Predicting NBA Playoffs Ranked Bracket

Ekisha Basu Connor Caldwell Yash Kothari Pranav Putta Lakshman Ravoori

Abstract

This project aims to use a deep neural network to predict the NBA playoffs' ranked bracket. The model will be tested against 2 baselines (Random Forest and Logistic Classifier). The majority of previous approaches to NBA predictions have been focused on generating the playoff spread. This project differentiates itself by focusing on the ranked bracket for the playoffs. At the end of this project, we aim to be able to predict the ranked bracket for the 2021 playoffs.

1. Introduction

The NBA is one of the top grossing sports in the United States, bringing in over $8 billion in revenue per year. As such, sports betting is a prominent aspect of American culture that engages millions of people nationwide. Engineering a prediction algorithm that can reliably produce insights about the outcomes of games would be of significance to consumers, the NBA franchise and opposition team analysis, as well as the gambling industry.

As is common with any sport, the NBA has a dense collection of data and statistics ranging from points scored all the way to the number of blocks, assists, and steals made by every player. The unpredictability of the NBA is very likely due to our inability to accurately quantify players and performance metrics on the field. Through this project, we hope to understand which performance metrics and statistics are the major contributors to the overall game outcome.

2. Problem Definition

The overarching problem can be defined as a decision tree and ranking problem.

[*]Equal contribution . Correspondence to: Pranav Putta <pputta7@gatech.edu>.

Proceedings of the 37^{th} International Conference on Machine Learning, Online, PMLR 119, 2020.

2.1. Application of Decision Tree and Ranking

Each individual basketball game over the course of an NBA season can be represented as a decision tree. There are 'N' number of input features - counting statistics - that can be traversed over to eventually reach a binary outcome - Win for Team A or Win for Team B. Taking the sum over all results over a season for an arbitrary team 'A', we can then "rank" teams in order of the number of total wins they accumulate.

2.2. Input Space

Basketball statistics data will be pulled from basketball-reference.com and espn.com/basketball/stats. Unique dataset will be created after performing statistical analysis of correlation between various stats and the impact on team win. Final dataset will be pre-processed using PCA and mean-centering to reduce dimensionality.

2.3. Output Space

$(x1, x2, x3, x4, x5, x6, x7, x8)$
$(x1_A, x2_A, x3_A, x4_A, x5_A, x6_A, x7_a, x8_A)$
2x8 Dimensional Matrix - ranked from x1-x8 for each conference (east and west)

2.4. Baseline 2: Logistic Classification

The probability of a team's win or loss is expressed as a linear combination of input parameters. The classification categories are a binary option of win for 'team A' or a win for 'team B.' The teams will then be ranked by the sum of their wins.

2.5. Assumptions

a. Team win percentage is based solely on individual performances and quantifiable statistics
b. Injuries to players unaccounted for (pre-season predictions)

3. Methods

We plan to use two baseline models to compare against our own model.

3.1. Baseline 1: Random Forest Model

The team's rank will be classified via the random forest model. This works by traversing through a pseudo-random decision tree to find game outcomes via a sequence of input parameters.

3.2. Baseline 2: Logistic Classification

The probability of a team's win or loss is expressed as a linear combination of input parameters. The classification categories are a binary option of win for 'team A' or a win for 'team B.' The teams will then be ranked by the sum of their wins.

3.3. Our Model: Feedforward Deep Neural Network

The team's rank will be classified via a feedforward deep neural network (DNN). The model will have 6 to 8 layers and consist of various layer types. Memory is important as we will need to remember each individual game prediciton and use it to base future predictions.

3.3.1. Architecture of Memory

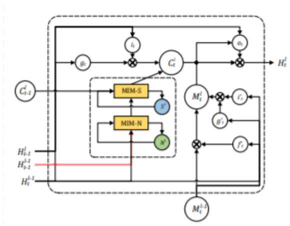

Figure 1. Memory in Memory Block

Instead of using a standard ST/LSTM unit, we will use a memory in memory unit, better for spatiotemporal predictions (our individual game predictions will need to be recorded based on both prediction space as well as time of prediciton)

3.4. Performance Metrics/Loss Functions

Recurrent unit is required within the deep neural network architecture for feedback from memory. Loss functions used will be two-fold:

3.4.1. Mean-Squared Error (MSE)

Distance between our predicted outcome of a game and the ground truth of the actual result

3.4.2. Mean Average Error (MAE)

Generally used as a variant of an error function for classification/regression problems. This has been used by our baseline models in the past and hence makes it easy for a comparative basis.

3.4.3. Memory Feedback

The memory of predictions for results of past games and the error in those predictions will be recurrently fed into our deep network as double-verification (since our network performs two functions simultaneously).

3.4.4. Past Performance

Currently, Random Forest $>$ Logistic Classifier in terms of performance (ie it has a lower MAE). Hence, we aim to improve upon the MAE of the Random Forest while also using MSE as an additional testing measure.

4. Potential Results and Conclusion

4.1. Potential Results

A bracket of playoff teams (generated in a visual form) would be an optimal representation of our results. This has been done in the past by various models and seems like a good baseline for our result representation. Our goal is to be able to use regular season win sums to predict the progression of the bracket through the NBA playoffs (predict the overall winner). A table/ranking column vector will also be a result we would like to display.

Figure 2. Potential Visual Representation

4.2. Conclusion

The project's success will definitely be measured based on how well our model can learn to predict results. However, the overarching goal of this project is to be able to come up with a more comprehensive result of statistical features that impact the outcome of a game. The usage of data as a basis for information is prevalent in sports. If we are able to identify a dataset of basketball counting statistics that can be said to span the most important features impacting winning, this would be a huge success. Apart from that, we hope to learn how to use and test various machine learning models to perform data analysis on sports results as an overall goal.

5. References

Cheng, Ge, et al. "Predicting the Outcome of NBA Playoffs Based on the Maximum Entropy Principle." Entropy, vol. 18, no. 12, 2016, p. 450., doi:10.3390/e18120450.

Kohli, Ikjyot Singh. "Finding Common Characteristics Among NBA Playoff Teams: A Machine Learning Approach." SSRN Electronic Journal, 2016, doi:10.2139/ssrn.2764396.

Hu, Feifang, and James V. Zidek. "Forecasting NBA Basketball Playoff Outcomes Using the Weighted Likelihood." Institute of Mathematical Statistics Lecture Notes - Monograph Series A Festschrift for Herman Rubin, 2004, pp. 385–395., doi:10.1214/lnms/1196285406.

reHOOPerate. "NBA Neural Networks: Quantifying Playoff Performance by Number of Games Won and Seeding." Medium, Re-HOOP*PER-Rate, 21 Aug. 2020, medium.com/re-hoop-per-rate/nba-neural-networks-quantifying-playoff-performance-by-number-of-games-won-and-seeding-9b8e9d2f6a55.