*Data Mining Final Paper*

Riley Halloran

Computer Science Major

Indiana University

Bloomington Indiana, USA

[rifhall@iu.edu](mailto:rifhall@iu.edu)

Jack Hilvert

Computer Science Major

Indiana University

Bloomington Indiana, USA

[jhilvert@iu.edu](mailto:jhilvert@iu.edu)

Dean Rosier

Computer Science Major

Indiana University

Bloomington Indiana, USA

[drosier@iu.edu](mailto:drosier@iu.edu)

1. Introduction

This paper is a follow up to a presentation given in CSCI-B 365 Data Mining and Analysis. This paper and presentation are part of the final project for the class, where we find a data mining problem and then analyze it. The goal is to bring together all the knowledge learned in the class to demonstrate the ability to understand and model a problem in this field.

1. The Problem

Our group decided that we should attempt to model the National Basketball Association (NBA) postseason for each team based on the regular season.

The NBA has 30 teams, broken down into 2 conferences and 6 divisions. Each team plays 82 games against the other teams, aiming for the highest record in their conference. The top 8 teams make the playoffs, and are seated in the tournament based on their record, with the best team playing the 8th best team, and so on. These 16 teams play in a single elimination tournament, with 4 rounds. Each round is a best of seven, with each team needing four games to win and move on to the next round. The final two teams face off in the finals, and the team with the best regular season record gets to start with home court advantage. The finals are always between a team from the Eastern Conference and the Western Conference, but as we will discuss later, our algorithm doesn’t take this into account.

Professional basketball is a complicated sport, and can be hard to predict. Compared to other popular US sports, it is extremely high scoring. In baseball or soccer, teams might only score a couple of runs or goals in a game. However, in the NBA, teams regularly score over 100 points a game. Basketball is also an interesting blend of a team sport and an individual sport. Unlike baseball, soccer, and football, which have many players on the field at a time, each basketball team only has 5 players on the court. This means that each individual player's impact on the game is extremely large, and it lets star players take over the game and carry their team. This can make it hard to predict the winner of individual games, because one player having an off night can change the entire outcome of the game. But basketball is still very much a team sport. One player can’t win alone, and how well a team works and plays together is a big factor in success. Star power and team chemistry are both things that are hard to see on the stat sheet, so we knew going into this assignment that it would be complicated to try and predict who would win the championship.

Our first decision to make was to decide how to determine postseason success. We figured out that the best way to do this is to try and predict the amount of post season wins. Because there are four rounds in the playoffs, and you need four wins to move on to the next round, the team with 16 wins is the team that won the championship. Our other option was to try and use a decision tree to classify each team into either a winning team or a losing team. This method didn’t work, as we discuss later in the report. Another benefit of predicting wins in the off season is that we not only get to see who won in the given year, but we also get to see who our model would predict would end up in second place, and so on.

1. Data Set

The data set used for this project is taken from the official NBA website [1]. Our group sorted the data by each team in alphabetical order and then copied the data into a csv file. A file was created for each season and season type (example: 2020-2021 Regular Season). We grabbed 20 years of data from this and trained our model.

Some problems we ran into when collecting data was the fact that the NBA has been changing over the years. Teams like the Hornets and the Brooklyn Nets used to be called the Bobcats and Jersey Nets respectively. This creates a problem where we have to make sure that these stay consistent throughout the model especially when the number of teams change for a season. A second problem is the fact that the number of regular season games has been changing in our data set. Most seasons have 82 games played for each team, but this has been subject to change based off of COVID and other things impacting the season. This means that the data is not all consistent and makes it harder to analyze. Because of these factors we limited the number of seasons gathered to the last 20 years.

1. Imports

We used SK learn as an import to use the decision tree, regression, and random forest modules. We used numpy as a way of dealing with arrays and doing math. We used pandas to help parse the data sets and get the information we needed in an easy to use way. In the decision tree, random forest, and regression files, we import a file titled season. This is an object that was created in order to read in data from different data sets and store that data for easier access in the tree, forest, and regression files. We also used sklearn.metrics in order to easily calculate our error and r2 scores for analysis of the algorithms.

1. Algorithm

We started out using linear regression in order to predict the total number of games won for each team in the postseason. This turned out to be a poor way of predicting the number of wins. We had horrible accuracy scores.

We changed the model to be a decision tree. Our classification would be how many games the team won in the postseason. This worked out well because in order to be the finalist, a team would have to win 16 games. We had much better error scores when implementing the decision tree from sklearn. These scores were actually improved once we limited the depth of the tree. We tried having no max-depth, a max-depth of 5, and a max-depth of 3. We found the max-depth of 3 to be the most accurate predictor of the number of postseason wins.

Because the decision tree worked so well, we decided to also try out a random forest. The object for the random forest was also provided by sklearn. We limited the number of estimators to 100 and tried to predict as before. We found no significant improvement on accuracy with 100 estimators. We did not try to use any other number of estimators.

As far as training data goes, we used all previous season data to train the algorithm, and we used the current season as our testing set. This turned out to work much better than using only the previous season. We also found it worked best when we had at least 10 seasons of training data, so we made sure that the earliest season we would test on would be 2012 because our earliest data set is from 2002.

Finally sklearn has a built-in function to determine what the most important splitting feature is for a decision tree. We will be discussing these in the analysis of our results. Being able to see the most important features for the trees gives us much more insight into the results we got, and why we got them.

1. Analysis of Results

Our decision tree algorithm actually did quite a good job at predicting general outcomes of the postseason. Typically one of the teams with the most number of wins would actually be a finalist, and the others would make it quite far in the tournament.

Sports predictions are quite hard to make, anything from injuries to personal player issues could happen at any point in the season: both things that would be almost impossible to predict. We feel that because of this our error rate is quite low. Another thing that we failed to take into account is the inclusion of conferences in the postseason.

In the NBA one team from each conference will be in the playoffs. Because of this it should be important to weigh results based on what conference the team is in. If the best team in the West is worse than the top three in the East, our program will rank the best team in the West as number 4. However, in the real life scenario, the best team in the West is much more likely to be ranked number 2 overall because they are more likely to beat out their conference and make it to the finals.

Not taking this into account definitely skewed our results. However, as mentioned before, our algorithms still did quite a good job at predicting how good different teams would perform overall. This could be because in the previous seasons the top two teams in the finals were from different conferences.

Another thing to take into account is smaller things like team morale at the time of playing, any trades that could’ve happened, or even external events for things like COVID-19. These things obviously cannot be predicted, some of them are almost impossible to quantify. They would definitely have a weight on the final output of our data, but they are not raw data and thus would be very difficult to use to train an algorithm like this.

Our results also showed that number of losses was typically the most important feature in predicting a winner. This is interesting at first glance, until you realize that losses have much more of an effect in the postseason than wins do. If you lose too many games, you no longer continue.

Another important feature was a statistic called PIE, the Player Impact Estimate. This also makes sense when put in context of the game. The more events that a team controls in the game, the more opportunities they have to win the game, the more likely they are to actually win the game.

1. Individual Work Summary

Riley contributed to the project by researching the data set and then figuring out how to extract the data from the NBA website to get a proper data set. He then worked on the presentation that was used in class while his teammates worked on the programming. He was able to help analyze the results of the model and posited changes that could be made to improve the program.

Jack did most of the coding for the project. It was on his local version of the code that the final project was made on. He worked in conjunction with Riley and Dean to make sure the data was in the right format for the code. Not only that he also wrote the program that cleaned the data for the model to then use. Lastly he contributed to the presentation by adding in pictures of his code and specifics on the work.

Dean was our top NBA consultant for this project. As the one who knew the most about the sport going in it was his knowledge that our data and programs were laid on top of. He was the one that pointed out problems with the data set, and did his own research about the postseason so that we would be better prepared when it came to processing our models results. He contributed to the slides with this knowledge, and worked closely with Jack to best tune the code.

1. Conclusion

Overall this project is an interesting look into sports modeling. Ever since the advent of sabermetrics in Baseball, sports analytics has been a popular topic for many people. The ability to use the numbers and objectively determine who the best players and teams are is a promising idea especially for those in the industry. Our project is nothing new but it follows in the footsteps of other statisticians who have done far better and understand the data well beyond what our team could produce in a short period of time. The interesting thing for us was the challenge to see what goes into creating a model to determine success and what are the most important factors when evaluating teams. This turned out to be a little lackluster as our best metrics for determining success was the number of losses and that is as useful as saying the team that lost the least will win.

If we could do the project again our team would want to focus on incorporating ranking metrics and conference divisions into the model. The “eye test” is an important tool for anyone who follows sports which is just by watching a team or player, how good you think they are. This concept is useful because it means that if you think a team looks good or bad then that is a determining factor into how these teams rank. One way to incorporate that into the data would be to collect team rankings from sports journalists and insiders to create another vector of data. If we wanted to build a more accurate postseason model then we would do well to make sure that conference divisions are intact so that teams from one side do not get rated higher than their counterparts. By getting closer to how the actual end of season results look, this project would be better equipped to model the NBA.

This project has been a learning experience for everyone involved in this group and was a fun way to merge two avenues of interest into one. We hope that this write up has been informative to your understanding of the details of our project and how we put it all together.

1. References

[1] NBA Team Stats Advanced website [Teams Advanced | Stats](https://www.nba.com/stats/teams/advanced/?sort=W&dir=-1)