What Makes a Championship Winning Team…Statistically?

GROUP 10 – Corey T., Carter H., Luis G., Paul C.

24 June 2021

Introduction

Every year, upwards of 15 million Americans come together to support their favorite college basketball team when the NCAA D1 MBB Tournament begins, affectionally dubbed "March Madness". What makes this tournament so special is that it is full of upsets and late game drama that provide galvanizing and inspiring games throughout the tournament. Furthermore, many Americans partake in predicting the outcomes of tournament games, often times with money on the line with their fellow peers. For example, according to Bloomberg News and Business (https://www.bloomberg.com/graphics/2015-march-madness-gambling/), 9 billion dollars' worth of cash flow was put into betting in the 2015 March Madness Tournament. Having piqued our interest as a group, we wanted to conduct research on 2 questions regarding college basketball teams and their success throughout the season and tournament: What variables are important in determining the win rate/proportion for a team in a given season? Additionally, how can the 2020 Tournament field be simulated with the data present?

To try to answer these questions, we looked at a College Basketball Dataset that provides information about college basketball teams within power 5 conferences, such as their record, field goal percentage, offensive and defensive efficiency, and much more. With respect to the first question, we wanted to dive into many of the different factors of basketball that lead to wins and losses. This question is very important, considering that the sport-betting industry is gaining traction and is being made legal in more states. Studying teams' tendencies can lead to my making better bets as well as creating a successful prediction of the winner of tournament games.

With respect to the second question, a significant part of the March Madness Tournament is deciding what teams get into the tournament, as well as what seed each team gets. Depending on how well a team plays during the regular season determines the seed you get in the post season. Using some of the analysis from answering the first question, we can try to simulate what the seeds will be for a mock season. The 2020 college basketball season is the season we wanted to look into because there was a full regular season, but due to COVID-19 there was no tournament. Using the data of the regular season we can attempt to predict the seeding arrangement for what would have been the 2020 tournament.

Data: Determining Win Rate

In our project, we where overwhelmed by a vast data set with an extensive amount of continuous variables related to performance and results for each team in a respective year, with some of them being of our interest due to most likely being able to predict and create a model for something of high interest such as win rate, as to find what made a "winning team". Win rate was not included in the original data set, so we had to mutate our data frame from the already included G and W fields, by dividing W(wins) by G(games). Our data was first divided into multiple files each corresponding to a certain year, but we decided to combine them together with some information from outside sources, such as 2021 results, which were not included in the original cbb21 file as to have all of the information from 2013-2021 in one file. In later parts of the

project, we decided to focus in ADJDE and ADJOE as we thought they would be useful at predicting win rate. We decided to create a density plot of win rate to check out the distributions and have an idea of what we could expect to find in the data. There were also multiple categorical variables in our data set such as TEAM,CONF,RESULT,SEED,POSTSEASON and YEAR, which we later excluded for one of the sections of our project so as to find those continuous variables that are significant and would result in a good model for calculating win rate.
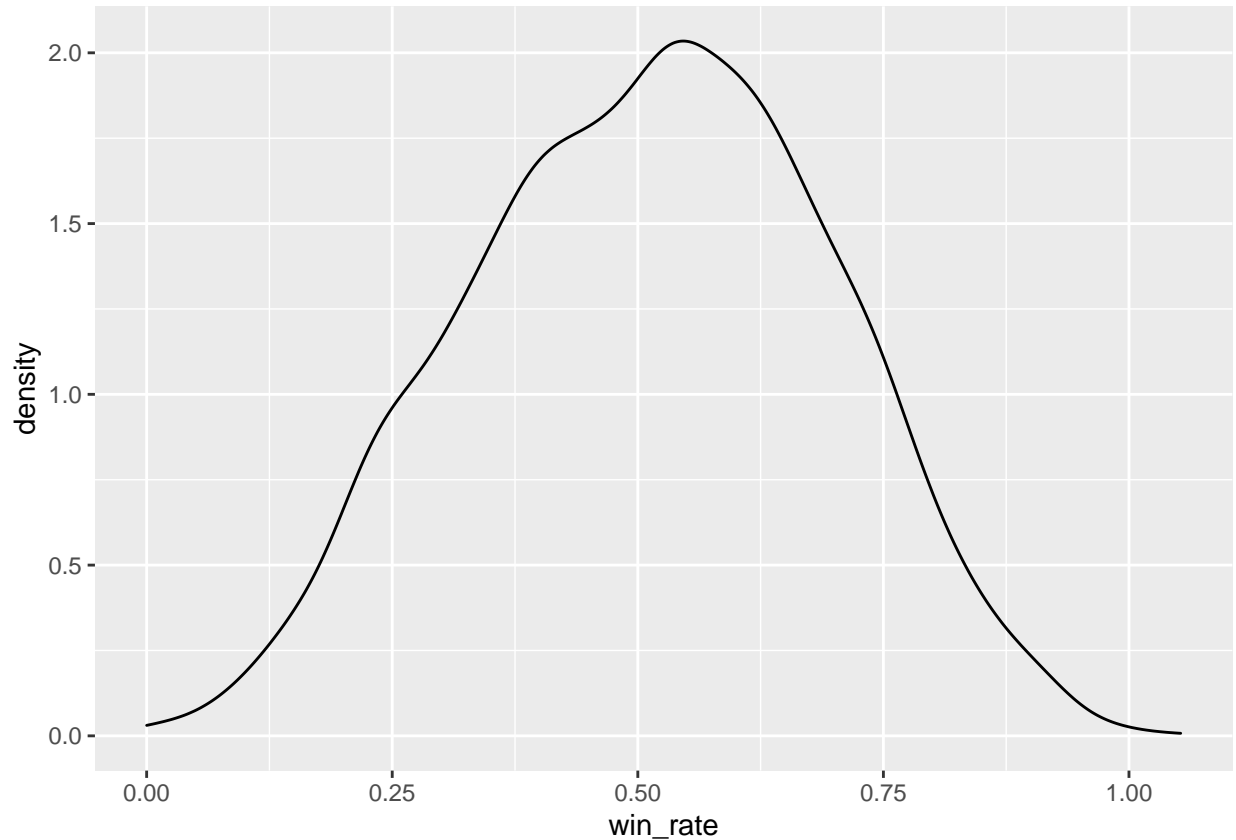

Figure 1: Density Plot of Win Rates of D1 MBB Teams, 2013-2021 (excluding 2020)

Some variables of interest:

```
## # A tibble: 9 x 2
##    Variable Definition
##    <chr>    <chr>
## 1 TEAM      Name of the team
## 2 CONF      Conference in which the team participates in
## 3 win_rate  Win rate, derivated from G and W
## 4 G         Amount of games played
## 5 W         Amount of wins
## 6 ADJOE     Adjusted Offensive Efficiency (Estimate of points scored 100 possesi~
## 7 ADJDE     Adjusted Defensive Efficiency (Estimate of points allowed per 100 po~
## 8 ORB       Offensive Rebound Rate
## 9 TOR       Turnover Percentage Allowed
```

Figure 2: Data Dictionary for Some Important Variables (a comprehensive dictionary can be found at the Kaggle link in the 'Data: Tournament Simulation' section)

Some of what is included in the modified data frame:

```
## 'data.frame':    2802 obs. of  24 variables:
##  $ TEAM      : chr  "Indiana" "Gonzaga" "Kansas" "Louisville" ...
##  $ CONF      : chr  "B10" "WCC" "B12" "BE" ...
##  $ ADJOE     : num  121 119 112 116 108 ...
##  $ ADJDE     : num  89.7 90.2 86.2 84.5 85 90.8 89.4 91.5 87.8 93.2 ...
##  $ BARTHAG   : num  0.969 0.96 0.951 0.974 0.938 ...
##  $ EFG_O     : num  54.7 54.9 53.3 50.6 51.1 51.9 50.6 53.9 50.6 49.6 ...
##  $ EFG_D     : num  44 44.9 41.5 44.8 43 45.7 45.4 45.5 44.5 46.2 ...
##  $ TOR       : num  19.3 17.2 20.3 18.3 20.1 16.7 16.2 15.6 20.8 20.9 ...
##  $ TORD      : num  20.9 20.8 18.4 27 22.4 18.4 20.7 20.4 19.2 19.5 ...
##  $ ORB       : num  39 37.8 33.8 38.2 30.4 31.3 31.8 28.8 36.1 38.4 ...
##  $ DRB       : num  31.4 29.8 29.3 33.3 31 28.4 28.2 32.3 27.6 33.1 ...
##  $ FTR       : num  45.8 40.8 39.5 40 36.8 34.6 35.5 39.2 36.6 40.9 ...
##  $ FTRD      : num  27 29.9 32 34.9 35.3 28.3 29.5 32.7 32.4 31.7 ...
##  $ X2P_O     : num  52 55 52.9 50.8 50.2 50.6 49.4 50.8 50.4 51.6 ...
##  $ X2P_D     : num  43.2 42.1 39.3 43.4 41.4 44 43.6 46.2 44.3 44.9 ...
##  $ X3P_O     : num  40.3 36.5 36.4 33.3 35.3 36.2 35.6 39.9 34.1 29.6 ...
##  $ X3P_D     : num  30.4 32.9 30.3 31.8 30.7 32.6 32.4 29 30.1 32.3 ...
##  $ ADJ_T     : num  67.8 65.1 67.7 67.1 62.5 63 65.3 67.8 64.4 64.6 ...
##  $ WAB       : num  7.8 7.6 7.5 9 6.6 6.3 7.2 7.5 6.7 4.5 ...
##  $ RESULT    : chr  "S16" "R32" "S16" "Champions" ...
##  $ SEED      : int  1 1 1 1 2 2 2 2 3 3 ...
##  $ POSTSEASON: int  1 1 1 1 1 1 1 1 1 1 ...
##  $ YEAR      : int  2013 2013 2013 2013 2013 2013 2013 2013 2013 2013 ...
##  $ win_rate  : num  0.806 0.912 0.838 0.875 0.781 ...
```

Figure 3: Table of Variables and Their Respective Data Types from Dataset used in Figure 1

Results: Determining Win Rate

In basketball, there are a lot of factors that go into a team becoming a winning team, but we thought that there would be certain stats that could measure how well a team performs in the court, with AD-JOE(adjusted offensive efficiency) and ADJDE(adjusted defensive efficiency) being the first ones to cross our mind. The purpose of this section is to compare what could matter most in having a winning team, with ADJOE(adjusted offensive efficiency) and ADJDE(adjusted defensive efficiency) being analyzed with respect to win rate, a statistic calculated from the original data as wins divided by games, as some teams had much more games played than others so the data would be biased towards those teams as more games played meant a bigger chance for much more wins. First, we decided to plot the win rates of all teams with respect to both ADJOE and ADJDE, and we expected those with higher ADJDE to win the less whereas those with higher ADJOE to win the most. There could be teams with a high enough ADJOE that could diminish a high ADJDE and still result in big wins, so we wanted to check whether a balance of both would be favored. First, we wanted to make sure what was more favored in terms to win rate, so by plotting ADJOE and ADJDE vs win rate we were able to see that a low ADJDE and a high ADJOE often resulted in higher win rates, which is as expected. A good defensive team could be expected to have a low ADJDE, while a good offensive team would be expected to have a high ADJOE, so we wanted to research both ends.

```
## 'geom_smooth()' using formula 'y ~ x'
```

Figure 4: Relationship between Adj. Offensive Efficiency, Win Rate

```
## `geom_smooth()` using formula 'y ~ x'
```

Relationship between Adjusted Defensive Efficiency and Win Rate
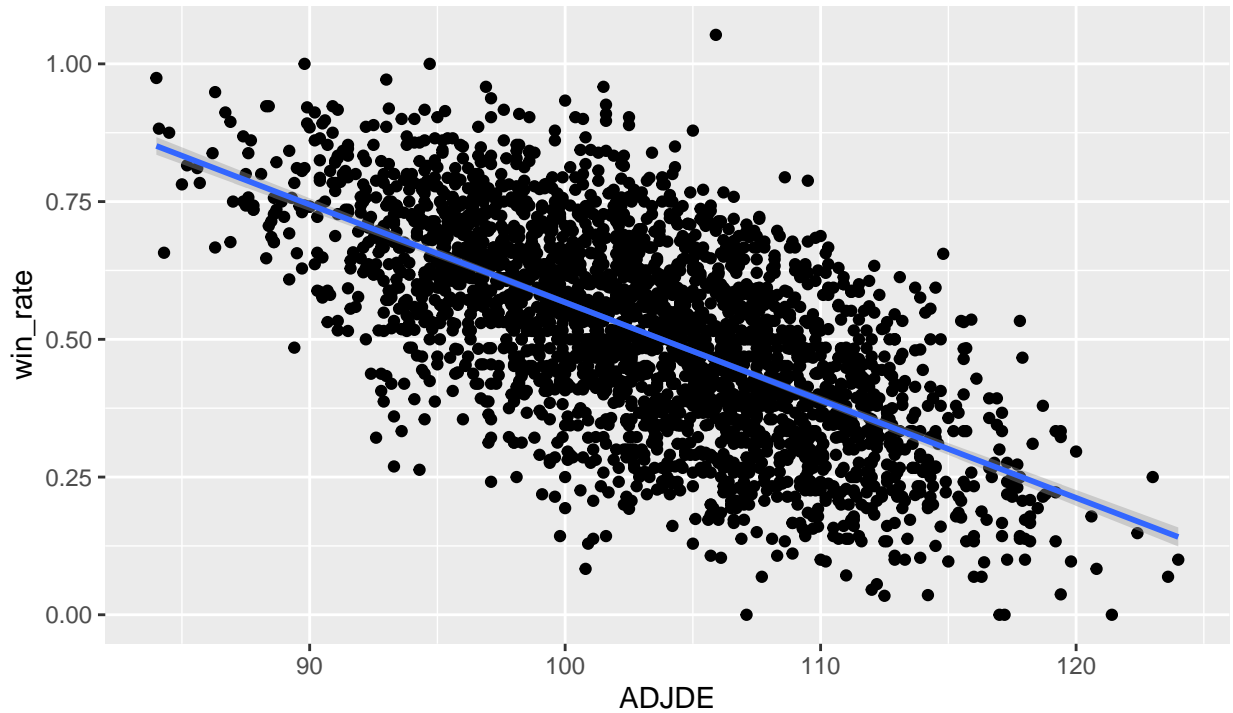Data shown is of teams in tournaments from 2013–2021.
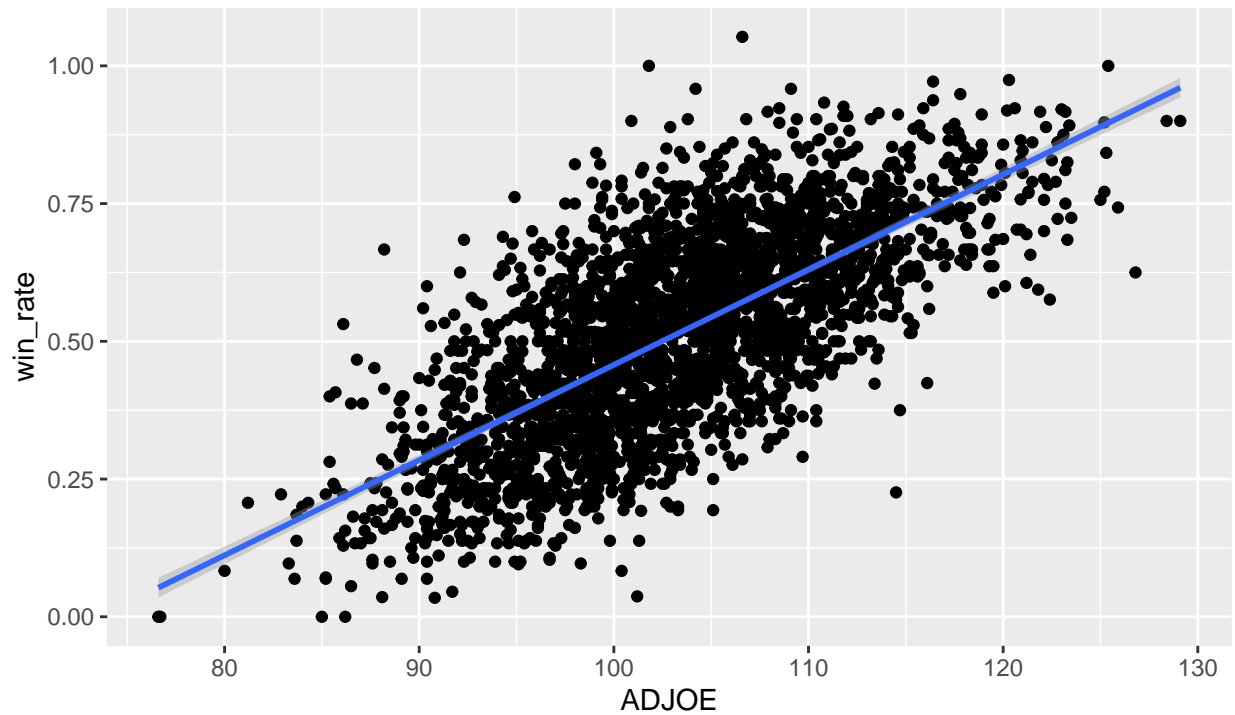
WR: Win Rate; ADJDE: Adjusted Defensive Efficiency: Points Allowed per 100 Possessions

Figure 5: Relationship between Adj. Defensive Efficiency, Win Rate

But still, this is not enough to answer our question, as we still do not know if teams could have both a high ADJOE and high ADJDE or low ADJOE and low ADJDE and still win, or if other variables are still more impactful (that is, additional variables include important explanations in the variability of Win Rates). So, we decided to categorize win rate and ADJDE into three categories each, where each category was an interval of that variable. We divided our graphs by facets using three levels of ADJDE with three box plots each representing a win rate interval. We also used anova to make sure that the means between groups are different, which is what we expected since the start.

```
## # A tibble: 3 x 8
##   .y.   group1      group2            p   p.adj p.format p.signif method
##   <chr> <chr>       <chr>         <dbl>   <dbl> <chr>    <chr>    <chr>
## 1 ADJOE >75         <75 and >50 1.14e- 34 1.1e- 34 <2e-16   ****     Wilcoxon
## 2 ADJOE >75         <50         4.50e-114 9  e-114 <2e-16   ****     Wilcoxon
## 3 ADJOE <75 and >50 <50         7.08e-167 2.1e-166 <2e-16   ****     Wilcoxon
```
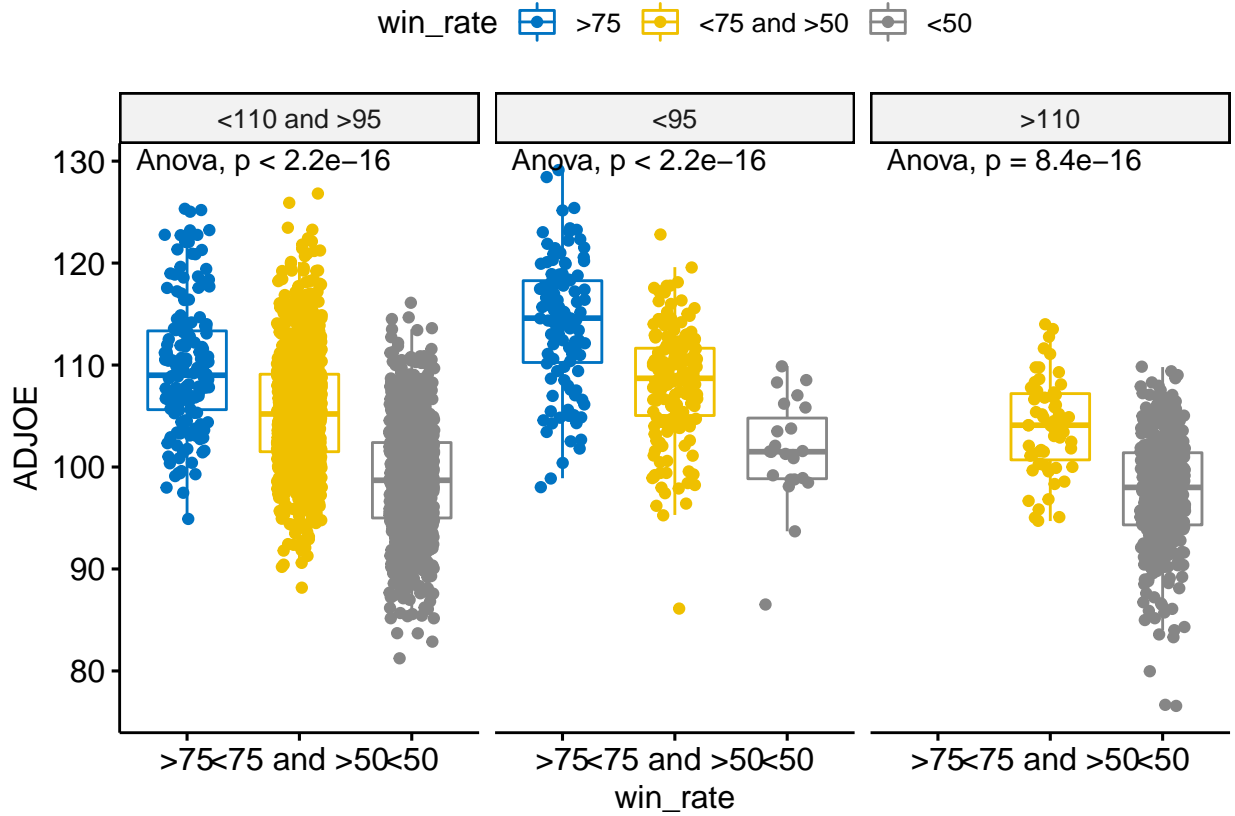
Figure 6: ANOVA Test for Difference of Means between Categorical Groups of Win Rates

By looking at the box plots, we can see more clearly that there is a relation between ADJOE and win rate, as the boxplots seem to be in a lower ADJOE range as the win rate decreases(which is marked by intervals). What is interesting is that we can eyesight several differences in the groups of ADJDE, with the most significant being that of no team having >75 winrate while also having >110 ADJDE. There also seems to be slight differences in quartiles and ADJOE highs in the box plots when comparing by ADJDE, with the most significant being the difference in highs in ADJOE in the <75 and >50 win rate interval when comparing all ADJDE groups. Another interesting thing is those teams that had a <95 ADJDE seemed to perform well, as most points in the group seem to be concentrated in the <75 and >50 and >75 win rate interval, with a really small fraction being in the <50 win rate interval. Even then, with a much less concentration in points in those ranges not >75, it still remains similar to >75 in magnitude of point presence but with higher ADJOE quartiles and highs, as there are points even in the 130 range. Through these box plots and the above mentioned graphs we were not only able to notice about the effect of ADJOE and ADJDE values and the respective win rates, but also of a suspecting interaction between ADJDE and ADJOE. Because of this, for the next step we decided to use lasso so that we could figure out the most significant variables which we may have missed due to tunneling in ADJOE and ADJDE. In addition, we could also check if there is an interaction between ADJOE and ADJDE, and graph it in case there was.

```
##   [1] "(Intercept)"    "ADJOE"         "ADJDE"          "BARTHAG"
##   [5] "EFG_O"          "EFG_D"         "TOR"            "TORD"
##   [9] "ORB"            "X2P_D"         "X3P_O"          "X3P_D"
##  [13] "ADJ_T"          "WAB"           "ADJOE:ADJDE"    "ADJOE:BARTHAG"
##  [17] "ADJOE:EFG_D"    "ADJOE:TORD"    "ADJOE:DRB"      "ADJOE:FTRD"
##  [21] "ADJOE:X3P_D"    "ADJOE:WAB"     "ADJDE:BARTHAG"  "ADJDE:EFG_O"
```

6

```
##  [25] "ADJDE:EFG_D"    "ADJDE:TORD"    "ADJDE:ORB"     "ADJDE:DRB"
##  [29] "ADJDE:FTR"      "ADJDE:X2P_O"   "ADJDE:X2P_D"   "ADJDE:X3P_O"
##  [33] "ADJDE:ADJ_T"    "BARTHAG:EFG_D" "BARTHAG:TOR"   "BARTHAG:ORB"
##  [37] "BARTHAG:DRB"    "BARTHAG:FTR"   "BARTHAG:X2P_O" "BARTHAG:X2P_D"
##  [41] "BARTHAG:X3P_O"  "BARTHAG:ADJ_T" "BARTHAG:WAB"   "EFG_O:EFG_D"
##  [45] "EFG_O:TORD"     "EFG_O:DRB"     "EFG_O:FTRD"    "EFG_O:X2P_D"
##  [49] "EFG_O:X3P_O"    "EFG_O:X3P_D"   "EFG_O:ADJ_T"   "EFG_O:WAB"
##  [53] "EFG_D:ORB"      "EFG_D:FTR"     "EFG_D:X2P_O"   "EFG_D:X2P_D"
##  [57] "EFG_D:X3P_O"    "EFG_D:X3P_D"   "EFG_D:ADJ_T"   "EFG_D:WAB"
##  [61] "TOR:TORD"       "TOR:ORB"       "TOR:DRB"       "TOR:FTR"
##  [65] "TOR:FTRD"       "TOR:X2P_O"     "TOR:X2P_D"     "TOR:X3P_O"
##  [69] "TOR:X3P_D"      "TOR:ADJ_T"     "TOR:WAB"       "TORD:ORB"
##  [73] "TORD:DRB"       "TORD:FTR"      "TORD:FTRD"     "TORD:X2P_O"
##  [77] "TORD:X3P_O"     "TORD:X3P_D"    "TORD:WAB"      "ORB:DRB"
##  [81] "ORB:FTR"        "ORB:X2P_O"     "ORB:X2P_D"     "ORB:X3P_O"
##  [85] "ORB:X3P_D"      "ORB:ADJ_T"     "ORB:WAB"       "DRB:FTRD"
##  [89] "DRB:X2P_O"      "DRB:X2P_D"     "DRB:X3P_O"     "DRB:X3P_D"
##  [93] "DRB:WAB"        "FTR:X2P_O"     "FTR:X2P_D"     "FTR:X3P_O"
##  [97] "FTR:ADJ_T"      "FTR:WAB"       "FTRD:X2P_O"    "FTRD:X3P_D"
## [101] "FTRD:ADJ_T"     "FTRD:WAB"      "X2P_O:X2P_D"   "X2P_O:X3P_O"
## [105] "X2P_O:X3P_D"    "X2P_O:ADJ_T"   "X2P_O:WAB"     "X2P_D:ADJ_T"
## [109] "X2P_D:WAB"      "X3P_O:ADJ_T"   "X3P_O:WAB"     "X3P_D:ADJ_T"
## [113] "X3P_D:WAB"      "ADJ_T:WAB"
```

Figure 7: List of Nonzero Coefficients from LASSO Model

```
##
## Call:
## lm(formula = win_rate ~ ADJOE + ADJDE + ADJOE:ADJDE, data = NONUMERIC)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.42821 -0.07878 -0.00239  0.07549  0.51567
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.804e+00  4.597e-01   8.276  < 2e-16 ***
## ADJOE       -2.093e-02  4.406e-03  -4.751 2.13e-06 ***
## ADJDE       -4.476e-02  4.466e-03 -10.023  < 2e-16 ***
## ADJOE:ADJDE  3.281e-04  4.304e-05   7.625 3.33e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1135 on 2798 degrees of freedom
## Multiple R-squared:  0.6126, Adjusted R-squared:  0.6122
## F-statistic:  1475 on 3 and 2798 DF,  p-value: < 2.2e-16
```
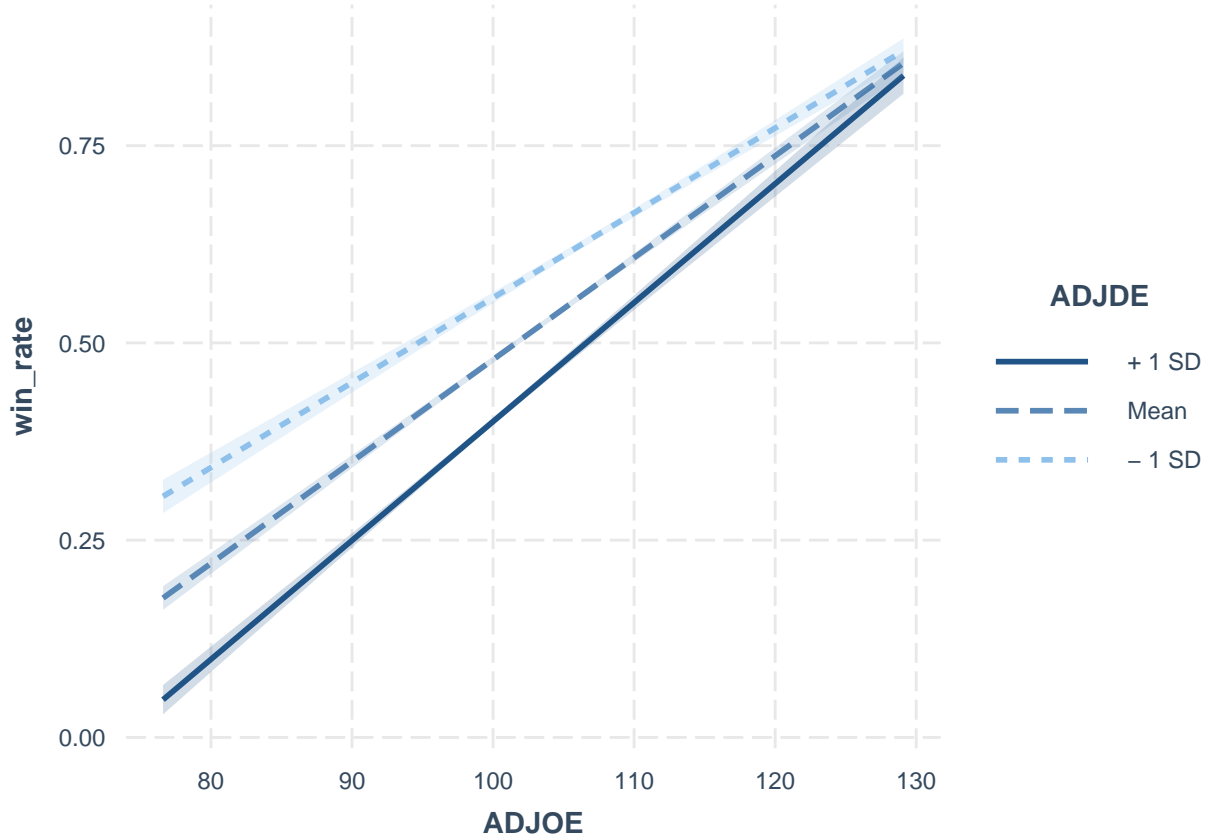
Figure 8: Interaction Plot between Adj. Offensive, Defensive Efficiency and Win Rate

After performing LASSO techniques, we were able to notice that that win rate was not something attributed to just a couple variables like ADJOE and ADJDE as we suspected, but a lot of variables also included in the data set and their interactions were also found to be significant. We were overwhelmed by the amount of significant variables outputted by lasso, as we did not consider them at the start of the project, but since they were a lot for the scope of this project we still decided to focus in ADJOE and ADJDE as their interaction mentioned before was found to be significant. In addition, we found that that just using ADJOE or ADJDE by themselves were significant enough, as shown by the table with the col names of those found to be significant. Next, we created an interaction plot of ADJOE with ADJDE and compared the three different lines categorized by different ADJDE standard deviations by win rate, and our results were what we expected as we found the line with an ADJDE of -1 SD to be the one with a higher intercept and higher win rate peak, and that with an ADJDE of +1 SD to be the one with the lowest intercept and lowest peak. However, as ADJOE increased, the win rate of all three lines also increased, and were fairly similar at where they peaked but still the difference of peaks could be seen clearly. In addition, ADJOE and ADJDE were found to be significant not only with their interaction, but also in interactions with other variables. Still, win rate was something much more complex that could not only be explained by just the interaction of ADJDE and ADJOE. A good model for calculating win rate could not only be created using ADJOE and ADJDE, but would also include a lot more variables, especially those such as WAB(wins above cutoff for making March Madness) or barthag(power rating).

Data: Tournament Simulation

Of particular interest in simulating the 2020 Tournament are datasets that encapsulate teams' summary statstics for any given season (2013-2021) and another that recounts the history of the Tournament by game. The first of these datasets contains a number of important summary statistics, many of which were discussed earlier in this section (e.g., ADJOE, ADJDE, ORB, TOR); each statistic is a representation of some aspect of the game of basketball, and higher/lower values of some statistics are often accompanied by higher win rates (see the Results section for additional commentary). Figure 9 provides a visualization of how this dataset is structured; note that, for 2020, the 'RESULT','SEED', and 'POSTSEASON' columns do not exist (these indicate how far a team went in the tournament, their seed that year, and whether or not they were invited, respectively).

	TEAM	CONF	G	W	ADJOE	ADJDE	BARTHAG	EFG_O	EFG_D	TOR
1	Michigan	B10	24	20	118.10	91.10	0.95	54.90	44.90	16.30
2	Baylor	B12	24	22	123.20	94.50	0.95	57.50	49.10	17.60
3	Illinois	B10	29	23	117.70	90.40	0.95	55.60	46.60	18.20
4	Gonzaga	WCC	26	26	125.40	89.80	0.98	61.00	47.50	16.10
5	Iowa	B10	29	21	123.50	95.70	0.95	54.60	48.30	13.30

Figure 9: Sample Summary Statistics from 2021 Season (some columns not shown due to cutoff)

Additionally, Figure 10 shows a small sample of games from the history of the NCAA Tournament (all games since 1985 were stored in this format).

Year	Seed1	Score1	Team1	Team2	Score2	Seed2	ScoreDiff	Upset
2013	1	79	Louisville	North Carolina A&T	48	16	31	0
2013	2	73	Duke	Albany	61	15	12	0
2013	3	65	Michigan St.	Valparaiso	54	14	11	0
2013	4	64	Saint Louis	New Mexico St.	44	13	20	0

Figure 10: Sample Data from History of NCAA Tournament Games

Sourcing Used Datasets

The following links or methods of obtaining the predominant resources used in the compilation of this report:

- Kaggle Repository containing dataset(s) depicted in Figure 1: **Link**

- ESPN Link to 2021 Tournament Data (merged with Kaggle Repository to include results from this year): **Link**

- Link to Warren Nolan's website containing historical BPI Data (merged with Kaggle Repository and ESPN data): **Link**

- Data.World Link containing datasets depicted in Figure 2: **Link**

Results: Simulating the 2020 Tournament

As the world grinded to a halt in March of 2020, college basketball, which had been preparing for conference tournaments and the "Big Dance," as the tournament to which the entire season culminates is coloquially referred, stopped alongside it. The cancellation brought about speculation of who would have comprised the elusive fields of 68, what "Cinderella stories" would have been, and who would have emerged the Champion.

Fortunately, advancements in machine learning and classification techniques, along with their practical applications in the blossoming field of data science, afford the ability to re-create college basketball's greatest tournament. The following sections will be dedicated to the discussion of how the 2020 NCAA D1 MBB Tournament can, and will, be simulated. The first section will be dedicated to the definition of the aforementioned "field of 68" teams, along with relevant terminology and slang phrases used to discuss the results of such tournaments. Next, the development of classification tools to structure the tournament field will be discussed, which include methods such as Support Vector Machines (SVM), K-Nearest Neighbors (KNN) or K-Means Clustering algorithms, and the use of Artifical Neural Networks (ANN). Finally, the same methods will be appropriated to creating methods of determining the victor of a particular match-up. These two facets comprise how the tournament progresses. Results of each round, along with the eventual champion (no virtual banner for UNC, unforunately), will be presented as well.

Relevant Terminology

Before discussing how any classification methods work or their respective performances, a framework for understanding them must be established, especially for audiences who are not acclimated nor familiar with the college basketball environment in recent years (i.e since 2011, when the 'First Four' games were included to broaden the field from 64 to 68 teams). Therefore, let the following list serve as a guide to the words and phrases repeatedly used throughout this analysis:

- *Field*: This is a coloquial term referencing the entire scope of teams who receive an invitation to the tournament.

- *Automatic [Bid]*: This term refers to teams who are conference champions within their contemporaneous and respective tournaments. Conference champions receive an invitation to the tournament, no matter from where they originate.

- *At-Large*: This term refers to teams who received an invitation to the tournament but were not automatically entered; a committee is responsible for determining these teams.

- *Seed*: A team's ranking within their region/pool, ranging from 1 (Best) to 16 (Worst).

- *Region*: The tournament is divided into four regions, and thus four of each seed enter the first round, dubbed the 'Round of 64.' (Six '11-seeds' and '16-seeds' are chosen, but four of each enter the 'First Four' games, likely determined by a committee.)

- *Upset*: Such refers to a game in which the result is a significantly lower-seeded team defeating a high-seeded team (e.g., see Duke-Lehigh in 2012, Duke-Mercer in 2014, Virginia-UMBC in 2019, Ohio State-Oral Roberts in 2021).

Other terms, such as the names given to different rounds of the tournament, should be self-explanatory; that is, the number in any respective name (aside of the Championship Round) indicates the number of remaining teams.

The Field of 68

Since 2011, in which the tournament has been held included 68 teams who shared one common goal: "Win or Go Home." But before such goals can be realized, they first are made aware of their initial opponent. A committee is appointed (as to who exactly comprises said group, the authors are unsure) to carry out the selection process, which involves the assignment of seeds 1-16 to teams based on a number of statistics (e.g., Strength of Schedule, or SOS, which is a ranking given to identify the team(s) with the most difficult slate of games over the entire season). Four teams are allotted each seed, with the aforementioned explanation of two additional seeds being offered to fill out the first games that are played the day before the tournament is scheduled to begin. As a result, "1-seed" teams play "16-seed" opponents, "2-seed" teams play "15-seed" opponents, and so on. Teams are then split into four regions, where each region is a microcosm consisting of a single team seeded 1-16. The winners of each region arrive at the 'Final Four' round, and the winners of those games compete for the title.

Methods

Accurately structuring any tournament field can be difficult, and March Madness is no exception. Bracketologists (such as Joe Lunardi), individuals who employ analysis of historical records and current performances to predict fields, labor throughout the season to communicate to the public their opinions on who will reach the postseason. This section allows the authors to assume this role (and, when the time comes, hopefully win their next office or family bracket competitions) with the help of aforementioned classification techniques.

The goal of this subsection is to structure the field of 68 teams that would have hypothetically competed for the championship. Such is a two-step process: first, the field will be generated by determining worthiness for a postseason berth. Afterwards, teams will be seeded according to a separate set of classifiers. Finally, regression techniques will be used to simulate matchups designed in a traditional tournament format (see Appendix for the full results)!

It is worth noting that all of the described methods were tested on data *independent* from data on the 2020 season beforehand to ensure validity of the process (e.g., prevent overfitting and poor classification amongst test data) and become acclimated with the programming necessary to achieve an end-goal. Some of the included output will be from randomly selected test data, others exclusively the 2020 season.

Postseason Berth

SVM

Simply put, the term "postseason berth" refers to any given team being included in the tournament, whether in the First Four or Round of 64. In order to simulate this for the 2020 season, classifiers were trained on data from the 2013-2021 tournaments (excluding 2020). No weighting nor mutation of the data was conducted as a result of how far a team progressed in the tournament, but instead only their presence was noted.

A train-test split ratio of 80/20 (that is, trained on 80% of the data, tested on 20%) was used in the generation of an SVM classifier. Hyperparameters, along with their respective results of Accuracy ("Acc"), Precision ("Prec"), and Sensitivity ("Sens"), common metrics used to describe SVMs and Confusion Matrices, were used to tune the classifier; the best set of aforementioned metrics belonged to the parameters $C = 1, \gamma = 1$, where C is the 'cost' parameter and γ the 'gamma' parameter.

Additionally, when tested, this SVM performed relatively well, as indicated by the confusion matrix

$$\begin{bmatrix} TP & FP \\ FN & TN \end{bmatrix} = \begin{bmatrix} 437 & 15 \\ 28 & 80 \end{bmatrix}$$

where TP signifies "True Positive," FN "False Negative," and the like.

Figure 11: Confusion Matrix of SVM Postseason Berth Classifier

Predictions for Postseason Births within Testing Data
Original Data of 2013-2021 Games, exc. 2020; See Caption for Color Guide

Blue: Team reached postseason.
Red: Team did not reach postseason.
Black: The actual results.
LARGER SIZES indicate False Positive (Red), False Negative (Blue).
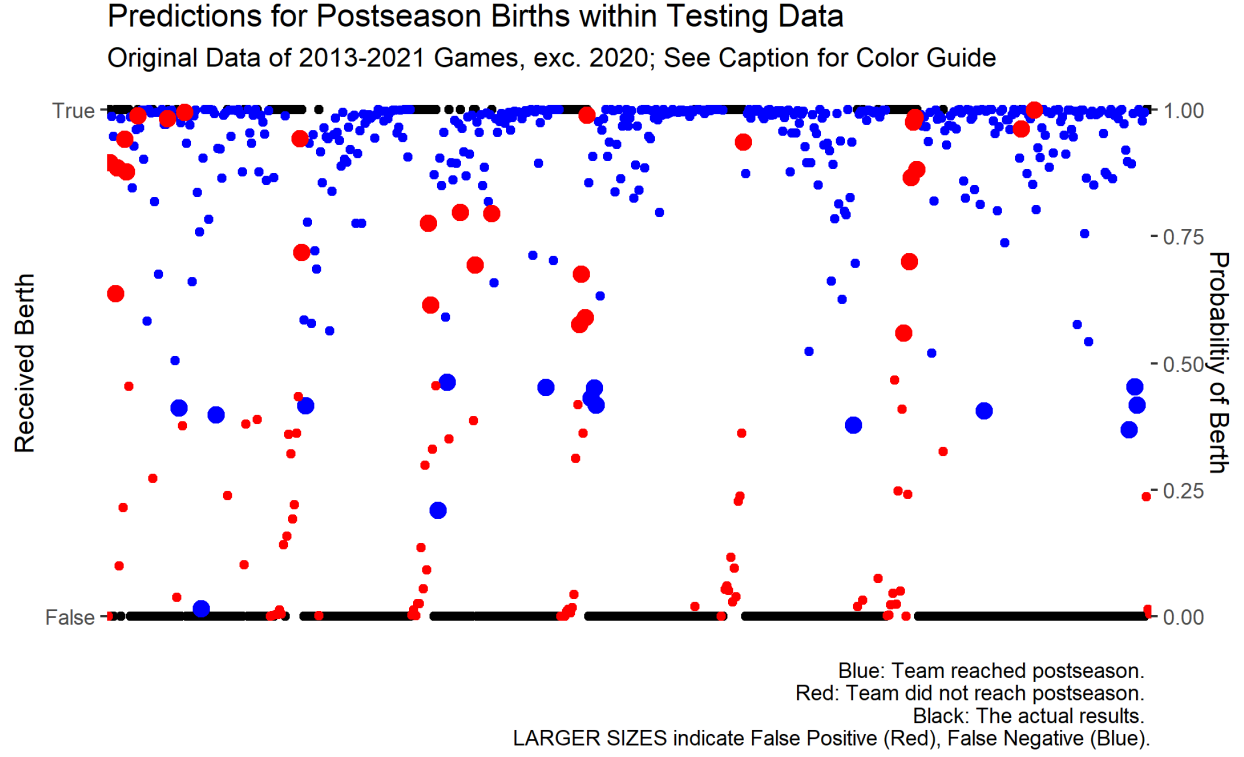
Figure 12: Probabilities of Postseason Berths, 2013-2019 and 2021)

As evidenced by Figures 11 and 12, the SVM was confident in many of its classifications of teams' eligibility for postseason appearances, so much so that it failed to identify an adequate-sized field when validated on data from 2020. To combat this shortcoming, any results marking a team eligible for an appearance were coupled with the necessary number of *low-confidence* predictions of a team *not* making it into the postseason (e.g., as 46 teams were automatically selected, 22 were chosen among those initially not selected).

KNN

In addition to an SVM, a K-Means Clustering approach known as "K-Nearest Neighbors" was introduced. To tune the hyperparameter k for this algorithm, classifiers tested on data from 2020 were compared with respect to $\mu_{pv!=av}$. Note that 'pv' and 'av' represent the predicted and actual values (e.g., the predictions and test dataset, respectively). Figure 13 on the following page shows that a value of $k = 6$ minimized this function and was thus used in the determination of postseason berths.
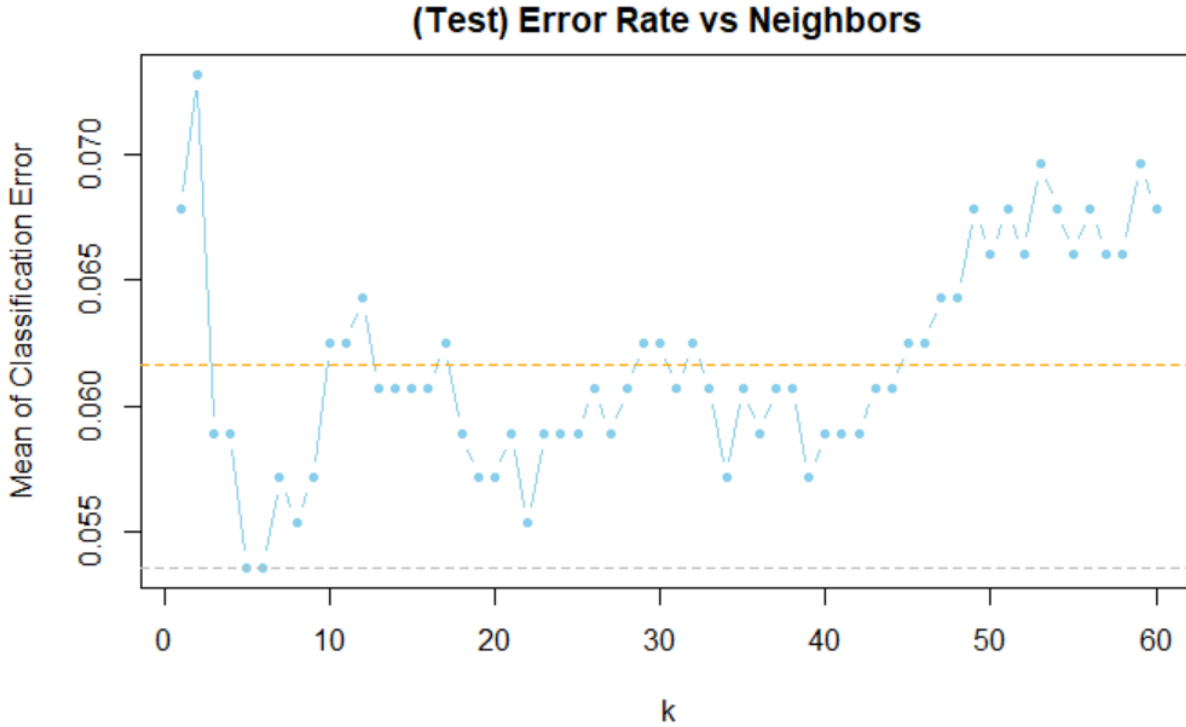
Figure 13: Plot of Hyperparameter Tuning for Postseason-Berth KNN Classifier)

When validated using data from 2020, a similar scenario occurred to that when using the SVM classifier; as a result, 49 teams received automatic berths, and 19 were chosen among those initially not selected.

ANN

Finally, an Artificial Neural Network was included for determining postseason eligibility. The model was trained and tested on the same data sources as the SVM and KNN before also being validated with data from 2020. Surprisingly, this network performed extremely well, boasting a "train-accuracy" of ~0.925 and "test-accuracy" of ~0.916.

The model structure is defined below (taken from output of the .py file from where the model originates).

```
Model: "sequential_1"

_____
Layer (type)                 Output Shape              Param #
=================================================================
dense_3 (Dense)              (None, 64)                1280

_____
dense_4 (Dense)              (None, 32)                2080

_____
dense_5 (Dense)              (None, 2)                 66

=================================================================
Total params: 3,426
Trainable params: 3,426
Non-trainable params: 0
_____
```

Figure 14: Table of Postseason ANN Structure)

13

Field Generation

When each of the aforementioned classifiers were validated using data from 2020, their predictions were used to each determine a set of teams to label with seeds. The code (under the 'Field Generation' section of the main script) for the following section employs filtered datasets from the validation data for each set of predictions. Rather than re-hash descriptions of each classifier for this different purpose (because accuracies cannot be measured, as there are no values to use as comparison), the results of each classifier will be presented, along with an explanation of necessary methods of imputation and analysis to arrive at a finalized tournament field.

For *each* classifier, its prediction indices for postseason berths were cross-referenced with team-names from the validation data to obtain a list of 68 teams. As mentioned before, if a classifier failed to predict the necessary number of berths, then additional values were imputed based on having the lowest remaining probabilities of *not* being given an invitation. Each list was then trained on a separate set of classifiers (of the same techniques as above) that labeled each team with a seed. As a result, each seed classifier yielded a *68x2* table represent its predicted field and seeds for each field.

The three tables were then joined, allowing for missing values, to arrive at a *105x3* table. Automatic bids were determined by selecting teams who had predicted values for all three classifiers (before imputation), ranking their average seeds, and seeding in chunks of 4 teams per increment. Missing values were imputed with the value 16 as any classifier who desired *not* to include them in predictions communicated that something about such teams' statistics did not warrant a postseason appearance and, as 16-seeds often have an early exodus (only *1* in the entire tournament history has won a game), such value acts as a penalty to ultimately match such teams with higher-seeds who are, from a historical perspective, more likely to win.

At-large bids were selected under the following process: first, the average seed of all teams not receiving automatic bids were computed and ranked in ascending order. Then, the highest *n* rankings were selected. It should be admitted that **this process is different than how selection committees normally proceed**, with conference champions being given automatic seeds and all other teams competing for at-large bids; the authors felt that the process used in this report incorporated a rigorous use of classification techniques and that such process would be better suited without the premature filtration of teams that were not conference champions in 2020.

The following table, beginning on the next page, shows the rankings of the 43 automatic bids and 25 at-large bids (the 'Rank' column starts anew because the two groups were compiled and ranked separately):

TEAM	KNN	SVM	ANN	means	Rank	Seed
Kansas	1	1	1	1	1	1
Gonzaga	1	1	1	1	2	1
Baylor	3	3	1	2.333333	3	1
Dayton	3	3	1	2.333333	4	1
San Diego St.	5	2	1	2.666667	5	2
Florida St.	3	4	1	2.666667	6	2
Duke	3	3	3	3	7	2
Creighton	5	3	1	3	8	2
Villanova	5	3	1	3	9	3
Maryland	6	3	1	3.333333	10	3
Michigan St.	6	3	3	4	11	3
Louisville	5	5	3	4.333333	12	3
Oregon	5	5	3	4.333333	13	4
Kentucky	6	4	3	4.333333	14	4
Virginia	5	5	3	4.333333	15	4
Ohio St.	6	5	3	4.666667	16	4
Penn St.	4	5	5	4.666667	17	5
Auburn	6	6	3	5	18	5
West Virginia	5	6	5	5.333333	19	5
Butler	6	7	3	5.333333	20	5
Seton Hall	9	5	3	5.666667	21	6
Michigan	5	8	5	6	22	6
Arizona	3	5	11	6.333333	23	6
BYU	6	7	6	6.333333	24	6
Wisconsin	9	7	3	6.333333	25	7
Houston	8	7	5	6.666667	26	7
Illinois	7	8	5	6.666667	27	7
Iowa	8	8	5	7	28	7
Rutgers	3	8	11	7.333333	29	8
Oklahoma	10	7	6	7.666667	30	8
Wichita St.	5	7	11	7.666667	31	8
Saint Mary's	11	7	6	8	32	8
USC	5	8	11	8	33	9
Colorado	7	8	11	8.666667	34	9
LSU	8	8	11	9	35	9
Providence	6	10	11	9	36	9
Cincinnati	9	8	11	9.333333	37	10
Utah St.	7	11	11	9.666667	38	10
Richmond	11	7	11	9.666667	39	10
Texas Tech	9	9	13	10.33333	40	10
Marquette	11	10	11	10.66667	41	11
Indiana	10	11	11	10.66667	42	11
Florida	11	11	11	11	43	11
Purdue	10	10	16	12	1	11
Georgia Tech	5	16	16	12.33333	2	11
Washington	5	16	16	12.33333	3	11
Texas	10	16	11	12.33333	4	12
Saint Louis	9	16	13	12.66667	5	12
Northern Iowa	11	16	11	12.66667	6	12
East Tennessee St.	12	16	11	13	7	12
Liberty	11	16	14	13.66667	8	13
Tennessee	9	16	16	13.66667	9	13

TEAM	KNN	SVM	ANN	means	Rank	Seed
Arkansas	11	16	15	14	10	13
Oklahoma St.	10	16	16	14	11	13
St. John's	10	16	16	14	12	14
VCU	10	16	16	14	13	14
San Francisco	11	16	16	14.33333	14	14
Davidson	11	16	16	14.33333	15	14
Arizona St.	16	16	11	14.33333	16	15
Rhode Island	16	16	11	14.33333	17	15
Stephen F. Austin	16	16	11	14.33333	18	15
SMU	12	16	16	14.66667	19	15
Northern Colorado	13	16	16	15	20	16
Memphis	16	16	13	15	21	16
Yale	16	16	13	15	22	16
South Carolina	14	16	16	15.33333	23	16
Iowa St.	15	16	16	15.66667	24	16
Louisiana Tech	16	16	15	15.66667	25	16

Figure 15: Rankings and Seeds of Simulated 2020 Tournament Teams

Tournament Simulation

In order to simulate the tournament with the above list of teams, a method to determine a victor for any given contest must first be derived. Using the "TEAM", "Team1", and "Team2" columns from the datasets shown in Figures 9 and 10 as primary keys, the following process was carried out:

- For all games from 2013-2019, compute the differences of all quantitative statistics (except 'SEED' and other categorical variables) between the two teams that competed in each contest (Figure 16, below).

- Create a regression model that predicts the difference in score (along with its sign, as it indicates which team emerged victorious).

- For teams in Figure 15, create and simulate games according to whether or not the regression model returned a positive or negative value (the magnitude represents a quasi-"betting line", how many points one team is expected to win by). Repeat this step for each region of the tournament and record the victor(s).

The regression model initially contained all of the aforementioned differences, and was minimized using step-wise regression under the *BIC* criterion. Ultimately, the following regression equation was used to simulate the 2020 Tournament: *ScoreDiff ~ ADJOE_DIFF + ADJDE_DIFF + TORD_DIFF + WAB_DIFF*.

Figure 7 includes sample differences from the games depicted in Figure 1 (the figures are snapshots of the same dataset).

ADJOE_DIFF	ADJDE_DIFF	BARTHAG_DIFF	EFG_O_DIFF	EFG_D_DIFF
25.3	-11.2	0.628	4.9	-0.4
17.2	-7.2	0.378	4.3	-2.8
3.6	-10.1	0.188	-5.1	-1.7
5.5	-7.8	0.219	1.1	0.1
5.5	-7.8	0.219	1.1	0.1

Figure 16: Differenced Statistics for Games from Figure 1 (most columns not depicted due to cutoff).

The difference statistics were created and passed to the regression model for each simulated game. Figure 17 shows the results of simulating each round of the tournament, including the victor and the difference in score (negative values indicate an "upset" victory).

ID	DIFF	TEAM	ROUND
Washington-Louisiana Tech	4.1	Washington	First Four
Georgia Tech-Yale	5.2	Georgia Tech	First Four
Kansas-Northern Colorado	12.2	Kansas	East - Round of 64
San Diego St.-Arizona St.	6.5	San Diego St.	East - Round of 64
Villanova-St. John's	1.6	Villanova	East - Round of 64
Oregon-Liberty	8.9	Oregon	East - Round of 64
Penn St.-Texas	8	Penn St.	East - Round of 64
Seton Hall-Marquette	2.6	Seton Hall	East - Round of 64
Wisconsin-Cincinnati	2.8	Wisconsin	East - Round of 64
Rutgers-USC	6.5	Rutgers	East - Round of 64
Kansas-Rutgers	7.1	Kansas	East - Round of 32
San Diego St.-Wisconsin	3.4	San Diego St.	East - Round of 32
Villanova-Seton Hall	0.6	Villanova	East - Round of 32
Oregon-Penn St.	1.8	Oregon	East - Round of 32
Kansas-Oregon	4.7	Kansas	East - Sweet 16
San Diego St.-Villanova	3.5	San Diego St.	East - Sweet 16
Kansas-San Diego St.	4.9	Kansas	East - Elite 8
Dayton-Iowa St.	10	Dayton	West - Round of 64
Creighton-SMU	9.7	Creighton	West - Round of 64
Louisville-Davidson	4.8	Louisville	West - Round of 64
Ohio St.-Oklahoma St.	7.5	Ohio St.	West - Round of 64
Butler-East Tennessee St.	6.8	Butler	West - Round of 64
BYU-Purdue	0.6	BYU	West - Round of 64
Iowa-Texas Tech	-3.2	Texas Tech	West - Round of 64
Saint Mary's-Providence	-3.4	Providence	West - Round of 64
Dayton-Providence	5	Dayton	West - Round of 32
Creighton-Texas Tech	1.6	Creighton	West - Round of 32
Louisville-BYU	1.2	Louisville	West - Round of 32
Ohio St.-Butler	5.7	Ohio St.	West - Round of 32
Dayton-Ohio St.	3.7	Dayton	West - Sweet 16
Creighton-Louisville	3.4	Creighton	West - Sweet 16
Dayton-Creighton	3	Dayton	West - Elite 8
Baylor-South Carolina	9.2	Baylor	Midwest - Round of 64
Duke-Stephen F. Austin	13.1	Duke	Midwest - Round of 64
Michigan St.-San Francisco	7.7	Michigan St.	Midwest - Round of 64
Virginia-Arkansas	-2	Arkansas	Midwest - Round of 64
West Virginia-Northern Iowa	8.3	West Virginia	Midwest - Round of 64
Arizona-Florida	4.1	Arizona	Midwest - Round of 64
Illinois-Richmond	2.4	Illinois	Midwest - Round of 64
Wichita St.-LSU	0.4	Wichita St.	Midwest - Round of 64
Baylor-Wichita St.	9.1	Baylor	Midwest - Round of 32
Duke-Illinois	7.5	Duke	Midwest - Round of 32
Michigan St.-Arizona	0.7	Michigan St.	Midwest - Round of 32
Arkansas-West Virginia	-1.4	West Virginia	Midwest - Round of 32
Baylor-West Virginia	5.4	Baylor	Midwest - Sweet 16
Duke-Michigan St.	2.8	Duke	Midwest - Sweet 16
Baylor-Duke	2.5	Baylor	Midwest - Elite 8
Gonzaga-Memphis	12.3	Gonzaga	South - Round of 64

ID	DIFF	TEAM	ROUND
Florida St.-Rhode Island	6.5	Florida St.	South - Round of 64
Maryland-VCU	-0.5	VCU	South - Round of 64
Kentucky-Tennessee	1.7	Kentucky	South - Round of 64
Auburn-Saint Louis	0.4	Auburn	South - Round of 64
Michigan-Indiana	5.2	Michigan	South - Round of 64
Houston-Utah St.	5.5	Houston	South - Round of 64
Oklahoma-Colorado	-0.3	Colorado	South - Round of 64
Gonzaga-Colorado	9.1	Gonzaga	South - Round of 32
Florida St.-Houston	-0.7	Houston	South - Round of 32
VCU-Michigan	-0.9	Michigan	South - Round of 32
Kentucky-Auburn	3.1	Kentucky	South - Round of 32
Gonzaga-Kentucky	9.5	Gonzaga	South - Sweet 16
Houston-Michigan	2	Houston	South - Sweet 16
Gonzaga-Houston	5.5	Gonzaga	South - Elite 8
Kansas-Dayton	0.6	Kansas	Final Four (East vs. West)
Gonzaga-Baylor	1.6	Gonzaga	Final Four (South vs. Midwest)
Kansas-Gonzaga	0.4	Kansas	Championship Game

Figure 17: Results of Simulating 2020 Tournament

Of particular interest are the games in which the regression model predicted an upset victory; there are a number of them occurring in the First Round (the "Round of 64"), one of emphasis being No. 10 Texas Tech's victory over No. 7 Iowa in the West Region. Over time, however, *higher seeds* were predicted to prevail, with all four one-seeds reaching the Final Four games and Kansas emerging as the Champion over Gonzaga (another astonishing runner-up finish for Gonzaga!). The authors would like to acknowledge this conclusion as an interesting one, considering that Gonzaga has reached the title game twice in the last 4 tournaments, losing most recently in 2021 to Baylor, and four years prior to North Carolina [in the Tar Heels' "Redemption Season", as many newspapers soon nicknamed it].

Conclusion

To recap, our group was focused on finding out what makes a championship basketball team – statistically! Our first question of interest was: what variables contribute most significantly to a high win rate, specially focusing on the adjusted offense and defensive efficiency (ADJOE & ADJDE). While our second question (or goal) of interest was to attempt to use the data we had to simulate the 2020 college basketball playoffs.

To summarize our findings, for our first question, we found that ADJOE & ADJDE has a statistically significant interaction, and statistically significant by themselves; the methodology we used to determine these results were ANOVA & Lasso. For our second question, we found out the we could indeed simulate the 2020 college basketball playoff field, but not very effectively with our current statistical skill-set; while our postseason berth models had high accuracy (e.g., the SVM had *~92%*, the ANN recorded *~91.5%*), seed classification models did not (e.g., an ANN classifier predicted *~30%* of the seeds of a test dataset correctly), the seed classifiers rarely agreed unanimously on seeding for most teams (outside of 1-seeds such as Gonzaga), which led to the use of average seeds to ultimately set the field. The methodological procedures included in simulating this tournament were SVMs, k-Means Clustering (i.e. K-Nearest Neighbors), and Artificial Neural Networks.

There does exist some room for improvement with respect to data collection, our group found it very difficult to find data on the strength of schedule for most teams (this was accomplished, however scraping and cleaning nine different datasets was an incredibly tedious process); this was not the only problem we ran into...a lack of computing power did not help when trying to run our neural network (which can simply be solved in

the future with the use of a more powerful GPU). This work could be expanded on by others in the future with a deeper analysis into two other important variables that contribute to a high win rate: TOR (turnover percentage allowed) and ORB (offensive rebound rate); these two variables exhibited similar behavior to ADJOE & ADJDE, as they were also significant by themselves and when interacting with each other, but they did not have significant interactions with ADJDE or ADJOE!

Our group thought of three real world implications for our analysis; those being: sports betting (i.e. winning the office bracket competition), team development / improvement, and a potential new ranking methodology. An obvious extension of this analysis into the real world is for sports betting; with rigorous enough models, one could predict the outcome of games, and therefore capitalize on a betting opportunity, which many do! With respect for team development and improvement, statistical analysis of a team can uncover what a team is doing well and what they are struggling with. Once a team figures out what they are struggling with they can focus on that variable to potentially improve their overall win rate. And finally, with regards to ranking methodology, a new model can be created from this type of statistical analysis to develop a more accurate and precise ranking measure, that is based on objective statistics and not subjective committees!