧 Three Point Accuracy

Columns: SUM(Offensive Efficie.., SUM(Tempo), SUM(Total Assists)
Rows: Team Name
Filters: Tournament Wins
Marks: All, Shape
Season
SUM(Offensive E...
SUM(Tempo)
SUM(Total Assists)

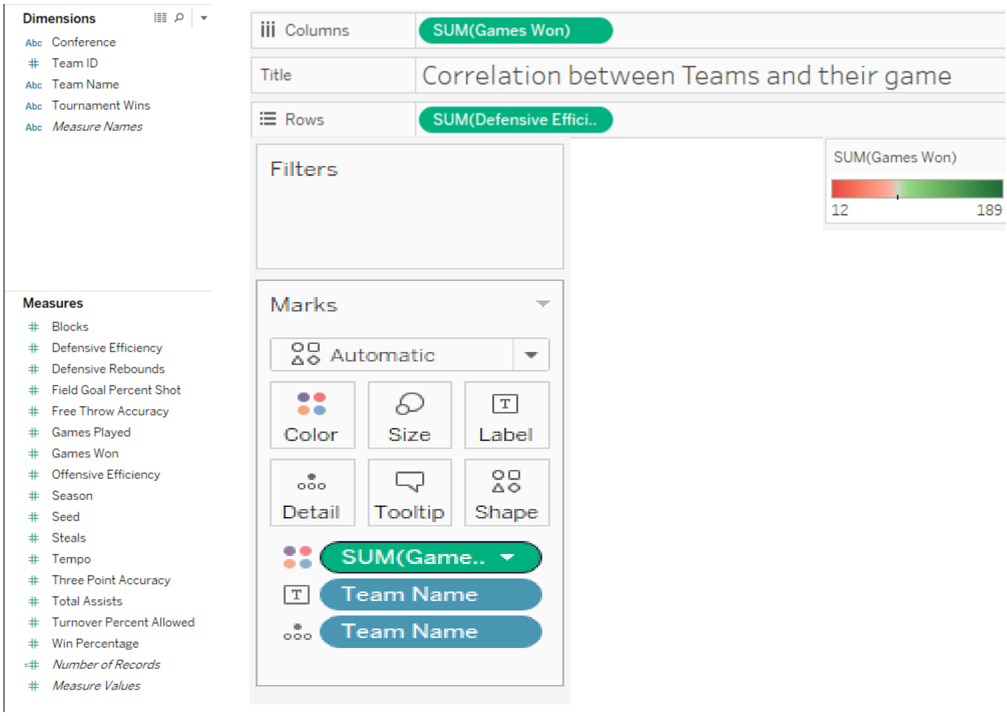
Season
🟦 2015
🟧 2016
🟧 2017
🟧 2018
🟦 2019

Figure 1 utilized the Tableau software to create the two-graph visual given in the report. On the far left, we can see the measures and dimensions available for use after connecting all the data we have from an Microsoft Excel file. The line graph in the figure instructions takes the "T" shape in the middle. The Columns section used the variable Season while the Rows section used Three Point Accuracy, Free Throw Accuracy, and Field Goal Percentage in a dual and synchronized axis format. The data was filtered with Tournament Wins equaling 6 to show the teams that won the tournament that year. The Measure Values part is the combination of the three shooting percentage types. The Team Name was put in the Tooltip and the graph was colored by the three different shooting percentage types as shown in the legend. The scatter plot graph in the figure instructions takes the "T" shape on the far right. The Columns section used the variables Offensive Efficiency, Tempo, and Total Assists while the Rows section used the Team Name. The data was filtered by teams that had 4, 5, or 6 Tournament Wins. The data points were colored by Season. Each pane of data includes an Average Line from the Data Analytics section of Tableau. The axises in the three pains do not include zero and are centered around the average line to save space and eye clutter. Both graphs utilized the Colorblind color palette. The final picture of the graphs were created by putting them into dashboard format with the addition of a small number of text boxes to provide more insight.

Figure 2

Figure 2 utilized the Tableau software to create the graph visual given in the report. Used the file csv file called all data. The heat map was generated by using the sum of turnover rates divided by wins. The map was sorted by team name starting with teams that have won 6 times in the Tournament wins column. Turnover ratio shows that those teams that are closer to 1 or over are the teams that are most likely to score better and even rank better than the teams that have lower than a 1 which is represented in the green color spectrum. The final heat map was created by putting them into dashboard format with the addition of a small number of text boxes to provide more insight.

Figure 3



Columns: SUM(Games Won)

Title: Correlation between Teams and their game

Rows: SUM(Defensive Effici..

Marks: Automatic

SUM(Game..

Team Name

Team Name

Figure 3 utilized Tableau software to create the visual in the report. To import the data, we used the csv file to get the data. The scatter plot was created by using the sum of games won and the sum of the defensive efficiency over the years of the file. As shown above, the games won is put into the columns and the rows contain defensive efficiency. The marks contain the SUM of the Games Won in color and the Team Name in detail as well as label. The teams with the higher defensive efficiency (on average) have a higher sum of games won throughout the years. The final scatter plot was created by putting them into dashboard format with the addition of a small number of text boxes to provide more insight.