Machine Learning project:
Predicting the outcome of an NBA game

Denis Malasi

Course of 2022-2023

1 Introduction

The National Basketball Association (NBA) has been at the forefront of sports entertainment for decades, captivating fans worldwide with its high-flying action and intense rivalries. With the rapid advancement of machine learning techniques and the availability of vast amounts of game data, predictive analytics has become an increasingly important tool for understanding the intricacies of the sport.
The approach taken in this paper for predicting the outcome of NBA matches involves the use of machine learning techniques, including Support Vector Machines (SVMs) and Random Forest (binary) classification models. The study is based on a comprehensive dataset [1] that contains information from several recent NBA seasons (from 2004 to 2018), which enables a thorough examination of the factors that impact the outcome of a game. The performance of these models is evaluated using both accuracy and AUC metrics. The aim of the research to determine which of these models provides the most accurate predictions and to demonstrate the potential of machine learning techniques in sports analytics.

2 Feature engineering

With an abundance of data at hand, the aim was to convert raw information into valuable insights. Merely compiling game-by-game rebound statistics for a team is insufficient unless it can be integrated into a more comprehensive analysis that ultimately enables us to predict the outcome of games - wins and losses. To achieve this, two distinct features were created to trace the team's performance throughout the current season and the previous ones: the NBA Elo rating [2][3] and the team performance throughout the season.
The NBA Elo rating is a unique tool for analyzing the performance of basketball teams over time. By using a mathematical model that considers game results and other factors, it provides an unbiased and objective way to rank teams and predict the outcome of future games. By analyzing the progress of teams season-by-season, the Elo rating offers valuable insights into the strengths and weaknesses of each team. It's calculated on a match-by-match basis and is a zero-sum calculation of the relative skill of a team (for example: if Team A gains 40 points over Team B, Team B will lose 40 Elo points). The formula for updating the team's rating is as follows

$$R'_A = R_A + K \cdot (S_A - E_A)$$

where: R'_A is the team's new rating after the game; R_A is the team's rating before the game; K is a constant weighting applied to each game; S_A is the result of the match, it equals 1 for a win or 0 for a loss; E_A represents the expected win probability of the team.
The K constant is defined as

$$K = 20 \cdot \frac{(MOV + 3)^{0.8}}{7.5 + 0.006 \cdot elo_{diff}}$$

MOV (Margin of victory) is the difference of points between the two teams and elo_{diff} is the difference of Elo rating of the two teams.

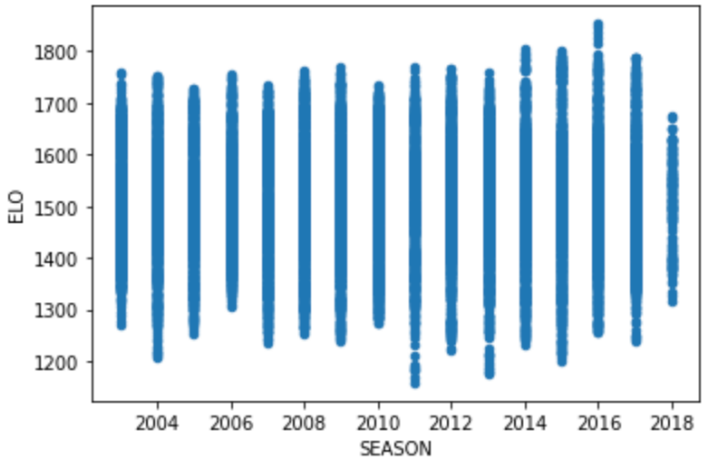
The expected win probability of the team is

$$E_A = \frac{1}{1 + 10^{\frac{R_B - R_A}{400}}}$$

R_B is the opponent's Elo rating and R_A is the current team's rating.
As said before Elo analyzes the progress season by season, so instead of resetting each team's rating when a new season begins, Elo carries over a portion (75%) of a team's rating from one season to the next.

$$R_{New\_Season} = \left(0.75 \cdot R_{End\_Season}\right) + (0.25 \cdot 1505)$$

1505 is the long-term mean Elo rating.



Elo distribution of the 30 NBA teams during the seasons

The second feature is a group of indexes showing the seasonal performance of each team.
Those indexes are:

- win streak
- seasonal wins
- seasonal average points
- seasonal average rebounds[1]
- seasonal average field goals[2] percentage
- seasonal average free throws[3] percentage
- seasonal average 3-point field goals Percentage
- seasonal average assists[4]

3 Metrics and models

Two different models were chosen to predict the outcomes of matches: a Random Forest and an SVM.
The Random Forest is like Tree bagging but when learning each tree, it removes some randomly chosen independent variables (n_{vars}) from the observations.
The model used in this project consisted of 500 trees with n_{vars} equal to the ceiling of the square root of the number of features ($\lceil\sqrt{P}\rceil$).
The second approach is an RBF (gaussian) kernel SVM. Hyperparameter tuning was conducted in order to find the optimal values for the hyperparameters C and gamma. The tuning process involved

[1] A statistic awarded to a player who retrieves the ball after a missed field goal or free throw.
[2] All shots taken during live game action.
[3] An unimpeded attempt at a goal awarded to a player following a foul or other infringement.
[4] A pass to a teammate that directly leads to a score by field goal

evaluating the performance of the SVM model on a validation set for a range of C and gamma values and selecting the values that resulted in the best performance. The optimal values were determined to be C = 0.6 and gamma = scale[5].
Prior to implementing the SVM model, a standardization process was performed on the dataset to ensure that all variables were rescaled to a common scale for optimal model performance.
Two evaluation metrics were utilized to assess the performance of the model: accuracy and Area Under the Curve (AUC). Accuracy is a commonly used metric in classification problems. It measures the number of correct predictions made by the model as a proportion of the total number of predictions. AUC represents the ability of a classifier to distinguish between positive and negative classes. The set of threshold values were calculated using midpoints of the sorted P_{Pos}^{i} with:

$$P_{Pos}^{i} = f_{Predict}'''(x^{i}, m) \quad f_{Predict}''': X \text{ x } M \rightarrow [0,1]$$

Where $f_{Predict}'''$ is the predicted output (as probability) of the single input feature.
Both models were then evaluated using 10-fold cross validation.

4 Results

The results in the table 1 show that Random Forest and SVM are better than the dummy classifier that predicts the most frequent class.
The accuracy of both models is higher than choosing the winner by just comparing the Elo.
SVM appears to be appears to be slightly better than the Random Forest with an accuracy of **66.55**%.

Model	**AUC**		**Accuracy**	
	Mean	Standard Dev	Mean	Standard Dev
Dummy Classifier	0.5	0	0.5892	0.01136
Comparing Elo Rating	/		0.6370	0
Random Forest	0.6917	0.009998	0.6545	0.008691
SVM	0.7050	0.01029	0.6655	0.009023

Table 1: Results of different models

5 Conclusions

In conclusion, this study aimed to predict the outcome of NBA matches by applying machine learning techniques such as Support Vector Machines (SVMs) and Random Forest (binary) classification models. The results showed that both models achieved high accuracy and AUC scores, but the SVM outperformed the Random Forest. The findings from this study demonstrate the potential of machine learning in sports analytics and provide valuable insights into the factors that impact the outcome of NBA games. The study also highlighted the importance of feature engineering in improving model performance and demonstrated the potential of machine learning techniques in sports analytics.
However, it is important to note that further research is needed to explore the limitations of the models and to incorporate additional features and data sources that may further enhance their predictive power.

References

[1] NBA games data *https://www.kaggle.com/datasets/nathanlauga/nba-games*
[2] Elo rating *https://it.wikipedia.org/wiki/Elo*
[3] How We Calculate NBA Elo Ratings *https://fivethirtyeight.com/features/how-we-calculate-nba-elo-ratings/*

[5] Default value used in python library "Sklearn": $\frac{1}{n_{Features} \cdot X.var()}$ $X.var()$ is the variance of the input features