**Design and Implementation of Analytics System**

*Determining Playoff Teams*

*Nick Smailer*

**School of Graduate Professional Studies**

Data Analytics

DAAN 888 – Design and Implementation of Analytics System

Fall, 2024

# Document Control

## Work carried out by:

|  |  |  |
| --- | --- | --- |
| **Name** | **Email Address** | **Task description** |
| Nick Smailer | nicksmailer@verizon.net | Created predictive modeling system to determine playoffs using win total. |

## Revision Sheet

|  |  |  |
| --- | --- | --- |
| **Release No.** | **Date** | **Revision Description** |
| 1.0 | 9/7 | Completed Lesson 2 Assignment |
| 1.1 | 9/15 | Added Data Dictionary |
| 1.2 | 9/22 | Described Cleaning and Introductory Modeling |
| 1.3 | 9/30 | Added Descriptive Statistics |
| 1.4 | 10/5 | Described Descriptive Statistics and all Cleaning |
| 1.5 | 10/12 | Discussed Warehouse Creation and Loading |
| 1.6 | 10/20 | Outlined Variable Selection and Data Transformation |
| 1.7 | 10/26 | Described Subset Creation and Modeling |
| 1.8 | 11/2 | Discussed Model Performance and Refinement Approach |
| 1.9 | 11/10 | Outlined Adjustments to Modeling |
| 2.0 | 11/16 | Detailed Final Conclusions |
| 2.1 | 11/23 | Added Final Visualizations |
| 2.2 | 11/30 | Adjusted Final Conclusions |
| 2.3 | 12/3 | Added Final System Instructions |
| 2.4 | 12/7 | Final Adjustments |

Table of Contents

[Document Control 1](#_Toc184493258)

[Work carried out by: 1](#_Toc184493259)

[Revision Sheet 1](#_Toc184493260)

[General Guidelines 3](#_Toc184493261)

[Lesson 2 Predictive Analytics System Group-Based Assignment 6](#_Toc184493262)

[Project Proposal: 6](#_Toc184493263)

[Proposal References: 8](#_Toc184493264)

[Task Schedule: 8](#_Toc184493265)

[Lesson 4 Predictive Analytics System Group-Based Assignment 9](#_Toc184493266)

[Data Dictionary: 9](#_Toc184493267)

[Lesson 6 Predictive Analytics System Group-Based Assignment 13](#_Toc184493268)

[Data Cleaning: 13](#_Toc184493269)

[Introductory Modeling: 14](#_Toc184493270)

[Descriptive Statistics Visualizations: 16](#_Toc184493271)

[Lesson 8 Predictive Analytics System Group-Based Assignment 19](#_Toc184493272)

[Variable Transformation: 20](#_Toc184493273)

[Feature Selection: 20](#_Toc184493274)

[Lesson 10 Predictive Analytics System Group-Based Assignment 22](#_Toc184493275)

[Variable Selection: 22](#_Toc184493276)

[Transformation: 22](#_Toc184493277)

[Modeling: 23](#_Toc184493278)

[Lesson 12 Predictive Analytics System Group-Based Assignment 24](#_Toc184493279)

[Parameter Testing: 24](#_Toc184493280)

[Modeling: 25](#_Toc184493281)

[Lesson 13 Predictive Analytics System Group-Based Assignment 29](#_Toc184493282)

[Research questions: 29](#_Toc184493283)

[Final Conclusions & Recommendations: 30](#_Toc184493284)

[Citations: 37](#_Toc184493285)

[Appendix: 38](#_Toc184493286)

[Lesson 14 Predictive Analytics System Group-Based Assignment 47](#_Toc184493287)

[Final Predictive Analytic System: 47](#_Toc184493288)

[Detailed Instructions to Run Application: 48](#_Toc184493289)

**General Guidelines**

1. To complete all the homework assignments for this course please use this template document.
2. Each assignment has to be submitting by Sunday 11:59 PM EST.
3. Each figure should be followed by a brief description about the figure.
4. The figures can be hand drawn and scanned in some circumstances, but the hand drawn figure should be clear and legible to obtain full credits. Unclear hand drawn figures will receive partial credits. For constructing figures and diagrams it is advised to use tools.
5. Figures and tables should have appropriate captions. For documenting and referencing styles please follow the APA or MLA writing style.
6. Please make sure that you provide a reference section.
7. Any material text or figure taken from books, journals or Internet should be referenced. If you have a sentence or a figure that does not belong (authorship) to you they need to be clearly referenced. If you fail to do so your report will be considered as a case for plagiarism. It is your duty to make sure that your report is free from any activity related to plagiarism. In case you are suspected of attempting plagiarism then you will be responsible for the cause. The penalty for plagiarism will be a “0” awarded to your report. So it is good to keep simple, always have the principle to acknowledge people for their contributions.

Please go through the following instructions before submitting the report

#### **Academic Integrity**

Academic integrity — scholarship free of fraud and deception — is an important educational objective of Penn State. Academic dishonesty can lead to a failing grade or referral to the [Office of Student Conduct](http://www.sa.psu.edu/ja/).

Academic dishonesty includes, but is not limited to:

* cheating
* plagiarism
* fabrication of information or citations
* facilitating acts of academic dishonesty by others
* unauthorized prior possession of examinations
* submitting the work of another person or work previously used without informing the instructor and securing written approval
* tampering with the academic work of other students

#### How Academic Integrity Violations Are Handled

In cases where academic integrity is questioned, [procedure requires an instructor to notify a student](http://www.psu.edu/oue/aappm/G-9-academic-integrity.html) of suspected dishonesty before filing a charge and recommended sanction with the college. Procedures allow a student to accept or contest a charge. If a student chooses to contest a charge, the case will then be managed by the respective college or campus Academic Integrity Committee. If a disciplinary sanction also is recommended, the case will be referred to the [Office of Student Conduct](http://www.sa.psu.edu/ja/title=).

All Penn State colleges abide by this Penn State policy, but review procedures may vary by college when academic dishonesty is suspected. Information about Penn State's academic integrity policy and college review procedures is included in the information that students receive upon enrolling in a course.

Additionally, Penn State students are expected to act with civility and personal integrity; respect other students' dignity, rights, and property; and help create and maintain an environment in which all can succeed through the fruits of their own efforts. An environment of academic integrity is requisite to respect for oneself and others, and a civil community.

#### For More Information on Academic Integrity at Penn State

Please see the [Academic Integrity Chart](http://www.campuses.psu.edu/CAO.pdf)  for specific college contact information or visit one of the following URLs:

* Penn State Senate [Policy on Academic Integrity](http://www.psu.edu/dept/oue/aappm/G-9.html)
* [iStudy for Success!](http://istudy.psu.edu/tutorials/) — learn about plagiarism, copyright, and academic integrity through an educational module
* [Turnitin](http://tlt.its.psu.edu/turnitin) a web-based plagiarism detection and prevention system

# Lesson 2 Predictive Analytics System Group-Based Assignment

**Purpose:**

To describe the purpose/objectives of the proposed predictive analytics system (project)

**Tasks:**

1. After the meeting with the instructor in week 1 the group will put together a proposal for designing a predictive analytics system.
2. Clearly list the task that will be completed in the following weeks. I should have provided the directions in regard to completing the tasks for different weeks during the week 1 meeting with the team.

## Project Proposal:

In the United States, one of the premier sports leagues is the National Basketball Association, also called the NBA4. Founded in the 1940s, the NBA has become a globally known league over the years, with fans in most countries4. Similar to all sports, the goal is to be the best, which in the NBA is achieved through winning the NBA Finals5. The Finals is a best-of-seven series between the top teams from each conference, East and West. To reach the Finals a team must first qualify for the playoffs, then move through three successive rounds5. Just over half of the league makes the playoffs, which is 16 out of 30 teams4. There are also sizable financial incentives for reaching the post-season. After making the playoffs, a team’s franchise sees an increase in merchandise sales, as well as a higher demand for tickets6. At the same time, the tickets to playoff games are more expensive6. All of these factors lead to an increase in revenue for the team6. Additionally, people travel to see these games, as they are often intense and action-packed, leading to higher overall spending in the respective city providing a temporary economic boost6. Continued playoff presence garners continuous interest as the team maintains relevance- Golden State Warriors are a great example3. Staying relevant and entertaining leads to a greater likelihood of larger profits in a variety of ways. More games will be broadcast, more fans will attend, and more merchandise will be sold6. A benefit from the player’s perspective is more respect/fame/accolades as a result of playoff experience and/or championships7. Taking all of these factors into consideration, it becomes evident that making the playoffs is advantageous for the players, franchises, and cities. Considering the investment by franchises into their players, and the massive financial implications, a clear problem arises: tons of resources are invested, there’s a lot on the line, but no way to guarantee success. Picking who will win the championship is a shot in the dark, but since half the teams make the playoffs, predicting who these might be becomes much more attainable, especially given 50+ years of data to work with.

Seeing the need for a way to predict team success (playoff berth) I decided to define a few research questions, that if answered can solve this problem. They are as follows:

* How do player statistics from the BEST teams compare to the WORST teams?
* How many wins “guarantees” a playoff berth?
* Can player statistics be used to predict a playoff berth (seasonal averages)?
  + 5 players with most minutes played
  + 5 players with most minutes played POSITIONALLY (PG, SG, SF, PF, C)
* Do advanced metrics provide better predictions than standard 5 categories (points, rebounds, assists, steals, blocks)?
* Are TEAM statistics more indicative than PLAYER statistics?
* How much impact do ALL-STARS have?

By comparing subsets of historically great teams to historically poor teams, it may become evident that certain statistical averages contribute to success. Basically, checking if patterns emerge within the subsets, and if there are clear dissimilarities between them. This will illuminate a few categories that could be critical and require further analysis. Upon early analysis, it became evident that 50 wins is fairly indicative of a playoff berth, this will be examined further to determine likelihoods by win total. There are 5 positions in basketball (PG, SG, SF, PF, C), and the lineup is constructed around these. That being said, there is often a combination of these not limited to 1 and only 1 of each, for example there could be 2 PF and no C. As a result, it is important to look at both the players who played the most minutes total and by position. Sometimes, these will be exactly the same, but not in all cases. Additionally, by looking at the top 8 players, we can include some bench contributors and see the impact of players in the regular rotation outside of the top 5. Finally, there are 5 classic statistical categories: points, rebounds, assists, steals, and blocks. In addition to these, in the past few decades advanced metrics have been added like +/- which shows the net gain or loss while a player was on the floor. I am interested in examining if including advanced metrics allows for better predictions. PCA may be helpful here. Overall, the goal is to define a playoff berth win threshold, then determine whether individual or team statistics can be used to predict wins and by extension playoff appearances.

Through extensive data exploration and cleaning, subsets will be generated for model creation. Player statistics will be divided into buckets to help with the continuous nature of the data. This is a classification problem, as the teams either make the playoffs (1) or don’t (0). Creating a data warehouse to store the data will allow subsets to be queried with relative ease based on user selected criteria. Following model generation and optimization, analysis will be conducted and conclusions made. From there, hopefully a few key metrics will be discovered, for example, if a team has two players that each average greater than X amount of points, that team goes to the playoffs Y percent of the time.

Throughout the process of answering the research questions above, the insights gained will determine what sets the great teams apart from the poor teams, how player and team statistics impact win total, and how win total impacts playoff contention. By answering these questions, teams can make more informed decisions regarding rosters and team construction in accordance with the combinations that have been proven to yield playoff berths. As mentioned above, the multifaceted benefits of a team making the playoffs provide more than enough incentive for the insights gained to demand serious consideration.

## Proposal References:

* + - 1. Cabarkapa, Dimitrije, et al. “Game Statistics That Discriminate Winning and Losing at the NBA Level of Basketball Competition.” *PloS One*, U.S. National Library of Medicine, 19 Aug. 2022, [www.ncbi.nlm.nih.gov/pmc/articles/PMC9390892/](http://www.ncbi.nlm.nih.gov/pmc/articles/PMC9390892/).
  1. Datta, Sumitro. “NBA Stats (1947-Present).” *Kaggle*, 6 July 2024, [www.kaggle.com/datasets/sumitrodatta/nba-aba-baa-stats](http://www.kaggle.com/datasets/sumitrodatta/nba-aba-baa-stats).
  2. “Golden State Warriors on the Forbes NBA Team Valuations List.” *Forbes*, Forbes Magazine, www.forbes.com/teams/golden-state-warriors/. Accessed 25 Nov. 2024.
  3. NBA. *The Official Site of the NBA for the Latest NBA Scores, Stats & News. | NBA.Com*, www.nba.com/. Accessed 8 Dec. 2024.
  4. “NBA Playoffs.” Wikipedia, Wikimedia Foundation, 11 Nov. 2024, en.wikipedia.org/wiki/NBA\_playoffs.
  5. “The Economics of the NBA Playoffs.” *ImpactFI*, 13 May 2024, impact-fi.com/blog/the-economics-of-the-nba-playoffs.
  6. Zweig, Marshall. “How to Earn Respect for Life in the NBA Playoffs.” *Bleacher Report*, Bleacher Report, 28 Aug. 2017, bleacherreport.com/articles/1616229-how-to-earn-respect-for-life-in-the-nba-playoffs.

## Task Schedule:

|  |  |  |
| --- | --- | --- |
| **Week** | **Task** | **Status** |
| Week 1 | Decide topic, introductory meeting with professor | Complete |
| Week 2 | Download data, begin exploration | Complete |
| Week 3 | Begin cleaning, early analysis, meet with professor | Complete |
| Week 4 | Finish cleaning, basic modeling | Complete |
| Week 5 | Draft warehouse, feature reduction, meet with professor | Complete |
| Week 6 | Determine features, create warehouse | Complete |
| Week 7 | Load data, meet with professor | Complete |
| Week 8 | Modeling | Complete |
| Week 9 | Discuss above steps in project report, meet with professor | Complete |
| Week 10 | Visualization | Complete |
| Week 11 | Analysis of results, meet with professor | Compete |
| Week 12 | Finalize project report | Complete |
| Week 13 | Submit report, schedule appointment | Complete |
| Week 14 | Presentation, peer review | Complete |

# Lesson 4 Predictive Analytics System Group-Based Assignment

**Purpose:**

To layout the plan for collecting data for the predictive analytic system.

**Tasks:**

1. After the meeting with the instructor in week 3 the group will put together a plan for collecting data for the proposed predictive analytics system.
2. List the different sources for data collection. Also update the progress on the data collection (if any).
3. Clearly list the task that will be completed in the following weeks. I should have provided the directions in regard to completing the tasks for different weeks during the week 3 meeting with the team.

**Data Collection and Data Dictionary**

**Data Collection:**

In order to find a suitable dataset for my goals, I browsed the Kaggle site which hosts tons of datasets spanning countless topics. I began by searching for basketball datasets, but I wasn’t seeing exactly what I had in mind, so I refined my search. Before long, I was able to find a data source containing quite a few data files (21 total) of historical NBA data. The data in the repository was previously extracted from the website Basketball Reference and stored on Kaggle in a CSV file format. Looking through the files, there was quite a bit of information, so it was crucial to determine what would be useful to answer my research questions. Considering my goals are mainly focused on player and team statistics, I chose to focus on four datasets:

Player Totals.csv, Team Totals.csv, Team Stats Per Game.csv, and All-Star Selections.csv. The remaining files consist of information regarding the player’s career, award selections, all-star teams, and more.

Since the data was already available in CSV format, I then downloaded the three files I chose and uploaded them into R to begin inspecting the values within. Before explaining my introductory findings, it is important to understand the makeup of the datasets.

## Data Dictionary:

The tables below provide an overview of the data files that will be used in my analysis.

Player Totals.csv

This file contains one record for each player for each season per team in the NBA between the years of 1947 and 2024. For example, if a player was traded during the season, they would have two records for that season: one for the team they were traded from and one for the team they were traded to. The columns consist of some personal information about each player and their seasonal totals in a variety of statistical categories. This dataset has 35 variables and 31,870 observations.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Column** | **Description** | **Data Type** | **Variable Type** |  | **Example** |
| player | Name of the player | Character | Nominal |  | LeBron James |
| player\_id | Unique ID | Numeric | Nominal |  | 3463 |
| birth\_year | Year player was born | Numeric | Ratio |  | 1984 |
| season | Year | Numeric | Interval |  | 2024 |
| seas\_id | Year ID | Numeric | Nominal |  | 31585 |
| lg | League | Character | Nominal |  | NBA |
| tm | Team | Character | Nominal |  | LAL |
| age | Player’s age | Numeric | Ratio |  | 39 |
| experience | Years in NBA | Numeric | Ratio |  | 21 |
| g | Games played | Numeric | Ratio |  | 71 |
| gs | Games started | Numeric | Ratio |  | 71 |
| mp | Minutes played | Numeric | Ratio |  | 2504 |
| fg | Field goals made | Numeric | Ratio |  | 685 |
| fga | Field goals attempted | Numeric | Ratio |  | 1269 |
| fg\_percent | Field goal percent | Numeric | Ratio |  | 0.54 |
| x3p | 3-point made | Numeric | Ratio |  | 149 |
| x3pa | 3-point attempted | Numeric | Ratio |  | 363 |
| x3p\_percent | 3-point percent | Numeric | Ratio |  | 0.41 |
| x2p | 2-point made | Numeric | Ratio |  | 536 |
| x2pa | 2-point attempted | Numeric | Ratio |  | 906 |
| x2p\_percent | 2-point percent | Numeric | Ratio |  | 0.592 |
| e\_fg\_percent.x | Efficient field goal percentage | Numeric | Ratio |  | 0.599 |
| ft | Free throw made | Numeric | Ratio |  | 303 |
| fta | Free throw attempt | Numeric | Ratio |  | 404 |
| fg\_percent | Field goal percent | Numeric | Ratio |  | 0.75 |
| orb | Offensive rebounds | Numeric | Ratio |  | 61 |
| drb | Defensive rebounds | Numeric | Ratio |  | 457 |
| trb | Total rebounds | Numeric | Ratio |  | 518 |
| ast | Assists | Numeric | Ratio |  | 589 |
| stl | Steals | Numeric | Ratio |  | 89 |
| blk | Blocks | Numeric | Ratio |  | 38 |
| tov | turnovers | Numeric | Ratio |  | 245 |
| pf | personal fouls | Numeric | Ratio |  | 78 |
| pts | points | Numeric | Ratio |  | 1822 |
|  |  |  |  |  |  |

Team Stats Per Game.csv

This file contains one record for each team for each season. The columns are made up of single game averages of a variety of statistics. These averages are created using the team totals which are the sum of all the team’s players’ statistics. This dataset has 31 variables and 1845 observations.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Columns** | **Description** | **Data Type** | **Variable Type** | **Example** |
| season | Year | Numeric | Interval | 2024 |
| lg | Leage | Character | Nominal | NBA |
| team | Team name | Character | Nominal | Miami Heat |
| abbreviation | 3 letter abbreviation | Character | Nominal | MIA |
| playoffs | True or False | Boolean | Categorical | TRUE |
| g | Games | Numeric | Ratio | 82 |
| mp\_per\_game | Minutes played per game | Numeric | Ratio | 240.1 |
| fg\_per\_game | Field goals per game | Numeric | Ratio | 39.2 |
| fga\_per\_game | Field goal attempts per game | Numeric | Ratio | 92.5 |
| fg\_percent | Field goal percent | Numeric | Ratio | 0.465 |
| x3p\_per\_game | 3-point made per game | Numeric | Ratio | 13.7 |
| x3pa\_per\_game | 3-point attempts per game | Numeric | Ratio | 37.7 |
| x3p\_percent | 3-point percent | Numeric | Ratio | 0.364 |
| x2p\_per\_game | 2-point made per game | Numeric | Ratio | 29.3 |
| x2pa\_per\_game | 2-point attempt per game | Numeric | Ratio | 54.8 |
| x2p\_percent | 2-point percent | Numeric | Ratio | 0.535 |
| ft\_per\_game | Free throws per game | Numeric | Ratio | 18.5 |
| fta\_per\_game | Free throw attempts per game | Numeric | Ratio | 23.2 |
| ft\_percent | Free throw percent | Numeric | Ratio | 0.797 |
| orb\_per\_game | Offensive rebounds per game | Numeric | Ratio | 12.5 |
| drb\_per\_game | Defensive rebounds per game | Numeric | Ratio | 32.2 |
| trb\_per\_game | Total rebounds per game | Numeric | Ratio | 44.7 |
| ast\_per\_game | Assists per game | Numeric | Ratio | 26.6 |
| stl\_per\_game | Steals per game | Numeric | Ratio | 7.5 |
| blk\_per\_game | Blocks per game | Numeric | Ratio | 4.5 |
| tov\_per\_game | Turnovers per game | Numeric | Ratio | 13.5 |
| pf\_per\_game | Personal fouls per game | Numeric | Ratio | 18.6 |
| pts\_per\_game | Points per game | Numeric | Ratio | 113.2 |
|  |  |  |  |  |

Team Totals.csv:

This file is similar to the Team Stats Per Game, except it is seasonal totals not per game averages. Additionally, it includes information on team performance during the respective season. This dataset has 31 variables and 1845 observations.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Column** | **Description** | | | **Data Type** | **Variable Type** | **Example** |
| lg | League | | | Character | Nominal | NBA |
| abbreviation | 3 letter abbreviation | | | Character | Nominal | PHX |
| team | Team name | | | Character | Nominal | Phoenix Suns |
| season | Year | | | Numeric | Interval | 2024 |
| playoffs | True or False | | | Boolean | Categorical | TRUE |
| age | Average age of players on team | | | Numeric | Ratio | 26.3 |
| w | Wins | | | Numeric | Ratio | 46 |
| l | Losses | | | Numeric | Ratio | 36 |
| pw | Pythagorean wins | | | Numeric | Ratio | 45 |
| pl | Pythagorean losses | | | Numeric | Ratio | 37 |
| mov | Margin of victory | | | Numeric | Ratio | -2.18 |
| sos | Strength of schedule | | | Numeric | Ratio | -0.2 |
| srs | Simple rating system | | | Numeric | Ratio | 2.16 |
| o\_rtg | Points scored per 100 possessions | | | Numeric | Ratio | 117.2 |
| d\_rtg | Points allowed per 100 possessions | | | Numeric | Ratio | 115.4 |
| n\_rtg | Net rating | | | Numeric | Ratio | 1.8 |
| pace | Estimate of possessions per 48 minutes | | | Numeric | Ratio | 100.1 |
| f\_tr | Free throw attempt rate | | | Numeric | Ratio | 0.251 |
| x3p\_ar | 3-point attempt rate |  |  | Numeric | Ratio | 0.408 |
| ts\_percent | True shooting percentage | | | Numeric | Ratio | 0.576 |
| e\_fg\_percent | Efficient field goal percentage | | | Numeric | Ratio | 0.539 |
| tov\_percent | Turnover percent | | | Numeric | Ratio | 11.6 |
| orb\_percent | Offensive rebound percent | | | Numeric | Ratio | 27.1 |
| ft\_fga | Free throws per field goal attempt | | | Numeric | Ratio | 0.200 |
| opp\_e\_fg\_percent | Opponent efficient field goal percentage | | | Numeric | Ratio | 0.572 |
| opp\_tov\_percent | Opponent turnover percentage | | | Numeric | Ratio | 12.4 |
| opp\_drb\_percent | Opponent defensive rebound percentage | | | Numeric | Ratio | 75.2 |
| opp\_ft\_fga | Opponent free throws per field goal attempt | | | Numeric | Ratio | 0.192 |
| attend | Total attendance | | | Numeric | Ratio | 696418 |
| attend\_g | Attendance by game | | | Numeric | Ratio | 16968 |
| arena | Name of arena team plays in | | | Character | Nominal | United Center |

All-Star Selections.csv:

This file contains the players selected as all-stars during each season.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Column** | **Description** | **Data Type** | **Variable Type** | **Example** |
| player | Name of all-star player | Character | Nominal | Lebron James |
| team | Conference player plays in | Character | Nominal | East |
| lg | League the player plays in | Character | Nominal | NBA |
| season | Year the player was selected as an all-star | Numeric | Interval | 2024 |
| replaced | Whether or not the player was replaced | Boolean | Categorical | TRUE |

# Lesson 6 Predictive Analytics System Group-Based Assignment

**Purpose:**

To layout the plan for preparing data for conducting predictive analytics.

**Tasks:**

1. After meeting with the instructor in week 5 the group will put together a plan for preparing the data for performing analytics.
2. Submit a report with details on data collection. Your data collection step should have been completed by now.
3. List all the steps for preparing the data for predictive modeling. I should have provided the directions in regard to completing the tasks for preparing the data during the week 5 meeting with the team.

## Data Cleaning:

Upon early analysis, I discovered a few inconsistencies in the datasets. In regard to all three, there was more than one league included, and I want to focus on the NBA specifically, as the others merged with the NBA to become what it is today. Each of the team specific datasets had additional problems I noticed. First, the amount of games varied throughout the years. This is a problem considering one of my goals is to predict win total using player statistics. Without a uniform number of games, this is like hitting a moving target. Additionally, there are rows that contain the league averages for each season, so these have null values in the win and loss columns, as they are not actual teams. This once again creates an issue for making predictions on wins.

Knowing the current season is 82 regular season games, I did some research to determine when this was implemented and any seasons that did not conform to this structure. I found the NBA moved to the 82-game schedule in 1967, and in the 1999 and 2012 seasons there were ‘lockouts’ that created a shorter season. The lockouts were due to disagreements between the players and the league preventing the season from beginning as planned. Seeing that the NBA has used the current schedule format for over 50 years now, I decided to eliminate all seasons prior to 1967, as well as 1999 and 2012. This allows for standardization of the season length across all datasets and creates a set target for predictions. Next, I eliminated all rows where the league was ABA instead of NBA and eliminated the league average rows from the team datasets. I created new datasets, allowing me to retain the original in case I want to revisit the league averages later in my analysis. In order to prepare for my analysis, I also created buckets in the player statistics dataset. I created buckets for points, assists, defensive rebounds, offensive rebounds, total rebounds, steals, and blocks. Points are in buckets by five, all other categories are by two. This helps to reduce the continuous nature of the dataset which will aide in predictions later on. To achieve this, I mainly used the mutate, seq, and cut functions. Through trial and error, I created a functioning workflow for the points bucket, then repeated with adjustments for the rest of the variables I wanted to segment.

Another area of interest for me was the minutes played by each player. Only the players who play a considerable amount of minutes will significantly impact the game, so it does not make sense to include players who barely got in the game, as they will not aid the analysis. As a result, I determined the 5 number summary of minutes played and decided to eliminate all values below the mean. This will also eliminate players who missed a significant amount of time due to injury, as they would have a disproportionate impact considering only partial availability. This will help to reduce the overall size of the data, leading to a dataset that is much more qualified for making predictions.

A graph of a number of minutes played

Description automatically generatedA graph of a number of minutes played

Description automatically generated

**A graph showing the time of a game

Description automatically generated with medium confidence**A graph of a bar chart

Description automatically generated with medium confidence

## Introductory Modeling:

After standardizing the datasets, I began putting together visualizations for both the player and team datasets. On the player side, I focused on the distribution of per game statistics using histograms, trying to determine the spread of the data. The results show that points per game and rebounds per game approximate a normal distribution skewed to the left, while the rest are heavily biased to the left. This is because lots of the players in the dataset are not major contributors on their team, meaning they play few minutes and record few statistics. I also looked into the per game averages based on position using boxplots. The results show that overall, each position has a very similar distribution of points per game. Point guards record more steals and assists than the other positions, while centers record more rebounds and blocks. Next, I investigated the correlation between different per game averages to determine if there were any notable relationships. Looking at the results, there is really one obvious takeaway. Average points per game has a correlation above 0.6 for average rebounds per game, average assists per game, and average steals per game. This implies that an individual who scores lots of points is more likely to record a higher number of points rebounds, assists, and steals, as well. This is not necessarily surprising, because if a player can score with ease, chances are they will be above average in other areas as well. That being said, it was interesting to see the data prove this point.

On the team side, I had already made a few scatterplots showing win totals over the years, so I wanted to explore some of the other features. I first examined the offensive and defensive rating and found both to follow a normal distribution. Next, I created a few scatterplots to show the trends of the league over the years: field goal attempts and field goals made, offensive rating and defensive rating, offensive rebounds and defensive rebounds, and finally steals and blocks. Taking each of the graphs into account, it becomes evident that the league has been trending away from defense. This is evident firstly through the offensive rating and defensive rating graph which shows both metrics trending up both all-time and currently. While it may seem like a good thing that defensive rating improved, defensive rating represents the amount of points a team gives up, showing that teams are both scoring and allowing more points. The steals and blocks graph echoes this sentiment. While blocks per game have been relatively consistent, steals per game have been slowly decreasing over time. Another interesting takeaway was the negative trend of offensive rebounds. Getting an offensive rebound can allow more shot attempts thus more points making this a desirable metric. From the 70s through the early 90s offensive rebounds per game remained pretty consistent, but since then, they have decreased by 1/3. One explanation for this could be the field goal graph, both attempts and makes steadily decreased from 1970 to the early 2000s, but for about the past 15 years, both have been increasing. Field goal attempts has much more drastic changes than field goals made, but the lines reflect each other showing the league has maintained a fairly constant shooting percentage.

Next, I wanted to learn more about the win totals of teams that made the playoffs, and if there seems to be a definitive threshold for making the playoffs year after year. To gain an understanding of the fluctuations over the years, I created two line graphs. The first shows the lowest win total each year of teams that made the playoffs and the second shows the average win total of all the teams that made the playoffs each year. Both graphs have a horizontal line displaying the average of the entire subset, showing whether years were above or below the average and how well the average represents the actual data. I also made a dataframe showing the average win total by playoff teams each year to show the exact value, as it can be a little hard to tell from graphs.

Looking at the results, I was able to determine a few things. First, the threshold to make the playoffs is around 40 wins- anything lower is very unlikely to make the playoffs. That being said, the average win total for playoff teams across the whole dataset was just under 49, showing that 40 wins does not guarantee a spot. Additionally, looking at the dataframe to examine the actual values, I was surprised to see the consistency of the averages. From 1989 to 2019, the average wins for playoff teams stayed between 48.1 and 52.3 wins. I find that to be a very tight margin for a 30-year timeframe, and while I can’t prove it, I believe the reason for the inconsistency in 2020 and 2021 is a result of COVID which upended the season and caused the NBA to play in what was called ‘the bubble’, an isolated campus of sorts at Disney World in Florida.

Looking at the results of my graphs and dataframe, I had a few thoughts. It seems to me that 40 wins is like the doorstep to the playoffs- sometimes you get in sometimes you don’t. On the other hand, if you reach 50 (roughly the average of all playoff teams) it’s basically a guarantee. Considering these early takeaways, I decided to create a couple basic models to see if they validated my thoughts. I first used a linear regression to determine the relationship between wins and playoff appearance. The results showed there was a strong association between number of wins and playoff appearance, but did not do a good job of using wins to predict playoffs. While the coefficients were certainly inaccurate, the p-values were incredibly low, showing a definite relationship between wins and playoff appearance. Confirming that there is a relationship between the two, I created a logistic regression to determine playoff likelihood by win total. Looking at the results, my thoughts from earlier were on the right track. The model predicts that winning 40 games leads to a 52.2% chance of making the playoffs, while winning 50 yields a 96.5% chance. Finally, I made a random forest model and a decision tree which drew almost the exact same conclusions. The random forest predicted that winning 41+ games always leads to making the playoffs, while 35 and below never makes the playoffs. The random forest made very definite predictions, likely due to the lack of features, leading me to prefer the predictions made by logistic regression. The decision tree doubled down on the predictions made by the random forest, identifying 41 wins as its branching criteria. The decision tree showed that winning at least 41 games leads to the playoffs 88% of the time, while winning less than 41 games leads to the playoffs 10% of the time. Considering the season is 82 games long, these early conclusions make it appear that winning greater than 50% of the games is a great indicator of playoff appearance.

## Descriptive Statistics Visualizations:

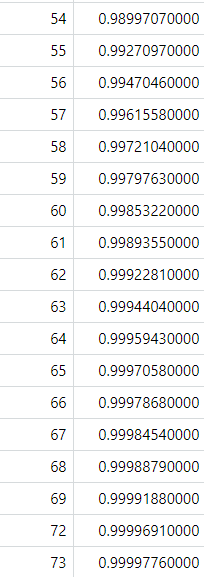
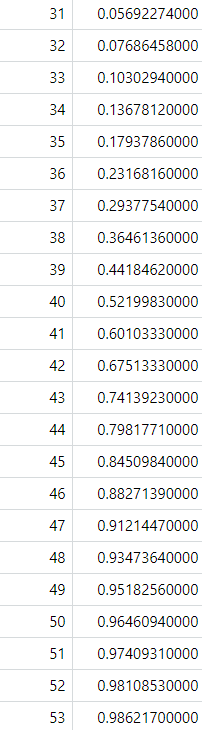
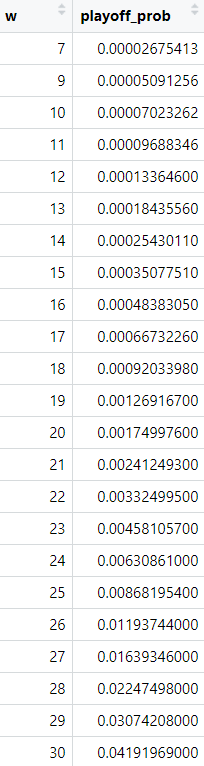
Lowest Win Playoff Qualifiers: Average Win Total of All Playoff Qualifiers:

A graph with blue and red lines

Description automatically generatedA graph showing the average win

Description automatically generated

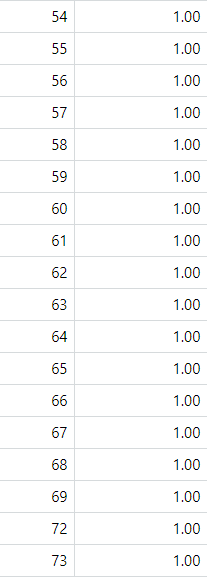
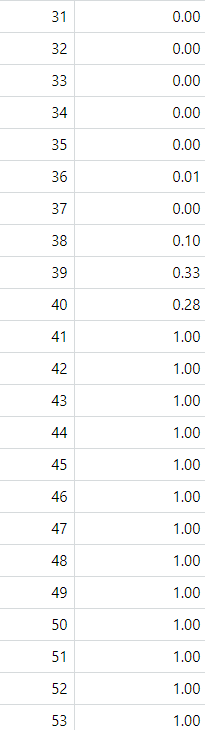
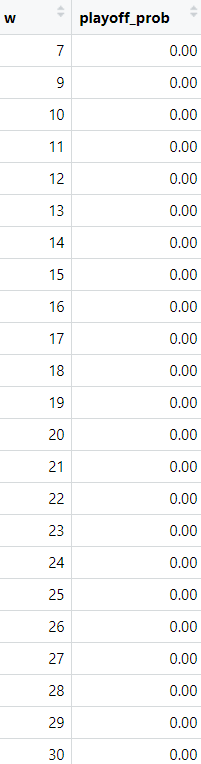
Linear Regression Summary: Logistic Regression Results:



A screenshot of a computer code

Description automatically generated

Random Forest Results:



Average Playoff Team Wins 1989-2019:

A number of years and numbers

Description automatically generated with medium confidence

Decision Tree:

A diagram of a graph

Description automatically generated with medium confidence

Next Steps:

Based on the takeaways from my initial modeling, it is evident that wins have a strong impact on playoff appearances. Additionally, my analysis has provided lots of insight on playoff likelihood based on win total, so a logical next step will be to determine which statistics can be used to predict win total. I also want to expand my analysis to include more variables as predictors. This will allow me to determine which team statistics have a high impact on wins, thus a high impact on playoff appearance.

I think it will be important to drill down into which team statistics have a strong impact on wins, as this will clue me in on areas to focus on for player statistics. I am intending on sticking to the schedule created during previously, as I am now just about ready to create my warehouses having completed cleaning and introductory modeling.

# Lesson 8 Predictive Analytics System Group-Based Assignment

**Purpose:**

To layout the plan for cleaning the data for conducting predictive analytics.

**Tasks:**

1. After meeting with the instructor in week 7 the team will complete the steps for data preparation.
2. List all the steps for variable (feature) selection and variable transformation. I should have provided the directions in regard to completing the tasks for variable selection and transformation during the week 7 meeting with the team.

Data Preparation:

The final step of data preparation was to store the datasets in their respective data warehouses. Each warehouse had been created during the previous weeks, ensuring the field names and data types matched. First, I saved the final datasets from R into xlsx files. From there, I imported them into Knime, where I created a workflow for each dataset. Each workflow first isolates the fields for the dimension tables, creates a unique key, and then uploads the data to the warehouses in DBeaver. Next, the foreign keys are joined to the fact table, which is then uploaded into DBeaver. This process was followed for both datasets, with the only difference being the team warehouse has two dimension tables and the player warehouse has three. Below is one of the workflows used:

A diagram of a diagram

Description automatically generated with medium confidence

A screenshot of a computer

Description automatically generatedIn DBeaver, I used the table creation and column add buttons rather than writing queries:

A screenshot of a computer

Description automatically generated

## Variable Transformation:

Both the player and team datasets started with about 50 fields. As a result, it would be quite tedious to inspect the distribution of each field, check for skewness, and then apply individual transformations. In order to reduce the overall time taken to normalize the data, I attempted to use the Box-Cox method of normalization. I started with the team dataset, which posed a few challenges, as some of the columns contain negative values, which the Box-Cox transformation cannot handle. An example is a field like net rating: offensive rating minus defensive rating. This can easily lead to values below 0, as long as the defensive rating is higher than offensive. This would mean the team in question allows more points than they score on average. I want to retain these values in my feature selection, so I had to move on from Box-Cox. This led me to do some research into other options, and I discovered the Yeo-Johnson method which draws from the Box-Cox method, except it can handle negative values. To conduct the transformation, I first removed the response variables from the selection and identified the strictly numeric columns. Then, I wrote a loop to apply the Yeo-Johnson transformation to all columns iteratively, which saved the results to a copied version of the data used for the transformation. Using a loop like this drastically reduces the time taken for transformation, as the whole dataset is effectively transformed in one shot.

While it was a little tedious, the process worked out well, so I decided to apply the same process to the player data. The player data does not contain negative values, but I decided to stick with Yeo-Johnson over Box-Cox for consistency purposes. Once again, I first removed the response fields and the nonnumeric fields, then looped through the dataset applying the transformation. The transformation was effective once again, so having transformed and normalized both datasets, I moved on to conducting feature selection.

## Feature Selection:

A screen shot of a computer code

Description automatically generatedThe team dataset began with 48 fields in total, two being the response: wins and playoffs. While this is not an exorbitant amount, it is a little high, so my goal with Boruta was to decrease the features while maintaining those with high importance. While conducting the Yeo-Johnson transformation, all fields had to be numeric, so I was able to reduce the number of fields to 41 right away. First, I performed Boruta on the transformed data using wins as the response variable. Then I conducted Boruta on the transformed data using playoff appearance as the response, and finally I used losses as the response. While I am really only concerned with the best predictors for wins and playoffs, since losses directly oppose wins, I was curious what the overlap would be. Each run of Boruta returned the all the fields from the data.

Since the results of the analysis were consistent for both fields I am looking to predict, I believe these features are important, but I wanted to pursue further analysis. I created a correlation matrix to examine the relationships between the variables and check for multicollinearity between the confirmed features. I found a few fields with very high correlation, greater than 0.9, so I had to examine the relationship and see if the fields are derived from each other or communicate the same information. One of the fields, mov (margin of victory), had a correlation of 0.99 with n\_rtg (net rating), these effectively communicate the same thing, so mov was eliminated as n\_rtg communicates the information in a better fashion, being derived from offensive and defensive rating. Pace was highly correlated with all ‘shooting’ metrics (2-point attempts per game, 2 point makes per game, 3-point attempts per game, 3 pt makes per game, etc). As a result, pace was removed, as well. Lots of the ‘shooting’ metrics were highly correlated with each other both in the positive and negative direction. For example, more 3-point attempts yields more 3-point makes and less 2-point attempts. These fields were retained, as they communicate different information, and it will be important to see how both attempts and makes effect wins. Finally, the srs (Simple Rating System) field was removed. Srs had a correlation of over 0.99 with n\_rtg and srs, as it is a combination of these metrics. Adding this reduction to the results from Boruta, I was able to bring the total down to 34 predictors. I plan to experiment with different fields in the models, seeing how offensive and defensive metrics specifically impact the response variables, in addition to modeling with all retained fields. Overall, I am happy to have reduced the predictors by over 10 from 46 to 34. This represents a 26% reduction overall features. Eliminating the irrelevant features will allow the models created to provide more robust and reliable predictions, as the “noise” has largely been removed. Having completed and examined the important features in the team dataset, I repeated the process with the player dataset.

The facts used as predictors from the player dataset number 46, slightly below the initial number of predictors from the team dataset. After removing numeric columns and response for Yeo-Johnson, 42 predictors remained. Once again, I applied Boruta to the transformed dataset, using both wins and playoffs as the response variable in separate evaluations. The Boruta for both wins and playoffs only identified one field that was not confirmed, which for me was not acceptable. I increased the number of iterations a few times, but the results did not change.

A screenshot of a computer program

Description automatically generated

As a result, I decided to pursue the correlation matrix again. Looking at the results, I realized there were fields for total stats, per game averages, and stats buckets. As a result, I decided to remove the totals, because the per game averages and buckets are much more informative considering the totals fluctuate vastly based on games played. This allowed me to remove the pts, ast, trb, orb, drb, stl, and blk fields. Additionally, I decided to remove 3-point attempts, 2-point attempts, and free throw attempts. 3-point makes, 2-point makes, free throw makes, and the percentages will be retained. Makes and percentage are more important than attempts for players, so I do not want to include these in my analysis. Making these changes, I was able to reduce the dataset by an additional 10 fields, trimming the total down to 30 fields.

On the whole, I conducted feature selection on both datasets using a combination approach. First, I applied the Boruta algorithm using two different response variables to determine the significant features. Then, I used a correlation matrix to inspect the confirmed features and remove features based on multicollinearity. Finally, I scrutinized the data to ensure the same information is not being conveyed in more than one field (for example mov and n\_rtg discussed above). Through this involved process, I was able to reduce the predictors in both datasets by over 25%.

Next Steps:

Having stored the data and conducted feature selection, I am now fully ready to begin modeling. Using the retained features, I now know the features to query for my models from my warehouses. Additionally, I was able to determine the most important features from each dataset. Next, I plan to model using all the retained features, then refine my models to subsets of the features, adding and removing features iteratively. A few models I plan to employ are logistic regression, decision tree, random forest, and neural networks with different architectures. I will likely use more than those listed, as well. Considering I created some early models using wins to predict playoffs, I will reflect and compare the results to the more advanced models which will use the predictors to predict both wins and playoffs separately.

# Lesson 10 Predictive Analytics System Group-Based Assignment

**Purpose:**

To detail the variable selection and Transformation steps

**Tasks:**

1. After meeting with the instructor in week 9 the team will complete the steps for variable selection and Transformation.
2. List all the steps for predictive analytics modeling. I should have provided the directions in regard to completing the tasks for predictive analytics modeling during the week 9 meeting with the team.

## Variable Selection:

In the previous milestone, I completed the majority of the variable selection and transformation. I applied the Yeo-Johnson transformation on each dataset in order to normalize the data. Then I used Boruta for variable selection, however only a small number of features were identified. In order to reduce the features further, I used a correlation matrix for each dataset and identified derived attributes and looked for multicollinearity. This allowed me to eliminate a few more features prior to modeling. For further details, please refer to section above where I go into further detail on the Yeo-Johnson process, Boruta, and the correlation analysis. Additionally, that section discusses the features that were eliminated from each dataset and the reason why.

## Transformation:

In order to begin modeling, I thought it would be important to transform the datasets into a few subsets for more nuanced analysis, as well as to compare the results to models which used the whole datasets. First, I identified the subsets I wanted to create based on the remaining features. I decided to begin with two subsets from the player dataset, player average statistics and player bucket statistics. For the average and bucket datasets, I removed the bucket fields from the average dataset, and the average fields from the bucket dataset. For the team dataset, I created two subsets, one for team offensive statistics and one for team defensive statistics. To create these subsets, I used the select function from Dplyr and removed each unwanted field using the column name. Having created five total subsets, I moved to the final steps before modeling, splitting the data.

First, I set a seed, then I identified the response variable, wins, and decided on a 75:25 split of training to test data. Then I applied the split to the datasets and saved the real results for wins into a separate variable that will be used for validation. Having completed variable selection, data transformation, and splitting the data, I completed all pre-modeling steps and began creating my first models.

## Modeling:

Previously I decided to begin with linear regression, random forest, and decision tree. I started by using each of these three techniques on the whole player and team datasets. I trained the models, then used the test datasets to make predictions. Then, I used mean average error and mean squared error to evaluate all models. I also used the r2 score and adjusted r2 for the linear regressions. Comparing the models, overall, the team dataset did a much better job of accurately predicting wins. This can be seen looking at the error metrics, as the mean average error and mean squared error are lower for all three of the team models. Also, the r2 and adjusted r2 for the linear regressions show substantially better performance from the model using the team data. The player linear regression summary showed the model was only able to explain less than 15% of the variability in wins, but the team linear regression explained greater than 90% of the variability. This was true for both the r2 and adjusted r2 scores. I also used a scatterplot to examine the residuals of the team linear regression, the results seem to be evenly distributed on both sides of 0, showing that the model’s predictions seem to be reasonable.

Having begun modeling with the complete datasets, I have a few early takeaways. First, the obvious conclusion is that the team dataset is much more insightful into predicting wins. This is clearly illustrated through the error metrics and the substantial difference compared to the player models error metrics. Next, I will not be continuing with the decision tree model, as the random forest operates on the same framework and achieved better results. My final takeaway is that the player dataset requires further refinement, as the results from this stage of modeling were extremely unsatisfactory. I will also explore hyperparameter tuning as I move forward with the modeling process.

Next Steps:

After creating the subset models, I can compare using the error metrics used above and determine right away if the subsets or full datasets are more informative. I also want to continue to tweak the datasets, adding and removing selected features iteratively to see the performance is affected. This may be especially impactful if it helps to increase the accuracy of the player models. One idea I am considering is creating a binary threshold for wins, then using a logistic regression. Considering I previously made predictions on playoff appearance based on win total, I can tweak the threshold based on percent likelihood of playoff appearance.

# Lesson 12 Predictive Analytics System Group-Based Assignment

**Purpose:**

To detail the modelling, evaluation and validation of the Predictive Analytics system

**Tasks:**

1. After meeting with the instructor in week 11 the team will complete the steps for modelling, evaluating and validation of the Predictive Analytics system.
2. Submit the report detailing about the modelling, evaluating and validation of the Predictive Analytics system.

## Parameter Testing:

In order to achieve the best possible results during modeling, I tested multiple different values for the mtry, number of trees, and node size. I conducted this process for both the player and team datasets. In each case, I started by determining the ideal mtry value, then kept the mtry constant and tested the number of trees. After determining each of those, I kept them the same and tested the ideal node size. In the player dataset, the values identified were an mtry of 4, 200 trees, and a node size of 5. In the team dataset, the values identified were an mtry of 7, 100 trees, and a node size of 10. Moving forward, all random forest models were conducted using these hyper parameters. Next, I determined the ideal number of clusters for my cluster analysis.

Since clustering is unsupervised, it is important to use the best number of clusters. In some of the scenarios, I clustered using subsets of the data (for example all-stars and non-all-stars) and I wanted to know the similarities between members, so I used one cluster. In other cases where I was looking for differences in the whole dataset (for example, clustering all players into all-star groups) I used the silhouette score and the within cluster separation score to determine the number of clusters. In each case, I tested values from 2 to 10. In the case of the best and worst players, two clusters was determined as the best option, while in the all player comparison, four clusters was identified.

Team Hyper parameters

A table with numbers and a yellow line

Description automatically generated A yellow line with black text

Description automatically generated A yellow line with numbers

Description automatically generated with medium confidence

Player Hyper parameters:

A yellow and black numbers

Description automatically generated with medium confidenceA yellow line with numbers

Description automatically generatedA yellow and black text on a white background

Description automatically generated

All players cluster count:

A graph with numbers and lines

Description automatically generatedA graph with a line

Description automatically generated

Worst players cluster count Best players cluster count

A graph with a line

Description automatically generatedA graph with a line

Description automatically generated

## 

## Modeling:

During the last milestone, I was able to relatively low mean average error when predicting wins as a continuous response variable using the team data. This result was not consistent when using the player data, so over the past two weeks I focused extensively on improving the accuracy of the player models and broadening the scope of my analysis to complement the team data instead of attempting to achieve matching results.

In my first attempts to increase accuracy, I used the subsets created previously to train my models and compare the results. While the models now explained more of the variability within the response, the error only decreased marginally. I then had an idea to boost the accuracy through aggregation. I created a loop first create subsets for each team for each season, then to iterate through that list and aggregate the statistics for the players who played the most minutes into a single record, then form a new dataset with the aggregated observations. I first used the three players who played the most minutes, then increased the number to five. The results showed a large increase in the adjusted r2 score, but no reduction in mean average error. These findings were almost identical when looking at the average and bucket subsets, as well. Seeing that the aggregation seemingly caused an impactful improvement, so I came up with another approach; I changed the type of the response variable first to binary, then to categorical.

I thought maybe the continuous nature of the response was holding the models back, as there were too many different outcomes to attempt to predict. For the binary conversion, I set a threshold at 42 which my previous analysis showed to be a win total indicating the low-end win total for playoff appearance. I also modeled using a 50-win threshold, as this indicated borderline guaranteed playoff appearance. For the categorical version, I repeated the process, this time defining low, medium, and high thresholds based on the first quartile, median, and third quartile values. I used logistic regression and random forest models on each dataset allowing for uniform comparison of results. For the logistic regression, I used mean average error and r2, and for the random forest I use the confusion matrix. Also, before modeling, I normalized the classes. In the binary version, the 0 class had over three times the samples of the 1 class, and in the categorical model, the medium classification had more observations than the other two classes combined. In order to make the predictions more reliable, I set the total number of observations for each group equal to the total number of observations for the minority group. The 42-win binary model was able to achieve over 90% accuracy in classification, while the 50-win model achieved an 83.58% accuracy. Additionally, the categorical model was able to achieve over 75% accuracy. Clearly, all of these models performed much better than the models operating using the continuous response variable. The 42-win binary model performed well at positive classifications, displaying a 98.6% accuracy while the 50-win model displayed a 91.8% accuracy. The categorical model performed well with the low and high win classes but struggled a bit with the medium class.

Based on these results, I would say defining a binary threshold to make predictions against is crucial for this dataset. I chose my thresholds based on my earlier analysis predicting playoff appearance based on win total and on the average win total for playoff teams historically. Considering the correct identifications, I am satisfied with the results, and I think they show the importance of defining a target and mitigating class imbalance. My earlier analysis, I could not explain much of the variability in wins or accurately predict win total using player statistics. After making these modifications, the results were quite the opposite. Having thoroughly explored the impact of player statistics on wins, I decided to shift my focus to descriptive statistics. It seems fairly obvious that teams with more players designated as ‘all-stars’ would win more, thus making the playoffs more often. As a result, since I was unable to effectively determine player statistics that lead to wins, I wanted to determine player statistics that lead to all star selections. First, I decided to confirm my thinking regarding all stars and playoffs. To do this, I needed to add a column to the player dataset indicating whether they were an all star for the respective season. After successfully adding the field, I aggregated the field based on team and season IDs, then appended the resulting column to the team data as a count of all star players from their team for each season. Finally, I printed the count of TRUE and FALSE observations for playoff appearance for each of the values represented in the all-star count. What I found is that teams with two or more all-stars make the playoffs more than 95% of the time. Additionally, teams with three of more made the playoffs 100% of the time. There were 55 instances of a team having three all stars and 8 instances where a team had four, and in each case the team made the playoffs. On the whole, even if a team had just one or more all-star, they made the playoffs more than 75% of the time. Also, if a team had zero all-stars, there is a 78% chance they do not make the playoffs.

Seeing the clear impact of all-star players on playoff appearance, I attempted to predict all-star selection based on player statistics. Currently, the selection process is relatively arbitrary, for example there is no defined guidelines like x points per game, y assists per game, z rebounds per game means you are automatically an all-star. As a result, I think it would be beneficial and at the least informative to determine the statistical commonalities between players who are selected all-stars. To begin my analysis, I ran a logistic regression on the average and bucket subsets and was able to achieve above 97% accuracy despite class imbalance. The class imbalance was about 90:10 of non-all star to all-star. To mitigate this, I again set the maximum samples equal to the number of samples from the minority class. Repeating the modeling, I was able to achieve a 92% accuracy and less mistakes at correctly the positive class.

In order to drill down further into the metrics contributing to all-star selection, I iteratively removed features. Ultimately, I was able to exceed 93% accuracy overall and 97% accuracy correctly classifying all-stars using only 12 features. The final features I used each showed a decrease in accuracy upon removal, so it seems these 12 are most indicative of all-star selection:

1. pts\_per\_game: average points the player records in a game
2. trb\_per\_game: average total rebounds a player records in a game
3. orb\_per\_game: average offensive rebounds a player records in a game
4. ast\_per\_per\_game: average assists a player records in a game
5. stl\_per\_game: average steals a player records in a game
6. blk\_per\_game: average blocks a player records in a game
7. fg\_percent: percent of both two- and three-point shots a player makes
8. ft\_percent: percent of free throws a player makes
9. x3p\_percent: percent of three point shots a player makes
10. e\_fg\_percent: effective field goal percent: percent of two- and three-point shots a player

makes, adjusted for the fact that three point shots are worth more

1. experience: how many years the player has been in the NBA
2. age: age of the player

Having seen the data clearly indicates a predictable difference between the classes, I moved on to a cluster analysis. My goal here was to determine the threshold so to speak of certain statistics to be deemed an all-star. What my introductory clustering showed was a substantial difference between the cluster centers of the two groups. The cluster centers indicate that all-stars record on average about twice as much of each of the five main stat groups besides blocks. When looking at the cluster distributions by stat, there are three stats where the differences becomes clear: points, assists, and steals. The distribution of points almost has no overlap between the two classes. The results show these are likely to be key metrics in all star selection, which speaks to something interesting. Recently offense has been taking over the game, so the displayer importance of steals along with points and assists shows that defense is still considered and emphasized. Based on these findings, I would have to say building a team around a guard who can become an all-star seems to be the most reasonable path to success. The distributions of points, assists, and steals by position shown earlier displayed that point guards lead the way in both steals and assists, while shooting guards lead the way in points and are second in assists and steals. This shows that these players likely become all stars more often than the other positions. To confirm this, I created a table showing the count of all-stars by position historically. I was a little surprised to see SF had the most, followed by SG, PF, C, and PG. While SG came in second, I definitely expected PG to rank higher given the descriptive statistics showing the distribution of average assists and steals among PGs has a higher range and a higher mean than the other positions.

Considering the previous findings regarding number of all-stars on a team and playoff appearance, it becomes evident that pursuing all-star guards in trades, or drafting quality guards who can develop into all-stars is crucial to making the playoffs. This assessment may even be refined to strictly SGs, as they have a much higher number of historical all-star selections than PGs. If a team with one or more all-stars makes the playoffs more than 75% of the time, it is beyond evident the importance of all-stars to winning games and making the playoffs. Points, assists, and steals are the stats where all-stars significantly outpace non-all-stars. The distribution of these stats has the highest upper bounds and average in the PG and SG positions. SGs have the second highest all-star selections of the five positions historically. Combining all of these facts, it seems very advantageous to ensure there is an all-star SG on your team. Additionally, taking into account the scatterplot of historic all-star selections by position, the count for PGs appears to be increasing steeply in recent years showing their value may be understated in the historic count. This could signal a coming change in which the importance of PGs is appreciated, making it important for teams to invest in all-star PGs, as well.

Next steps:

Having reached fairly strong conclusions regarding player statistics, all-star selections, and wins, my next steps are to tie these results in with the results I achieved with my team dataset in order to create a comprehensive discussion on how the pieces fit together. The analysis on all-stars represents a major breakthrough, as they are definitively linked to team success providing the link I have been looking for between individual and team success.

# Lesson 13 Predictive Analytics System Group-Based Assignment

**Purpose:**

To complete all the modules of the Predictive Analytics System

**Tasks:**

1. Submit the final project report of your team’s Predictive Analytics system by the end of week 13. In this week the team is supposed to complete the visualization of the Predictive Analytics system.
2. The team should set up an appointment with me to discuss about the project report feedback before the final project report submission. The team meeting has to be scheduled as early as Friday and no later than Sunday.
3. More instructions about the final submission of the project report, datasets, scripts etc. will be provided by the instructor over the e-mail.

## Research questions:

* **How do player statistics from the BEST teams compare to the WORST teams?**
* Clustering results using one cluster for players from best and worst teams showed the players from the best teams outperformed the players from the worst teams across the board.
* Players from the best teams shoot considerably more shots, both field goals and free throws (1051 vs 944 for FGs and 358 vs 290 for FTs) suggesting better teams play at a faster pace, emphasizing offense and more shot attempts.
* Coaches may need to adjust strategies to account for these findings- encourage shooting more shots per game.
* **How many wins “guarantees” a playoff berth?**
* Historical data shows 42 wins to be a good indicator of playoff appearance (586/642), and 50 wins is almost a certainty (322/327).
* The decision tree also identified 42 as a viable threshold.
* From 1989-2019 the average win total of playoff teams remained between 48.1 and 52.3, once again showing 50 wins to be very indicative of playoff appearance.
* **Can player statistics be used to predict a playoff berth (seasonal averages)?**
  + 5 players with most minutes played
    - Yes, using both the 42- and 50-win thresholds, the models using the top 5 players by minutes played were able to achieve greater than 80% accuracy. This was the highest accuracy achieved by the player models, showing that by eliminating the less impactful players, the data becomes more conclusive and reliable.
  + 5 players with most minutes played POSITIONALLY (PG, SG, SF, PF, C)
    - Accuracy was lower than the results from just the 5 players who played the most minutes. Enforcing the positional criteria had a negative impact on the accuracy, showing the players who played the most overall were more insightful than the players who played the most by position.
* **Do advanced metrics provide better predictions than the standard 5 categories (points, rebounds, assists, steals, blocks)?**
  + Yes and no. Using the team datasets, the inclusion of the advanced metrics yielded marginally better accuracy (93.75% vs 92.36% and 91.89% vs 91.22% for the 42- and 50-win thresholds respectively) and for the player dataset, better results were achieved using the “simple” statistics across the board modeling against the 50-win threshold.
  + The results show that the “simple” statistics are driving the predictions, considering the consistency and in some cases improvement upon the removal of the advanced statistics. While the advanced statistics are insightful in other ways, it is clear they are not as indicative of wins as the “simple” statistics.
* **Are TEAM statistics more indicative than PLAYER statistics?**
  + Yes, in each test (both thresholds and categorical) the team models performed better than the player models. The team models performing consistently better proves that team statistics are more indicative of wins than player statistics, no matter how the player statistics are prepared.
* **How much do ALL-STARS influence playoff appearance?**
  + A great deal. Historically, having one or more all-star on a team yields playoff appearance 75% of the time, while two or more yields playoff appearance 95% of the time.

## Final Conclusions & Recommendations:

Based on historical data, it is clear that 50 wins should be a target win total for all teams, while 42 wins can be used as a less ambitious target. Additionally, when predicting the win total, team statistics clearly yield the best results and when using player statistics, only the most impactful players should be included. These players can be determined by the number of minutes they played throughout the season. Additionally, when predicting the win total, the best results are achieved using per game averages and shooting percentages rather than advanced statistics.

My findings also show that offense is very highly valued, and the better teams attempt more shots, leading to more points, even if the shooting percentages are the same. Additionally, all-stars have an obvious and dramatic impact on playoff appearance and therefore win total. As a result, the importance of having at least one all-star player on your team cannot be understated. In order to maximize win total, which will increase the likelihood of making the playoffs, the main focus should be obtaining an all-star player who can both score and assist. Considering the positions that are the best at this are shooting guards and point guards, my recommendation is to attempt to acquire a point guard or shooting guard of all-star caliber. This will have a cascade effect, as making the playoffs will yield an increase in revenue and the worth of the franchise. In order to both become a better team and generate more revenue; the first priority must be adding an all-star player to the roster.

**How do player statistics from the BEST teams compare to the WORST teams?**

To compare player statistics from the best and worst teams, I first had to create groupings by team to centralize the players. I did this using the team and season IDs. From there, I created two subsets, one for players whose teams won less than the first quartile value for wins, and one for players whose teams won more than the third quartile value for wins. While this certainly excludes a large chunk of data from the middle, comparing opposite sides of the spectrum like this allows for the highest possible information gain. Initially, I looked at the five number summary for the five main stats. The results showed the best teams had a slight edge across the board, but not as much as I expected. As a result, I refined the data to only include the five players who played the most minutes for each team. After adjusting, the differences became more significant. Next, I did a cluster analysis on both datasets. I used the silhouette score to determine the best number of clusters for each dataset and the results showed that two clusters would be the best. Looking at the cluster centers for the first cluster, the players from the best teams vastly outperformed the players from the worst teams. Within the second cluster, the players from the worst teams had the edge in points per game, but the rest of the fields were about equal. When using only one cluster, the best teams had a significant advantage. Based on the clustering results, it is clear that the players on the best teams perform much better than the players on the worst teams. The main difference is in points, rebounds, and assists- the players from the best teams record much more of those statistics. The values for steals and blocks were fairly similar between the players from both groups, showing that offensive metrics are driving the difference between the two groups. Finally, I created a bar chart to visualize the cluster center comparisons, which makes the distinction very clear.

Outside of the five main stats, there were a few other very notable differences. One that really caught my attention was the difference in shot attempts. First, the players from the best teams averaged 1051 field goal attempts, while the players from the worst teams averaged only 944 field goal attempts. This extended to free throw attempts, as well. The average for the players from the best teams was 358, while the average for the players from the worst teams was only 290.

A white rectangular box with black numbers

Description automatically generated

This partially explains the gap in points per game, because the shooting percentages were not that different. To me this indicates that the better teams likely play at a higher pace, focusing on shooting as much as possible and trying to draw fouls rather than emphasizing defense. Considering the league has been becoming more offense based, this is not surprising on the whole, but the disparity in shot attempts is definitely surprising. Based on these findings, my conclusion is that the worst teams are not necessarily the worst due to player performance but likely due to strategy, as they are not attempting anywhere near as many attempts than their counterparts on the best teams.

**How many wins “guarantees” a playoff berth:**

In order to determine how many wins a team needs to achieve in order to make the playoffs, it is important to first define some context. In the NBA season, there are two conferences, east and west, and 82 total regular season games. There are thirty total NBA teams split between the two conferences, and ultimately, eight teams from each conference make the playoffs. This means just over half the total number of teams advance to the postseason. As discussed previously, there is an extreme positive to making the playoffs, both immediate financial success and in an overall increase in relevance leading to more exposure which through continued success can make the organization a mainstay in NBA conversations.

A great example of this is the Golden State Warriors from about 2014 to today. Prior to that, throughout the early 2000s, they had little success, then suddenly their team shot to the top of the league, reaching the NBA finals in five consecutive seasons and winning three of the five. This continuous domination led to their team becoming a household name, as well as a global attraction. It is not uncommon to see people in Steph Curry jerseys (the player who led their team during those years) all over the world. Additionally, in 2013, just prior to their massive success, they were the 8th most valuable team with Forbes estimating their worth at $555 million2. Today, the Warriors are the most valuable team in the NBA with a Forbes valuation of $8.8 billion1. This quick case study illustrates the undeniable power winning games (and championships) has at creating worldwide relevance and generating massive amounts of money. Additionally, some context is required for both sides of the spectrum. The worst team to make the playoffs in my analysis had a record of 30-52 in 1986, while the best record was 73-9 in 2016 (achieved by the Warriors during the stretch described above). Clearly, while making the playoffs, vast variation in regular season win total is possible. As a result, my goal is to use historical data and machine learning to determine a reliable threshold for the total number of regular seasons wins needed to reach the playoffs.

Immediately, I checked the win distribution for playoff teams, which showed in addition to the values mentioned above, the average win total was 49.64, just below 50. Seeing this, I then ran a count of the results for teams that achieved 50+ wins. The results show that historically, teams winning greater than 50 games have made the playoffs 322/329 times, showing this threshold is very reliable for playoff appearance. Next, I made a few models using solely the number of wins to predict playoff appearance. I used logistic regression and random forest, and their results were similar. The decision tree identified 42 wins as a good indicator, while the logistic regression showed a sharp increase in playoff likelihood with win totals greater than 42. As a result, I decided to investigate the historic playoff appearance of teams achieving at least 42 wins. The results showed that 586 of 642 teams achieving 42+ wins made the playoffs. While this is less compelling than the results of 50 wins, it is still greater than 90%. Finally, I determined the average win total for all playoff teams for each season, and the results were striking. From 1989 to 2019, the average win total for all playoff teams stayed between 48.1 and 52.3, with most of the values being very close to 50. I believe this further validates my earlier findings that 50 regular season wins basically guarantees a playoff berth. Also, based on my previous findings, it appears 42 wins may be a dependable low-end threshold, a win total that likely leads to a playoff berth, and each subsequent win past 42 further boosts the likelihood.

**Do advanced metrics provide better predictions than more “simple” metrics:**

When comparing the prediction accuracy with advanced metrics and “simple” metrics, the team models performed marginally better with the advanced, but the player models performed better with the “simple”.

The way I was able to test this was by iteratively removing features on my models and testing the accuracy. What I mean by this specifically, is that from the player and team datasets, I created subsets retaining only per game averages and shooting percentages. For the team dataset, I also retained net rating. This reduced the team dataset from 37 predictors to 23, and the player dataset from 27 predictors to 15. Many of the advanced metrics were also highly correlated with other fields, as they were derived statistics based on combinations of other statistics. Additionally, I created my own fields which I called “buckets” for each of the five main stats in order to reduce the continuous nature of the average fields. I thought that by somewhat standardizing these fields, it would allow for more accurate predictions, but ultimately the results were very similar with the results using the average data slightly outperforming the bucket data. Considering this were user created fields, based on distinctions I created, in the rest of my analysis, while I continued modeling using the buckets, I relied more on the actual per game averages, as my distinctions could be considered arbitrary and therefore insignificant.

Looking at the results from the team dataset, the models using advanced metrics were able to achieve 93.75% and 96.15% accuracy for the 42- and 50-win thresholds respectively. Modeling with the “simple” subset 93.06% and 93.75% accuracy for the 42- and 50-win thresholds respectively. Considering the large reduction in features compared to the marginal decrease in accuracy, it becomes clear that the “simple” metrics are driving the predictions much more than the advanced metrics. Additionally, considering the models were each able to exceed 90% accuracy with only the “simple” metrics, it shows they allow for very robust and reliable predictions.

In regard to the player averages dataset, the initial models for both buckets and averages achieved an 81% accuracy predicting against the 42-win threshold, and 76% and 78% accuracy predicting against the 50-win threshold. Upon modeling again using the refined dataset predicting against the 42-win threshold, the models achieved 76% and 79% accuracy for the average and bucket datasets respectively. Finally, predicting against the 50-win threshold, the models each achieved 80% accuracy. Comparing the results, there is a marginal decrease in accuracy using the “simple” metrics to predict against the 42-win threshold. On the contrary, the results improved when predicting against the 50-win threshold. The results were very similar to the team data counterpart, except the models outperformed the advanced metrics using the 50-win threshold. Once again, the marginal reduction in accuracy suggests the “simple” metrics are driving the predictions, and the results predicting against the 50-win threshold seem to confirm this finding. On the whole, the consistency of the results both with and without the advanced metrics shows that they do not enhance the predictions accuracy in a significant manner. As a result, I would be inclined to suggest modeling using the “simple” metrics, as they are easy to interpret and yield roughly the same results.

**Are TEAM statistics more indicative than PLAYER statistics:**

My analysis proved team statistics to be more valuable than player statistics when predicting wins as a continuous response, but when a binary threshold is defined, player statistics are more valuable to predicting wins.

I reached this conclusion because the team dataset was able to achieve lower error metrics than the player dataset in a variety of experiments. First, I attempted to predict the win total using all retained fields from each dataset. I used a few different machine learning algorithms like linear regression, random forest, and neural network. Working with the team dataset, the models were able to achieve a mean average error of around 2, while the player dataset had a mean average error of above 7. Additionally, when investigating the regression results, the adjusted r2 for the team dataset was greater than 0.9, while the adjusted r2 for the player dataset was about 0.15. The residuals for the team dataset showed the model was a good fit, as well.

Seeing these results, I was pleased with the team modeling results, but fairly disappointed with the results from the player data. Due to the really low adjusted r2 score, it seemed to me the player models really struggled to explain the variability in the continuous response. To mitigate this, I created both binary and categorical versions of the player data. I created one binary dataset using 42 as the threshold and one using 50 as the threshold. These values were selected based on my analysis above regarding how many wins “guarantees” a playoff berth. For the categorical data, I created low, medium, and high win groups based on the 1st quartile value, median, and 3rd quartile value. I also repeated the process with the team data in order to continue comparing their performance. Having created these groupings, I was then able to use confusion matrices to evaluate the accuracy of my models. I also used the mean average error to compare the models, but mainly relied on the confusion matrices because they provided insight into where the models did well and where the models struggled. Upon validation, I noticed severe class imbalance in both the binary and categorical, so I standardized the classes based on the class with the minimum number of observations.

Following the class standardization, I reran the models described above and saw a slight drop in accuracy, but an increase in the accurate identification of the positive class in the binary models. After standardization, the team Random Forest for 42 wins achieved a 91.89% accuracy, and the Random Forest for 50 wins achieved a 93.75% accuracy. This suggests that the team data may be better suited to a higher threshold. Using the player averages, the model was able to achieve 81.53% and 79.97% accuracy for the 42-win and 50-win thresholds respectively, while the models using the player buckets achieved 81.51% and 75.92%. The accuracy was fairly consistent across both classes, as well. Looking at the categorical models, the team model had an 89.91% accuracy, while the average and bucket models only reached 77.84% and 79.08% respectively. Having thoroughly compared the model’s performances, I worked to remove features in order to determine which were really driving the predictions, which is discussed in more detail in the above section.

A table with numbers and a number on it

Description automatically generated

A table with numbers and text

Description automatically generated

A table with numbers and a number of objects

Description automatically generated with medium confidence

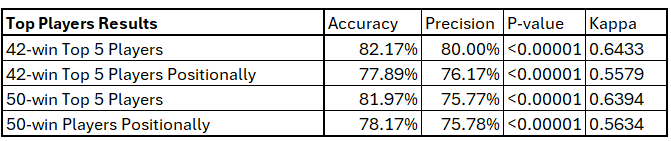
Based on these results, there are three clear takeaways. First, the team data is better suited to predicting wins across the board. It was already determined that the team data yielded better results with the continuous response, and these findings were consistent with both binary thresholds and the categorical response. While the player models were almost always able to exceed 75% accuracy, the team models outperformed them by at least 10% each time. Next, the binary response is better than the categorical response. The accuracy of both the team and player models was higher using the binary response than the categorical. It seems that the models all really struggled to predict the medium class, with lots of false identifications, incorrectly classifying the medium class as both low and high about evenly. Finally, it is clear that the 50-win threshold is easier to accurately predict against than the 42-win threshold, as the team, player average, and player bucket models all had a higher percentage of accurate predictions using the 50-win threshold. As a result, my conclusion is that the best results come from using the team data to predict against a 50-win binary threshold. With an accuracy above 96%, this was clearly the highest performing model.

**Can player statistics be used to predict a playoff berth (seasonal averages)?**

* + **5 players with most minutes played**
  + **5 players with most minutes played POSITIONALLY (PG, SG, PF, SF, C)**

Yes, they can but only to a certain degree of accuracy, and aggregation did not help. Building on the section above, my findings showed that upon adjusting the response variable to binary, the player statistics can achieve an accuracy greater than 75% at predicting wins. Based on the threshold, the win total thus indicates playoff appearance, especially when the threshold is set at 50. Initially, I aggregated the records by the three players who played the minutes then increased the number to five. To do this, I had to define which fields should be summed, and which should be averaged in order to keep the values consistent with my previous datasets. The aggregated models did not show any real improvement compared to the models discussed in the previous section. As a result, I decided to repeat the process, just without aggregation. I created two subsets, one containing the five players who played the most minutes for each team for the respective season, then one doing the same thing, except enforcing a criterion of one for each position (PG, SG, PF, SF, C).

When limiting the data to the five players who played the most minutes for each team in each season, the models were able to achieve 80% accuracy with both the 42- and 50-win thresholds. This is an improvement over the results seen using all players, suggesting that refining the dataset to the most common players provides more valuable insight than when the entire team is considered. Next, I moved on to the five players who played the most minutes positionally for each team for each season. Once again, I modeled using both the 42- and 50-win thresholds.



The models were able to achieve an accuracy of about 78% for each threshold, just below the results from the five players who played the most minutes regardless of position. This shows that overall, minutes played by position is not as consequential as minutes played without the restriction. Additionally, the fact that both subsets of models were able to surpass the prediction accuracy of the models which used all players, not just the top five based on minutes played, shows that by including all players from each team, the impact is decreased. Selecting only the top five players by minutes played allowed for more flexibility, and the model’s performance reflected that. Crossing the 80% mark is the highest accuracy out of any of my models using player data, and the models achieved it with both win thresholds. This proves that refining the data to exclude less impactful players had an effect, and that playoff appearance can effectively be predicted using player stats, through win total as a surrogate.

**How much do ALL-STARS influence playoff appearance?**

Having noticed the impact of player statistics on wins, I wanted to investigate further into the impact of players designated as all-stars on team success. Each year, twelve players are selected from each conference, based on their current season statistics, to be all-stars. There is no defined criteria for all star selection, so sometimes the selection process seems arbitrary. I had two main questions during my investigation here: what do players selected as all-stars have in common and how do all-stars impact winning. I decided to tackle the second question first.

To answer the question, I created an aggregated count field in the team dataset showing the total number of all-stars on the team for the season. Then, I created a table showing the count of teams that both made and missed the playoffs for each existing number of all-stars. The number of all-stars on a team ranged from 0-4, and the results were striking. While it is somewhat intuitive that having all-stars will lead to team success, the amount of influence was incredible. Having one or more all-star on your team yields a 75% chance of making the playoffs, while two or more yields a 95% chance. Clearly, having all-star players is almost a necessity to make the playoffs. Next, I attempted to determine patterns between all-star players using a cluster analysis. I chose to look specifically into per game averages for the five main stats.

In order to determine patterns between the all-star and non-all-star groups, I created two subsets of the data based on the whether or not the player was an all-star. I then used k-means clustering with one cluster to determine the commonalities. The results showed that all-stars record significantly more stats, specifically points, assists, rebounds, and steals. All-stars averaged more blocks, as well, but the margin was not as significant as the other three stats. In order to drill down further into the differences, I repeated the process of creating subsets by all-star designation, this time based on position. This yielded ten subsets, two for each of the five positions. The results were very consistent with the initial clustering, although a few of the positions had even greater disparities between points per game. The results here once again show the emphasis of the NBA is clearly offense.

These findings raised another question in my mind. Are there positions that frequently average more points, rebounds, assists, and steals? Early in my analysis, I created box plots of per game averages for each of the five statistics for each of the five positions. Looking back on these, I was able to determine that yes, there are positions that frequently average a higher amount. Point guards and shooting guards lead the way in points, assists, and steals, while centers average the most rebounds. In order to see if the historical selections match with the data, I created a table of all-star count by position. Shooting guards ranked second, but I was shocked to find point guards ranked last and centers were second to last. This was counterintuitive to what I expected to see based on the information provided by the cluster distributions and the box plots. This kind of confused me, so I plotted the historic trends of all-star selections by position over the years. What I found showed centers plummeting in recent years, while point guards have been rising fairly steadily for about the past 15 years. Combining that with the cluster and distribution findings, I would say the historic count undervalues point guards and their worth is beginning to be fully appreciated. Another aspect that needs to be considered is the change in the center position. Traditionally, on offense centers play under the basket, but in recent years, their role has changed to include more 3-point shots which was never the case in the past. This seems to coincide with the decline in all-star selections showing that they may have been more valuable doing their original job. Combining all my findings from this section, I strongly believe the data suggests acquiring a most importantly a shooting guard or secondarily a point guard who has all-star potential in order to increase the likelihood of reaching the playoffs.

In addition to my findings regarding all-stars and playoff likelihood, I also modeled using player statistics to predict if the player would be an all-star. The models were able to achieve a high level of accuracy right away, displaying 93.5% accuracy. Similar to the modeling I did previously predicting wins, I began to remove features and model again in order to determine which features were driving the predictions. Ultimately, I achieved the best results using the exact same fields which had the most success using twelve predictors, which were almost the exact same as the predictors which achieved the best results during my win modeling. The only differences were the exclusion of the wins field and the exclusion of two shooting percentage fields. This model was able to achieve 93.4% accuracy, and 96.75% accuracy at predicting all-stars. While it is fairly intuitive that the stats a player records will be responsible for whether or not they are selected to be an all-star, the high level of accuracy displayed by my models shows how well stats can be used to predict who the all-stars will be. Tying these findings together with my previous comments regarding the impact of continuous playoff appearances, the financial implications of having an all-star player are clear given the relationship with playoff appearance. Additionally, the accuracy of the models would allow people in management to assess whether their players are likely to become all-stars, or if they should look elsewhere. I believe my findings could be applied by teams as a minimum criterion to determine whether they should or should not pursue certain players based on the skillset they will bring to their team. This specifically refers to comparing their stats to the stats identified in the cluster analysis, especially points, assists, and steals. By pursuing trades and acquisitions in this way instead of simply looking for big name players (who are likely all-stars, but may not fit the criteria, for example a player who scores lots of points but does not record assists or steals) or cost savings, teams could apply data driven analytics and insights to their decisions.

A white rectangular sign with black text

Description automatically generated

## Citations:

1. “Golden State Warriors on the Forbes NBA Team Valuations List.” *Forbes*, Forbes Magazine, www.forbes.com/teams/golden-state-warriors/. Accessed 25 Nov. 2024.
2. Releases, Forbes Press. “Forbes Releases 2013 NBA Team Valuations.” *Forbes*, Forbes Magazine, 21 June 2013, www.forbes.com/sites/forbespr/2013/01/23/forbes-releases-2013-nba-team-valuations/.

## Appendix:

**Best vs Worst:**

A graph of different colored bars

Description automatically generatedA number and text on a white background

Description automatically generatedCluster centers: Bar graph of cluster centers

A screenshot of a computer

Description automatically generatedShot attempts

**Playoff “Guarantee”:**

A number of years and numbers

Description automatically generated with medium confidenceHistoric playoff counts, yearly average win total for playoff teams, decision tree, and win distribution for playoff teams

A diagram of a person's choice

Description automatically generated

**A screenshot of a computer code

Description automatically generated**

A close up of a number

Description automatically generated

**Team vs Player Statistics & Advanced vs “Simple” stats:**

Initial Player Modeling

A black text on a white background

Description automatically generated

A white background with black text

Description automatically generated

A screen shot of a computer program

Description automatically generated

Initial Team Modeling

A black numbers on a white background

Description automatically generated

A white background with black text

Description automatically generated

**A computer code with blue text

Description automatically generated**

Residuals for regression:

A graph of a graph showing a number of dots

Description automatically generated with medium confidence

A screenshot of a computer code

Description automatically generated**A screenshot of a computer code

Description automatically generated**Team 50-win threshold Team 42-win threshold

Team “simple” 50-win threshold Team “simple” 42-win threshold

**A screenshot of a computer

Description automatically generated**

**A screenshot of a computer code

Description automatically generated**

A screenshot of a computer code

Description automatically generatedPlayer average 42-win threshold Player average 50-win threshold

**A screenshot of a computer code

Description automatically generated**

A screenshot of a computer code

Description automatically generatedPlayer bucket 42-win threshold Player bucket 50-win threshold

A screenshot of a computer code

Description automatically generated

Player “simple” average 42-win threshold Player “simple” bucket 42-win threshold

A screenshot of a computer code

Description automatically generatedA screenshot of a computer code

Description automatically generated

**A screenshot of a computer code

Description automatically generatedA screenshot of a computer code

Description automatically generated**Player “simple” average 50-win threshold Player “simple” bucket 50-win threshold

**A screenshot of a computer

Description automatically generated**Team categorical Player average categorical

**A screenshot of a computer

Description automatically generated**

Player bucket categorical

**A screenshot of a computer

Description automatically generated**

**Top 5 players:**

**A screenshot of a computer code

Description automatically generatedA screenshot of a computer

Description automatically generated**Top 5 by minutes played, 42-win threshold Top 5 by minutes played, 50-win threshold

**A screenshot of a computer

Description automatically generatedA screenshot of a computer code

Description automatically generated**Top 5 by position, 42-win threshold Top 5 by position, 50-win threshold

**All-star Impact:**

**A screenshot of a computer code

Description automatically generatedA screenshot of a computer code

Description automatically generated**Playoff appearance by all-star count All-star predictions

**A screenshot of a computer code

Description automatically generatedA screenshot of a computer code

Description automatically generated**All-star predictions, refined data All-star predictions refined data

All-star clustering boxplots by statistic

**A graph with a red and blue line

Description automatically generated with medium confidenceA graph with a red and blue rectangle

Description automatically generated**Points: Assists:

Rebounds: Steals:

**A graph with different colored squares

Description automatically generated with medium confidence**

**A graph with red and blue squares

Description automatically generated**

All-star positional clustering results

**A screenshot of a computer program

Description automatically generated**

Statistic distributions by position

Points: Rebounds:

A graph of a bar chart

Description automatically generatedA graph of points per game by position

Description automatically generated

Assists: Steals:

A graph of a bar chart

Description automatically generatedA graph of a chart

Description automatically generated with medium confidence

3-point attempts by Center position

**A graph with blue dots

Description automatically generated**

# Lesson 14 Predictive Analytics System Group-Based Assignment

**Purpose:**

To provide a demonstration of your team’s Predictive Analytics System

**Tasks:**

1. The team should prepare to demonstrate the designed Predictive Analytics System in this capstone course.
2. The demonstration should be video recorded and shared with the instructor through the box/Canvas.
3. Every team will have an opportunity to go through the demonstration of the Predictive Analytics system designed by different teams. I will share the video recording from different teams using the box.
4. The team should set up an appointment with me to discuss about the feedback on the project report and on the demonstration video in this week, one day after the final submission is done.
5. Team members should submit the peer evaluation form latest by Thursday 11:59 PM EST in week 14.
6. More instructions will be provided by the instructor over the e-mail.

## Final Predictive Analytic System:

Taking all of my findings into account, I designed an application which relies on user input to predict which teams will surpass a certain number of wins for a certain season. By adjusting the threshold to a value like 50 wins, which results in a playoff berth greater than 97% of the time, users can predict which teams will make likely the playoffs. The application uses the team data to train a random forest, then make predictions for a selected season provided by the user. This allows for comparison to real results since the predicted events already occurred. I decided to only include the team data because my models were able to achieve greater than 90% accuracy in all test scenarios using the team data, while the player data only eclipsed 80% accuracy in the best cases. To be employed in the best possible fashion, the model would be equipped with live updated data from the current season, allowing for predictions on the current season as more data is accumulated. I created the application in R Markdown, and it has two main components, the UI and the server. The UI is simply the organization of the buttons, their titles, the front-end. The server contains the logic related to each button, the back end.

Upon running the application, the user first needs to click the Import Data button which reads the data in from JSON files. In my application, the file path points to where these files are stored on my local device, however anyone could download these files and simply adjust the file path to run the application successfully. Next, there is a win threshold response field in which the user enters the number of games they want to use as the binary threshold. Upon setting the threshold, there is a button which calculates the historic likelihood of making the playoffs at that win total. Next, the user can adjust the win column to binary based on the threshold and then standardize the classes using two separate buttons. This allows users to skip class standardization if desired. Additionally, there is a subset button which creates the subset of data I achieved the best results with. This should be used prior to modeling, as it eliminates at least one text field. The next button trains a Random Forest model using the adjusted dataset. Finally, there is a drop down menu where the user can select the year they desire. After entering the year, there is a predict button which when clicked outputs the teams predicted to eclipse the threshold for the selected year. The application also provides output messages for each step, letting the user know the operation was successful.

I tested the application a bunch of times, comparing predictions from years that have passed to the teams that actually made the playoffs. What I found is interesting, however not surprising. On the whole, the teams provided were above the threshold almost every time. The main issue was not all teams were returned. For example, maybe 12 teams were actually over the threshold, and the model predicted 10. The 10 teams predicted would all be correct, but a couple teams were not identified. Additionally, the model had better predictions using the 42-game threshold, or just a lower threshold in general. The higher the threshold, the lower the class size, thus less training data. As a result, when the threshold is too high, the model begins to have a hard time. For example, when setting the threshold to 50, the model may identify 3 teams when 6 actually eclipsed the threshold. Basically, the issue of not identifying all teams becomes accentuated with a higher threshold.

On the whole, I am pretty pleased with the application, as it has a high level of functionality, operates on user input, and can be run by anyone as long as the file paths are adjusted. I think it would be interesting to build on the existing application to enable it to include player data allowing for even further user customization. At this time, that was not realistic or a worthwhile inclusion, but just for fun that would be an interesting inclusion.

The video will be attached as a link in the submission.

## Detailed Instructions to Run Application:

1. In order to run the application, all that is required is RStudio and the JSON files. The data files will be included in the zip file. Following the steps provided in the link will show how to download RStudio: <https://rstudio-education.github.io/hopr/starting.html>
2. Adjust file paths to reflect where the JSON files are stored after downloading

A screenshot of a computer program

Description automatically generated

1. A white background with blue text

   Description automatically generatedInstall packages listed at the beginning of the scripts and then import them from library:
2. Run code to build the app, then run the final line separately.

A black and white text

Description automatically generated

RED NUMBERS ON USER INTERFACE INDICATE STEP NUMBER

1. Click the import data button to read the JSON files in.
2. Enter the desired win threshold into the input field.
3. Calculate playoff percentage if desired
4. Adjust win column based on the threshold- defaults to 42
5. Standardize the class distribution
6. Subset the data to remove features prior to modeling
7. Train Random Forest
8. Select year desired for predictions- dropdown menu populated by existing seasons
9. Click the predict button to make final predictions
10. In order to make new predictions, the process must start from scratch because the response has been converted to binary, so attempting to convert to binary again will set all values to 0

A screenshot of a computer

Description automatically generated

**13**

**12**

**11**

**10**

**9**

**8**

**7**

**6**

**5**

Application Citations:

“Welcome to Shiny.” *Shiny*, shiny.posit.co/r/getstarted/shiny-basics/lesson1/. Accessed 20 Nov. 2024.