Sarthak Jain

STAT-6650

# Predicting the points total of an NBA game

## Abstract

I recently came across an article that broke down gambling revenue by each state. As a sports/data science nerd, I asked myself two questions. First, if using a statical approach would turn a profit? Second, how a machine learning model would measure up against a regular bettor? Through this project, I will attempt to answer the previous two questions. The way to answer these two questions is to build machine learning models that would predict the scores of the NBA games correctly.

## Introduction

The National Basketball Association is the largest basketball association in the world and brings in an estimated eight billion dollars a year. As a collective whole, the NBA generates immense revenue and data. The data pertains to teams and players alike. This data is beneficial for analytics and game strategy. Sports teams and organizations have adapted data analytics and modeling to gain every competitive advantage they can get. A lot of decisions made in a basketball game are determined by statistics and analytics. These analytics could very well make the difference in the outcome of a game.

Sport prediction is usually a classification problem to determine wins and losses, but one can also attempt to determine the margin of victory, a continuous target. A lot of unique features play into depicting the outcome such as the historical performance of the teams, results of matches, data on players, and so on. These predictions are pivotal in the betting process as bookmakers, media, fans, bidders, and stakeholders are all deeply involved.

When creating a model, it is important to have domain knowledge and a functional understanding of the data. This will go a long way and allow the data scientist to understand what data is important. This paper discuss the methodology, the analysis and the resulting conclusion.

Sport predictions have been becoming more relevant in the industry. Before every game, analysts will display a prediction as to who will win and the margin of victory. Another industry that has stemmed from sport predictions is the betting industry. Legal betting platforms have been appearing and expanding for years. An accurate model is imperative to predict games and quantify all metrics in a basketball game to ensure the betting process is equitable. There are many uses to machine learning models in sports and their usefulness is only growing.

## Methods and analysis

The best performing NBA models achieved an upper bound between 66 - 72% accuracy in predicting the winner of an NBA game. During the regular season of NBA basketball, the upset rate is 32.1%. This means that the non-favorite team will complete an upset about 32% of the time. This upset rate makes sense, as the best performing models have managed to predict games with about 70% accuracy. In this review, NBA prediction models will be examined and compared to one another to understand what goes into creating a successful model.

The most well known model to predict NBA games is the FiveThirtyEight model. They use a metric called Elo ratings in their model. Elo ratings keep track of the final score of each game, where the game was played, and when the game was played. A 1500 Elo rating is the average and starting rating. This score changes as the season progresses adding or subtracting to account for the team's performance. The Elo rating also carries over from season to season as good teams tend to remain good and the same applies for bad teams. The model also uses a metric called RAPTOR, which uses a blend of basic box score stats, player tracking metrics and plus/minus to estimate a player's effect (per 100 possessions) on his team's offensive or defensive efficiency. Both the RAPTOR score and the Elo rating are both updated after each game to reflect the performance of the team and the players. Not only does the model account for team and player metrics, but also other variables such as home court advantage, fatigue (back-to-back games), travel (distance traveled), historical playoff performance, and altitude. The measurements are also adjusted in the playoffs to account for the seeding of teams. The model is complex and uses a variety of feature engineered statistics and has proven to be the gold standard for NBA prediction models. (FiveThirtyEight).

"Which NBA Statistics Actually Translate to Wins" by Chinmay Vayda goes into detail on the specific statistics that are most important to the outcome of a game. The statistics were a team's Offensive Rating, Defensive Rating, Rebound Differential, and 3-Point %. This model used all the traditional box score stats as well as Pace and PIE (Player Impact Estimate). The model implemented a variety of machine learning models such as Linear SVC (support vector classifier), KNeighborsClassifier, SVC, Bagging Classifier, Random Forest Classifier, and XGB Classifier. A bagging classifier is an ensemble meta estimator that fits base classifiers each on random subsets of the original dataset and then aggregate their individual predictions to form a final prediction. XGBoost is a decision-based ensemble machine learning algorithm that uses a gradient boosting framework. The model that showed the best results was the support vector machine SVC classifier. The model performed with great accuracy on the test data but did not perform as well on predicting new games. The model suffered from bias and overfitting on the training set as they had a limited dataset.

Another model that predicted NBA games was one done by Josh Weiner on Towards Data Science. This was an interesting case as they used a variety of feature engineered columns and player data. They created two separate models: one based on team data and Elo rating and another on player data. This model used the Elo ratings of the teams, the recent team performance of the team (last 10 games), recent player performance, player season performance, and player efficiency ratings. Player season performance is the average player stats over the entire season and player efficiency rating is a rating of a players per minute productivity. The first model created using the team stats and Elo rating used Logistic Regression and RandomForestClassifier as two potential models. The models performed well with 66.95 - 67.15% accuracy. The second model created used individual player statistics and scoring to make their game outcome prediction. It uses stats such as the players average performance over the past 10 games and how many points a player will score in each game. This specific model used Linear Regression model to predict the score of the game based on the points scored by the players. This final game result is then used to predict the outcome of the game. This model did not perform as well and received a 58.66% which was the expected outcome. Aggregated player performance has a high variability and is inconsistent as one of the sole metrics in predicting basketball games. One of the main takeaways from the article is that the time spent optimizing their parameters was not worthwhile, as it was time consuming and computationally costly for only a marginal improvement in the accuracy

The most common features used in predicting an NBA game were the home team advantage, win percentage, rebounds, assists, turnovers, steals, blocks, plus/minus score, offensive rating, defensive rating, and true shooting percentage. This combination of stats gives a thorough understanding into team performance and can be used to accurately predict games. Most of the models scored from about 58 – 70% range, a score within that range would be acceptable for the model.

Since models in the past saw promising result, this project random forest classifier and XG boost was used. As I explained in the project proposal the model would be considered profitable only if it predicts at least 53% of the scores correctly.

Random Forest Classifier: A random forest is an ensemble learning method for classification and regression. It builds multiple decision trees and merges them together to get a more accurate prediction. In a random forest, only a subset of the features are taken into consideration by the algorithm for splitting a node. The model will also rank the importance of each feature in making the final decision. The figure below is a random forest with two trees (Tutorials Point).

XGBoost: It is an optimized distributed gradient boosting library designed to be highly efficient. It uses the Gradient Boosting framework and supplies a parallel tree boosting. Gradient Boosting is a prediction model in the form of an ensemble of weak prediction models, typically decision trees. The model is built in a stage-wise fashion and allows the optimization of an arbitrary differentiable loss function. (Sklearn).

## Conclusion

The goal for building moles was to predict the total score of the games which would help betting over/under for any given. When compared to the betting outcomes XGBoost predicted 6/10 games, and Random Forest classifier predicted 5/10 games correctly. Adding a rolling ELO rating, and team efficacy rating to the model should help improve the accuracy. As mentioned the proposal the model would be considered successful if it predicts over 53%, which was achieved.

References

Houde, Matthew *"Predicting the outcome of NBA games."*

Bryant University, April 2021

Kandell, Jake. *"JakeKandell/NBA-Predict."*

GitHub, GitHub, 2019, github.com/JakeKandell/NBA-Predict.

*"Machine Learning Tutorial: Machine Learning with Python - Javatpoint."*

Www.javatpoint.com, JavaTPoint, 2018.

Silver, Nate. *"2019-20 NBA Predictions."*

*FiveThirtyEight*, FiveThirtyEight, 12 Feb. 2020, projects.fivethirtyeight.com/2020-nba-predictions/.

Torres, Renato Amorim. "*Prediction of NBA games based on Machine Learning Methods."*

*University of Wisconsin, Madison* (2013).

Betting odds: https://www.sportsbookreview.com/betting-odds/

Alrurad, Chris. "Legal US Sports Betting Revenue."

https://sportshandle.com/sports-betting-revenue.