Revisiting the Correlation of Basketball Stats and Match Outcome Prediction

Samuel Li
Shanghai American School
2588 Ying Gang East Road #338
China
86-18019466700
Samuel02px2020@saschina.org

ABSTRACT
Professional basketball games, high-stake in nature, have garnered great attention from the research community to model, simulate and predict game outcomes. Despite all the advanced metrics and models developed over the years, predictions remain unreliable and upsets still occur frequently. This paper reinvestigates whether these complex stats perform significantly better over simple statistics, and whether simple statistics benefit from common machine learning generalization and robustification processes.

CCS Concepts
• Computing methodologies→Supervised learning by regression • Computing methodologies→Feature selection • Computing methodologies→Learning linear models

Keywords
Game Prediction;Machine Learning;Feature Selection;Composite Statistic

1. INTRODUCTION
In recent years, advances in technology have populated many traditional sports fields with large amounts of data, ranging from the traditional field statistics to difficultly accumulated composite statistics. Sparked by interests varying from game prediction to team building, different parties are now analyzing available data, attempting to answer the question, "What makes a team win? "

Traditionally, teams have used player statistics and box scores as a means of weighing performance, in hopes of finding the players that best fit their teams. The advent of the internet and mass commercialization of sports has added an increased desire to finding the best team and understanding why they are so. Many people believe that predicting sports at the highest level is a trivial task, yet even the best performers are only able to correctly predict game outcomes around seventy percent of the time, notwithstanding all the information that is available. At the highest level of professional basketball, the National Basketball Association (NBA) represents a playing field with the highest stakes, for betters and players alike. Due to the rigor of games and the closeness in team strength, small factors like injuries and morale often have led to strong correlations to winning or not, yet are underrepresented in data and increasingly difficult to extrapolate from existing statistics. The main goal of this paper is to assess different machine learning methods on varying statistics from the common place box scores to expert crafted composite statistics. We discuss the results of these methods in the context of predicting games and analyzing their performance on games during the season.

ICMLC 2020, February 15–17, 2020, Shenzhen, China

ACM ISBN 978-1-4503-7642-6/20/02…$15.00
DOI: https://doi.org/10.1145/3383972.3383980

2. LITERATURE REVIEW
Becker, Wang, and Papamichael [1] used popular machine learning techniques and previous match data in the 90's to predict game victory with 73% accuracy. Their work concludes that linear regression can be a sufficient algorithm at predicting basketball games. However, later work in [2] notes that there is a glass ceiling in game prediction with machine learning methods. They reasoned that most existing attributes cannot capture the intangibles including factors like experience, leadership, and luck. Another challenge one has in predicting games comes in the form of context bias [3]. One example is recency bias, where humans can easily extrapolate events like injuries from watching recent games, while algorithms must be presented with specially constructed data in order to weigh statistics from recent games more strongly than those in the past. Another is understanding the fundamental inequalities of winning, as a strong team smashing a weak opponent verses one top team edging out their closest opponent carry different implications.

Richardson, Wang, Zhang, and Yu [4] instead emphasize use of the Regularized Adjusted Plus Minus (RAPM) statistic, which is composed of previous player statistics. They argue that use of this sophisticated feature in training is important for understanding how strong offensively and defensively a team can play. They conclude that RAPM is more effective at predicting games compared to box score information due to its composition of high-fidelity game events in its calculation. Similarly, Yang [5] instead calculates the PER statistic for each player, a singular measure of an individual's contribution to a game. He then calculates team wise PER ratings to predicts games using traditional machine learning methods to some success. On the other hand, Torres instead applies machine learning methods to very simple features like win rate at home and away, and achieved decent prediction accuracy [6].

Finally, we can also see similar methods applied to other large team sports such as soccer and rugby. Soccer, as another traditional sport, tends to differentiate from basketball in the difficulty and amount of scoring, which makes it more difficult to predict due to the vast increase in draws. As an example, Ulmer and Fernandez have applied many machine learning methods similar to those above to past game statistics in soccer, but are only able to achieve up to 52% error rate [7]. Rugby on the other hand, is also a volatile sport to predict due to a more isolated audience (primarily in the states) as well as much greater injury odds. Patterson applied many dataset transformations before using the aforementioned methods to predict the margin of victory, getting a best error window of 12.5 points, which he claims is reasonable given that most rugby wins are less than 14 points apart [8].

Nonetheless, all their works mention the difficulty in encompassing complexities like injury information within. In our paper, we aim to replicate the popular machine learning methods on more recent seasons of data, manipulating sliding windows of past team games to predict season games in different ways.

3. EXPERIMENTS

3.1 Dataset and Glossary

We leverage a dataset consisting of NBA statistics from 2012-2018, submitted by Paul Rosetti on Kaggle, an online data science community which allows its users to find and publish data sets as well as explore machine learning [9]. The dataset contains a CSV file that reported each NBA game in the perspective of officials, teams, and players, which gave us the game history in chronological order. It included data from the 2012-2017 NBA regular seasons. Each team plays 82 games in each season, and there are 30 teams in the league, which equals to 2460 total games per season. Since ach game is played between two teams, the number is reduced to 1230 games per season, and 7380 games across 6 seasons. The CSVs not only contains conventional statistics like the number of three pointers made, but also composite statistics referring to different basketball jargon. The following section explains some of the popular terms used in the CSV.

First off, let p denote the amount of points the home team scores and po denote the amount of points the opposing team scores. Then, we have the point differential, which refers to the difference in points the home team scores and the points the opposing team scores.

$$p_d = p - p_o$$

Next, we have possession P, which describes a chance for a team to score, calculated as follows:

$$\text{Field Goals Attempted} = a_{fg}$$
$$\text{Offensive Rebounds} = r_o$$
$$\text{Turnovers} = t$$
$$\text{Free Throws Attempted} = a_{ft}$$
$$P = a_{fg} - r_o + t + (0.4 * a_{ft})$$

Finally, with possessions, we can calculate the offensive and defensive ratings of a team as follows:

$$\text{Offensive Rating (ORtg)}\ o_r = \frac{100 * p}{P}$$
$$\text{Defensive Rating (DRtg)}\ d_r = \frac{100 * p_o}{P}$$

Through these composite statistics we are able to learn relative strengths and weaknesses of teams. In the following section, we mutate the existing dataset to create sliding window averages of each team. Table 1 and 2 gives a glimpse into the elementary and composite statistcs we manipulate.

Table 1. First few rows of elementary features in the dataset

teamPTS	teamFGA	teamORB	teamTO	teamFTA
84	90	18	13	20
94	79	18	21	22
107	75	7	16	28
120	79	5	8	32
99	85	9	12	18

Table 2. First few rows of composite features in the dataset

teamP	teamOrtg	teamDrtg
0.4251	94.4447	105.6882
0.5300	105.6882	94.4447
0.6127	112.6515	126.3381
0.6446	126.3381	112.6515
0.5327	108.1034	99.3678

3.2 Dataset Mutation

Given that the challenge of predicting games comes in extrapolating information before the actual game, we chose to compute a new dataset given the years of NBA statistics. To access the strength of a team before a game, we averaged a sliding window of their past 15 games, creating features representing their past averages, to balance the implications indicated by a single game. Taking a smaller window of games will increase variance, as the individual contribution of each game increases. This causes the corresponding averaged statistics to exaggerate events like wins over weak opponents or shuffled rosters. On the other hand, using a larger window of games also has its consequences, such as making it more difficult to reflect events like recent player injuries. Since the corresponding statistics no longer proportionally represent critical events, classifiers will likely magnify the insignificant. We choose a window of 15 games, as it has tested to be long enough for noisy games to be represented yet not magnified. To prevent duplication of training data, we only consider games from the perspective of the home team, instead of training the classifiers on both home team and away team data.

3.3 Algorithms

In order to predict the result of games, we train 3 different machine learning classifiers, using linear regression, logistic regression, and support vector machines. To do the training, we partition the dataset for games during 2012 to 2014 and assess their accuracies on statistics that build up to games in the 2015, 2016, and 2017 seasons. The accuracies of the models were calculated using the metrics.accuracy_score function from the Sckit-Learn Library and they are shown in Table 3.

I. Linear Regression - A popular method, due to its simplicity, for solving regression problems that require the prediction of a continuous quantity. To predict, it assumes a linear weight for each input feature and calculates its linear combination with the input vector to produce a continuous output. For categorical features, we discretize the categorical values from 0 to (N – 1), where N is the number of different values the feature value can assume. Hence, we represent the actual outcome of a loss or win with 0 and 1. As linear regression outputs a continuous value, we assume that outputs greater

than 0.5 constitute the prediction of a win, and vice versa for a loss.

II. Logistic Regression - Another popular method for solving binary classification problems that classify the elements of a given set into two groups. Unlike linear regression which produces continuous outputs, logistic regression uses log odds to produce a probability fixed between 0 and 1. Hence, we assume probabilities greater than 0.5 to constitute a win, and probabilities below that to mean a loss.

III. Support Vector Machine (SVM) - Support vector machines can solve both regression and classification problems, unlike linear and logistic regression which can only solve one of the two. This paper focuses on its use in classification problems, since prediction is a binary classification problem. SVMs are based upon the idea of finding a hyperplane that best divides a dataset between multiple classes, two in our case. Hence, the hyperplane is simply a line that separates the data into two classes. The support vectors are the data points closest to the hyperplane. Support vector machines are designed to maximize the separation between the support vectors and the hyperplane so that future predictions regarding which class a data point should belong to can be made with more certainty.

Table 3. Prediction Accuracies of Different Methods

Year	Linear	Logistic	SVM
2015	0.66911	0.67154	0.66341
2016	0.63798	0.63334	0.63739
2017	0.64309	0.64228	0.64228

3.4 Feature Selection

Since the purpose of this study is to explain the implications of different statistics on match prediction, we explore a few feature selection methods to find strong correlations to winning or losing. We apply the following methods to our linear model. The first of such methods is LASSO (Least Absolute Shrinkage and Selection Operator), a form of regularization. LASSO regularization is used for primarily two purposes: regularization and feature selection. LASSO regularization puts a constraint on the sum of the absolute values for the weight coefficient of each feature. In order to do so, the method applies a shrinking process where by it penalizes the coefficients of the features, shrinking some of them to zero. The features with non-zero coefficients at the end of the LASSO are selected to be a part of the model. By reducing the amount of features that contribute to the overall prediction, we are able to address the overfitting caused by a large number of relatively insignificant variables. Using a different strength coefficient, we perform LASSO and assess accuracies across 3 different seasons when different numbers of features remain. Table 4 shows the average accuracy alongside the number of features, with the accuracy peaking at 14 features.

Table 4. Average Accuracies of 3 years using LASSO

Number of Features	Average Accuracy
1	0.58375
2	0.64553
3	0.64797
4	0.64851
5	0.64878
6	0.64688
7	0.64852
8	0.64445
9	0.64661
10	0.64472
11	0.64579
12	0.64905
13	0.64916
14	0.65041
15	0.64986
16	0.64959

The second feature selection method used was a correlation matrix, which shows the correlation coefficients between sets of variables (refer to Figure 1). Heat maps — graphical representations of data where the individual values contained in a matrix are represented as colors — were used to display the matrix. Referring to the color scale, the color next to the 0.0 represents two unrelated variables. As the color progressively gets lighter from that point, there is a stronger positive correlation between the two variables. As the color progressively gets darker from that point, there is a stronger negative correlation between the two variables. Due to a heat map containing all the variables being too large and complex, we draw this heat map using 14 variables, as 14 features result in the best average accuracy. The 14 variables post LASSO that have the highest correlation coefficient with the variable we predict, which is a win or loss. The left and top edge of the heat map represents the correlations that each of these variables have with the game result. It is in descending order from top to bottom or left to right depending on whether the left or top edge is considered. The overall shade indicates that the correlations between each of these variables and the game result is not very strong.

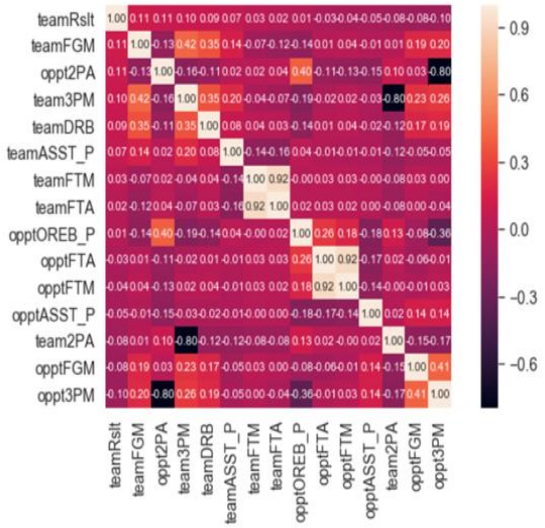

Figure 1. Heatmap of Post Lasso Features

This is expected as an NBA game is extremely complex and not one factor is indicative of the result of a game. Though the correlations are not strong independently, a linear combination of these variables resulting from linear regression will often be far more meaningful and predictive. Through Lasso regression, we select 14 features we believe to be important influencers on the actual prediction. The heat map shows each variables' correlation with each other, which we observe because high correlations between variables would indicate that they are redundant. Most of the heat map indicates weak correlation between the variables,

which means that there is little redundancy as these features are not linear combinations of one another.

Referring to our heat map, near the diagonal spanning from the top left to the bottom right, there are two squares that show a high correlation between the amount of free throws attempted (teamFTA/opptFTA) and the amount of free throws made (teamFTM/opptFTM) by both teams. After careful consideration using intuition and logic, both feature variables were kept within the model because of the unique insights they present. Free throws made contribute to the total points scored — evidently an important indicator for wins. Free throw attempted offers additional knowledge on the ability of the team to draw shooting fouls (free throws are awarded to players who are fouled during shooting motion), which can frustrate the opponent and also force opponent players off the court (a player is no longer allowed to play a game when they reach 6 fouls in a game). Furthermore, the combinations of the two statistics is free throw percentage (free throw percentage=free throws made / free throws attempted), which is highly indicative of a team's offensive strength.

Through feature extraction, we are able to gain unique insights into what contributions each variable (or a combination of the two) can bring to the table for prediction that are unable to be represented by a single variable. As the two feature selection methods provided complementary results on the best variables for a predictive model, making it a relatively easy decision to include all 14 of them. After the features were selected, a postprocessing method was implemented to prune our constructed dataset for selected features. We then rerun linear regression to see the improvements of feature engineering, which is evident in Table 5.

Table 5. Prediction Accuracies After LASSO

Year	Pre-Lasso	Post-Lasso
2015	0.66911	0.67236
2016	0.63798	0.63927
2017	0.64309	0.65447

4. RESULTS AND DISCUSSIONS

4.1 Basketball Stats that Correlates with Match Outcome Prediction

After feature selection, the 14 features with greatest impact are home team field goals made, home team two pointer attempts, home team three pointers made, home team free throws attempted, home team free throws made, home team defensive rebounds, home team assists, opponent field goals made, opponent two pointers attempted, opponent three pointers made, opponent free throws attempted, opponent free throws made, opponent assists, and opponent offensive rebounds. This selection of statistics contains 7 features from the home team and 7 features from the opposing team that show strong correlation to basketball game results. The number of field goals, two pointers, and three pointers made can provide insight into both teams' offensive strength and offensive strategy, whether they prioritize layups and short distance shots verses taking long distance shots. The amount of free throw attempts gives insight on the foul-drawing ability of the teams, while free throws made provide information on the shooting ability of the team. The only statistic that differs between the two teams is rebounding, as the algorithm selected defensive rebounding for the home team and offensive rebounding for the away team. Defensive rebounding records how a team obtains a possession after a miss on their court, while offensive rebounding measures how well a team can maintain possession by stealing a rebound from a missed shot on the opponent's basket. Defensive rebounds are much easier to obtain compared to offensive rebounds because the team on defense is generally closer to the basket. It is expected for the defensive team to get the rebounds most of the time. Offensive rebounds are rare and can act as a huge tempo swing for the offensive team. This shows the importance of home court advantage as the prioritization of defensive rebounds for the home team indicates that as long as the home team guards their basket adequately, they will take the game. On the other hand, for the away team to win, they must rely on stealing rebounds and being extremely aggressive.

4.2 How much does Machine Learning Help with the Prediction?

Although machine learning models are unable to perform as successfully compared to the best experts in the field, whom can consistently predict around 70% of games, they are able to assess a general majority of games accurately. Using our sliding window approach, our predictors are very good at assigning teams high odds of victories when they show consistent victories over stylistically different teams, usually favoring teams like the Golden State Warriors, whom have streaked for long in the regular season. However, a challenge in basketball prediction has always come in the form of upsets, which unfortunately, our predictors using sliding window statistics are not able to predict as well, such as the Lakers' landslide victory over Golden State Warriors on 03/06/2016 or the Denver Nuggets lucking out Golden State Warriors on 01/03/2016.

4.3 Beyond Machine Learning

A long time interest of basketball fans and experts alike is the possibility of using a single statistic to measure the relative strengths of a team. An example of this is the concept of an ELO score [10], a metric commonly used to measure difference between two players in a zero sum game like chess. Given that machine learning algorithms produce a weighted vector of coefficients that used in conjunction with features results in a variable representing win and loss, we decompose the feature weights post selection between the 7 features of the home and away team. As shown in Figure 2, for each game of the season, we multiply the corresponding features of the home team with the home team coefficients, and the corresponding features of the away team with the away team coefficients, culminating in a number α. If α is greater or equal to 0.5, a ceiling function is used, producing a value of 1 which symbolizes a home team win. If α is lesser to 0.5, a floor function is used, producing a value of 0, or a home team loss. The logic is further illustrated in Figure 3. We find that a naïve predictor that uses this singular "matchup strength" metric for both teams, is able predict basketball games with a decent accuracy.

$$\alpha = 0.039*HomeFGM + 0.019*Home2PA + 0.026*Home3PM + 0.022*HomeFTA + 0.032*HomeFTM + 0.018*HomeDRB + 0.003*HomeAST - (0.044*AwayFGM + 0.022*Away2PA + 0.019*Away3PM + 0.009*AwayFTA + 0.016*AwayFTM + 0.001AwayAST + 0.004*AwayORB)$$

$$if\ \alpha \geq 0.5,\quad x = \lceil \alpha \rceil$$

$$if\ \alpha < 0.5,\quad x = \lceil \alpha \rceil$$

Figure 2: Algorithm behind our statistic

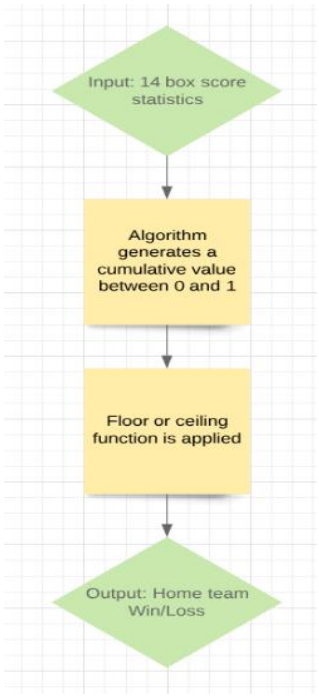

Figure 3: Flowchart of Algorithm

Table 6. First few rows of our statistic as a feature

Our Statistic	Actual Result	Our Result
0.63674	Loss	Win
0.59008	Win	Win
0.41658	Win	Loss
0.55299	Win	Win
0.47444	Loss	Loss

Table 7. Prediction Accuracies using our statistic

Year	Naive
2015	0.63021
2016	0.65120
2017	0.62501

5. CONCLUSION

All in all, our results exceeded the predictive accuracy of a randomizer, which would predict either true or false for every game and theoretically achieve 50% accuracy, and even prediction accuracy guarantees listed on websites that provide betting advice, which claimed an approximate accuracy of 62%. Although our results do not exceed the predictive accuracy of established NBA experts, whom predict with around 71% accuracy, there are still interesting conclusions gained and additional areas of interest.

Using sliding window averages, we were able to teach models to exhibit recency bias and to prefer teams that traditionally did well in the past by providing data for a large collection of features. Performing feature selection using lasso regularization and heatmap visualization allows us to realize accurate predictions of 67% using only 14 features. Interestingly enough, compositing our weights and features into a singular sliding strength value is even able to predict with up to 65% accuracy, as shown in Table 7. However, our method did not generalize well for events like player injuries and sudden upsets. An injury of a star player can turn a great team into an average one, yet even the use of a sliding window did not fully capture and extrapolate such information from previous game data. A simple adjustment could be for player injuries to be included as two feature for each team. Still, it would be difficult for the model to extrapolate the full consequence of an injury, as the injury of a mostly benched rookie verses the injury of a star player cannot be gained from integers.

Even with these deficiencies, the model was able to predict with an accuracy of up to 67%. There are several improvements that can be made in addition to the paper. One would be the application of sliding windows on player statistics, which may better hint at team performance and potential injuries. However, it would require additional implementation details to deal with player hiatuses and new rookies entering the league. Another potential point of interest is the inclusion of playoff statistics into the picture, as teams will often play their opposing teams multiple times in a short span of time. With these improvements, it may be plausible to perform near the standard of NBA experts, which is extremely impressive given that they have spent years watching and understanding the game. Due to the inherent randomness and unquantifiable nature of basketball, the NBA will remain an interesting challenge to predict and an area of machine learning application in the near future.

6. REFERENCES

[1] M. Beckler, H. Wang, M. Papamichael, “NBA Oracle,” Technical Report, 2009

[2] A. Zimmermann, S. Moorthy, and Z. Shi, “Predicting college basketball match outcomes using machine learning techniques: some results and lessons learned,” Technical Report, Oct 2013.

[3] E. Heit, P. Price, and G. Bower, “A Model for Predicting the Outcomes of Basketball Games,” *Applied Cognitive Psychology, Vol-8 621-639*, Jan. 1994

[4] L. Richardson, D. Wang, C. Zhang, X. Yu, “NBA Predictions,” Technical Report, Dec. 11 2014

[5] S. Yang, “Predicting Regular Season Results of NBA Teams Based on Regression Analysis of Common Basketball Statistics,” Honors Thesis, May 2015

[6] R. Torres, “Prediction of NBA Games based on Machine Learning Methods,” Technical Report, 2013

[7] B. Ulmer, M. Fernandez, “Predicting Soccer Match Results in the English Premier League,” Technical Report, 2014

[8] G. Patterson, “Predicting International Rugby Scores,” Technical Report, 2017

[9] P. Rossotti, 2018, “NBA Enhanced Box Score and Standings (2012 - 2018),” Retrieved from https://kaggle.com/pablote/nba-enhanced-stats.

[10] FiveThirtyEight, 2018, “The Complete History of The NBA,” Retrieved from https://projects.fivethirtyeight.com/complete-history-of-the-nba.