Kevin O’Donnell

Dr. Lupinacci

STAT 5905-002

4 September 2022

# Predicting the NBA MVP

## Introduction

As an aspiring basketball analyst, I decided to structure my project around the NBA. I have always been curious how sportsbooks create the odds for certain events, like who will win the Most Valuable Player (MVP) award and would like to see if I too can reliably predict the winner of that award. What makes this goal particularly engaging and challenging is the lack of certainty surrounding the award’s definition. The NBA has no formal definition for the award or qualifications that the candidates must meet to earn votes. A group of panelists, made up of sportscasters and reporters from the United States and Canada, meet and give votes to the players they believe deserve MVP. They rank their top five candidates, with the first-place vote counting for 10 points, the second-place vote counting for seven, the third counting for five, the fourth counting for three, and the fifth counting for one. The winner is the player with the most total points from voting (Corvo). This is also the player with the highest award share. Award share is defined as the number of points a player receives from voting divided by the total possible points from first-place votes (Yoo). Although the voting system has been standardized since the 1980-81 season, the criteria for these votes are far from defined. The panelists crown the winner at their discretion.

Each year, there is considerable debate about who should win the award and what winning means. Some contend that it should be given to the best player in the league. Some argue that it should be the most valuable to their team. Others think it should be the best player on a great team. Still others support a heavily qualitative approach that gives the award to a great player with the best narrative, like leading a weak team to the playoffs or returning from injury. Within each of these arguments, a multitude of additional questions arise. How does one define the best player? What statistics would you use? Are there certain individual statistics that matter more than others? Is there a win threshold that a team must meet for its players to be considered? Questions like these abound in the basketball world, but I would like to see if I can surmount this uncertainty and find a combination of variables that accurately predicts the MVP.

## Research Questions

The primary research question driving my project is: what factors are the best predictors of who will win the most valuable player award in a given year? I am also interested in accurately predicting next year’s NBA MVP. Therefore, the secondary research question guiding my analysis is: which model provides the most accurate MVP predictions? My initial goal is to accurately predict a season’s MVP at the beginning of the season before any games are played, so I begin by utilizing statistics from the previous year to predict MVP candidacy. However, I will also explore predicting the MVP with current player statistics to make more reliable predictions.

## Data Sources

My initial hypothesis is that advanced individual statistics, like win shares (WS), usage rate (USG%), player efficiency rating (PER), and value over replacement (VORP), will be the strongest predictors. Basic individual statistics, like points (PTS), true shooting percentage (TS%), and minutes (MIN), will be important, but of less value than the advanced statistics. To test this hypothesis and determine which variables most strongly affect the MVP award winner, I have gathered a variety of player and team statistics. I downloaded the players’ physical attributes from a dataset on Kaggle. Omri Goldstein uploaded [this dataset](https://www.kaggle.com/datasets/drgilermo/nba-players-stats?resource=download) with a variety of player statistics, of which I am only using player height and weight. I also gathered an MVP voting dataset from Danchyy on Kaggle. [This dataset](https://www.kaggle.com/datasets/danchyy/nba-mvp-votings-through-history) includes every player that has received MVP votes from 1947-2018. It includes each player’s award share, total points from voting, total points from first-place votes, and the maximum points from first-place votes in each year. I will utilize only the records from the 1980-81 season onward. I pulled the remainder of the data from the nbastatR package in R, which scrapes data from [basketball-reference.com](http://basketball-reference.com/). This data includes the MVP of each season, basic per-game statistics (on the team and individual levels), advanced individual statistics, draft position, age, and position data. It ranges from the 1954-55 season to the most recent NBA season: 2021-22.

Since I pulled voting data and the MVP of each season, I have multiple potential response variables that could allow me to predict the award winner each year. The basic MVP data is a single binary response column that is 0 when a player did not win the award and 1 when they did. There are 65 MVPs and 23,451 non-MVPs in the dataset. This will present challenges in my classification model because it is a very imbalanced data source. The model, therefore, is likely to misclassify winners as losers because predicting exclusively losers would generate an overall “accurate” model. However, this model would be useless for my purposes. I address these problems and their solutions in more detail in the [Model Building](#_Model_Building) section.

I will also consider continuous response variables. The primary continuous alternative to the binary MVP response is award shares, which I will model using a variety of regression models, starting with multiple linear regression. This will help balance the dataset and improve the reliability of my predictions. I explain the challenges still present after these changes later.

After gathering the response variable, I retrieved my explanatory variables. The individual basic statistics dataset includes identifier variables, namely player names and seasons, which I used to merge with the remaining data. It also includes 26 quantitative variables that describe player performance on a standard, per-game level. Examples of these common box score statistics are points, assists, rebounds (offensive and defensive), turnovers, minutes, effective field goal percentage, free throw percentage, and total games started. Note that true shooting percentage, which is an alternate measure of shooting efficiency that includes three-pointers, two-pointers, and free throws with appropriate weights, is included in the “advanced statistics” data table initially, but I will treat it as a basic statistic in my analyses. Since it is similar to effective field goal percentage and is a standard measure, I will treat it as such. Many of these typical statistics should help predict the MVP for a given season since they are easily interpretable and display a player’s impact on different facets of the game. They are also trusted metrics with considerable longevity, unlike some advanced statistics that emerged in the past decade. For these reasons, voters should typically consider them in their determination of the MVP.

In addition to standard per-game statistics, I am considering advanced metrics in my model and initial visualizations. I will explain these variables briefly below, but for more detailed descriptions and formulas, see [Appendix B](#_Appendix_B_–). This data table houses the same identifier variable as above and contains 20 advanced measures. All of these measures are quantitative and have a variety of scales and units. Three-point rate, free throw rate, offensive rebound percentage, defensive rebound percentage, total rebound percentage, assist percentage, steal percentage, block percentage, turnover percentage, and usage rate are all measured like a traditional percentage: between 0 and 100 percent. Statisticians measure win shares—total, offensive, defensive, and win shares per 48 minutes—in “wins” because they are the estimated number of wins a player contributes on offense, defense, or overall. Similarly, offensive, defensive, and box plus-minus statistics are measured according to the estimated points contributed by an individual player. Player Efficiency Rating is a standardized measure developed by John Hollinger, Senior NBA columnist at The Athletic, of per-minute production centered around a league average of 15. This statistic attempts to adjust performance to a per-minute measure so players with different average minutes per game can be compared. It also accounts for a team’s pace, so it does not penalize players on slow-playing teams compared to players on teams with more possessions per game. Finally, Value Over Replacement Player is the points per 100 possessions that a player contributes above what a typical replacement-level player in the NBA would contribute. Many of these advanced metrics are designed to encapsulate a player’s overall performance or to identify their impact on a game. As a result, I would expect these measurements to have a strong relationship with winning the MVP.

In addition to in-game performance measurements, I also will include demographic descriptors in my analysis. These include draft position, age, and position. Of these, I anticipate that age will have the most noticeable effect on winning the award. As seen in Graph 1 from [Appendix A,](#_Appendix_A_–) most MVPs are between 26 and 28 and there is a considerable drop-off in the number of MVPs once players turn 31. There are only 8 MVPs who have won the award ages 31 or above since 1955 and none since Steve Nash won the award in 2006. It appears that younger players are more likely to win the MVP than older players, so if a player is reaching the end of their career, their chances at MVP are probably much lower than they were at the beginning of their career.

Draft position also plays a role in MVP voting. Since the advent of the award, Moses Malone is the only undrafted player to win MVP. Aside from Malone, Nikola Jokic is the lowest draft pick to win MVP at pick 41. All other MVPs were drafted 15th or earlier. From these results, I would expect a player’s draft pick to weigh heavily on their odds of being voted the most valuable player in the NBA.

Finally, a player’s position could play a role as well and may have different effects based on the era we are considering. As seen in Graph 2 from [Appendix A,](#_Appendix_A_–) centers have the most trophies by a large margin–27 of 65–but tallied most of those trophies in the early years of the NBA. They dominated the 1960s and 1970s, but, until Nikola Jokic in the 2020-21 season, a center had not won MVP since Shaquille O’Neal in the 1999-2000 season. In the 2010s, point guards and small forwards won the award most often, suggesting a possible shift in preference from frontcourt to backcourt and hybrid players. As a result, position could inform the model of a player’s likelihood to earn MVP votes.

Since the award may not be purely individual, I have included a subset of basic team statistics to account for the team’s potential role in MVP voting. The team statistics dataset contains seven statistics as well as identifying variables to merge with the existing repository of individual metrics. The identifier variables are season and team. The statistics included are winning percentage, points per game, assists per game, rebounds per game, turnovers per game, field goal percentage, and free throw percentage. I selected these metrics because they collectively describe basketball analytics pioneer Dean Oliver’s four factors of basketball success: shooting, turnovers, rebounding, and free throws. According to Oliver, these factors are the most important predictors of team wins (“Glossary”). I am including these statistics, therefore, to measure a team’s overall success and determine if that affects a player on that team winning MVP. I hypothesize that individual factors will be more influential than team factors since the response is an individual accolade, but team statistics could improve the accuracy of the model.

Now that I have gathered and joined the data, I will move forward with further exploratory data analysis. I will also pay attention to redundant variables that would lead to singularities in my final model. An example of these data is field goals attempted, field goals made, and field goal percentage. I only need two of these three variables to fully describe a player’s shooting, since field goal percentage is simply the division of the former two variables. Including all of these variables in my final model is unnecessary and could prevent the model from converging. I explain how I address this in the [Data Visualization](#_Data_Visualization) and [Model Building](#_Model_Building) sections.

To address the class imbalance, I am considering award share as a response variable. Considering an average of 16.76 players receive a vote each year in a league of approximately 450 players, the class imbalance will improve, but the ratio of majority to minority class observations is still suboptimal for least squares regression. Many of the observations will be 0 and the response will be heavily right-skewed. I will consider alternatives to least squares regression that may handle imbalanced data better.

Beyond balance concerns, my data filtering will also include removing all observations prior to 1980. Before the 1980-81 season, players decided the MVP votes. Since this system was different from the current system of voting by broadcasters, data from early in the award’s history may not be representative of current data. Therefore, I will filter the data to include only the 1980-81 season and beyond. With the filtered data, I will decide whether to model a binary or continuous response and choose the appropriate regression techniques.

## Data Cleaning

Before performing statistical tests and building predictive models, I joined and cleaned my data sources and continued exploratory data analysis. After joining the data sets explained above, my full player statistics data set contained 70 variables. This data set contained all desired predictors and response variables for the seasons between 1954-55 and 2021-22. Given the NBA changed the award’s format in 1980, I filtered out all observations before the 1980-81 season. Additionally, because the award share data was only available through 2018, I could not use data past the 2017-18 season. This limited range reduced the number of observations in my data but improved the ratio of complete to incomplete cases since many advanced statistics were not available for players in the early years of the NBA.

Following this preliminary filtering, I elected to “lag” the player statistics to align the data with my goals. Since my primary objective is to predict the MVP of a season before it begins, I will use each player’s previous season’s data to predict their likelihood of winning the award. Of course, all static values, like height, weight, draft pick, and position, will not be affected. I lagged all of the player statistics that varied from year to year, meaning each row of data would contain a player, the year, their static attributes, and their statistics from the prior year. Using the previous year’s statistics ensures that the final model will use information that the end user of the model could acquire before the season. With that being said, I also created an identical dataset with current statistics that align a player’s statistics with the year of their award share. This data is more representative of a player’s performance in a given year, so the models on this data will perform better, as seen in the Model Building section.

After filtering for the correct years and imposing information availability restrictions on my data, I addressed data inaccuracies. Upon examining the dataset for anomalies, I noticed 81 rows that resulted from duplicated name and season combinations. Some players had multiple observations for the same year. The root cause of this issue was duplicate draft records. Some players were drafted more than once because they did not join the league upon their first selection due to extenuating circumstances. Therefore, the draft data recorded their name more than once and each duplicated player had multiple values for their draft position. When the draft data was joined with the complete player statistics dataset, the result was a record for each player, season, and draft pick when the desired result was a unique record for each player and season only. For example, the Atlanta Hawks drafted Arvydas Sabonis 77th overall in 1985, but the league discovered he was under 21 years old, the minimum age to be drafted at that time, so the pick was voided. He was drafted 24th overall by the Portland Trail Blazers the following year (Winn). As a result, Sabonis had two observations for every year he played in the NBA which were identical in every way except for draft pick. To resolve this issue, I downloaded a [draft history dataset](https://www.kaggle.com/datasets/hrfang1995/nba-drafts-of-19472018) from user hrfang on Kaggle which included draft data from 1947-2018 and filtered for the duplicated players in the prior draft data. I then grouped the data by each player and selected the rows with the latest year to correspond to each player’s official pick. I merged this dataset with the original draft data and replaced the existing draft pick variable with the correct value for the offending observations. Then, I merged the corrected draft data with the full player statistics data. This removed all duplicated player and season combinations.

## Exploratory Data Analysis

With the cleaned and joined data, I began exploring relationships between predictor variables and potential response variables. The potential response variables I considered are award share (award\_share), which is continuous, and whether a player won the MVP or not (isMVP), which is binary.

### Correlations Between Predictors and Award Share

First, I examined the correlations between all 50 numeric predictors and award share. The table of correlations can be found in [Appendix C – Exploratory Data Analysis Tables](#_Appendix_C_–). This analysis gave me insight into which variables are most and least related to the proportion of points from first-place votes a player earns. Value Over Replacement Player (ratioVORP) has the strongest positive correlation with award share of any numerical predictor with a value of 0.44942. This is followed by the total number of MVP awards a player has going into the season (totalMVP), Offensive Win Shares (ratioOWS), Win Shares (ratioWS), and free throws attempted per game (ftaPerGame).

As I predicted, advanced statistics seem to be stronger predictors than basic statistics because many advanced statistics appear in the top five strongest correlations. Value Over Replacement Player, Offensive Win Shares, and Win Shares are all advanced statistics that attempt to encapsulate a player’s overall impact on the game and are moderately positively correlated with award share. This means that as award share increases in value, these three statistics tend to increase as well. Of the 50 numeric variables, 42 have positive correlations, but many are close to zero and represent weak positive relationships between the predictors and award share. Since the three listed statistics have moderate positive correlations, I expect them to be significant predictors in regression models for award share.

Their strong correlations relative to other advanced statistics also give insight into which advanced statistics are better than others for predicting the award winner. Many of the advanced percentages, that attempt to quantify the percentage of available statistics a player records while on the court, have weak relationships with winning MVP votes. For example, offensive rebound percentage (pctORB), block percentage (pctBLK), turnover percentage (pctTOV), and steal percentage (pctSTL) have weak, negative relationships close to zero. These advanced statistics, therefore, may not be useful at all in predicting the award winner, while VORP, OWS, and WS may be significantly more useful.

Some basic statistics have strong positive correlations as well. As aforementioned, free throws attempted has the fifth highest correlation value. The sixth and seventh highest values come from free throws made (ftmPerGame) and points per game (ptsPerGame). It is unsurprising that points per game increase with award share, because, since 1980, MVPs have averaged 26.95 points per game while non-MVPs have averaged just 8.41 points during the same time frame. Also, points are a very direct measure of success because scoring more points than the opponent means winning the game. It would make sense, then, for points to be at least moderately associated with votes received because points scored are fundamental and easily available as a statistic.

The overwhelming presence of offensive statistics at the top of this list suggests there may be a preference for offensive performance over defensive excellence. Defensive Win Shares (ratioDWS) have the 8th highest correlation with award share but are the first defensive statistic to appear on the list. Remember that Offensive Win Shares rank third on this list and overall Win Shares rank fourth. Since DWS and OWS are the same measures for different sides of the ball, it seems that offense seems to be more closely associated with winning MVP votes than defense.

Additionally, free throws attempted and made have higher correlations than the comprehensive defense measure. The prominence of these two measures is surprising considering Chicago gleans the highest proportion of points from free throws per game at just 20.8% this season. For comparison, the highest proportion of points from two-point shots comes from Atlanta at 59.2%. For three-point shots, Indiana earns the highest proportion with 38.9% (“NBA Team Percent of Points from Free Throws”). Therefore, I would expect two and three-pointers to weigh more heavily on players’ MVP votes because they account for a higher proportion of points.

It is possible that free throws have a stronger positive relationship because the number of attempts indicates how aggressively a player is defended, which in turn could indicate the perceived skill of the offensive player. MVP-caliber players may get fouled more often in the act of shooting to prevent them from converting shots with higher points per make. This could explain the relatively high correlation between award share and free throws attempted. Once they get to the line, it is important for them to take advantage of the opportunity for an uncontested shot. Those who do are higher efficiency free throw shooters, earn more points per game from the line than inefficient shooters, and therefore, may be considered for MVP more than their inefficient counterparts or those who make fewer visits to the charity stripe. Of course, this is speculative, but these high values nonetheless indicate that free throws may be an important factor in predictive models.

Finally, with regard to correlations, usage rate (pctUSG) has a weaker relationship with award share than expected. Considering MVPs average a usage rate of 30.49%, compared to 18.85% for non-MVPs, I would expect usage rate to be higher in this table. Typically, teams give their best players the ball most often. Also, a high usage rate allows for strong raw production, in terms of points, assists, and other offensive measures, so it should be a strong predictor for MVP voting. Its weak relationship with award share is surprising and will be monitored in the model-building phase.

### Relationships Between Predictors and Winning the MVP

In addition to examining correlations between predictors and the continuous response, award shares, I also built a series of simple logistic models to determine the relationship between each predictor and the binary response, winning MVP. Each of these models used one explanatory variable to predict whether a player would win MVP or not. I gathered the p-values for the explanatory variable in each model and arranged them in ascending order in Table 2 of [Appendix C – Exploratory Data Analysis Tables](#_Appendix_C_–). Using an alpha level of 0.05, I determined if each predictor variable was marginally significant. All p-values were below 0.05 and, therefore, all variables were marginally significant predictors of winning MVP except for a player’s team (slugTeamBREF), age (agePlayer), offensive rebounding percentage (pctORB), three-point field goal percentage (pctFG3), specific position (slugPosition), teamwide turnovers per game (tovPerGameTeam), and teamwide free throw percentage (pctFTPerGameTeam).

Considering that the award is individual, and that team success varies with time, it makes sense that the team a player is on is insignificant. Specific teams go through irregular periods of success and failure, so the name of the team a player is on will not inform their chances of winning the award. However, some team statistics, specifically points per game (ptsPerGameTeam), assists per game (astPerGameTeam), and winning percentage (pctWins), may help predict the winner because they are marginally significant. These team statistics give more pertinent and complete information about team success than the name of the team, so they are significant. Additionally, because the team variable has over 30 different values, which would complicate and slow down model creation, it will be left out of any final statistical models.

Similarly, a player’s specific position (i.e., point guard, shooting guard, small forward) has little effect on if a player wins MVP or not. This may be surprising considering shooting guards dominated the 1990s and power forwards dominated the 2000s. However, when we consider that all four wins from shooting guards in the 1990s belong to Michael Jordan and two of four wins from power forwards in the 2000s belong to Tim Duncan, it seems that position has less to do with winning the MVP than player skill, which is explained by a combination of the other variables in the data.

The insignificant numeric variables—age, offensive rebounding percentage, and three-point field goal percentage—also have very weak correlations with award share and likely will not be strong predictors of the binary or continuous response. From a marginal perspective, these variables are not closely tied to winning the MVP.

Since winning the MVP is a function of award share, many of the variables with relatively strong positive correlations with award share also have relatively small p-values. Variables like VORP, OWS, WS, total MVPs, free throws attempted, and free throws made all have relatively strong correlations and small p-values, so they are marginally significant predictors of who will win the MVP award.

### Comparing Statistics for MVPs and non-MVPs

After determining which variables are significant on their own for predicting the most valuable player, I decided to determine which statistics are significantly different between MVPs and the rest of the league. I analyzed each individual statistic to determine where its mean is significantly higher for MVPs than non-MVPs from 1980 to 2018. Once again, I used the standard alpha equals 0.05 threshold to determine statistical significance. As expected, MVPs had significantly higher values for nearly every individual statistic. The results, shown in Table 3 in [Appendix C](#_Appendix_C_–), corroborate many of the observations made above in the analysis of the relationship between predictors and response variables; many of the marginally significant variables display significant differences between MVPs and non-MVPs.

However, there are some notable observations from this analysis. Primarily, the players’ attributes—height and weight—are not significantly different between the two groups. I anticipated that these measurements would differ between MVPs and the rest of the NBA. I hypothesized that being a taller and larger player would make it easier to score, given that taller players are naturally closer to the 10-foot hoop and heavier players should be able to displace defenders more easily than lighter players. I would expect these players to earn more MVP votes than smaller players because they should be able to score at a higher rate. However, it appears that the average height and weight are not actually different between MVPs and non-MVPs.

Another interesting observation is that the averages for steal percentage (pctSTL) and offensive rebounding percentage (pctORB) are significantly different, but MVPs actually have lower percentages. Since having a higher steal and rebounding percentage is better, I would expect MVPs to have greater percentages than the rest of the league. Their subpar performance on the offensive glass and in the steal category may suggest that these variables are not actually influential. Regardless of the significant difference, I would not expect that having a low steal or ORB percentage would make a player more likely to win the MVP. This is counterintuitive because we would not predict a player with worse statistics to have a better chance of being named the best player in the league. Above, it was determined that both variables have very weak correlations with award share and offensive rebounding percentage is not marginally significant for predicting the MVP. Therefore, I expect that these variables are simply inconsequential and happen to have lower means for MVPs because they are not considered in voting.

### Data Visualization

Having explored relationships between predictors and response variables, it was necessary to examine how closely predictor variables were tied to each other. I created color-coded correlation matrices to visualize the correlations between all numeric variables in the dataset. Due to the large number of predictors, I split the visualization into three charts. These plots are located in [Appendix A - Data Visualization](#_Appendix_A_–_1).

Chart

Description automatically generated

The presence of possible multicollinearity in future model building is apparent, given the number of strong correlations between explanatory variables. Across all plots, one trend I observed was that advanced statistics tend to have high correlations with one another. For example, in the first plot, it is apparent that Player Efficiency Rating (PER) is closely related to Offensive Box Plus Minus (OBPM) and all measures of win shares. This trend is to be expected since, as aforementioned, advanced statistics are typically attempting to capture holistic player performance. Therefore, their formulas or the models that drive their predictions take similar information into account, including a variety of basic statistics included in this dataset. This is a point of concern for model building and I will address how the collinearity of advanced statistics, and multicollinearity in general, affects model interpretation in that section.

Another glaring example of a high correlation between predictors is visualized in the second plot, shown above for reference. This plot displays a variety of variables that either record that same facet of the game (i.e. shooting percentage) or combine to perfectly predict another. An example of the first case is the correlation between field goal percentage (pctFG in the plot) and effective field goal percentage (pctEFG in the plot). Effective field goal percentage is an extension of field goal percentage that weighs 3-point shots 1.5 times as heavily as 2-pointers due to their higher point value. Therefore, there are only slight differences between these two variables.

An example of the second case is the grouping of 2-point field goals made, 3-point field goals made, and total field goals made. These variables have strong correlations with each other because the first two combine to form the third. This occurs for the following groups of statistics:

1. ORB%, DRB%, TRB%
2. OBPM, DBPM, BPM
3. FT%, FTM, FTA
4. FG%, FGM, FGA
5. FG2A, FG3A, FGA
6. FG2M, FG3M, FGM
7. ORB, DRB, TRB
8. WS, OWS, DWS

To resolve this, I will ensure that these combinations do not appear in full in any of the following models. If they do enter during variable selection, I will remove the necessary variables. I will address a different solution to this issue for my final model build in [XGBoost for Award Share](#_XGBoost_for_Award).

In addition to these strong positive correlations, there is one noticeable negative relationship between block percentage (BLK%) and rebounding percentages.

Chart

Description automatically generated with medium confidence

BLK% has a strong negative correlation with total rebounding percentage (TRB%) and defensive rebounding percentage (DRB%). It is also slightly negatively related to offensive rebounding percentage (ORB%). These relationships are unintuitive because forwards and centers should have high rebounding percentages (because they are positioned by the hoop and are taller than guards) as well as high block percentages for the same reason. Similarly, guards should have lower values for both of these statistics. To investigate this strange relationship further, I plotted BLK% versus TRB%.

Chart

Description automatically generated

This plot shows that the unique distribution of BLK% is affecting the line of best fit and the correlation. Since so few blocks occur in a game—4.9 on average—if a player earns even one block, their percentage will be very high, compared to rebounding percentage, since there are 43.6 rebounds per game on average. This explains the stratification of BLK% values along the y-axis. The negative correlation is strong because, strangely, few players with high BLK% (above the large concentration of players near the x-axis) also have abnormally high TRB%. This pulls the line of best fit downward for high values of TRB%. However, it is difficult to determine a relationship between these two variables because there is such a high concentration of players with low BLK% and because blocks are infrequent enough to produce block percentages that are almost discrete in nature. Ultimately, since BLK% is not strongly correlated with award share or a strong marginal predictor for winning the MVP, I do not expect BLK% to appear in my linear or logistic models or to heavily impact my final XGBoost model.

After investigating variable correlations, I elected to visualize the variables with the strongest marginal relationships to winning the MVP in relation to one another and the MVP award to uncover further insights. I began by plotting each player’s award share and Value Over Replacement Player (VORP) from the 1980-81 season to the 2017-18 season.

**Chart, scatter chart

Description automatically generated**

The graph shows that, as VORP increases, award share generally increases as well. In general, having a higher VORP means a player is more likely to receive votes and win the award. This makes sense considering VORP is supposed to quantify how valuable a player is compared to a replacement level player, which is defined as a player with a box plus minus of -2 (see Appendix B for further details). However, there are some important observations to make and some players that deviate from this trend.

First, since only 4.1% of players in this dataset received MVP votes and have an award share greater than 0, most datapoints are concentrated along the x-axis. These players have a range of VORP values, but none greater than 7.4.

Additionally, certain players stand out in the dataset as potential outliers that defy the general trend. Three players—Chris Paul in 2008-09, LeBron James in 2004-05, and Kevin Garnett in 2004-05—have VORP readings above 9, which is better than 99.85% of the observations in this data. However, these 3 players received an award share below 20%. Given that many players with similar VORP values received much higher award shares, these appear to be outliers. Additionally, many players had significantly worse VORP values, but received large award shares. One of these players is Steve Nash in 2004-05, who outcompeted LeBron and Garnett in MVP voting despite having a VORP below 5. Nash won the MVP award in that year, which explains the surprisingly low award shares for LeBron and Garnett. Of course, since VORP is not a perfect predictor of award share, these instances are bound to occur. The 2004-05 season is one example of VORP not being a strong indicator of votes received.

Chris Paul in the 2008-09 season received fewer votes than expected for a different reason. His competition was Lebron James, a player with a historically high VORP of 11.8 in this season. LeBron’s performance was undoubtedly strong in this season, so Paul received comparatively fewer votes despite having a strong VORP statistic himself.

These examples illustrate that, while there is a moderate positive correlation between award share and VORP, using VORP alone to predict the MVP is insufficient. I expected VORP to be a significant predictor in models for award share and winning MVP, but I do acknowledge that it is not a comprehensive predictor and will have to be included alongside other independent variables in a multivariate model.

Chart, scatter chart

Description automatically generated

Given that offensive and defensive win shares were both highly correlated with award share compared to other variables and were significant marginal predictors of the award winner, I elected to explore their relationship to each other and the MVP award more closely. I was also eager to verify the apparent preference toward offensive win shares in MVP voting.

The graph above plots OWS against DWS and shows that they have a weak positive correlation with each other. There is a slightly positive trend but having a high DWS does not make a player more likely to have a high OWS, or vice versa. As expected, many of the MVPs have high values for at least one, if not both, of these metrics.

Notably, OWS seems to influence voting more since a greater proportion of OWS outliers than DWS outliers won the MVP. The highlighted points, which are outlined and bolded in the graph, represent the players with offensive and defensive win shares in the top 0.1% of all players in the dataset. There are 19 players in the top 0.1% for each statistic. Of the 19 with abnormally high OWS ratings, eight were MVP winners. Only two of 19 were winners on the defensive side. This supports my prior assertion that offensive statistics may weigh more heavily on MVP voting than defensive ones.

Chart, scatter chart

Description automatically generated

Finally, I investigated the relationships between free throws attempted, free throws made, and winning the MVP. Of course, FTM and FTA are strongly positively correlated as the more times a player goes to the free throw line, the more times they will score. The horizontal and vertical lines on the graph represent the 75th percentile for FTM and FTA, respectively. The slope of the diagonal reference line is the 75th percentile for free throw percentage (81.8%). All MVPs exceed the 75th percentile threshold for FTM and FTA, but a considerable number fall short of the free throw percentage cutoff. This suggests, consistent with correlation results from above, that volume of shots attempted and made matters more than the rate at which these shots are converted.

Earning foul shots is an important proxy for player skill and usage. Of course, the more a player is on the floor, the more often they can get to the line and make foul shots. As expected, minutes per game is strongly correlated with FTA (r=0.773) and FTM (r=0.775). Free throw frequency can also indicate, to some extent, how often a player is used for scoring opportunities since shot attempts have a strong correlation with free throw attempts (r=0.836) and free throws made (r=0.851). Furthermore, offensive players may be fouled more often if they are perceived to be an offensive threat by the defense and are tightly contested on shot attempts as a result.

Free throw percentage, on the other hand, is an efficiency metric without regard for the volume of scoring chances. These metrics tend to be suboptimal predictors on their own for the MVP because being considered for the award is heavily dependent on how often a player is used and has opportunities to positively impact the game. For example, a player could play 10 games, attempt 10 free throws in total, and convert nine of those attempts. This would be an extremely efficient clip, but with such a small sample size of games and shots, it is unlikely that this player is impactful enough to earn MVP votes. Efficiency could be an important consideration, but only given a certain volume of usage.

## Model Building

### Training and Test Set Creation

After exploring marginal relationships and running statistical tests to determine significant differences between MVPs and the rest of the NBA population, I moved into the model building phase. The data used was once again from the 1980-81 season through the 2017-18 season, because this included complete award share data under the modern MVP voting system. In total, 15,809 observations are included, 38 of which are MVPs. I designated observations between 1980-81 and 2007-08 to the training set and observations from 2008-09 to 2017-18 to the test set. This means approximately 70 percent of the observations are used for training the subsequent models and the remaining 30 percent is used for testing. Note that the split is not perfectly 70/30 because I elected to keep all observations from the same year together instead of sampling the data randomly for each set.

### Evaluation

Considering the class imbalance, some traditional methods for evaluating the following models are unreasonable. For example, logistic models are typically evaluated according to specificity, sensitivity, and misclassification rate. Since the following logistic models are trained on heavily imbalanced data, they will tend to predict an extremely large proportion of non-MVPs. This will result in a low misclassification rate, since most observations are not MVPs and the models will predict large numbers of those players. It will also result in a high specificity for this reason. However, despite being “accurate,” sensitivity will be low, meaning the model’s ability to predict the award winners using the typical probability cutoff of 0.5 will not be sufficient.

To rectify this and to be able to compare across logistic and linear models, which typically rely on different evaluation criteria, I created a new, standard metric to evaluate the models’ accuracy. This metric assumes that the model predicts that player with the highest predicted probability of winning the MVP or the highest predicted award share (depending on the response variable) will win the award. If the player with the highest predicted response is the true MVP of his season, the model has succeeded. If not, the model has failed. As such, the accuracy, as I define it, it the total number of successful predictions on the test set divided by the total number of MVPs in the test set, which is 10. All mentions of accuracy will refer to this equation unless otherwise specified.

### Models Using Prior Year’s Statistics

To begin the model building process, I created a linear model for award share that included all possible predictors. This model used statistics from the previous year to predict MVP for the upcoming season. I used backward selection by AIC (Akaike Information Criterion) to determine the best combination of explanatory variables in the model. I chose backward selection, instead of forward selection, to include more variables and, therefore, produce a higher R2 value and explain more variability in award share. This will help the model predict more accurately than a model with fewer variables built using forward selection. When I attempted forward selection, the resulting AIC was higher than the model generated from backward selection, so its predictability was worse. When I investigated the VIF values, both models possessed substantial multicollinearity, so I elected to use backward selection for stronger predictability.

AIC evaluates the maximum value of the likelihood function for the model with respect to the number of parameters. In other words, unlike R2, it penalizes models for having too many variables and would favor a model with fewer predictors over a model with more predictors if each had the same predictive power. Backward selection removes variables from the model until AIC is minimized, which is the desired result.

Backward selection resulted in a model for award share with the following explanatory variables:

* Age
* 3-point Rate
* FT Rate
* DRB%
* AST%
* USG%
* WS
* BPM
* VORP
* FT%
* Minutes per game
* FGM per game
* FTA per game
* DRB per game
* AST per game
* PTS per game
* Team PTS per game
* Team TOV per game
* Team FG%

Note that multicollinearity does exist because some explanatory variables are highly correlated with one another. For example, the correlation between minutes and field goals made per game is 0.74. However, I elected to leave all variables in the model because reducing multicollinearity to acceptable levels, where all variance inflation factors (VIFs) are below five, drastically reduced the predictability of the model on the training set according to AIC values. I chose to use this combination of variables for each subsequent model to allow for like comparisons in accuracy.

The model building stage of this project is focused primarily on optimizing the accuracy of my predictions. Due to the presence of multicollinearity, coefficients for each variable may be susceptible to small changes in the model and, therefore, it will be difficult to discern which variables are conditionally significant. Luckily, I have already addressed my primary research question by running simple logistic models to determine which variables are marginally significant predictors of winning the MVP and computing correlation values to reveal which variables are most closely linearly related to award share. Moving forward, models will be evaluated and chosen based on their accuracy, not their ability to reveal relationships with award share or winning the MVP award.

The linear model for award share performed with 50% accuracy on the test set, meaning it correctly predicted five out of 10 MVPs from the 2008-09 season to the 2017-18 season. The R2 value was .2407, meaning 24.07 percent of the variability in award share is explained by the predictor variables. The model’s results, along with results for all other models, can be found in [Appendix D – Model Results](#_Appendix_D_–_1). The model correctly predicted the MVP for the four seasons that LeBron James won the award because it predicted that James would win for the first six years in the test set. The model favored “the King” during this stretch and correctly predicted the winner for four out of those six years. It also correctly predicted that Russell Westbrook would win the MVP in 2017, when he averaged a triple-double on the season. Notice, however, that the model underpredicts the response variable. For example, while Lebron James earned an award share of 0.969 in 2009, the model predicted he would only win a share of 0.222.

The residual plots also display a lack of normality and a lack of fit for certain values, but this will be addressed in conjunction with the residual plots of other linear models in [Problems with Linear and Logistic Models](#_Problems_with_Logistic).

I then built a logistic model for the binary response, isMVP, using the same training set and predictors. This model’s accuracy was noticeably worse, at just 20 percent. Like the linear model, this model underpredicted the response. For example, the model correctly predicted that LeBron James had the highest chance to win MVP in 2009-10. However, the model only predicted the probability for James to take home the trophy was 15 percent. This suggests that the model’s confidence in its predictions is low, considering no players had a high enough win probability to be classified as a probable MVP under a typical cutoff of 50 percent. The logistic model was likely less accurate than its linear counterpart because of the data’s class imbalance. Only 0.23 percent of observations overall are MVPs, while 4.1 percent earned a positive award share.

### Models Using Current Year’s Statistics

Given the initial models did not exceed 50 percent accuracy, I decided to try the same models using current data. While a player’s previous year may be a predictor for the following season’s performance, there are many factors that can influence play, such as injuries and natural progression or regression. Using the same year’s statistics to predict the MVP is less enticing, because it prevents one from using the model to forecast the MVP at the beginning of the season. However, the models with up-to-date statistics delivered better results and is more logical than the models using lagged statistics, since the MVP award is tied to a player’s performance in that year, not the year prior. Additionally, with a large enough sample of games, these models could be applied as the season is ongoing to track the fluctuating likelihood of winning the MVP or fluctuating predicted award share as time passes, giving the end user insight into the strongest candidates at any given time.

Using the same training and test split, I constructed a linear model for award share. This model boasted an improved R2 of 0.2935, compared to .2407 in Model 1, and an improved accuracy on the test set of 70 percent. This model predicted the same correct winners as the initial linear model, but also correctly predicted that Kevin Durant would win in 2013-14 and that Stephen Curry would win in 2015-16. Again, the model underpredicts the actual values for award share, but its accuracy did improve.

The logistic model also improved with current data, improving its accuracy from 20 to 40 percent. This model delivered higher probabilities for LeBron James’ MVP wins in 2008-09 and 2009-10 than either correct prediction in the previous model. It makes sense that the model would improve its accuracy and its confidence in certain predictions while utilizing current data instead of prior information. This is still a considerably worse than the linear model for award share, however, which outperformed the logistic models on both the prior and current statistics. Therefore, moving forward, I will focus on predicting award share instead of the binary response.

### Basic versus Advanced Statistics

Finally, I created two more logistic models that included only basic statistics and only advanced statistics to compare their performance and judge which unit, as a whole, predicts the award winner with higher accuracy. The variables used are listed in [Appendix D](#_Appendix_D_–_1). Both models performed poorly, but the advanced statistics were stronger predictors as a group. The basic statistics model had 10 percent accuracy while the advanced model had 20 percent accuracy. These results, although neither model is strong, provide more evidence to support my hypothesis that advanced statistics are stronger predictors for winning the award because they are more comprehensive measures than basic statistics.

### Problems with Logistic and Linear Models

The above models for award share and winning the MVP award were promising steps toward testing my hypotheses, but they possess a variety of problems surrounding accuracy and validity. The test set accuracy for the logistic regression models in particular is suboptimal because they are correct less than half of the time. The linear models for award share are stronger, but they fail to meet some assumptions for linear regression. Notably, the residuals for each linear model are heteroscedastic, meaning they do not have constant variance at every level of the explanatory variables as is required. This can be seen in the “Residuals vs Fitted” plots in [Appendix D](#_Appendix_D_–_1). For small fitted values, there is little variation in the estimates, but there is increasing variance as fitted values increase. This creates a trend in the plot, which is undesirable given that a random distribution of residuals across fitted values is optimal.

Additionally, the residuals should be normally distributed, but this assumption does not hold for any of the above linear models. This can be seen in the Normal QQ plots. Ideally, the points should follow the diagonal line, but there is a considerable deviation from this line in both the model with current data and the model based on prior data. Specifically, the residuals fall farther from the expected values at larger theoretical quantiles than at smaller quantiles. This indicates that the response variable, award share, is right-skewed, which was already known since most players have an award share of zero. This leads to abnormal residuals.

Neither the logistic nor the linear models in their current states are optimal regression models. Therefore, I took the following next steps to produce more reliable predictions.

### Machine Learning Models - XGBoost

I investigated machine learning models to uncover a method that is more robust to imbalanced data. Since the logistic models performed worse than the linear models, I decided to focus my attention on modeling award share. Additionally, the linear models fail to meet necessary assumptions, so moving toward a different model to predict award share is necessary.

To improve my predictions and address the shortcomings of the prior models, I created an XGBoost model. XGBoost is short for eXtreme Gradient Boosting, which is an open-source machine learning algorithm developed by Tianqi Chen that builds upon decision tree models (Brownlee, “XGBoost for Regression”). Gradient boosting is a type of machine learning algorithm that can be used for classification or, in this case, regression. In short, it involves building decision tree models, adding them to the gradient-boosted fit, and minimizing error as more trees are added by optimizing a loss function, like squared error (Brownlee, “A Gentle Introduction to the Gradient Boosting Algorithm for Machine Learning”). As the model is built, the overall error is reduced, and the predictions become more accurate. Extreme gradient boosting is a variation of gradient boosting that runs quicker and is more computationally efficient (Brownlee, “XGBoost for Regression”).

Since its inception, models using XGBoost have dominated data science competitions because of their speed and accuracy (Brownlee, “XGBoost for Regression”). Additionally, XGBoost addresses many of the aforementioned issues with ordinary least squares regression. In this case, XGBoost utilizes regression trees to predict a continuous response, but the method of layering regression trees does not use linear regression and, therefore, many assumptions for linear regression do not apply. Instead, the algorithm performs a series of splits on the predictor variables and computes a continuous score on each terminal node. The splits are determined by minimizing the sum of squared residuals, or the difference between predicted and actual values. For each consecutive tree, the model minimizes a regularized objective, which is a combination of a loss function and a complexity function. Since this process does not use linear regression and instead uses a tree-based learning algorithm, the response variable does not have to be linearly related to the predictors and neither the response nor the residuals need to be normally distributed.

Furthermore, XGBoost handles imbalanced data much better than ordinary least squares models, usually producing accurate and precise results with severe class imbalance (Brownlee, “How to Configure XGBoost for Imbalanced Classification”). XGBoost can also be configured to respond better to class imbalance by tuning its class weight parameter (Brownlee, “How to Configure XGBoost for Imbalanced Classification”). By increasing this value, the algorithm will weigh the minority class (players without an award share) more than the majority class (those with an award share). For example, if the parameter is set to three, the algorithm will give classification errors made on the minority class three times more importance than errors made on the majority class. Therefore, it will correct these errors more. Parameter tuning is a considerable advantage of machine learning models over traditional statistical models in terms of predictability, which is the ultimate goal of this model.

Finally, XGBoost uses k-fold cross-validation to reduce the model’s bias and prevent overfitting the model to the training data. My XGBoost model for award share uses the standard 10 folds, meaning the data is split into 10 groups. One group is chosen as the holdout, or test set, and the other two are used to train the model. This is repeated until each group is used as the test set. Without k-fold cross-validation, the model may conform extremely well to the training set but fail to apply to unseen data, reducing its accuracy on the test set and future data (Brownlee, “A Gentle Introduction to K-Fold Cross-Validation”).

There are drawbacks to utilizing an XGBoost model to predict players’ award shares. A notable shortcoming of this model is the algorithm’s restriction on categorical variables. XGBoost only allows numeric predictor variables (Brownlee, “XGBoost for Regression”). However, this is not a major problem in this model, since the only possible categorical predictors are team, position, and grouped position. Of these, only grouped position is significant and, given that guards are typically smaller than forwards, height and weight could be used as a proxy for this variable.

Additionally, because it is a machine learning algorithm and builds layered decision trees, the model does not produce coefficients for each variable like linear regression does. Therefore, one cannot determine whether the impact of a variable is positive or negative (Brownlee, “XGBoost for Regression”). However, the model does produce an importance report, which indicates which variables were used most to make predictions. Using XGBoost instead of a linear model sacrifices some interpretability for improved predictability. However, given that many of the predictor variables in my dataset and the above linear models are collinear, the coefficients’ interpretability was already compromised. Additionally, as explained prior, the objective of this model is to accurately predict the player with the highest chance of winning the MVP. Multicollinearity does not dramatically affect predictability and neither does the ability to discern the exact conditional effect of each predictor. Therefore, I am progressing with XGBoost as the dominant model.

## XGBoost for Award Share

I constructed the XGBoost model for award share by including all relevant numeric variables as predictors for award share. To address the issue of grouped variables explained in the [Data Visualization](#__Data_Visualization) section, I have removed the variable within each grouping with the weakest correlation with award share. These are the removed variables:

1. ORB%,
2. DBPM
3. FT%
4. FG%
5. FG3A
6. FG3M
7. ORB
8. DWS

Instead of using the combination of variables used for the previous models, I included all relevant possible predictors because the algorithm, while iterating over multiple decision trees, determines which variables are the best predictors and scores them based on how valuable they were in constructing the trees. I did not want to omit potentially impactful variables from the model by using the same subset as above. Furthermore, because the algorithm selects variables for regression trees and not traditional linear regression, the variable selection process is different.

I used the same training and test sets as I did with linear regression but utilized k-fold cross-validation with 10 folds to prevent overfitting. I iterated through a variety of parameter combinations to tune the model and determine the best combination of rounds, learning rate (eta), maximum tree depth, minimum loss reduction (gamma), minimum child weight, and subsample ratio. The optimal set of parameters is listed in [Appendix D](#_Appendix_D_–_1) under “Model 7 – XGBoost for award share.” In addition to these generic parameters, I added the class weight parameter (scale\_pos\_weight) to address the class imbalance. I set this parameter equal to the ratio of majority to minority observations in the training set, which is 16.5497. Compared to a model run without this parameter, the model with this class weight parameter had a higher R2 value by .017 and a smaller RMSE by 0.0005.

The best model details are displayed in [Appendix D](#_Appendix_D_–_1), including the summary statistics, test accuracy, correct predictions, and importance plot. The importance plot displays the explanatory variables’ “gain,” which is the contribution of the variable to the model (Abu-Rmileh). If one variable has a higher gain than another, it means that variable is more important for predicting (Abu-Rmileh). The importance plot shows that Win Shares were favored most by the model by a considerable margin. The winning percentage of a player’s team and a player’s VORP also factored heavily. This plot provides some transparency into which variables were used to make splits in the decision trees constructed by the algorithm. However, since many variables are collinear, I cannot determine conditional significance from this plot. If two variables are highly correlated, the algorithm will choose one to split on based on the order it is put into the data. For example, since Win Shares is before VORP in the dataset and the variables have a correlation of 0.905, the algorithm will favor Win Shares (Abu-Rmileh). For prediction, this is largely insignificant, since splitting on either variable would likely lead to similar predictions, especially after iterating through many decision trees, but it confounds inference on these variables. This is similar to how p-value and coefficient estimates can be unstable for individual predictors in OLS regression if there is multicollinearity present. Once again, this does not concern the goal of predicting the MVP accurately but reinforces the inability to discern conditional variable significance from these models.

Concerning predictability, the best model after parameter tuning resulted in 100 percent accuracy on the test set, which is an improvement from 70 percent on the best linear model. In addition to a perfect performance on the test data, the XGBoost model had a substantially higher R2 value than the linear model. Its R2 value is 0.6823, compared to the linear model’s R2 of 0.2935. This is partially due to the number of variables included in each model. The XGBoost model includes nearly every numeric variable, so its R2 is necessarily higher. However, it is important to note that the R2 value for a linear model with the same variables is only 0.3234. Therefore, the XGBoost captures more of the variability in award share than the linear models could.

Finally, the XGBoost model improved the Root Mean Squared Error (RMSE) from 0.0513 to 0.0368. This means that not only did the machine learning model produce more accurate MVP predictions, but it also provided more accurate estimates for award share. Its estimates were closer to the actual award share than the linear model’s estimates on average. The reduced residuals are apparent in the “Correct Predictions” tables for Model 3 and Model 7 in [Appendix D](#_Appendix_D_–_1). In general, the predicted award shares are closer to the actual values for the XGBoost model. For example, for the first season in the test set, LeBron James won the MVP with an award share of 0.969. XGBoost predicted an award share of 0.833, while the linear model severely underpredicted that value by predicting James would earn a share of 0.271.

Considering the XGBoost model is better than the linear model in accuracy, R2, and RMSE, I am more confident in the results of this model and will use it to predict the 2022-23 MVP.

## 2022-23 MVP Predictions

The following table displays the top 10 players by predicted award share and compares these rankings to the current NBA MVP ladder as of 12/10/22 (“Kia MVP Ladder: Jayson Tatum Seizes Top Spot in Latest Rankings.”).

| **Player** | **Predicted Award Share** | **My Rank** | **KIA MVP Rank** |
| --- | --- | --- | --- |
| Giannis Antetokounmpo | 0.36678 | 1 | 4 |
| Luka Doncic | 0.19797 | 2 | 2 |
| Jayson Tatum | 0.18820 | 3 | 1 |
| Ja Morant | 0.12540 | 4 | 8 |
| Damian Lillard | 0.07036 | 5 | N/A |
| Donovan Mitchell | 0.06688 | 6 | 9 |
| James Harden | 0.05897 | 7 | N/A |
| Trae Young | 0.05763 | 8 | N/A |
| Kevin Durant | 0.05159 | 9 | 7 |
| Tyrese Haliburton | 0.05082 | 10 | N/A |

At the moment, my model favors Giannis Antetokounmpo with Luka Doncic following at a distance as the second most likely MVP. Jayson Tatum rounds out my top three most likely MVPs at the quarter-season mark. The predicted award shares are noticeably lower for these players than the actual award shares have been for MVP winners. This is likely because teams have only played between 20 and 25 games, so the MVP race is much more contested than it is at the end of a season. Usually, by the end of a season, it is clear which few players will be considered for the award. For example, in 2021-22, Nikola Jokic and Joel Embiid were the clear frontrunners for the award as the season neared its close. I expect this to happen for this season as well, so there will likely be a few players with very high predicted award shares while the rest of the NBA will post predicted award shares close to zero.

For now, Antetokounmpo is the most likely MVP according to my model. He is fourth on the Kia MVP ladder. I predict that Doncic is the second most likely MVP, while the NBA has him as the runner-up. Tatum, who is the current frontrunner according to the NBA, is the third most likely. There are notable similarities between my predictions and the MVP ladder, which is encouraging. It appears that, in general, the most competitive MVP candidates are included in my top 10. Six of my top 10 candidates also made the top 10 in the Kia MVP ladder. Predicted ranks for Tatum, Doncic, and Durant were either identical or within two spots.

However, there are some discrepancies between who my model believes will win and who the NBA currently favors. For example, my model predicts Antetokounmpo will dominate the award, but the current ladder has him at rank four. I expect this difference is due to the importance XGBoost places on team winning percentage. The Milwaukee Bucks are second in the Eastern Conference with a 17-6 record. Not only is Antetokounmpo averaging 31.9 points and 11.3 rebounds, leading the league in usage rate, and boasting the fourth-best PER in the league, but his team is also excelling behind him. Likewise, Tatum’s Celtics are leading the East with 20 wins while he averages 30.8 points and 8.3 rebounds and registers in the top 10 for PER, VORP, and Win Shares.

Both of these players are deserving of a top 10 position in the rankings, but it is curious that Luka Doncic has not usurped Antetokounmpo. Doncic leads the league in points per game, Win Shares, OBPM, and VORP. He is also second in BPM, OWS, and PER. Given his dominance of advanced statistics, one would expect him to top the rankings. Advanced statistics, specifically WS and VORP, are strongly related to MVP awards. The reason Luka is probably second, as opposed to first, is due to his team’s lackluster performance thus far. Sitting at 13-11, the Mavericks are stagnant aside from their superstar point guard. Since the model weighs team winning percentage heavily, Doncic is penalized. This may also explain the difference between my predictions and the NBA’s rankings for Donovan Mitchell and Ja Morant whose teams are performing well but are not ranked as highly by the NBA media.

## Conclusions

My primary research question was as follows: what factors are the best predictors of who will win the most valuable player award in a given year?

I answered this question through a variety of marginal analysis techniques, including correlations between numeric explanatory variables and award share and simple logistic models for winning MVP with each explanatory variable. I followed this analysis with data visualization to explore the relationship between strongly correlated and highly significant predictors. Through this research, I determined that prior MVP performances (described by total MVP awards and the previous year’s award share), and advanced statistics (specifically Win Shares and Value Over Replacement Player) were the best predictors for the MVP award. Free Throws attempted and made were the basic statistics with the strongest relationships. If one were to examine a few statistics individually to determine MVP likelihood, these are strong candidates.

This marginal analysis also revealed that, per my initial hypothesis, advanced statistics are generally better predictors on their own than basic statistics. Additionally, comparing logistic models for winning MVP, one with only advanced predictors and one with only basic predictors, it seems advanced statistics provided more accurate predictions. This makes sense because these statistics generally aim to be holistic descriptors of performance, while basic statistics cover only small aspects of the game.

My secondary research question was as follows: which model provides the most accurate MVP predictions?

This question can be divided into two parts:

1. Which response provides more accurate results: winning MVP or award share?
2. What type of model is most accurate?

Concerning part one, it became clear that modeling award share was the better approach because, due to class imbalance, models based on award share had improved test set accuracy compared to logistic models.

With regard to part two, using current data, I determined an XGBoost model was the best solution and used parameter tuning to optimize its accuracy. This model correctly predicted nine of 10 MVPs from 2008-09 to 2017-18.

I initially wanted to conduct this analysis using the prior season’s data, but I shifted to current data for multiple reasons. Primarily, as expected, predictions improved when using current data. Additionally, MVPs are evaluated on their current performance, not past performance, so using past data is not representative of a player’s chances at the award.

As a result of this investigation, I was able to determine which variables are strong marginal predictors for the MVP award winner and produce an accurate model for predicting the MVP.

## Future Analysis

Since I now have a model that will output predictions given current player and team data, I can use the model to track predictions throughout the 2022-23 season. My goal is to create an automated process that gathers up-to-date player and team statistics, enters them into the model, generates a top 10 most likely MVPs list (as shown above), and compares the list to the Kia MVP ladder as it is updated. Since the ladder updates weekly, I would like to automate my model to create the list on the same day. This would allow me to track the model’s predictions in relation to the media and determine where, and perhaps why, my model’s predictions may deviate from media preferences.

At the end of the season, I will check which player has the highest predicted award share and see if that player wins. Ideally, I will do this every year to determine if the model performs well on additional holdout data and, specifically, more current data, since my model was trained on data ending in 2009.

Concerning further model creation, I would like to investigate how to create simpler models without sacrificing accuracy. My final model relies on 56 variables. As seen in the importance plot, many of these variables are not important for the XGBoost predictions. It is possible that variables could be removed to simplify the model and limit the possibility of null predictions due to missing values. Given time constraints, I was not able to iteratively remove variables and compare model performances to arrive at an optimal model. Investigating a method to automate this process is an enticing opportunity.

Finally, I would like to investigate features related to media coverage and add them to the model. My data describes how well players and teams perform but does not cover the qualitative trends in positive and negative media attention. I would like to quantify the amount of attention a player receives in some way, such as the number of Google searches or Twitter mentions for each player. This could be divided further into positive and negative mentions. Parsing out the tone or intent of a mention could be a very challenging machine-learning problem that I am not currently equipped to solve but would be an exciting addition to this project. In the future, I hope to investigate the narrative aspect of the MVP award in more detail and bolster my model by including these features.

# Appendix A – Data Visualization

# Chart, histogram Description automatically generated

Chart, bar chart

Description automatically generated

Chart, bar chart

Description automatically generated

Chart

Description automatically generated with medium confidence

Chart

Description automatically generatedTimeline

Description automatically generated with medium confidence

Chart, scatter chart

Description automatically generated

Chart, scatter chart

Description automatically generated

# 

# Appendix B – Advanced Statistics Glossary

All definitions and formulas are from the Basketball Reference Glossary

* Three-point rate (3PRate) – percentage of a player’s shots taken from 3-point range
  + 3PA / FGA
* Free Throw Rate (FTRate) - a measure of both how often a team gets to the line and how often they make them
  + FT / FGA
* Offensive Rebound Percentage (ORB%) – an estimate of the percentage of available offensive rebounds a player grabbed while he was on the floor
  + 100 \* (ORB \* (Team Minutes Played / 5)) / (Minutes Played \* (Team ORB + Opponent DRB))
* Defensive Rebound Percentage (DRB%) - an estimate of the percentage of available defensive rebounds a player grabbed while he was on the floor
  + 100 \* (DRB \* (Team Minutes Played / 5)) / (Minutes Played \* (Team DRB + Opponent DRB))
* Total Rebound Percentage (TRB%) - an estimate of the percentage of available total rebounds a player grabbed while he was on the floor
  + 100 \* (TRB \* (Team Minutes Played / 5)) / (Minutes Played \* (Team TRB + Opponent TRB))
* Assist Percentage (AST%) – estimate of the percentage of teammate field goals a player assisted while he was on the floor
  + 100 \* AST / (((Minutes Played / (Team Minutes Played / 5)) \* Team FG) - FG)
* Steal Percentage (STL%) - an estimate of the percentage of opponent possessions that end with a steal by the player while he was on the floor
  + 100 \* (STL \* (Team Minutes Played / 5)) / (Minutes Played \* Opponent Possessions)
* Block Percentage (BLK%) - an estimate of the percentage of opponent two-point field goal attempts blocked by the player while he was on the floor
  + 100 \* (BLK \* (Team Minutes Played / 5)) / (Minutes Played \* (Opponent FGA – Opponent 3PA))
* Turnover Percentage (TOV%) – an estimate of turnovers per 100 plays
  + 100 \* TOV / (FGA + 0.44 \* FTA + TOV)
* Usage Rate (USG%) - an estimate of the percentage of team plays used by a player while he was on the floor
  + 100 \* ((FGA + 0.44 \* FTA + TOV) \* (Team Minutes Played / 5)) / (Minutes Played \* (Team FGA + 0.44 \* Team FTA + Team TOV))
* Player Efficiency Rating (PER) - a rating developed by ESPN.com columnist John Hollinger. In Hollinger’s words, "The PER sums up all a player's positive accomplishments, subtracts the negative accomplishments, and returns a per-minute rating of a player's performance."
  + Refer to <https://www.basketball-reference.com/about/per.html> for the calculation
* Win Shares (WS) - an estimate of the number of wins contributed by a player
  + Refer to <https://www.basketball-reference.com/about/ws.html> for the calculation
* Win Shares per 48 Minutes (WS/48) - an estimate of the number of wins contributed by the player per 48 minutes (league average is approximately 0.100)
  + Refer to <https://www.basketball-reference.com/about/ws.html> for the calculation
* Offensive Win Shares (OWS) - estimates the number of wins a player produces for their team due to their offensive ability
  + Refer to <https://www.basketball-reference.com/about/ws.html> for the calculation
* Defensive Win Shares (DWS) - estimates the number of wins a player produces for their team due to their defensive ability
  + Refer to <https://www.basketball-reference.com/about/ws.html> for the calculation
* Box Plus Minus (BPM) - a box score estimate of the points per 100 possessions that a player contributed above a league-average player, translated to an average team
  + Refer to <https://www.basketball-reference.com/about/bpm2.html> for the calculation
* Offensive Box Plus Minus (OBPM) – the offensive component of BPM
  + Refer to <https://www.basketball-reference.com/about/bpm2.html> for the calculation
* Defensive Box Plus Minus (DBPM) – the defensive component of BPM
  + Total BPM - OBPM
* Value Over Replacement Player (VORP) - a box score estimate of the points per 100 team possessions that a player contributed above a replacement-level (-2.0 WS) player, translated to an average team and prorated to an 82-game season. Multiply by 2.70 to convert to wins over replacement
  + Refer to <https://www.basketball-reference.com/about/bpm2.html> for the calculation

# 

# Appendix C – Exploratory Data Analysis Tables

**Table 1 - Correlation of Each Numeric Variable with Award Share**

|  | **Variable** | **Correlation with Award Share** |
| --- | --- | --- |
| 1 | ratioVORP | 0.44942 |
| 2 | totalMVP | 0.39655 |
| 3 | ratioOWS | 0.37699 |
| 4 | ratioWS | 0.37353 |
| 5 | ftaPerGame | 0.30497 |
| 6 | ftmPerGame | 0.30464 |
| 7 | ptsPerGame | 0.27479 |
| 8 | ratioDWS | 0.27454 |
| 9 | fg2mPerGame | 0.26683 |
| 10 | fgmPerGame | 0.26377 |
| 11 | fg2aPerGame | 0.2479 |
| 12 | fgaPerGame | 0.23987 |
| 13 | tovPerGame | 0.22156 |
| 14 | ratioPER | 0.21273 |
| 15 | drbPerGame | 0.20279 |
| 16 | astPerGame | 0.19095 |
| 17 | pctUSG | 0.18728 |
| 18 | trbPerGame | 0.18479 |
| 19 | ratioOBPM | 0.18312 |
| 20 | stlPerGame | 0.18241 |
| 21 | ratioBPM | 0.18084 |
| 22 | countGamesStarted | 0.16661 |
| 23 | minutesPerGame | 0.16574 |
| 24 | ratioWSPer48 | 0.1591 |
| 26 | blkPerGame | 0.15015 |
| 25 | pctWins | 0.13271 |
| 27 | pctAST | 0.12672 |
| 28 | orbPerGame | 0.1202 |
| 29 | ratioDBPM | 0.09253 |
| 30 | countGames | 0.09021 |
| 31 | pctFGPerGameTeam | 0.08429 |
| 32 | pctFTRate | 0.07959 |
| 33 | pctTrueShooting | 0.07738 |
| 34 | pfPerGame | 0.06953 |
| 35 | pctFG | 0.06944 |
| 36 | pctDRB | 0.06892 |
| 37 | pctFG2 | 0.06212 |
| 38 | fg3mPerGame | 0.06031 |
| 39 | pctEFG | 0.0599 |
| 40 | fg3aPerGame | 0.05969 |
| 41 | pctTRB | 0.0521 |
| 42 | pctFT | 0.05048 |
| 43 | astPerGameTeam | 0.04625 |
| 57 | weight | 0.04157 |
| 44 | ptsPerGameTeam | 0.03266 |
| 45 | pctFG3 | 0.02982 |
| 56 | height | 0.02550 |
| 46 | drbPerGameTeam | 0.02001 |
| 47 | agePlayer | 0.01867 |
| 48 | pctFTPerGameTeam | 0.00011 |
| 49 | tovPerGameTeam | -0.00127 |
| 50 | pctORB | -0.00867 |
| 51 | pctSTL | -0.0184 |
| 52 | pctBLK | -0.01848 |
| 53 | pctTOV | -0.02668 |
| 54 | pct3PRate | -0.02823 |
| 55 | numberPickOverall | -0.09373 |

**Table 2- P-Values from simple logistic models for each predictor variable**

|  | **Variable** | **P-Value** | **Significant Marginal Relationship at alpha=0.05?** |
| --- | --- | --- | --- |
| 1 | totalMVP | 0 | Yes |
| 2 | award\_share | 0 | Yes |
| 3 | ratioWS | 0 | Yes |
| 4 | ratioVORP | 0 | Yes |
| 5 | ratioOWS | 0 | Yes |
| 6 | ratioDWS | 0 | Yes |
| 7 | ftaPerGame | 0 | Yes |
| 8 | trbPerGame | 0 | Yes |
| 9 | ftmPerGame | 0 | Yes |
| 10 | ptsPerGame | 0 | Yes |
| 11 | fg2mPerGame | 0 | Yes |
| 12 | fgmPerGame | 0 | Yes |
| 13 | fg2aPerGame | 0 | Yes |
| 14 | fgaPerGame | 0 | Yes |
| 15 | ratioPER | 0 | Yes |
| 16 | astPerGame | 0 | Yes |
| 17 | ratioWSPer48 | 0 | Yes |
| 18 | minutesPerGame | 0 | Yes |
| 19 | ratioBPM | 0 | Yes |
| 20 | ratioOBPM | 0 | Yes |
| 21 | drbPerGame | 0 | Yes |
| 22 | blkPerGame | 0 | Yes |
| 23 | tovPerGame | 0 | Yes |
| 24 | stlPerGame | 0 | Yes |
| 25 | countGamesStarted | 0 | Yes |
| 26 | pctAST | 0 | Yes |
| 27 | ratioDBPM | 0 | Yes |
| 28 | countGames | 0 | Yes |
| 29 | numberPickOverall | 0 | Yes |
| 30 | orbPerGame | 0 | Yes |
| 31 | pctFTRate | 0 | Yes |
| 32 | pctFG | 0 | Yes |
| 33 | pctTrueShooting | 0 | Yes |
| 34 | pctFG2 | 0 | Yes |
| 35 | pctEFG | 0 | Yes |
| 36 | pfPerGame | 0 | Yes |
| 37 | pctWins | 0 | Yes |
| 38 | pctFGPerGameTeam | 0 | Yes |
| 39 | height | 1e-05 | Yes |
| 40 | groupPosition | 7e-05 | Yes |
| 41 | drbPerGameTeam | 0.00015 | Yes |
| 42 | pctUSG | 0.00015 | Yes |
| 43 | pctTOV | 0.00018 | Yes |
| 44 | astPerGameTeam | 0.00041 | Yes |
| 45 | fg3mPerGame | 0.00046 | Yes |
| 46 | weight | 0.00072 | Yes |
| 47 | ptsPerGameTeam | 0.00074 | Yes |
| 48 | fg3aPerGame | 0.00162 | Yes |
| 49 | pctFT | 0.00492 | Yes |
| 50 | pctDRB | 0.00998 | Yes |
| 51 | pctTRB | 0.01257 | Yes |
| 52 | pct3PRate | 0.01467 | Yes |
| 53 | pctBLK | 0.02146 | Yes |
| 54 | pctSTL | 0.03223 | Yes |
| 55 | slugTeamBREF | 0.10443 | No |
| 56 | agePlayer | 0.18008 | No |
| 57 | pctORB | 0.19397 | No |
| 58 | pctFTPerGameTeam | 0.19469 | No |
| 59 | tovPerGameTeam | 0.62301 | No |
| 60 | pctFG3 | 0.80104 | No |
| 61 | slugPosition | 0.84501 | No |

**Table 3 -** **P-values from t-tests comparing MVPs to non-MVPs since the 1980-81 season**

|  | **Variable** | **P-Value** | **Is Mean for MVPs Greater or Less Than Mean for non-MVPs?** | **Significant at alpha=0.05?** |
| --- | --- | --- | --- | --- |
| 1 | countGamesStarted | 0 | Greater | Yes |
| 2 | minutesPerGame | 0 | Greater | Yes |
| 3 | ratioPER | 0 | Greater | Yes |
| 4 | ratioOBPM | 0 | Greater | Yes |
| 5 | ratioWSPer48 | 0 | Greater | Yes |
| 6 | ptsPerGame | 0 | Greater | Yes |
| 7 | ratioWS | 0 | Greater | Yes |
| 8 | ratioBPM | 0 | Greater | Yes |
| 9 | fgmPerGame | 0 | Greater | Yes |
| 10 | fgaPerGame | 0 | Greater | Yes |
| 11 | ratioOWS | 0 | Greater | Yes |
| 12 | ratioVORP | 0 | Greater | Yes |
| 13 | countGames | 0 | Greater | Yes |
| 14 | numberPickOverall | 0 | Less | Yes |
| 15 | tovPerGame | 0 | Greater | Yes |
| 16 | ftmPerGame | 0 | Greater | Yes |
| 17 | ftaPerGame | 0 | Greater | Yes |
| 18 | pctUSG | 0 | Greater | Yes |
| 19 | fg2mPerGame | 0 | Greater | Yes |
| 20 | fg2aPerGame | 0 | Greater | Yes |
| 21 | ratioDWS | 0 | Greater | Yes |
| 22 | pctTrueShooting | 0 | Greater | Yes |
| 23 | pctFG | 0 | Greater | Yes |
| 24 | pctFG2 | 0 | Greater | Yes |
| 25 | drbPerGame | 0 | Greater | Yes |
| 26 | pctEFG | 0 | Greater | Yes |
| 27 | pctFT | 0 | Greater | Yes |
| 28 | stlPerGame | 0 | Greater | Yes |
| 29 | ratioDBPM | 0 | Greater | Yes |
| 30 | astPerGame | 0 | Greater | Yes |
| 31 | trbPerGame | 0 | Greater | Yes |
| 32 | pctAST | 0 | Greater | Yes |
| 33 | pctFTRate | 0 | Greater | Yes |
| 34 | pfPerGame | 0 | Greater | Yes |
| 35 | pctDRB | 1e-05 | Greater | Yes |
| 36 | blkPerGame | 2e-05 | Greater | Yes |
| 37 | orbPerGame | 3e-05 | Greater | Yes |
| 38 | pctTRB | 0.00028 | Greater | Yes |
| 39 | pctTOV | 0.00032 | Less | Yes |
| 40 | pctFG3 | 0.00122 | Greater | Yes |
| 41 | fg3aPerGame | 0.00212 | Greater | Yes |
| 42 | fg3mPerGame | 0.00311 | Greater | Yes |
| 43 | agePlayer | 0.01326 | Greater | Yes |
| 44 | pctSTL | 0.03818 | Less | Yes |
| 45 | pctORB | 0.04836 | Less | Yes |
| 46 | pct3PRate | 0.09628 | Less | No |
| 47 | weight | 0.10829 | Greater | No |
| 48 | height | 0.20256 | Greater | No |
| 49 | pctBLK | 0.44776 | Less | No |

# Appendix D – Model Results

Model 1 - Linear Model for Award Share (Prior Year’s Data):

Accuracy = 50%

R2 = 0.2407

RMSE = 0.0512

p< 2.2\*10-16

Correct Predictions:

| **Player** | **Season** | **Award Share** | **Predicted Award Share** |
| --- | --- | --- | --- |
| LeBron James | 2008-09 | 0.969 | 0.22230 |
| LeBron James | 2009-10 | 0.98 | 0.24871 |
| LeBron James | 2011-12 | 0.888 | 0.17402 |
| LeBron James | 2012-13 | 0.998 | 0.16891 |
| Russell Westbrook | 2016-17 | 0.879 | 0.16001 |

Chart, scatter chart

Description automatically generated

Chart, line chart

Description automatically generated

Model 2 – Logistic Model for Winning MVP (Prior Year’s Data):

Graphical user interface

Description automatically generated with low confidence

Accuracy = 20%

Sensitivity = 0%

Specificity = 99.97%

Correct Predictions:

| **Player** | **Season** | **Predicted MVP Probability** |
| --- | --- | --- |
| LeBron James | 2009-10 | 0.15028 |
| LeBron James | 2012-13 | 0.10930 |

Model 3 – Linear Model for Award Share (Current Year’s Data)

Accuracy = 70%

R2 = 0.2935

RMSE= 0.0513

p< 2.2\*10-16

Correct Predictions:

| **Player** | **Season** | **Award Share** | **Predicted Award Share** |
| --- | --- | --- | --- |
| LeBron James | 2008-09 | 0.969 | 0.27148 |
| LeBron James | 2009-10 | 0.98 | 0.24705 |
| LeBron James | 2011-12 | 0.888 | 0.18490 |
| LeBron James | 2012-13 | 0.998 | 0.23591 |
| Kevin Durant | 2013-14 | 0.986 | 0.21917 |
| Stephen Curry | 2015-16 | 1.000 | 0.18221 |
| Russell Westbrook | 2016-17 | 0.879 | 0.25220 |

Chart, scatter chart

Description automatically generated

Chart, line chart

Description automatically generated

Model 4 – Logistic Model for Winning MVP (Current Year’s Data)

Graphical user interface

Description automatically generated with medium confidence

Accuracy = 40%

Sensitivity = 20%

Specificity = 99.94%

Correct Predictions:

| **Player** | **Season** | **Predicted MVP Probability** |
| --- | --- | --- |
| LeBron James | 2008-09 | 0.55698 |
| LeBron James | 2009-10 | 0.52699 |
| LeBron James | 2011-12 | 0.10976 |
| Kevin Durant | 2013-14 | 0.36880 |

Model 5 - Basic Statistics Model:

Variables:

TS%, FG%, 3PT%, 2PT%, EFG%, FT%, FGM per game, FGA per game,

3Pt FGM per game, 3Pt FGA per game, 2Pt FGM per game, 2Pt FGA per game,

FTM per game, FTA per game, ORB per game, DRB per game, TRB per game, AST per game, STL per game, BLK per game, TOV per game, PF per game, PTS per game

Graphical user interface, text, application

Description automatically generated

Accuracy on test = 10%

Sensitivity = 0%

Specificity = 99.97%

Correct Predictions:

| **Player** | **Season** | **Predicted MVP Probability** |
| --- | --- | --- |
| LeBron James | 2012-13 | 0.29459 |

Model 6 - Advanced Statistics Model:

Variables:

PER, 3Prate, FTRate, ORB%, DRB%, TRB%, AST%, STL%, BLK%, TOV%, USG%, OWS, DWS, WS, WS/48, OBPM, DBPM, BPM, VORP

Graphical user interface, text, application

Description automatically generated

Accuracy on test= 20%

Sensitivity = 50%

Specificity = 99.37%

Correct Predictions:

| **Player** | **Season** | **Predicted MVP Probability** |
| --- | --- | --- |
| LeBron James | 2009-10 | 0.99249 |
| Stephen Curry | 2015-16 | 0.97928 |

Model 7 – XGBoost for Award Share (Current Year’s Data):

Accuracy = 100%

R2 = 0.6823

RMSE = 0.0368

Correct Predictions:

| **Player** | **Season** | **Award Share** | **Predicted Award Share** |
| --- | --- | --- | --- |
| LeBron James | 2008-09 | 0.969 | 0.83278 |
| LeBron James | 2009-10 | 0.980 | 0.91833 |
| Derrick Rose | 2010-11 | 0.977 | 0.58619 |
| LeBron James | 2011-12 | 0.888 | 0.58310 |
| LeBron James | 2012-13 | 0.998 | 0.82773 |
| Kevin Durant | 2013-14 | 0.986 | 0.80155 |
| Stephen Curry | 2014-15 | 0.922 | 0.93008 |
| Stephen Curry | 2015-16 | 1.000 | 0.89623 |
| Russell Westbrook | 2016-17 | 0.879 | 0.72330 |
| James Harden | 2017-18 | 0.955 | 0.77518 |

Parameters:

ETA = 0.05

Max Depth = 4

Gamma = 0

Column Sample by Tree = 1

Minimum Child Weight = 1

Subsample = 1

Number of Rounds = 500

Class Weight Parameter = 16.5497

Chart

Description automatically generated

Works Cited

Abu-Rmileh, Amjad. “The Multiple Faces of ‘Feature Importance’ in XGBoost.” *Medium*, 2 Sept. 2021, <https://towardsdatascience.com/be-careful-when-interpreting-your-features-importance-in-xgboost-6e16132588e7>.

Brownlee, Jason. “A Gentle Introduction to K-Fold Cross-Validation.” *MachineLearningMastery.Com*, 22 May 2018, <https://machinelearningmastery.com/k-fold-cross-validation/>.

Brownlee, Jason. “A Gentle Introduction to the Gradient Boosting Algorithm for Machine Learning.” *MachineLearningMastery.Com*, 8 Sept. 2016, <https://machinelearningmastery.com/gentle-introduction-gradient-boosting-algorithm-machine-learning/>.

Brownlee, Jason. “How to Configure XGBoost for Imbalanced Classification.” *MachineLearningMastery.Com*, 4 Feb. 2020, <https://machinelearningmastery.com/xgboost-for-imbalanced-classification/>.

Brownlee, Jason. “XGBoost for Regression.” *MachineLearningMastery.Com*, 11 Mar. 2021, <https://machinelearningmastery.com/xgboost-for-regression/>.

Chen, Tianqi, and Carlos Guestrin. “XGBoost: A Scalable Tree Boosting System.” *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ACM, 2016, pp. 785–94. *DOI.org (Crossref)*, <https://doi.org/10.1145/2939672.2939785>.

Corvo, Michael. “How Voting Is Done for the NBA MVP and Its Evolution.” *ClutchPoints*, 1 June 2021, <https://clutchpoints.com/how-voting-is-done-for-the-nba-mvp-and-its-evolution>.

Frost, Jim. “Multicollinearity in Regression Analysis: Problems, Detection, and Solutions.” *Statistics By Jim*, 2 Apr. 2017, <http://statisticsbyjim.com/regression/multicollinearity-in-regression-analysis/>.

“Glossary.” *Basketball-Reference.Com*, <https://www.basketball-reference.com/about/glossary.html>. Accessed 15 Nov. 2022.

Goldstein, Omri. *NBA Players Stats since 1950*. <https://www.kaggle.com/datasets/drgilermo/nba-players-stats>. Accessed 15 Nov. 2022.

*HoopR-Package: HoopR*. <https://rdrr.io/github/saiemgilani/kenpomR/man/hoopR-package.html>. Accessed 15 Nov. 2022.

hrfang. *NBA Drafts of 1947-2018*. <https://www.kaggle.com/datasets/hrfang1995/nba-drafts-of-19472018>. Accessed 15 Nov. 2022.

“Kia MVP Ladder: Jayson Tatum Seizes Top Spot in Latest Rankings.” *NBA.Com*, <https://www.nba.com/news/kia-mvp-ladder-dec-2-2022-edition>. Accessed 8 Dec. 2022.

“NBA Team Percent of Points from Free Throws.” *TeamRankings*, <https://www.teamrankings.com/nba/stat/percent-of-points-from-free-throws>. Accessed 15 Nov. 2022.

Winn, Luke. “The Old College Try: The Complicated Recruiting Sagas of Arvydas Sabonis and His Son Domantas.” *Sports Illustrated Longform*, <http://www.si.com/longform/sabonis/index.html>. Accessed 15 Nov. 2022.

Yoo, David. “Predicting the next NBA MVP Using Machine Learning.” *Medium*, 6 May 2022, <https://towardsdatascience.com/predicting-the-next-nba-mvp-using-machine-learning-62615bfcff75>.