Predicting NBA Games Based on Machine Learning Methods

Zihao Li, Shuo Yang

Abstract—In this report, we predict the results of basketball games in the National Basketball Association (NBA) using machine learning algorithms. Based on the dataset, an accuracy rate upper bound is given. Moreover, we create four different classifiers: Support Vector Machine (SVM), K-Nearest Neighbor(KNN), Random Forest (RF) and our own combined model. Our prediction is in one of two classes for each game: win or loss. Our best accuracy rates were with our combined model: regular season is 67.6% and playoff is 63.4%.

Index Terms—Machine learning, National Basketball Association, Prediction.

I. Introduction

A. Motivation

NATIONAL Basketball Association (NBA) is a men's professional basketball league in North America, composed of 30 teams (29 in the United States and 1 in Canada). It is one of the four major professional sports leagues in the United States and Canada, and is widely considered to be the premier men's professional basketball league in the world. In China, basketball is the most popular sport, so NBA is so popular among Chinese. As the league involves a lot of money and fans, it is meaningful to simulate winning teams, to analyze player's performance and to assist coaches. The prediction on other popular sports, such as soccer and American football have been considered, however, predicting NBA games has seldom been considered before. Therefore, this work is of great necessity. Through the years, a lot of data have been collected based on NBA and each day the data become richer and more detailed. However, it is still very challenging to predict the game, even though almost all of the data are available.

Machine learning (ML) is one of the intelligent methodologies that have shown promising results in the domains of classification and prediction. In essence, it provides us the perspective to address problem by data-driven method, which is proven to be effective. There are many classical theory and algorithms; see, e.g.,[2],[11]. Therefore, we hope to achieve better predictions by using machine learning methods, which is exactly the objective of this report.

This work is instructed by Prof. Enmei Tu.

Zihao Li and Shuo Yang are with Department of Automation, Shanghai Jiao Tong University, Shanghai 200240, China. {lizihao, xiang-yang}@sjtu.edu.cn.

B. Our Contributions

In this report, we predict both the regular season and playoff of NBA by using KNN, Random Forest, SVM, neural network and our combined model. Also, we use some methods to do feature selection, such as PCA. Moreover, we give the accuracy rate upper bound analysis of our dataset. Finally, our best prediction rate of regular season is 67.6% and best prediction rate of playoff is 63.4%.

C. Related Works

Sports results prediction by using machine leraning has drawn many attentions recently in the literature, see, e.g., [14], [9], [1], [4],[10],[5],[12]. However, most of the work on predicting sports result focus on soccer and basketball has seldom been considered. [5] and [12] are two of the few papers considering predicting basketball games, which just predicting regular seasons and the accuracy rate is not high. For example, in [12], only regular season has been predicted and the best accuracy rate is just 67.0%.

In the machine learning literature, the algorithms of neural network, K-Nearest Neighbors, Random Forest, Support Vector Machine and Principal Component Analysis (PCA) have also been studied extensively in the past few years; see, e.g.,[8], [6], [3], [7].

D. Organization

The rest of this report is organized as follows. In Section II, we introduce some necessary preliminaries. Then we discuss feature cap analysis, dataset preparation, feature selection in Section III. In Section IV, we propose our models and algorithms design. Finally, we conclude by Section V. Section VI is acknowledgement.

II. Preliminary

A. NBA

In NBA, there are 82 games for a team during regular season, 41 at home and 41 away. The current league organization divides thirty teams into two conferences of three divisions with five teams each, which is shown Figure 1. Each team hosts and visits every other team at least once every season.

The NBA playoffs begin in April after the conclusion of the regular season with the top eight teams in each conference, regardless of divisional alignment, competing for the league's championship title. The playoffs follow a tournament format. Each team plays an opponent in a

Division	Eastern Conference			Western Conference		
	Central	Atlantic	Southeast	Pacific	Northwest	Southwest
Team Quantity	5	5	5	5	5	5

Fig. 1. Distribution of NBA teams

best-of-seven series, with the first team to win four games advancing into the next round, while the other team is eliminated from the playoffs. In every round, the best-of-7 series follows a 22111 home-court pattern, meaning that one team will have home court in games 1, 2, 5, and 7, while the other plays at home in games 3, 4, and 6. The final playoff round, a best-of-seven series between the victors of both conferences, is known as the NBA Finals, and is held annually in June.

B. Dataset

Our dataset contain the results of 19 seasons (from the 2000-01 season to the 2018-19 season) of the NBA. In all, there are about 20000 games in regular seasons and 1500 games in playoffs. For each game, our dataset includs the home team, the away team, the winner, Field Goal Attempt, Free Throw, Assists and so on. We obtain our dataset from the website [13].

III. Feature Engineering

A. Feature Cap Analysis

We believe that data and features determine the upper limit of machine learning, and models and algorithms are used to approach this upper limit. Therefore, the best classification effect that can be obtained on a set of features is significant to us, which means we can have a clearer evaluation of our models and algorithms.

For a given dataset

$$\{y_i, x_i | y_i \in Y, x_i \in X\}, i = 1, 2 \dots k \tag{1}$$

where Y is the set of category tags and X is the set of all sample feature values.

The classification performance of all classifiers must not exceed the upper limit determined by this data set. We consider a classifier with the best classification performance, which is denoted by C_{best}. First of all, we know that C_{best} is essentially a function map, and has an unique output value for any input. In the real environment, our collection of continuous values of features is limited by the measurement accuracy of the measurement tool or artificial regulations (we often keep the decimal places). Actually, the values obtained are strictly discretized, and we denote the minimum division d. Note that there exists a limited value L, and we have $d^{-1} < L$.

Based on the actual conditions, we can construct a continuous function f_c, so that we can classify according to this function to obtain the same performance as C_{best}, the construction method is as followed,

$$f(x_i) = class(y_i)/N_{class}, i = 1, 2 \dots, k \tag{2}$$
$$f(x') = f(x_1) + d(x') * (x' - x_1) \tag{3}$$
$$d(x') = (f(x_2) - f(x_1))/(x_2 - x_1) \tag{4}$$
$$x_2 = min\{x_i | x_i > x'\}, x' \notin X \tag{5}$$

Where $class(y_i)$ indicates the class that C_{best} classifies the samples with value of x_i according to y_i; N_{class} is the total number of classes for all samples.

Then, let us discuss the classification strategy based on f_c. For a sample x, if

$$f_c(x) \in [(k - 0.5)/N_{class}, (k + 0.5)/N_{class}) \tag{6}$$

then it will be divided into the class k. Obviously, for any sample in X, the classifier based on f_c will give the same classification result as C_{best}, which means the this classifier has the same classification performance as C_{best} on dataset X.

Theoretically, neural network has the ability to fit any continuous function. Because we have guaranteed that f_c is a continuous function, we must be able to fit f_c through the neural network. In other words, the optimal classifier C_{best} must can be directly expressed in the form of a neural network.

In our specific problem, because the NBA game format (game rules, common tactics) has not changed much in the past two decades, the distribution of game data in each season is very similar. We collected the game data for the past 19 seasons, which is the S we mentioned above. It can be considered that this dataset basically reflects the true distribution of the total sample space, so our best classifier C_{best} should have the similar classification performance on S_0 and S, which means that the classification accuracy is roughly the same. Generally speaking, a good classifier must have strong generalization performance, so the classification accuracy of C_{best} on S should be slightly less than the training classification accuracy obtained by overfitting S with a neural network (training accuracy), so we can design a slightly more complex neural network to train on the data set S until overfitting or the training accuracy no longer changes greatly. At this time, the training accuracy of our data is our estimation of the upper limit of the set features, and this estimation is slightly higher than the true classification accuracy of C_{best}.

B. Dataset Preparation

1) Dynamic Features: Based on the playoffs schedule table, we generated the home and away wins and losses of each team in the NBA playoffs in the first round of knockout. In the data set, $['HW','HL','VW','VL']$ is used. This field represents the home team's home wins and losses and away wins and losses in this round of knockouts. This set of characteristics is called "dynamic characteristics" because as the schedule advances, these four data for a team are constantly changing. The reason we chose these

four data as the characteristics to learn is that in the playoffs, the relationship between the games that have ended in a round will greatly affect the outcome of the next game. There are statistics, take The team that plays "Tian Wang Shan" (the fifth game in the first four games with a 2-2 draw) has a 80% probability of winning the sixth game.

2) Static Features: The static feature consists of two parts. The first part, which is denoted by $['HM'_r,'HL'_r,'VW'_r,'VL'_r]$, is similar to the dynamic feature. It is the playing situation between these two teams in the regular season. The second part is the average performance of the two teams in 82 regular season games, including average field goal percentage, rebounds, blocks and other data. Because these data will not change in the playoffs, so we name it "static features". We selected these data because they could reflect the overall ability of the two teams and the relationship between the styles of these two teams.

C. Features Selection

When processing classification problems, we hope that the feature dimensions be as small as possible and this will help us understand the data distribution better and design a more reasonable algorithm. However, this happens under the premise that these features can provide us with sufficient evidence for sample classification, so we need to give a feature upper bound analysis for the generated dataset.

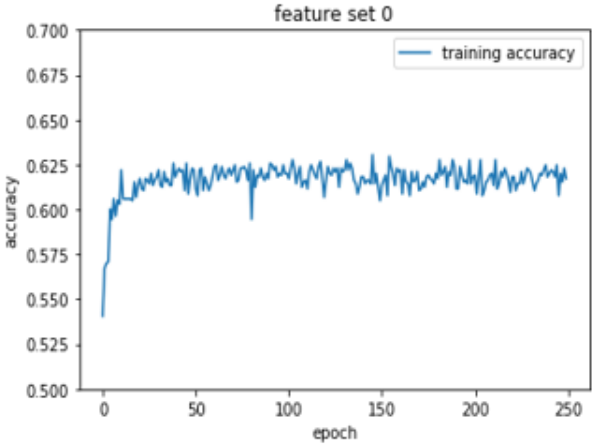


Fig. 2. feature set0 upper bound

1) Upper Bound Analysis of Dynamic Features (4 Dimensions): By using the method of overfitting neural network for the upper bound analysis, we estimate that if only four-dimensional dynamic features are used, then the classification accuracy rate finally achieved will not exceed 63 %. Also, we use SVM to perform 19-fold cross-validation and get 62.4 % prediction accuracy, which is very close to the upper limit of our prediction. Therefore, if we want to improve the performance of the classifier, we need to add more features.

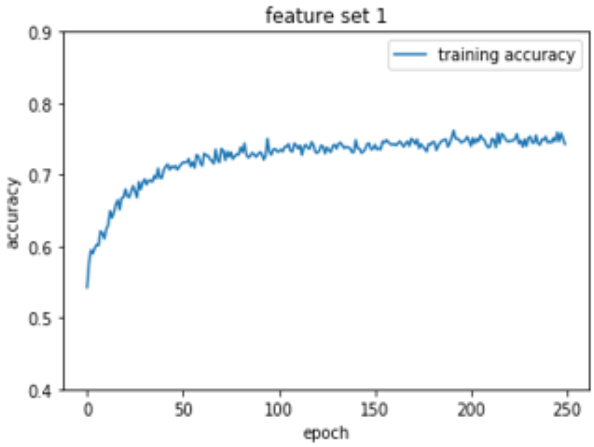
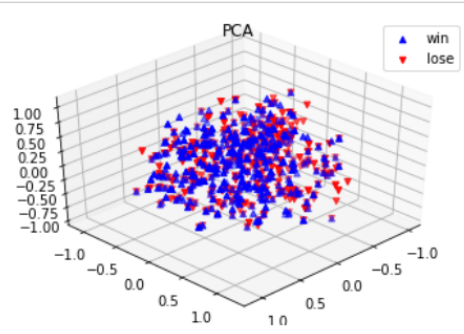


Fig. 3. feature set1 upper bound

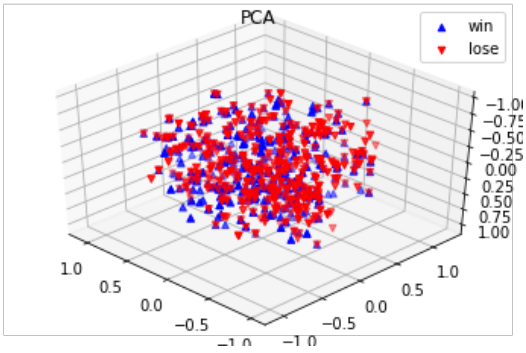
2) Upper Bound Analysis of Dynamic Features and Static Features (38 Dimensions): We add 34-dimensional data of static features on the basis of 4-dimensional dynamic features. As shown in Figure 3, our estimate of the upper limit has reached above 75%, which actually is a so high value of predicting competitive sports. Note that our final algorithm and model training are based on this 38-dimensional data.

3) Analysis of Problem Type: About 1500 playoff games in the last 19 seasons have been collected, and the features dimension is 38, which means that we had better have dimensionality reduction. Our purpose is to generate interpretable classifier. In order to extract useful advanced features we start to reduce the dimensionality of static features. The results of PCA are shown in Figure 4 and Figure 5. After reducing dimensionality to 3 by PCA, based on the result of the overall discus distribution, we find that the color of sample point is single along with a straight line. Therefore, we believe linear classifier is suitable for static features.



4) Dimensionality Reduction:

a) Degree of Divergence: Standard deviation of the 30 features, which represent divergence degree, are computed hoping to eliminate features with little variance. The result is shown in Figure 6. However, these features have similar divergence, so dimensionality can not be reduced by this way.




```
#特征选择
import copy
Traindata_fs = copy.deepcopy(Traindata)
from sklearn.feature_selection import VarianceThreshold
selector = VarianceThreshold()
selector.fit(Traindata_fs[:,8:])
print(selector.variances_)
```

```
[0.04882495 0.0329095  0.02821657 0.02866035 0.02549758 0.04062591
 0.03851396 0.03313992 0.04079221 0.03225576 0.02836147 0.03453203
 0.0291991  0.03411871 0.03322675 0.04944571 0.03342002 0.02812403
 0.02899111 0.02571799 0.0422552  0.03963887 0.03236769 0.04142644
 0.03187817 0.03005684 0.03513378 0.03041924 0.03057977 0.03198715]
```

Fig. 6. variance

b) Correlation: We use Lasso regression to perform correlation analysis. By adjusting the weight of the l1 penalty term, we retain different numbers of features, and then use these features to perform linear regression classification to compare the effect of the number of retained features on classification accuracy. We find that as the number of dimensions decreases, the accuracy of our cross-validation generally decreases. In the end, all 34-dimensional features retain the best effect, so we decided not to perform dimensionality reduction processing, and directly use these 34-dimensional features to build linear classifier. The result is shown in Figure 7.

5) Timeliness: When adjusting the parameters of the linear classifier, we found that if the data of the 19 seasons were divided into different parts in chronological order and trained separately, the classification performance of our linear classifier on static features would be improved. The result is shown in Table III-C5.

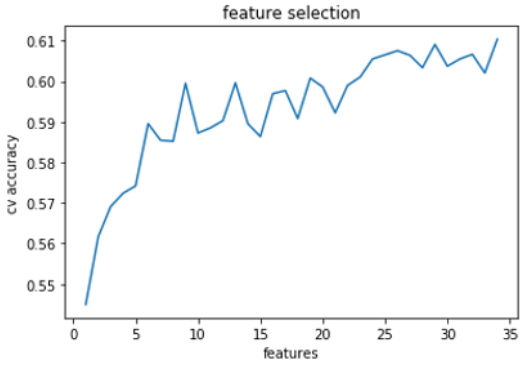


Fig. 7. lasso regression

Cross-Validation accuracy for four periods

season	CV accuracy
00-18	60.3%
00-06	61.2%
07-13	60.6%
14-18	65.1%

We observed the 19 years of playoff data and found that the types of players who played a key role in the playoffs have probably changed from high-height centers to guard and forwards.The style of play in the NBA has also changed from the main inside transition to run and blast tactics in recent years. We used the chi-square independence test to examine the factors most relevant to the outcome of the game over the three periods. The results are shown in Figure 8, Figure 9 and Figure 10.

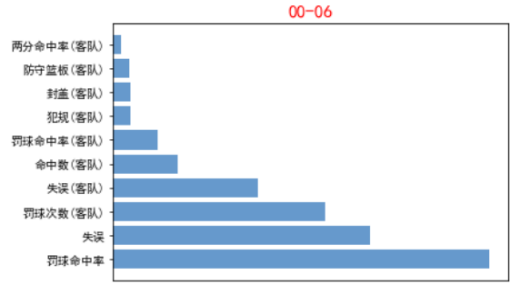


Fig. 8. 00-06 correlation

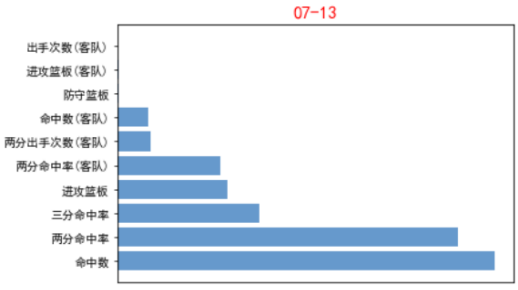


Fig. 9. 07-13 correlation

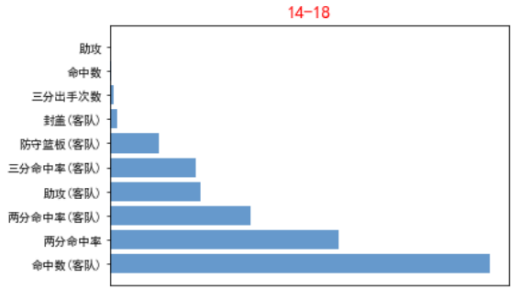


Fig. 10. 14-18 correlation

It can be seen from these three pictures that the correlation between the characteristics of each of the three periods and the results of the game has changed

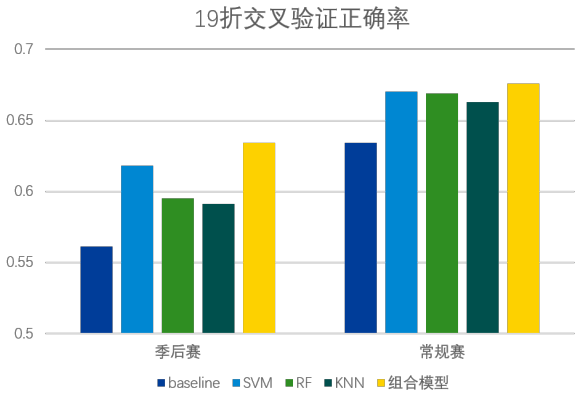


Fig. 11. CV accuracy

greatly. These results can help us better understand the key factors on the basketball court in reality, and can provide suggestions for team managers on team play and style building; on the other hand, it also shows that when we are performing data mining, full consideration must be given to the timeliness of the data.

IV. Models and Algorithms Design

In Section III-C1, we mentioned that the classification effect of approaching the upper limit has been achieved with SVM on dynamic features, where we used a polynomial kernel and achieved very poor results when trying to use a linear kernel. Obviously, we should not use a linear classifier to consider all the features together. Therefore, ensemble learning is used to classify the classification results, which are obtained by learning from dynamic features, and the classification results, which are obtained by learning from static features, as refined high-level features. Moreover, the refined high-level features are the input of another classifier, which finally generates the category prediction of the samples as the output. Note that the learning from dynamic features is done by a nonlinear classifier.

We take the probabilistic output of SVM (dynamic feature) and Logistic Regression (static feature) as the input of KNN. Finally we obtain $K = 100$ after selecting the optimal parameters through cross-validation. The results are shown in Figure 11. Note that all results in our project are based on 19-fold cross-validation.

V. Conclusion

In this project, we make judgments on the type of problem, feature selection, and algorithm design under the guidance of data analysis. Each step is well-founded and a highly explanatory forecasting model is generated. This forecasting model is more than a single classification model. The classification accuracy has improved significantly, and it has certain guiding significance for us to understand the actual competition of NBA. In addition, we also made a bold exploration of the feature cap of the data set, and concluded a relatively simple method to get a rough estimate of a set of feature caps.

In the future, we should also consider the correlation between the features and abandon the linear assumption to estimate the competitiveness of each team in the playoffs through non-linear processing. Also, there are many factors on the basketball court that have not been taken into account, such as the player's age, schedule (whether there is a back-to-back game), etc. Moreover, We will add more characteristics for analysis in future work. Acutally, non-competitive factors in sports will have a great impact on the result of the game, such as the short-term injuries or rumors of player before the game, which cannot be directly obtained from the database. We may need to use natural language processing (NLP) and other technical methods to crawl all previous games news and extract the information we want from it, which may greatly improve our forecast accuracy.

VI. Acknowledgement

We would like to thank Professor Enmei Tu and the TA Xiao Han for their contributions to our project.

Shuo Yang and Zihao Li contribute equally to this project. We write codes and write report together.

References

[1] S M. Arabzad, ME Tayebi A., S Sadi-N., and N. Ghofrani. Football match results prediction using artificial neural networks; the case of iran pro league. Journal of Applied Research on Industrial Engineering, 1(3):159–179, 2014.

[2] B. E Boser, I. M Guyon, and V. N Vapnik. A training algorithm for optimal margin classifiers. In Proceedings of the fifth annual workshop on Computational learning theory, pages 144–152. ACM, 1992.

[3] L. Breiman. Random forests, machine learning 45. Journal of Clinical Microbiology, 2:199–228, 2001.

[4] D Buursma. Predicting sports events from past results towards effective betting on football matches. In Conference Paper, presented at 14th Twente Student Conference on IT, Twente, Holland, volume 21, 2011.

[5] C. Cao. Sports data mining technology used in basketball outcome prediction. 2012.

[6] C.-C. Chang and C.-J. Lin. Libsvm: A library for support vector machines. ACM transactions on intelligent systems and technology (TIST), 2(3):27, 2011.

[7] T. Cover and P. Hart. Nearest neighbor pattern classification. IEEE transactions on information theory, 13(1):21–27, 1967.

[8] S. Haykin. Neural networks: a comprehensive foundation. Prentice Hall PTR, 1994.

[9] C. HERBINET. Predicting football results using machine learning techniques. 2018.

[10] A. Joseph, N. E Fenton, and M. Neil. Predicting football results using bayesian nets and other machine learning techniques. Knowledge-Based Systems, 19(7):544–553, 2006.

[11] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. nature, 521(7553):436–444, 2015.

[12] D. Miljković, L. Gajić, A. Kovačević, and Z. Konjović. The use of data mining for basketball matches outcomes prediction. In IEEE 8th International Symposium on Intelligent Systems and Informatics, pages 309–312. IEEE, 2010.

[13] F. Sean and K. Justin. Basketball statistics and history. https://www.basketball-reference.com/. Accessed April, 2000.

[14] B. Ulmer, M. Fernandez, and M. Peterson. Predicting soccer match results in the english premier league, 2013.