NBA Regular Season Game Prediction

Julian Bayram
Fordham Univeristy
jbayram@fordham.edu

Matthew Spalding
Fordham University
mspalding1@fordham.edu

***Abstract*—** Competitive sports are near and dear to the hearts of many. We love to cheer on and predict wins for our favorite teams, but not always using methods that are based in statistics and reason. Considering the significant amount of wealth transferred just between organization, sponsors, and players, betting scenes grow in kind with exposure and wealth within the sport. Our project sought to create a means of predicting the winner of a competitive sports match through several different predictive classifiers. Specifically, we looked at the NBA in order to build our model due to not only our personal love for the game but because of the unique qualities of the NBA within the data world. By refining our choices of classifiers, we sought to incrementally improve the accuracy of our prediction, which would allow for us to create a method of choosing classifiers that can generally be applied to other competitive sports. One method we examine is using established historical data trends as classifiers. An Example of this is that NBA games for the past sixty years on average are decided by a difference of ten to eleven points. Considering this, we can make a much more accurate prediction in a game where there is a difference of 10 points or more. Another method is using the scoring data from the sport in order to create classifiers.

We utilize the score of a given match to determine the winner, but certain biases behind the scores give us an even deeper understanding. For example, a home team in the NBA has in general a higher chance of winning than the away team in that match. Along with home advantage, teams on a winning streak also have a higher chance of winning their next matches. Through utilizing this we can then weigh certain victories higher than others.

I. Introduction

The NBA dataset we operated on from basketball reference includes 30 teams with 15 players. There are 16 attributes that describe player performance throughout a season. One of the reasons we chose the NBA is because it is a particular data rich sport with 1,230 games being played in the season before playoffs. In terms of predicting the winner of a competitive game through decision trees, the amount of data is generally the biggest issue. If there aren't many games played in a season, then the accuracy of the prediction will be off from overfitting the data. Due to the season-based contracts of players, on top of injuries and other problems that can arise in a highly competitive environment, most of the data we want to use is based in one season. Originally, we wanted to see if we could predict the outcome of the 2019-2020 season, but with not enough data of the actual teams facing off it would have to rely on mostly data about individual players. Thus, our primary objective is to raise our prediction accuracy as high as possible using public data that anyone can access and download from the NBA. Through this we can not only judge the winner, but we can figure out which classifiers are more valuable.

II. IMPLEMENTATION

A. Methodology

The dataset was retrieved from the basketball-reference website. We focused on the data from the 2015 to 2018 NBA regular season consisting of 1,230 games in a csv file from basketball-reference.com. The dataset contains 9,841 rows from all the regular season games played. After loading the dataset in pandas, we organized the data and fixed the missing rows and headings. Subsequently, we had to create new attributes from the existing data to increase the accuracy of our results. The first feature we implemented is who won the last game between the 2 teams, and who won their previous game;

we extrapolated these features in python as a way to determine which team is in better form at any given time.

B. *Experiment Methodology*

Given that basketball is an easier sport to predict, we used win percentage as our baseline prediction method. Just by comparing team win percentage we can accurately predict 61.2% of NBA games. Initially, we created a 16 feature vector for every game that compared different statistics between the two teams (points scored, 3pt made, 3pt att, FT made, FT att, etc.); nevertheless, our models suffered from overfitting. Following this dilemma, instead we focused on 7 features (points scored, fg att, def rebounds, assists, turnovers, team record, recent record). For our feature selection we gathered statistics for each team's first half on the season and tested the model on the remaining half of the 2018-19 season. For the feature selection we maximized the number of relevant variables and minimized irrelevant features. In order to reduce variance we used the extra tree classifier and maximized our features at 32. The resulting score for the ensemble represents the average of measures of trees for the group. The extra tree classifier uses each vector as a variable and models its effect on predicting the value of the target variable.

C. *Techniques*

We implemented 3 supervised learning classification models: logistic regression, random forest, and support vector machines (SVM). After isolating our target variables, we split our dataset to 80% training and 20% test set. We sampled 70 games for our logistic model. Logistic regression here trains the coefficients for the individual features within our feature vector to output the probability that a given team will win. In this case, since our outcome is binary, we use 0 and 1 to denote wins and losses, with the home team winning the game if the predicted value is ≥ 0.5

Our random forest model constructed 2000 trees and used sklearn to train and test our subsets as well as set the seed for replication. The random forest model classified wins on the testing subject, testing 500 trees with 1 variable tried for each split and gave us an accuracy of 61.2%. Subsequently, after the accuracy and confusion matrix are calculated we convert the win variable to a factor and classify wins on the entire dataset. Similar to the ROC curve analysis, the most important feature for the 2017 NBA season was team win percentage; consequently, a team's overall record is more impactful than their isolated play at home. The other dominant performance indicator for our prediction algorithm was home field goal percentage.

Our SVM model used the training and test data as a subset of our dataset, including 100 training games and 12 test games. SVM is efficient even when the number of samples is smaller than the number of features, it is also able to define multiple kernels and work with support vectors. We developed our SVM using different kernels using a sklearn library. We were able to deal with the different kernels without representing the weights, where σ illustrates the distance between x(y) and x(z) additionally, we used the Gaussian Basis Function kernel, which gives us a modified polynomial kennel and the decision rule.

III. RESULTS

After using historical data from previous seasons, we realized that it did not make that big of an impact on our accuracy. The table below shows what we calculated based off that season and the seasons before it.

Seeing as it only rises a few tenths of a percent, we can tell that data from previous seasons is not as important as data directly within the season. Individual players can significantly influence and impact the result of a game, as a result, trades and injuries often alter the performance of NBA teams from one season to the next, making it difficult to use historical data. Furthermore, as some teams draft and release players as a way to free up cap space or modify strategy, we concluded that we should test individual seasons separately.

Algorithm	**2015-16**	**2016-17**	**2017-18**
Log. Regression	64.2%	64.1%	64.2%
SVM	62.8%	62.6%	62.3%
Random Forest	61.2%	61.1%	61.3%
Baseline	61.7%	61.4%	61.5%