NBA prediction

Jun-Jie Yang(80%)
109550046
NYCU CS
yang9187.cs09@nycu.edu.tw

Wei-Jie Huang(20%)
109550056
NYCU CS
huangweij.cs09@nycu.edu.tw

1. Introduction

Basketball is a well-known sport, and NBA, known as the most prestigious basketball league, attracts the top-notch basketball players around the globe.In the final project, we want to use the skills and techniques learned from the course to predict the winner of NBA games.

We want to use different machine learning algorithm and datasets. With the comparison of these results, finding the one that has the best accuracy of prediction. If we can predict more accurately, we can actually apply the knowledge learned in this course and get a sense of accomplishment.

2. Related Work

While implementing our goal, we decompose the final project into four steps. We scraped our data from available information at nba.api [1] which contained detailed team data in each regular season from season 2016 to 2021 and saved to csv files. Our next step was to read in all this data, and clean the duplicate data because most advanced data are extensions of tradition data into formulas, and then we saved these new data to the new csv files. Next we import dataset into the different classification methods. In addition, we create two different features: Elo Ratings and Recent Team Performance, and apply these features into different classification methods. Finally, we get the accuracy of each classification algorithm.

Inspired by these two references [2,3], we understand the general process of designing a module and make some improvements. The first reference is insufficient for number of data types(column),and the second reference only discusses traditional data, so we decided to include more advanced data.

3. Methodology

3.1. Baseline

I chose to import Scikit-learn, given the ease of implementation for a variety of classification algorithms. We choose the following four algorithms.

1. Logistic Regression

   Logistic regression belongs to the family of supervised machine learning models. Unlike Linear Regression model which predicts outcomes on a range of values between 0 and 1, Logistic regression models the probabilities for classification problems with two possible outcomes. Since we are predicting wins and losses, this type of classification fits perfectly.

2. Support Vector Classifier

   Support Vector Classifier works by mapping data points to a high-dimensional space and then finding the optimal hyperplane that divides the data into two classes.

3. Random Forest Classifier

   Random Forest Classifier combine multiple CART trees and add randomly allocated training data to greatly improve the final calculation result. Each individual tree in the random forest produce a class prediction and the class with the most votes becomes our model's prediction.

4. Gradient Boosting Classifier

   In Gradient Boosting, each predictor tries to improve on its predecessor by reducing the errors, and it actually fits a new predictor to the errors made by the previous predictor. Gradient boosting models are suitable for us because of their effectiveness at classifying complex datasets.

3.2. Main Approach

1. Scrapping the Data Scraping tradition data and advance data from nba.api from season 2016 to 2021.

2. Processing the Data

   Combine tradition data with advanced data, and remove some duplicated data columns.The only difference between PACE and PACE_PER40 is that PACE_PER40 uses 40 minutes as the denominator.

The functionalities of them are too close, so we delete PACE_PER40. In addition, since we have FG percentage, FG made and FG attempt can be deleted. Not only that, TS% is advanced data extended from the formula of weighted value of the FG percentage, 3PT percentage and FT percentage, so they can be deleted,too. Finally, In some games, the data of rebound percentage is not recorded, which will result in NaN in this column, so the rebound percentage is removed.

3. Creating second dataset

   The first dataset focuses on the impact of single-game data on the results of the game. And the second dataset we want to make predictions from the team's performance in the last ten games and introduce Elo rating.

   (a)Elo Rating

   Elo rating system can be used to calculate the quality and relative skill of teams in a league. Elo rating are either given or subtracted points based on the final score of each game and where it was played with weights being given to point difference, upsets, and location.

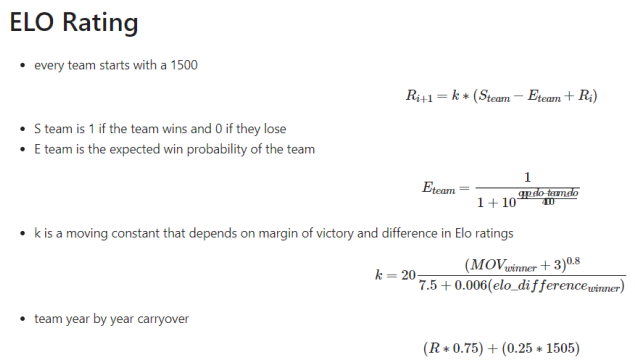


Figure 1. We set the parameter k of elo rating to 20 and home team advantage to 100, according to the reference. [4]

   (b)Recent Team Performance

   How a team has been performing recently is likely a good indicator of how they will perform in an upcoming game. Thus, we will keep track of the averages of each teams stats over their previous 10 games.

4. Classifying

   Use datasets made in two different ways with different classifiers to measure accuracy.

4. Experiments

1. performance

   We found that the accuracy of the first method is better than the second method , and among these classification methods, SVC and Random Forest Classifier perform the best, about 72% accuracy. We think that the reason why the second dataset performs not as good as we expected is that:

   (a)The accuracy of Elo rating: We calculated that when two teams played against each other, the team with the higher Elo rating had only a 62% chance of actually winning the game, which is similar to our model.

   (b)Insufficient sample size for last ten games: The performance of the past ten games is affected by player rotation and injury, which may not fully reflect the strength of the team, thus increasing the error of prediction.

   (c)Lack of sample: The amount of data in six seasons is too small.

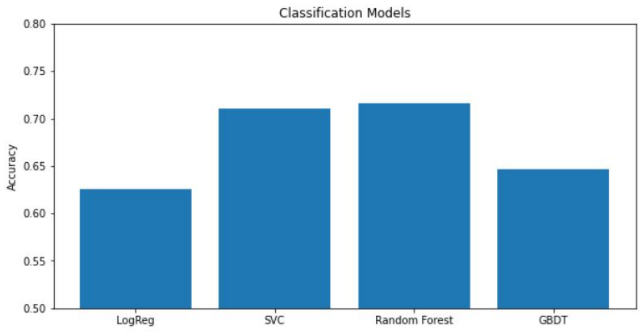


Figure 2. dataset 1

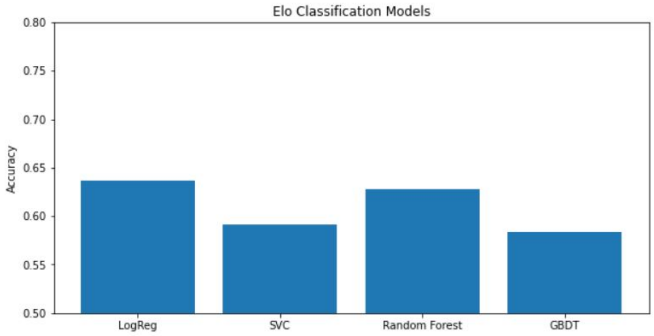
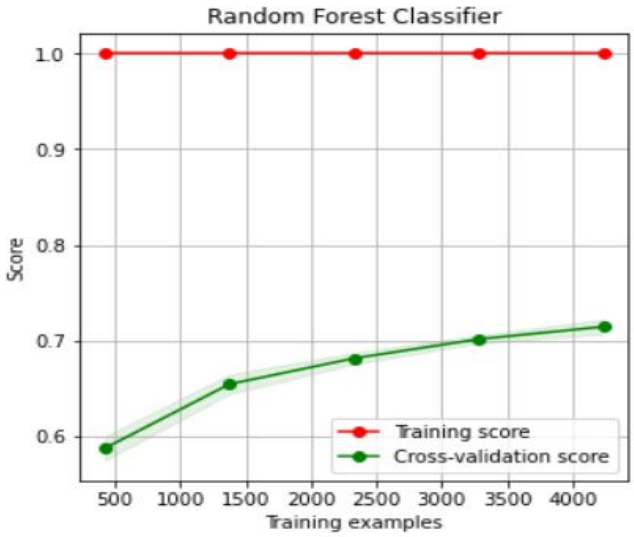


Figure 3. dataset 2

2. training and validation curve

   (a)the training score of the random forest classifier is much greater than the validation score, and the validation score keeps rising. Adding more training samples will most likely increase generalization.

(b)Both the validation score and the training score converge to a value that is quite low with increasing size of the training set. This means that the model may be too simple in structure, resulting in underfitting.



References

[1] https://github.com/swar/nba_api. 1

[2] https://github.com/jduannn/MachineLearningModel/blob/main/Final_Duan_James.ipynb. 1

[3] https://github.com/mhoude1/NBA_Model/blob/master/full_code.ipynb. 1

[4] https://fivethirtyeight.com/features/how-we-calculate-nba-elo-ratings/#:~:text=Here's%20the%20formula%3A%20Take%20the,and%20then%20divide%20by%2028. 2

5. Github report:

https://github.com/yangalt/nba-prediction