Machine Learning Methods for Basketball Game Prediction

Kevin Cleary, Hicham Derkaoui, Jared Elinger, Siddarth Mehta, Ziyi Wang

Georgia Institute of Technology

1 Project Summary

With the legalization of sports betting, there is a growing interest in predicting the outcome of sporting events. This adds a new revenue stream to the already massive amount of money involved in sports [1]. Thus, there is a natural and predictable desire to be able to accurately predict the outcome of games for both the team owners/management who are trying to maximize the team's success, and thus likely revenue, as for the independent bettors. Development of a more efficient machine learning approach to develop a classifier could not only lead to potentially improved betting accuracy, but also equip franchises with better analysis tools to provide a competitive advantage in identifying what are the key factors in order to be able to win games. Such insight could directly affect the decisions made in player drafting/recruiting.

There have long been questions and discussion about the difference between the NBA, NCAA (college) basketball and the EuroLeague. Applying a classifier trained on NBA data and observing its performance on other leagues may provide insight into the similarities and differences between them. Further insight could be gained by training classifiers on each of the data sets and observing differences in weights as well as what features in the feature vectors are most important. The methods for developing the classifiers are discussed in the next section. Our goal is to develop a classifier to predict who will win a game based on only data that had "already occurred". We will replicate techniques previously explored as well as experiment with a newly proposed feature vector. We will then use the NBA trained classifier on data from other leagues to try and see its accuracy and potential to be used on other leagues as NBA data is far more abundant.

In order to achieve these tasks, we began by using a published methodology for NBA game prediction in [2]. We used the feature vector suggested in [2], and tested a series of different classifiers (LDA, QDA, Logistic Regression, SVM, PLA, and MLP). Each of these classifiers accuracy was recorded. We then used the classifiers trained on NBA data to test on NCAA and EuroLeague Data to see the feasibility of training on NBA data for testing on other leagues. We then also trained classifiers using the NCAA and EuroLeague data as a baseline comparison. To attempt to improve on the classifiers, we attempted to use a new feature vector that used the four factors to try and predict the winner by using the average four factor value of each team leading into the game. [3] We calculated a correlation between the four factors and winning when having the post game results as well to see the validity of using the four factors. All of the same classifiers trained on NBA data were tested and compared to natively trained classifiers with all results presented within.

2 Data

2.1 NBA Data

The NBA data was prepared by Siddarth and Pat (Ziyi). The NBA data was taken from a Kaggle submission (https://www.kaggle.com/ionaskel/nba-games-stats-from-2014-to-2018). The dataset contains team stats from every game from the 2014-2015 season to the 2017-2018 season. The data is structured by having team one's raw stats, team two's raw stats, and team one's outcome of the game. There are 30 teams in the NBA who each play 82 games. Over 4 seasons, we have a total of 9840 entry points. However some of these are "soft repeats" where there is another entry with the team stats are presented in the opposite order and outcome is flipped. So in reality, we really have actually 4920 unique data points and the other half are the "soft repeats". The NBA data can be viewed in the 'nba_data.csv' file submitted with our code.

2.2 NCAA Data

The NCAA data was prepared by Jared. Like the NBA data, the NCAA data was taken from a Kaggle submission (https://www.kaggle.com/c/mens-machine-learning-competition-2018/data). It contained data

for the 2017 season. The data was broken down by season and contained data formulated as team one stats, team two stats, (including a team name identifier) and then a set of logicals for if team 1 was the home team and if team 1 won or not. There were 12008 data points, but each game had a redundant data point with each team as team 1 and each as team 2, so there were approximately 6000 unique datapoints. More seasons were not used as the code to order and doctor the data to be usable was quite time intensive and suffered from dimensionality issues as unlike the NBA, teams can play varying numbers of games against various opponents. The dataset will be included in the submitted files. Even with the one season the number of data points was similar to the NBA data set size.

2.3 EuroLeague Data

The EuroLeague Data was prepared by Kevin. The EuroLeague data was scraped from the league's official website: www.euroleague.net. The scraper was provided by Horvath [4]. We used 2012-2016 seasons as training data and used 2017-2018 seasons as testing data. Over the course of the 6 seasons we have 1761 unique games. The format of the EuroLeague has changed over the years. In 2016-2017 it adopted the current model. 16 club teams from European states play each other once, totalling a 30 round regular season. The top 8 teams make the first round of the playoffs which is a best of 5 game series. The Final Four and Championship game are single elimination games. Considering the scarcity of data, we decided to include both regular season and playoff games from pre-2016 format and after. The EuroLeague data can be viewed in the 'Euroleague 2012-2019.csv' file submitted with our code.

3 Features

3.1 Wisconsin

The Wisconsin classifier code was written by Pat (Ziyi). The Wisconsin features were used in [2], where the authors used them to predict NBA game results from the 2006-2007 season to 2012-2013 season with different classifiers. The features are:

1. Win-loss percentage (Home team)
2. Average point differential (Home team)
3. Win-loss percentage in previous 8 games (Home team)
4. Win-loss percentage at home (Home team)
5. Win-loss percentage (Visitor team)
6. Average point differential (Visitor team)
7. Win-loss percentage in previous 8 games (Visitor team)
8. Win-loss percentage as visitor (Visitor team)

The Wisconsin features were the first choice for this project since they are commonly used in traditional sports analytics and easy to interpret. Additionally, their previous usage in [2] provide us with a performance baseline to compare our results against. The features capture a lot of intuitively valuable data about a team. The previous 8 game win rate weights how the team has been doing recently. The point differential gives a better measure of to what degree the team won or lost. The win rate home or away takes into account the importance of home field advantage on the game.

3.2 Four Factors

The Four Factors analysis code was performed by Siddarth. The four factors were coined by Dean Oliver as the four stats most critical for basketball teams to win games. The four factors are shooting, turnovers, rebounding, and free throws. The statistics used to quantify these factors are effective field goal percentage (eFG%), turnover percentage (TOV%), offensive rebounding percentage (ORB%), and free throw rate (FTR). Their derivations are shown below

$eFG\%: \frac{(FGM + 0.5 * 3FGM)}{FGA}$ where FGM is field goals made, 3FGM is three point field goals made, and FGA is field goals attempted.

$TOV\%: \frac{TOV}{FGA + 0.44 * FTA + TOV}$ where TOV is turnovers and FTA is free throws attempted

$ORB\%: \frac{ORB}{ORB + oppDRB}$ where ORB is offensive rebounds and oppDRB is opponent's defensive rebounds

$FTR: \frac{FTM}{FGA}$ where FTM is free throws made

We first wanted to see how strong the correlation was between the four factors and actually winning basketball games. Given both teams' four factors from a game, could we learn and predict who won the game? The results of being able to predict games from the actual four factors using a variety of classification algorithms are shown below.

Algorithm	Accuracy
Logistic Regression	93.3%
LDA	94.9%
QDA	94.6%
PLA	94.7%
SVM	93.0%

Table 1: Four factors winning correlation

The table shows that there is a very strong correlation between four factors and winning and therefore four factors might be a mechanism worth exploring for game prediction. However, if we are trying to predict the outcome for a game, we will not have any teams' four factors from the game beforehand. Therefore we must use an **estimate** of the four factors and see if we can learn to predict the outcome of basketball games from this **estimate**. In our project we used each team's seasonal average of the four factors going into the games as our estimation. The x_n and y_n for every data point in our four factor data set adhered to the following format:
$x_n = [team1\_4factors, team2\_4factors]$

$$y_n = \begin{cases} 1 & \text{if team1 wins} \\ -1 & \text{if team1 loses} \end{cases}$$

4 Methods

4.1 Data Processing

Before implementing the classifiers, the data was first preprocessed into the desired format. All data files were originally organized as game logs and preprocessed by: (1) discarding repeated entries; (2) calculating Wisconsin features and four factors from game statistics; (3) organize each data entry as [Home team

Wisconsin/Four factor features, Visitor team Wisconsin/Four factor features, Home team game result]. Additionally, principle component analysis (PCA) was done on the Wisconsin features, and the data was projected onto a low dimensional subspace.

4.2 Classifiers

The classifiers used in this project include: Linear Discriminamt Analysis (LDA), Quadratic Discriminant Analysis (QDA), Logistic Regression (LR), Support Vector Machine (SVM), Perceptron Learning Algorithm (PLA), Multi-layer Percepron (MLP). We chose a comprehensive classifier list available in sklearn to investigate the effectiveness of different decision boundaries on a classification task that should in theory be highly nonlinear.

4.3 Training

To train the classifiers, the data was first shuffled and separated into training, validation and testing sets. All the hyperparameters of the classifiers were found using grid search within the validation set. The test set was used to report the prediction results.

5 Results

The result tables and figures were created by Hicham along with the poster creation.

5.1 Baseline

Before testing the trained classifiers, we first took a naive approach to the game prediction problem by always predicting the team with higher overall win rate as the winner. Our goal is to outperform the naive approach. This approach was tested on only the test data set and achieved prediction rates of:

League	Prediction rate
NBA	64%
EuroLeague	66%
NCAA	62%

Table 2: Baseline prediction rate

5.2 Wisconsin

Before implementation of the classifiers, PCA was performed on the Wisconsin features, and the singular values of the features are (ranked from highest to lowest) $[298.7, 289.4, 12.6, 12.0, 6.5, 6.0, 2.6, 2.5]$. Using this information, the data was projected onto a 4-dimensional subspace (captures 97% of variance in data). The data was first used to test the effectiveness of the classifiers when training and predicting on the same data set:

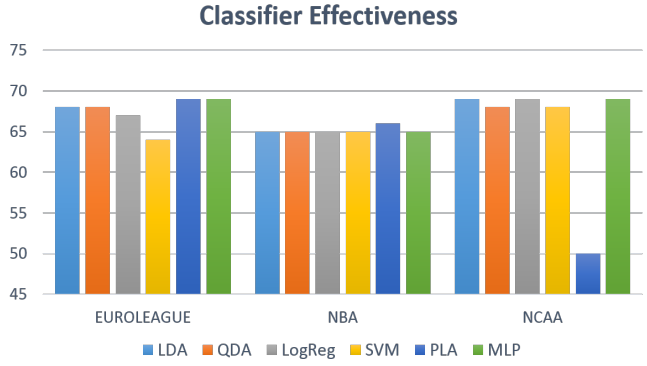

Figure 2: Classifier behavior on Wisconsin features

Classifier Effectiveness (%)				
	Euroleague	NBA	NCAA	*Mean*
LDA	68	65	69	67.3
QDA	68	65	68	67
LogReg	67	65	69	67
SVM	64	65	68	65.7
PLA	69	66	50	61.7
MLP	69	65	69	67.7

Table 4: Classifier Effectiveness

As we can see, the classifiers performed better than our naive approach (except for PLA on NCAA). They all tend to have the same level of effectiveness, with MLP being a bit better on average.

We then went one step further in trying to generalize our approach by looking at the classifier effectiveness when training on the NBA data set and predicting on the two others:

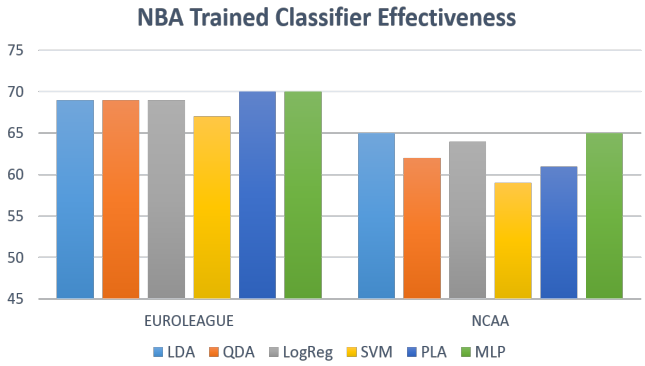

Figure 4: NBA trained Wisconsin classifiers on other leagues

NBA Trained Classifier Effectiveness (%)			
	Euroleague	NCAA	*Mean*
LDA	69	65	67
QDA	69	62	65.5
LogReg	69	64	66.5
SVM	67	59	63
PLA	70	61	65.5
MLP	70	65	67.5

Table 6: NBA Trained Classifier Effectiveness

Here the results proved to be very interesting. The classifiers trained on NBA data all generalize well to EuroLeague and NCAA data sets, with PLA and MLP giving even better results than when training and predicting on the same data set. LDA and logistic regression are also very consistent.

5.3 Four Factors

We then performed these two same approaches on the four factors:

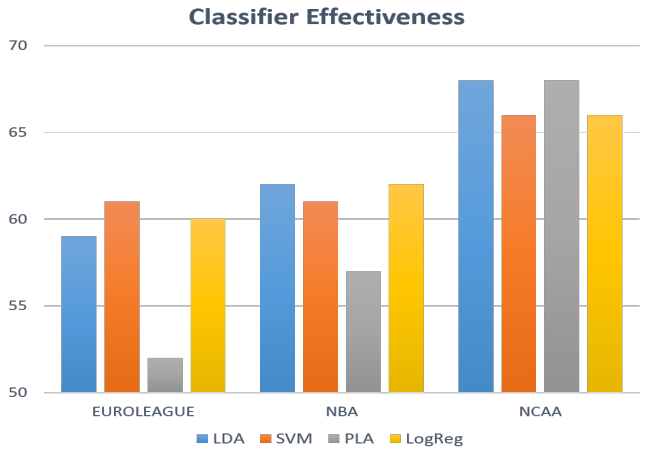

Figure 6: Classifier behavior on Four factors features

Classifier Effectiveness (%)				
	Euroleague	NBA	NCAA	*Mean*
LDA	59	62	68	63
SVM	61	61	66	62.7
PLA	52	57	68	59
LogReg	60	62	66	62.7

Table 8: Classifier Effectiveness

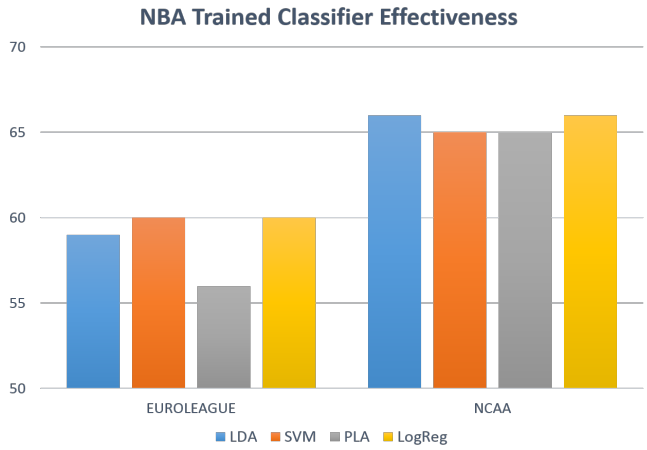

Figure 8: NBA Four Factors classifiers on other leagues

NBA Trained Classifier Effectiveness (%)			
	Euroleague	NCAA	*Mean*
LDA	59	66	62.5
SVM	60	65	62.5
PLA	56	65	60.5
LogReg	60	66	63

Table 10: NBA Trained Classifier Effectiveness

The results observed overlap with those on the Wisconsin features except that NCAA predictions are overall better with four factors.

6 Conclusion

Based on the results from the experiments of this paper we were able to see multiple interesting conclusions. The first is that training on NBA data for testing on other leagues was effective as there was only a couple of percentage points decrease in effectiveness in the classifier accuracy. This is helpful as NBA data is far easier to come by and sort then accumulating NCAA or EuroLeague data, which would make studying/prediction far easier and allow for a far larger training set. Additionally, we saw that there was not a significant difference in the accuracy of the classifier used, but in general MLP and LDA were the most effective followed by logistic regression and lastly was SVM and PLA. Finally, we saw that prediction on NCAA games and EuroLeague was far more effective than the NBA. This likely stems from the fact that there is far less parity in the college game and EuroLeague than in the NBA. If the analysis was redone on only instances where top tier teams played each other in the NCAA or EuroLeague, the results would likely approach the results seen from the NBA. Finally, we saw that the Four Factors had a very high correlation with predicting who would win the game if post game stats were known apriori. Thus, perhaps a better estimator of the four factors than the running average up until the start of the game would increase the effectiveness of a classifier based on projected four factor values. However, even with the simple approximation of the four factor values the classifier had a similar accuracy to the previously suggested Wisconsin feature set.

References

[1] Rubino, Daniel. How much money do Americans bet on sports. [Online]. Available: https://www.legalsportsbetting.com/how-much-money-do-americans-bet-on-sports/

[2] Torres, Renato Amorim and Hu, Y.H., "Prediction of NBA games based on Machine Learning Methods," *University of Wisconsin Madison*, Dec 2013.

[3] Kotzias, Konstantinos. (2018, march) Four Factors of Basketball as a Measure of Success. [Online]. Available: https://statathlon.com/four-factors-basketball-success/

[4] Horvat,Tomislav Job,Josip Medved,Vladimir. (2018, August) Prediction of Euroleague Games based on Supervised Classification Algorithm k-Nearest Neighbours. [Online]. Available: https://www.bib.irb.hr/969257/download/969257.K-BioS_2018_8_CR.pdf