Which Statistics Best Predict the Results of NBA Basketball Games?

Daniel Glynn

October 16th, 2020

Abstract

The goal of this project is to determine which statistics, basic or advanced, are the best at predicting the success of a team each game, as well as how good of a prediction model can be created by only looking at one team's stats. Data was obtained from official NBA box score statistics. After removing data that wouldn't be relevant to the project, models were created that looked at basic team statistics and models were created that looked at advanced team statistics. A random forest classifier and a k-nearest neighbor model were both used for classification. Ultimately, the best models were just over 80% accurate at predicting whether the team won or lost. Furthermore, using the random forest classifier, the most important basic statistic was found to be field goal percentage and the most important advanced statistics was found to be team total rebound percentage. Overall, the most relevant statistics in both categories related to efficiency in creating shots and points.

Overview

When watching an NBA game, it is common to hear from the announcers “this team is winning even though they have 15 turnovers in the half, ” or “this team is giving up too many offensive rebounds to stay in the game, ” or something along those lines. In this project, I set out to determine which statistics are truly the most important in predicting whether a team will win or not. When looking at an NBA box score, I want to know which areas or metrics I should focus on when determining how well they played.

Of course, the points of the home and away team could predict the result of every game with 100% accuracy because that’s how the sport works, but in this project, only one team’s statistics are being looked at. How well a team can limit the opposing team is very important without a doubt, but for the sake of this project, the opposing team’s stats are going to be ignored. Additionally, basketball is a sport that is more influenced by individual players than most others, given how important 1-on-1 defending and attacking is. However, for the following models and analysis, only the team’s statistics as a whole will be looked at.

Data Acquisition

The dataset I am working with is from kaggle.com posted by the user Paul Rosotti[1]. It contains official NBA box score data from 2012 to 2018. This includes data from each game like the game start time and date, team name, opponent, referees, game result and plenty of statistics about the game, such as points, rebounds, assists, steals, turnovers, blocks, shots, various shooting percentages, fouls, offensive rating, defensive rating, pace, various efficiency metrics, and more, as well as similar stats about their opponents during that game. There was also a separate file containing statistics and information specific to each player on the team, which I will not be using for my analysis, as I am only interested in overall team statistics. The dataset only included regular season games, not the playoffs. So, since each team plays 82 games a year and there are 30 teams in the NBA, there are a total of (82 * 30) / 2 = 1,230 games each season. The division by 2 is because there are 2 teams in each game. There are 6 seasons in the dataset which makes 7,380 total games that will be considered. The data was structured in a couple csv files with columns indicating features/statistics, and three rows for each game.

Data Preprocessing

My initial data cleansing included removing some features that weren’t relevant for my project, including official names, team abbreviation, team division, points by quarter, among others. I also changed the dataset so that each game was represented by 1 row in the dataset instead of 3. The original reason each game took up 3 rows was only to account for 3 different official names, but since that isn't relevant, I kept just one row for each game. I also removed games that went to overtime from the data as they have an increased number of shots/points/etc. The effect would be small since there are so many games to look at, but I found in the data that roughly 6% of games went to overtime, so it is not a totally insignificant number. Additionally, I changed the team Result column which contained either 'Win' or 'Loss' to instead contain a 1(for a win) or a 0(for a loss). This will be helpful for later when calculating accuracy scores. The team result will be the target for the model, of course.

I graphed the relationship between various statistics and whether or not the team won that game. Below I have included 4 of those graphs: assists vs. points, pace vs. points, 3P% vs. FG%, and TREB% vs. TS%. A red triangle indicates a loss and a blue square indicates a win. The graphs are very busy since so many games were included in the dataset but we can still see overall relationships and patterns.

Figure 1

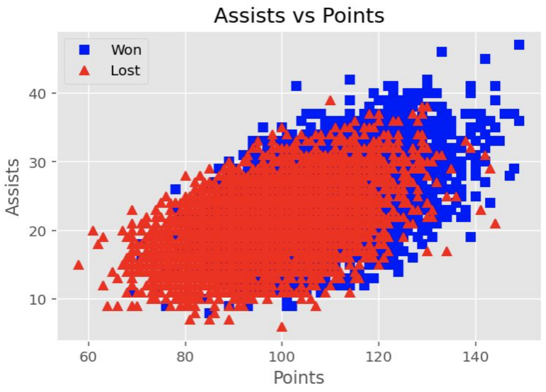


Figure 2

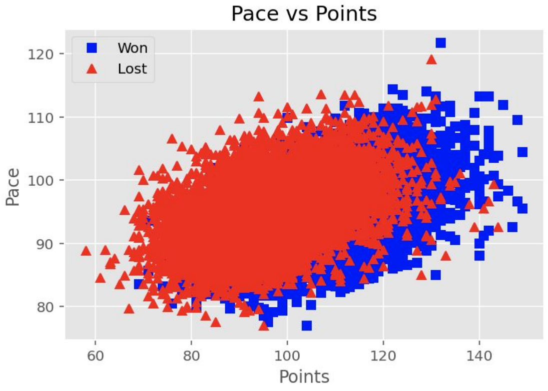


Figure 3

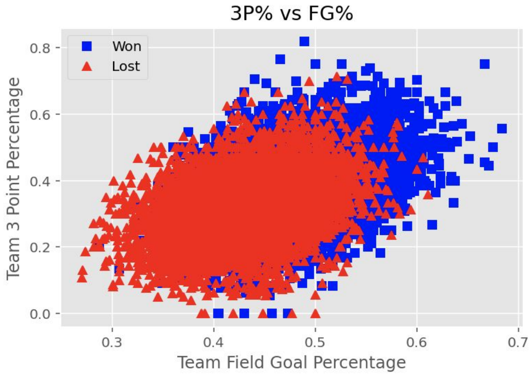


Figure 4

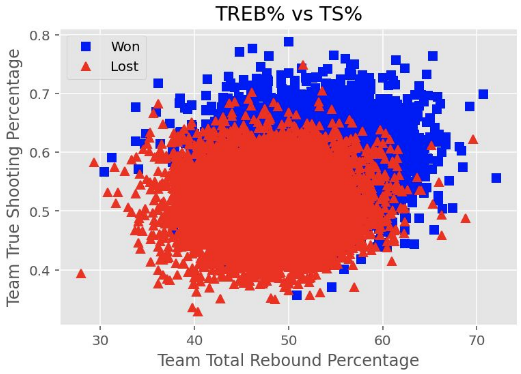


Figure 1 shows a linear relationship between assists and points and also shows that when teams have a high amount of points/assists, they win more often. Figure 2 shows a weaker linear relationship between pace and points, and it seems pace has less effect on the result of the game. In Figure 3, we also see a linear relationship between field goal percentage and 3 point percentage, which makes sense as a team that shoots well from 3 point range most likely also shoots well from everywhere. The high shooting percentages also seem to be common in wins. In figure 4, there appears to be no relationship between total rebound percentage and true shooting percentage, however, it appears in the games where the team had a high true shooting percentage, they were more likely to win.

Feature Selection

Before model selection, I hand selected both the basic and advanced statistics that I will use for each model. I separated it into two categories like this because I believe it makes more sense to look at basic statistics in relation to other basic statistics and same with advanced statistics. These selections were made based on looking at graphs similar to the ones, but were overall not very exclusive The features used for the basic model are listed here(see appendix for abbreviation meanings):

[teamPTS, teamAST, teamTO, teamSTL, teamPF, teamFGA, teamFG%, team3PA, team3P%, teamFTA, teamFT%, teamORB, teamDRB, teamTRB]

And the features used for the advanced model are listed here:

[teamTREB%, teamASST%, teamTS%, teamEFG%, teamOREB%, teamDREB%, teamTO%, teamSTL%, teamBLK%, teamPPS, teamPlay%, teamAR, teamAST/TO, teamSTL/TO, pace]

Some basketball fans may notice offensive efficiency and defensive efficiency are missing from the list. The reason for this is that offensive efficiency is just points per 100 possessions and defensive efficiency is the opposing team's points per 100 possessions, therefore they basically give you the score of the game. If you were to give a model these two features, it could predict the game 100% of the time, and would be counterproductive for the purposes of this project. They are still useful statistics to look at in general, but won't be used here. Now that we have selected the features and target for our two models.

Model Selection

I will use 2 separate algorithms in my project, both are supervised learning algorithms. Both deal with looking at a team's basic/advanced statistics from a certain game and predict whether they won that game or not. Therefore this is a classification problem.

The machine learning algorithm I am using for the first part is a random forest classifier. The reason I chose this algorithm is because random forests are less susceptible to overfitting, can be very accurate, and can handle many features. However, the main reason I chose this algorithm is because it helps me accomplish one of the goals I set at the beginning of the project. Random forests can give estimates of what the most important features are in the classification. Since I am trying to find out which statistics are most important in a basketball game, this is a great algorithm to use.

The second algorithm I will use is a k-nearest neighbor classifier. The reason for this algorithm is that I wanted another simple algorithm to make a model and see how accurate that model could be.

Results and Evaluation

For the first part of my analysis, I created two random forest models and trained them on my NBA games dataset. In the first model, I only used basic statistics for the features, which were listed above, but they are the types of statistics you see on the box score right after the game. In the second model, I only used advanced statistics for the features. These, on the other hand, are usually a little more complicated and have a formula based on basic statistics. For both models, I used 100 estimators and a max depth of 20, which I came to after messing around with those values and finding what worked best.

In the basic stats random forest model, I calculated the accuracy score to be 0.805 and the f1 score to be 0.806. In the advanced stats random forest model, I calculated the accuracy score to be 0.803 and the f1 score to be 0.798. The models performed very similarly to each other, both around 80% accurate, with the model based on basic statistics to have the slight edge.

Additionally, with the random forest models I found the most important features of both models. I then graphed that information in the following two bar graphs.

Figure 6

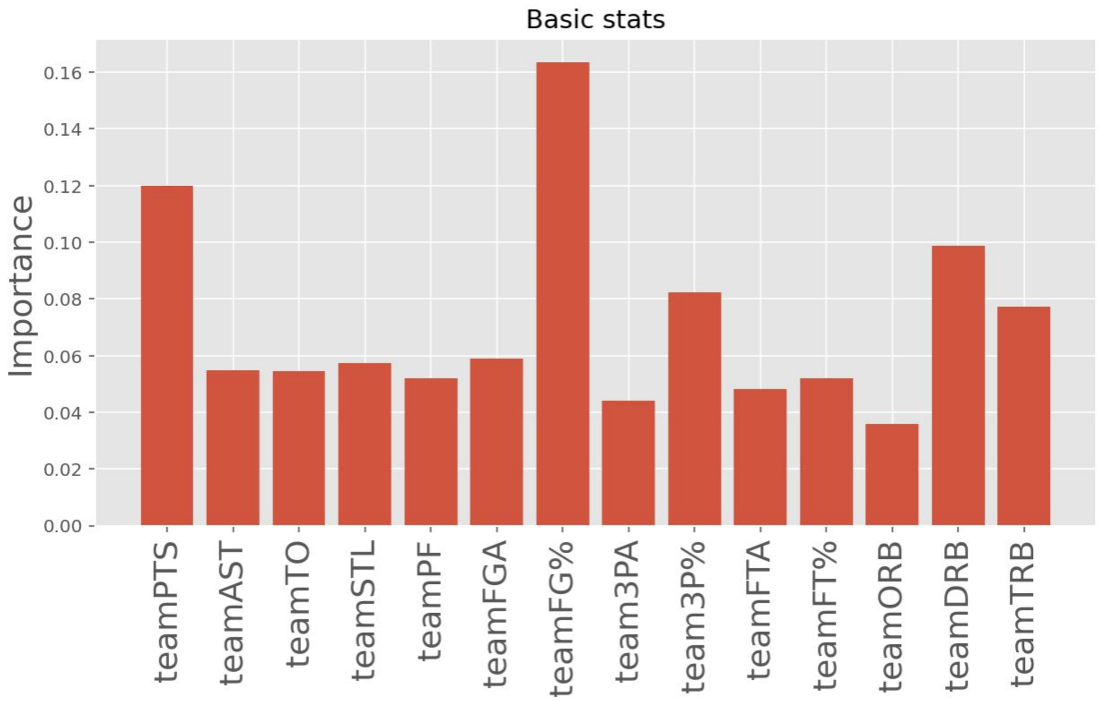
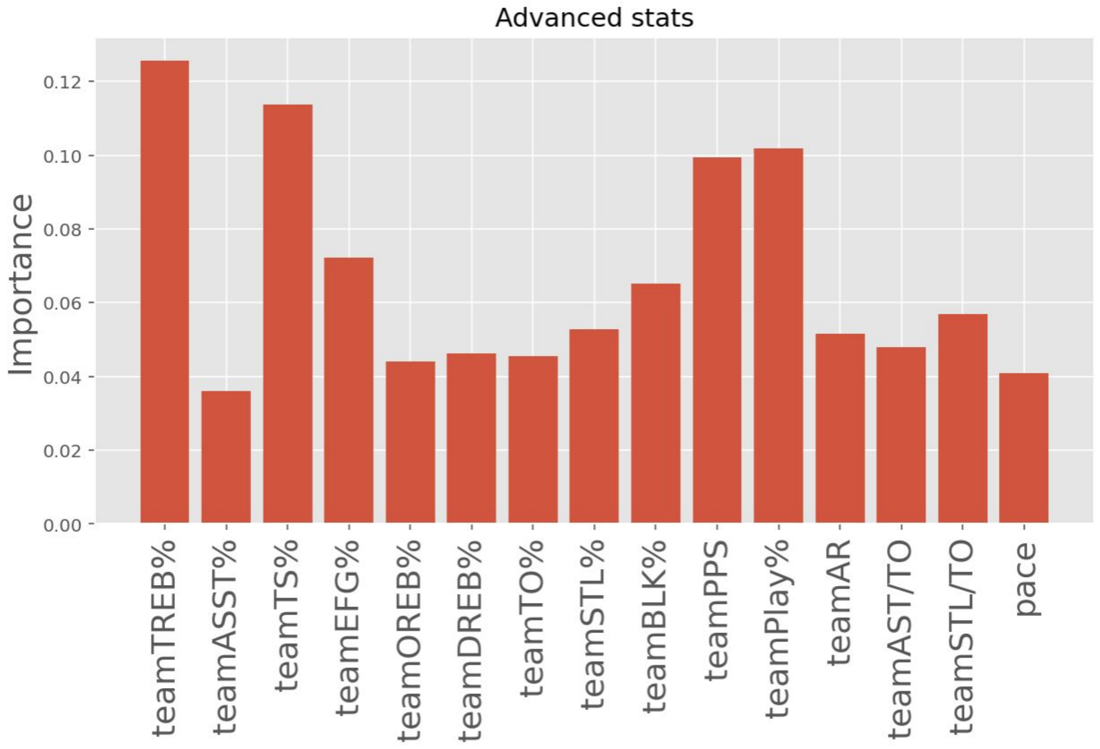


Figure 7



As you can see, in the basic stats model(Figure 6), the model found team field goal percentage to be the most important followed by team points and team defensive rebounds. I would've guessed team points to be the most important, since the team with the higher points always wins, but with this model, it turns out that field goal percentage, or the number of shots that a team shoots that go in, is the most important. It seems to be more important to be efficient with your points than simply get a lot of them.

In the advanced stats model(Figure 7), of the statistics I chose, the model found team total rebound percentage(percentage of the total rebounds in the game a team gets) and true shooting percentage(an advanced metric determining overall shooting accuracy including 2s, 3s, and free throws) to be the most important, closely followed by team points per shot and team play percentage(percentage of plays resulting in a shot). I found it interesting how the most important metric was one regarding rebounds instead of shot percentage or shot production. One possible reason for this is rebounds can partly control how many shots your team gets off as well as how many shots the other team gets off. This is because if you get an offensive rebound, you get another chance at scoring, and if you don't let the opponent get an offensive rebound, they don't get a second chance.

On top of the random forest models, I also created two k Nearest Neighbor models and trained them with the same dataset. This was done to get a perspective of the same problem from a different algorithm. Again, I used one for basic stats and one for advanced stats. In the basic stats kNN model, I calculated the accuracy score to be 0.799 and the f1 score to be 0.803. In the advanced stats kNN model, I calculated the accuracy score to be 0.725 and the f1 score to be 0.720. While the accuracy score of the basic model was very similar to that of the random forest models, the score of the advanced model was 7 to 8 percentage points lower in this case. I also found the confusion matrices for both models:

```
Basic confusion matrix              Advanced confusion matrix

      [1157  306]                         [1097  404]

      [ 285 1205]                         [ 407 1045]
```

As you can see, the model is clearly missing on a lot of data points, which goes to show the overall volatility and unpredictability of NBA games, or perhaps the models' unreliability. Overall, however, I was impressed that an 80% accurate model could be created, just by looking at one team's stats and not the others. The most interesting of the project, though, was finding out which statistics had the biggest impact on the 80% accurate model.

Future Work

The main issue with this project is that the models can only predict the results of a game after it has happened. This isn’t exactly interesting, since you can just look at the score to the result, you don’t need a model. Therefore, if I could create a different model building off this one, I would maybe look at statistics halfway through the game and predict the game in real time from there. Another idea was to look at a team's statistics from their previous games in the season to predict how well they might do in a future game. Of course, in these future cases, I would also want to look at the opponent/prospective opponent’s stats to predict the game. All this is not to say that the model I created here was useless, though. When making a predictive model, it would be helpful to know which stats are actually important in predicting games, which was mostly accomplished in this project.

Appendix: Statistic/Metric descriptions

Basic statistics:

- teamPTS: points scored by team
- teamAST: assists made by team
- teamTO: turnovers made by team
- teamSTL: steals made by team
- teamPF: personal fouls made by team
- teamFGA: field goal attempts (field goal = any shot, except free throw)
- teamFG%: field goal percentage
- team3PA: three pointers attempted
- team3P%: three point percentage
- teamFTA: free throws attempted
- teamFT%: free throw percentage
- teamORB: offensive rebounds made by team
- teamDRB: defensive rebounds made by team
- teamTRB: total rebounds made by team

Advanced statistics:

- teamTREB%: total rebound percentage by team
  - Calculation: teamTRB * 100 / (teamTRB + oppTRB)
- teamASST%: assisted field goal percentage by team
  - Calculation: teamAST / teamFGM
- teamTS%: true shooting percentage by team
  - Calculation: teamPTS / (2 * (teamFGA + (teamFTA * 0.44)))
- teamEFG%: effective field goal percentage by team
  - Calculation: teamFGM + (0.5 * team3PM) / teamFGA
- teamOREB%: Offensive rebound percent by team
  - Calculation: teamORB * 100 / (teamORB+ opptDRB)
- teamDREB%: Defensive rebound percent by team
  - Calculation: teamDRB * 100 / (teamDRB+ opptORB)
- teamTO%: Turnover percentage by team
  - Calculation: teamTO * 100 / (teamFGA + 0.44 * teamFTA + teamTO)
- teamSTL%: Steal percentage by team
  - Calculation: teamSTL * 100 / Poss
- teamBLK%: Block percentage by team
  - teamBLK * 100 / Poss
- teamPPS: Points per shot by team
  - Calculation: teamPTS / teamFGA
- teamPlay%: play percentage by team

    - Calculation: teamFGM / (teamFGA – teamORB + teamTO)
- teamAR: assist rate for team
    - Calculation: (teamAST * 100) / (teamFGA – 0.44 * teamFTA + teamAST + teamTO)
- teamAST/TO: assist to turnover ratio for team
    - Calculation: teamAST / teamTO
- teamSTL/TO: Steal to turnover ratio for team
    - Calculation: teamSTL / teamTO
- Pace: pace per game duration
    - (poss * 48 * 5) / totalMin

References

[1] https://www.kaggle.com/pablote/nba-enhanced-stats