Predicting NBA Game Results: A Machine Learning-Based Approach

Meheresh Yeditha

INTRODUCTION

The National Basketball Association (NBA) is a professional men's basketball association based in North America and is considered to be the world's premier men's basketball association. The simplicity of the game of basketball, combined with the international and diverse nature of the game, has quickly led to basketball becoming one of the world's most popular sports. It has also made the NBA one of the world's most followed, highest-grossing sports leagues.

In addition to its lucrative financial nature, the game of basketball also contains many unique components that make it more conducive to statistical analysis and reasoning than other major sports. This has led to a statistical revolution that has transformed the way NBA players and teams are analyzed and utilized. It has also led to recent exploration into using machine learning techniques to predict and analyze the results of games and player performances. In this vein, this paper seeks to build a statistical model based on machine learning software that can be used to predict which team will win each basketball game in a given season using classifiers, as well as predict the margin of victory using indicative statistical measures using regression.

This paper discusses the features utilized and feature space that led to the construction of this model, including team wins, opposing team wins, winning and losing streaks, and the location and time of a game, all for the 2015-2016 NBA season. This paper also includes a discussion of the machine learning approaches used to obtain these results, including algorithms like linear regression, logistic regression, and support vector machines, and techniques like parameter tuning and feature selection. The rest of the paper is comprised of a summary of parameter tuning used in these approaches and an error analysis.

RELATED WORK

There has been significant recent effort to attempt to predict NBA game results through machine learning.

"Prediction of NBA Games based on Machine Learning Methods", by Torres, outlines a classifier-based approach based on a feature space that included win-loss percentage, point differential per game (margin), and win-loss percentages of each team based on whether the game was home or away. Torres also uses a variety of approaches in his paper, tunes his parameters in a variety of ways, and incorporates a diverse feature space and a

variety of seasons. Using a linear regression-based approach, Torres obtains a prediction accuracy of 0.7009. In contrast to the approach described in this paper, Torres solely focuses on classification, and uses a small number of correlated features in his feature space, compensating for this by applying principal component analysis. [1] This paper attempts to solve the aforementioned issue through careful feature selection, but retains Torres' approach to designing the feature space.

"Various Machine Learning Approaches to Predicting NBA Score Margins", by Avalon, et. al, outlines a classifier-based approach to predicting the outcome of games, as well as a regression based approach to predicting margins, just as this paper does. Avalon et. al's experimentation with linear regression, Gaussian discriminant analysis, and support vector machines with principal component analysis shows their classifier-based system obtaining an accuracy of .6196, as well as a precision of .6404 and a recall of .8022, and a comparatively good performance in the regression task. Like this paper, Avalon et. al looked at one season (2013-2014), but incorporated almost 218 features, very few of which were independent. Furthermore, the different playing styles of numerous teams in the league that manifest themselves in these statistics may have resulted in overfitting. I attempt to solve this problem by condensing the feature space and creating a compact, highly accurate model. [2]

DATA SET

This project involved collecting win-loss and statistics on every game in a given NBA season. The raw game information data was collected from basketball-reference.com, a website dedicated to statistically documenting the NBA. The format of the website's statistics on game outcomes for each team in a given season is illustrated in Figure 1, with information on the game number under G, date, time, box score, opponent name, status as a win or loss, team score and opponent score, record for the team after the game, and streak as given features in the table.

Regular Season Share & more ▾ Glossary

G	Date			Opponent		Tm	Opp	W	L	Streak
1	Wed, Oct 28, 2015	7:30p ET	Box Score	Philadelphia 76ers	W	112	95	1	0	W 1
2	Fri, Oct 30, 2015	7:30p ET	Box Score	Toronto Raptors	L	103	113	1	1	L 1
3	Sun, Nov 1, 2015	3:30p ET	Box Score	San Antonio Spurs	L	87	95	1	2	L 2
4	Wed, Nov 4, 2015	7:00p ET	Box Score	@ Indiana Pacers	L	98	100	1	3	L 3

Figure 1: Sample data from basketball-reference.com

For the purposes of this paper, the 2015-2016 season was examined. The data was subsequently downloaded as a CSV for each NBA team. From this original list of features, only the day of the week, opponent, win-loss status, scores, win-loss records, and streaks were kept, and manipulated into a statistics-friendly form by changing win streaks (denoted as "W x" in the Streak category) to positive numbers, and changing losing streaks (denotes as "L x" in the Streak category) to negative numbers. The team score was then subtracted from the opponent score to get the margin of victory/defeat. The box score links and time of the game were also removed, while the date field was mostly removed except for the day of

the week the game was played. The sixth column denoting whether the game was played at home or against was transformed into a nominal value of H for home and A for away, with blank spaces indicating a game was played at home and "@" indicating the game was played at the opponent's home arena, or away. Finally, the win-loss numbers after each game were converted to percentages. [3]

Name	Type	Explanation
Date	Nominal	Day of week of the game
Location	Nominal	Home (H) or away (A)
Margin (Regression Only)	Numeric	Point differential between team and opposition
Win/Loss (Classification Only)	Nominal	Whether team won the game over opposition
Win-Loss Record, Previous Year, Team	Numeric	Win-loss percentage of team in the previous NBA season (2014-2015)
Win-Loss Record, Previous Year, Opposition	Numeric	Win-loss percentage of opposition in the previous NBA season (2014-2015)
Team Wins	Numeric	Win-loss percentage of team up to current game
Opposition Wins	Numeric	Win-loss percentage of opposition up to current game
Streak	Numeric	Number of games won or lost consecutively for team
Opponent Streak	Numeric	Number of games won or lost consecutively for opposition

Figure 2: Explanation of features in data set

In addition to this initial data, each instance was initially augmented with the team's record from the previous year, which was done to compensate for a paucity of data about each team near the beginning of the season. A Python script then parsed this data to augment each instance with the opponent's record up until that point in the season, and the opponent's record in the previous season. Finally, the data was represented in two different fashions, with one serving as the regression data with the win/loss nominal value eliminated in favor of the margin of victory/defeat, and one serving as the classification data with the margin feature eliminated in favor of the win/loss nominal value. At the end of this process, I ended up with 2460 instances in each of the data sets, corresponding to 1230 games total.

At this point, the set was split into cross-validation, testing, development, and holdout sets. The order was randomized and split in Weka using the RemovePercentage filter, with the first split at 80% to create the holdout set, inverting the selection and splitting again at

25% to create the development set, and inverting the selection again and splitting at 66% to create the test set, and inverting the selection to create the training set. This obtained a split of 40%-20%-20%-20% of the original data for the training set, development set, test set, and holdout set, respectively.

At this point, I switched to Lightside for error analysis to determine problematic features. I standardized the streak feature for the development set for regression and applied the same average and standard deviation of the development set to standardizing the training and test sets. I then trained on the training and test sets combined and evaluated on the development set. I examined frequency, average value, and a high horizontal and low vertical difference. I saw that the opposition team's record column had by far the highest horizontal difference and by far the lowest vertical difference for the both the classification and the regression set.

Consequently, I added one extra feature to give some more context to the opponent's record coming into a game: the opponent's streak. This gave me the features illustrated in Figure 2.

Note that due to the binary nature of the classification data (two classes, W or L), I will rely on accuracy and kappa as the primary determinants of performance for the remainder of this paper, while for regression, I will rely on the correlation coefficient to convey the value of the results.

DATA EXPLORATION

First I take a look at my development set to understand my data a bit better and see what kind of performance I should expect. Several machine learning algorithms were run with this set with a 10-fold cross validation, the results of which can be seen in Figure 3 and Figure 4. I observe from this initial run that logistic regression has the best performance on the classification set, while Gaussian processes has the best performance on the regression set. I also note that when running this, the features with the highest weights were team wins, opponent wins, team streaks, and opponent previous year record.

Algorithm	**Accuracy**	**Kappa**
Naïve Bayes	63.8%	0.2756
Logistic	66.26%	0.3252
SMO	65.7%	0.3126
J48	65.2%	0.3048

Figure 3, development set performance with 10-fold cross-validation, classification

Algorithm	**Correlation coefficient**
Linear Regression	0.3842
Gaussian	0.3933
SMOReg	0.3813
M5P	0.3856

Figure 4, development set performance with 10-fold cross-validation, regression

I noted that the correlation coefficients were on the lower side of what is expected based on previous papers in the regression table, and suspected that this was due to the relatively small sample size. I then ran the bagging meta classifier with each of these algorithms, which yielded the results in Figure 5.

Algorithm	**Correlation coefficient**
Linear Regression	0.4575
Gaussian	0.4536
SMOReg	0.4537
M5P	0.457

Figure 5, development set performance with 10-fold cross-validation and bagging, regression

BASELINE PERFORMANCE AND ERROR ANALYSIS

I subsequently performed a baseline analysis using Weka, training on the training data and testing on the testing data. From the data exploration step, I see that the highest performing algorithms are likely to be Gaussian processes for the regression data set and logistic regression for the classification data set. Before this, though, I run a OneR test using just the value of opposition wins over the course of the season to estimate a naïve baseline solution and understand what kappa and accuracy should look like. This led to 56% accuracy with a kappa statistic of 0.1232.

With logistic regression on the classification data, I see an accuracy of 68.73%, along with a kappa value of 0.3747. When I run Gaussian processes on the regression data, I get a correlation coefficient of 0.4546, with a mean absolute error of 9.6 points.

I then switched to using Lightside to conduct a final error analysis in a similar fashion as described above. Just like I did then, I saw that the opposition team's record had the highest horizontal difference and the lowest vertical difference for the both the classification and the regression set, but this time the streaks also had a relatively high horizontal difference and a relatively low vertical difference, indicating that perhaps streaks are not as good judges of games as the model thought.

Examining this output to determine what exactly went wrong in the regression and classification tasks, however, was quite a challenge. In most of the incorrect instances, game outputs were quite unintuitive, showing no clear correlation to any of the data given. Looking up the corresponding games to the instances in the models sometimes yielded unexpected results as well, with some of the best teams in the league losing to some of the worst on any given game day. This is perhaps reflective of the game of basketball itself, and will be discussed further in the Discussion section.

OPTIMIZATION

In my final step, I attempted to tune the parameters of the algorithms I had chosen for each data set. I observe that for the classification data set, the parameter with the largest impact on performance was maxIts, which outlines the maximum number of iterations to perform for the given regression. To accomplish this, I utilized the CVParameterSelection meta-classifier combined with logistic regression, and tested maxIts with values from -1, the default value, to 16. I saw that 10 was the optimal value, which led to a kappa of 0.3828 and correctly classifying 69.12% of instances. I also ran CVParameterSelection on maxIts with values ranging from -10 to -1, to make sure that there were not lower values with better performance. I saw that 69.12% with a kappa of 0.3828 was still the optimal value. These results indicate that the new parameters only led to an improvement of two correctly classified instances, which is not a statistically significant difference.

Similarly, for the regression task, I chose to tune the exponent of the PolyKernel and the noise level in the Gaussian Process, as done on prior assignments in this class. After experimenting with various combinations of exponents and noise, I found that the combination that worked the best was an exponent of 2 and a noise level of 4.0. With these parameters, I observe a correlation coefficient of 0.4563, which is not statistically significant compared to the baseline.

RESULTS

To obtain my final results, I combined my training and test sets and ran it against the holdout set for each problem for both the regression and classification data sets using Weka defaults for logistic regression on the classification data set and Weka defaults for Gaussian processes on the regression data set. Using logistic regression with parameters of R = 1.0E-8, and M = -1, this yields 68.5% of instances that are correctly classified, with a kappa statistic of 0.3704 on the classification data set. This marks a significant improvement over running a majority-vote algorithm such as OneR, which yields a correctness rate of 56.1% with a kappa of 0.1232 (a chi-squared value of 16.0998 and a p-value of .00006, making this result highly statistically significant). This result is comparable to results in previous papers. Furthermore, using Gaussian processes with parameters L = 1.0, N = 0, and exponent E = 1, I see that the correlation coefficient is 0.4494 on the regression data set. This indicates that this regression model can predict NBA games with a correlation coefficient of almost 0.45.

DISCUSSION AND CONCLUSION

Over the course of this paper, a process for modeling and outlining how to classify and predict NBA game outcomes and point differentials/margins has been discussed and demonstrated. This process includes data extraction and manipulation, feature space design, baseline observation, error analysis, and parameter tuning. The value of this work lies in both the condensed and thought-out feature space, as well as the presence of both a classification and regression model.

From the error analysis, we see that there are an enormous number of underlying factors that go into each game, some of which are simply non-quantifiable, such as injuries during the game or having an off-shooting night or other idiosyncrasies unique to the nature of basketball, or quality of coaching or matchup issues that are very difficult to quantify. Interestingly, optimization of the parameters also indicates that the nature of the regression problem may be under fitted by any kind of linear model, or even the quadratic model that was eventually chosen, and indeed our results confirm that this is most likely the case. Another aspect of this problem that may be worth looking into is the fact that opposition wins, chosen by OneR as the most predictive factor and achieving 56% accuracy based on this alone, is also the problematic feature as identified through the error analysis portion. Diversifying and fleshing out this feature may help achieve even more accurate results in the future.

Work into this area in the future should find a way of creating more expressive and less dependent features that are not related to box statistics like rebounding, assists, or the like (indeed, the roughly equivalent performance of this model to ones that do incorporate these elements indicates that these are not necessarily needed to predict games) and instead reflect intangibles and perhaps ways of representing the players themselves, like comprehensive player ratings optimized for a machine learning task, or modeling other intangibles on the team such as quality of management or coaching. Although there are certainly some aspects of the game that will prevent highly accurate predictions, pursuing more accurate predictions in this new area at the intersection of sports analytics and machine learning is an intellectually – and financially – interesting exercise.

REFERENCES

[1] *Torres, Renato A. Prediction of NBA games based on Machine Learning Methods. University of Wisconsin, Madison. Computer Aided Engineering Center. December 2013. Accessed May 13, 2017. https://homepages.cae.wisc.edu/~ece539/fall13/project/AmorimTorres_rpt.pdf.*

[2] *Grant, Avalon, Bali Batuhan, and Guzman Jesus. Various Machine Learning Approaches to Predicting NBA Score Margins. Stanford University. CS229 Machine Learning Autumn 2016. Accessed May 13, 2017. http://cs229.stanford.edu/proj2016/report/Avalon_balci_guzman_various_ml_approaches_NBA_Scores_report.pdf.*

[3] *"Basketball Statistics and History." Basketball-Reference.com. Accessed May 15, 2017. http://www.basketball-reference.com/.*