Multiple Machine Learning Algorithms-based NBA Team Playoffs Prediction

Manho Yeung

Statistics and Data Science, University of California Santa Barbara, Santa Barbara, USA

Abstract. With the rapid development of data analytics in sports, it is vital to use machine learning methods to make decisions and predictions. This study focuses on predicting NBA playoff qualifications using machine learning techniques. By utilizing team-level statistics from 1947 to 2024, the paper implemented models such as Logistic Regression, K-Nearest Neighbors, Random Forest, and Elastic Net Regression. The data was preprocessed by scaling, centering, and handling missing values, followed by rigorous 5-fold cross-validation to ensure robust evaluation. Among the models, Random Forest outperformed the others, achieving the highest ROC-AUC score of 0.841. Its ensemble approach allowed for the effective capture of complex feature interactions, making it the most accurate model for predicting whether a team would qualify for the playoffs based on team performance. The research demonstrates the power of machine learning in improving prediction accuracy, providing insights for future sports analytics, and offering a foundation for integrating more complex data like player metrics or strategic factors. This work contributes to advancing predictive modeling in sports.

1 Introduction

In the past few months, the exciting NBA (National Basketbal Association) playoffs have already finished, which has attracted plenty of attention from basketball fans all around the world. Among the whole NBA season, playoffs are one of the most exciting and arousing phases. The importance of entering the playoffs is to win the title, which is often considered the greatest honor for the games of the NBA. The reason why machine learning method is used to predict the NBA playoffs qualification is that it can transform complex, large datasets into manageable models, which can be used to predict the outcome of NBA playoffs. Also with the development of machine learning and data mining, it becomes gradually appropriate to apply ML methods and its implications to analyze the sport data [1]. Furthermore, by following the trend of machine learning applications, more and more players or companies attempt to enhance their performance or profits on or off-court [2].

There are plenty of researches that apply machine learning to predict results in the field of sports. However, the accuracy of some previous research does not achieve an excellent level. Horvat uses the algorithms of logistic regression, naive bayes, decision tree, multi-layer perceptron neural networks, K-nearest neighbours and logit boost to predict basketball

Corresponding author: myeung@ucsb.edu

game outcomes. Among these algorithms, KNN has the best performance, which has the highest accuracy of 59%, while the decision tree has the worst performance, which has the lowest accuracy of 53.5%. Madhavan applies Hidden Markov models (HMMs) to predict the match results, then the model is able to reach an accuracy of 73% [3,4].

The purpose of this research is to develop a predictive model that can forecast whether an NBA team will enter the playoffs based on a dataset called “Team Stats Per Game.cvs”, and implement multiple machine learning techniques to yield the most accurate model for this binary classification problem.

These variables will be used to predict the binary response variable, indicating playoff qualification (TRUE/FALSE). The paper will partition our data into training and test sets and create a recipe encapsulating all preprocessing steps. The dataset will be prepared for 5-fold cross-validation to ensure the robustness of our model evaluations [5,6].

Various classification models were employed, including Logistic Regression, KNN, Random Forest, Elastic net regression, LDA (Linear Discriminant Analysis) and Boosting Trees to model our training data [3]. The breadth of this model selection is to capture both linear and non-linear relationships and offers a balance between bias and variance. After deriving and comparing the outcomes from all models, it is vital to pick the one that exhibits the best performance. This chosen model will then be applied to the test dataset to see how well the model performs.

2 Methods

2.1 Data Collection and Preparation

The dataset utilized in this study was sourced from the Kaggle repository titled "NBA Stats (1947-present)." Specifically, the subset "Team Stats Per Game.csv" was employed, which includes performance metrics for NBA teams from the 1947 to 2024 seasons. The dataset initially contained 28 variables across 1,845 records, representing various statistics such as field goals per game, three-point percentages, assists, rebounds, and points per game.

Data cleaning involved removing variables with significant missing values and those that demonstrated high multicollinearity [7,8]. To detect multicollinearity in the dataset, a correlation matrix was initially employed to identify pairs of variables with high correlation coefficients. Variables showing strong correlations were flagged for potential removal. Furthermore, Variables such as "abbreviation," "season," "games played (g)," and "minutes played per game (mp_per_game)" were excluded due to irrelevance in predictive modeling. Variables with strong correlations, such as "two-point field goal attempts (x2pa_per_game)" and "free throw attempts (fta_per_game)," were also removed to prevent redundancy and overfitting. The final dataset included 11 variables deemed most relevant for predicting playoff qualification.

The dataset exhibited missing values, particularly in older records where certain statistics, such as blocks per game and steals per game, were not consistently recorded. To address this, KNN (K-Nearest Neighbors) imputation was applied, using $k = 5$, which considers the relationships between existing data points to fill in the missing values accurately. This method was chosen to preserve the dataset's integrity while minimizing the introduction of bias.

2.2 Feature Engineering and Data Transformation

From the original 28 variables, the paper selected 11 key features that may affect the predicted results. These features included "free throws per game (ft_per_game)," "total

rebounds per game (trb_per_game)," "assists per game (ast_per_game)," "points per game (pts_per_game)", and etc. The target variable, "playoffs," was a binary indicator where "TRUE" represented a team making the playoffs, and "FALSE" indicated otherwise.

To prepare the data for modeling, categorical variables such as "team" were transformed into dummy variables using one-hot encoding. Continuous variables were centered and scaled to standardize the dataset, ensuring that all predictors had equal weight in the models. Additionally, predictors with near-zero variance were identified and removed, streamlining the dataset and improving model efficiency.

2.3 Model Preparation

The paper splits the dataset into training and testing sets to evaluate the predictive models' performance. A 75/25 split was implemented, with stratification based on the response variable "playoffs" to preserve the distribution of playoff and non-playoff teams in both sets. The resulting training set comprised 1,383 records, while the test set contained 462 records—each split designed to ensure robustness in our model evaluations [9,10].

A single recipe was created to handle preprocessing steps. This recipe included dummy encoding for categorical variables, centering and scaling for all predictors, and KNN imputation for missing values. The recipe was prepped and baked. Specifically, the recipe included dummy encoding for categorical variables (like team names), centering and scaling numerical predictors for standardization, and using KNN imputation to handle missing values. Additionally, near-zero variance predictors were removed to avoid irrelevant variables. This consistent approach across all models ensured that they were trained on a uniform dataset, reducing bias and making the comparisons between model performances more reliable [11].

2.4 Model Building

In the pursuit of constructing robust and reliable models for predicting NBA team playoff qualifications, this study rigorously employed six distinct machine learning algorithms: Logistic Regression, KNN (K-Nearest Neighbors), Random Forest, Elastic Net Regression, LDA (Linear Discriminant Analysis), and Boosting Trees. Each model was meticulously developed and trained to discern patterns within the comprehensive NBA team performance dataset, which had been carefully preprocessed to enhance model accuracy and stability. The models were assessed using cross-validation and fine-tuning techniques to ensure their effectiveness in handling the complexities of this binary classification problem [12].

In this project, six machine learning models were employed to predict NBA playoff qualifications, each carefully tuned to maximize performance.

Logistic Regression served as our baseline model due to its simplicity and interpretability. The key hyperparameter, regularization strength (C), was tuned to balance bias and variance. Exploring various C values can prevent overfitting while ensuring robust performance, ultimately establishing a reliable benchmark for comparison.

The KNN algorithm, known for its non-parametric approach, relies heavily on the number of neighbors (k). The paper conducted a grid search to find the optimal k, balancing sensitivity to noise and generalization. This tuning ensured that KNN accurately classified teams based on their proximity in the feature space.

Random Forest, an ensemble method, was employed to capture complex interactions between features. Key hyperparameters included the number of trees, maximum tree depth, and minimum samples to split a node. Through extensive tuning, the paper optimized these parameters to maximize the model's ROC AUC (Receiver-operating Characteristic Curve Area under the Curve) score, ensuring a strong balance between model complexity and performance.

Elastic Net Regression combines Lasso and Ridge regression to handle correlated features. The two critical hyperparameters—mixing parameter (l1_ratio) and regularization strength (alpha)—were tuned to achieve the best balance between feature selection and regularization. This approach allowed the model to maintain flexibility while preventing overfitting.

LDA was included for its efficiency in handling linearly separable data. Given its straightforward nature, LDA did not require extensive tuning. Instead, the model was deployed to provide insights into linear relationships within the dataset, serving as a robust benchmark.

Finally, Boosting Trees were employed to incrementally improve prediction accuracy by focusing on errors from previous models. The main hyperparameters—learning rate (learning_rate), number of boosting rounds (n_estimators), and tree depth (max_depth)—were carefully tuned. A lower learning rate improved generalization, while the optimal number of boosting rounds and tree depth ensured the model's ability to capture complex patterns without overfitting.

Each model followed a consistent workflow, from setup and hyperparameter tuning to training and evaluation. Performance was assessed using metrics like accuracy, precision, recall, and ROC AUC, with results stored for further analysis. This thorough approach allowed us to compare the models effectively and identify the best-performing techniques for predicting NBA playoff outcomes.

3 Results and Discussions

Table 1 compares the ROC-AUC of different models: logistic regression, K-Nearest neighbors, linear discriminant analysis, elastic net regression, random forest and boosting trees. The higher ROC-AUC score a model presents, the more accurate its prediction is. The table also highlights the key hyperparameter settings, such as regularization strength in Logistic Regression, the number of neighbors in KNN, and the number of trees and maximum depth in the Random Forest model.

Table 1 Results for each model

Model	Best ROC-AUC	Key Hyperparameters
Logistic Regression	0.7384	Regularization (C) - No significant tuning required
K-Nearest Neighbors	0.7332	Number of Neighbors (k) - Tuned within range 1 to 10
Linear Discriminant Analysis	0.7385	Default parameters
Elastic Net Regression	0.7386	L1 Ratio: 0.3333 - 0.7778, Regularization Strength: 0.001 - 0.1, Mixture - Range of (0, 1)
Random Forest	0.8213	Number of Trees: 400-500, Max Depth: 8-12, Min Samples Split: 8-12, Number of Predictors: 4
Boosting Trees	0.7905	Learning Rate, Number of Trees (200-600), Max Depth - Specific tuning required to optimize model performance

The Logistic Regression model achieved a ROC-AUC score of 0.7384, indicating a moderate ability to predict playoff outcomes. This model did not require significant hyperparameter tuning, relying on the default regularization parameter CCC. While the model was straightforward to implement and interpret, its linear nature limited its ability to capture complex relationships within the data. The ROC-AUC score suggests that logistic regression may not be sufficient for more nuanced predictions where interactions between variables are critical.

KNN achieved a ROC-AUC of 0.7332, slightly lower than Logistic Regression. This model's performance was tuned by adjusting the number of neighbors (k) within the range of 1 to 10. Despite the fine-tuning, KNN struggled with the high dimensionality of the dataset, which likely affected its performance. The relatively lower ROC-AUC score reflects KNN's challenge in effectively distinguishing between playoff and non-playoff teams, especially in more complex data environments. The model's simplicity in considering only the nearest neighbors limited its predictive power.

LDA, with a ROC-AUC of 0.7385, performed on par with Logistic Regression. The model operated under its default parameters, making it an easy-to-implement option for linear classification tasks. However, like Logistic Regression, LDA's linear approach may have limited its ability to capture the intricacies of the data, leading to similar performance metrics. The model's strength lies in its simplicity and interpretability, but its application is better suited for datasets where linear separability is assumed.

Elastic Net Regression showed a slight improvement over the simpler models with ROC-AUC of 0.7386. By combining the properties of Lasso and Ridge regression, this model was able to handle multicollinearity better, which may have contributed to its marginally higher performance. The model was fine-tuned across a range of L1 ratios (0.3333 to 0.7778) and regularization strengths (0.001 to 0.1), reflecting its flexibility in balancing model complexity and bias. Despite these adjustments, the ROC-AUC suggests only a marginal improvement over Logistic Regression, indicating that while Elastic Net offers more robust regularization, it may not significantly outperform simpler models in this context.

The Random Forest model has a quite high ROC-AUC of 0.8213, which demonstrates its effectiveness in capturing the complex interactions within the dataset. The model's performance was significantly enhanced by tuning several hyperparameters, including the number of trees (400-500), maximum depth (8-12), and the number of predictors considered at each split. Random Forest's ability to aggregate the predictions of multiple decision trees allowed it to reduce overfitting while maintaining high accuracy, making it the best-performing model in this study.

Boosting Trees also performed well, achieving a ROC-AUC of 0.7905. This model required specific tuning of its learning rate, the number of trees (200-600), and maximum depth to optimize its performance. Boosting Trees iteratively improved upon previous predictions, making it robust in handling complex patterns in the data. Although its ROC-AUC score is slightly lower than that of the Random Forest, Boosting Trees excels in iterative error correction, making it highly effective in scenarios that involve complex patterns and intricate data structures. While Random Forest is better at handling general prediction tasks, Boosting Trees stands out in cases where incremental learning and refinement are crucial, offering a competitive advantage in such applications despite its marginally lower overall performance.

4 Conclusion

After plenty of trying research, data mining and various models including Logistic Regression, K-Nearest Neighbors, Logistic Regression, Boosting Trees, Elastic Net Regression and Random Forest, the conclusion can be drawn that the Random Forest model

is the most effective. The Random Forest model performed very well with a ROC AUC score of 0.8404, indicating a high capability to determine whether a team will enter the playoffs or not. The strong ROC AUC score achieved by the Random Forest model highlights its effectiveness and dependability in managing the intricacies and subtleties present in the dataset.

This model's success can largely be attributed to its capacity to accommodate a vast number of predictors while effectively capturing correlations among the variables. The performance of the Random Forest model underscores the significance of ensemble methods in predictive modelling, particularly when dealing with a combination of numeric and categorical data. Although the results are encouraging, future work could involve integrating additional variables, such as player performance metrics, injury reports, and coaching strategies, to boost the model's predictive accuracy. Moreover, exploring advanced techniques like neural networks and hybrid models that combine multiple algorithms might offer further improvements in accuracy. Continuous refinement and validation with updated datasets will be crucial in maintaining the model's relevance and accuracy in predicting NBA playoff outcomes.

Reference

1. K. Apostolou, C. Tjortjis, Sports Analytics Algorithms for Performance Prediction. In Proceedings of the 10th International Conference on Information, Intelligence, Systems and Applications (IISA), Patras, Greece, 2019. SCITEPRESS.
2. W. Tichy, 2016. Changing the Game: 'Dr. Dave' Schrader. SCITEPRESS.
3. M. Beckler, H. Wang, M. Papamichael, 2013. NBA Oracle. Zuletzt besucht am, 17(2008-2009.9). SCITEPRESS.
4. W. Cai, et al., 2019. A Hybrid Ensemble Learning Framework for Basketball Outcomes Prediction. Physica A: Statistical Mechanics and its Applications, 528. SCITEPRESS.
5. W. Chen, Z. Shen, Y. Tao, 2013. Big Data Series: Data Visualization. Beijing: Electronic Industry Press. SCITEPRESS.
6. T. Horvat, L. Havaš, D. Srpak, 2020. The Impact of Selecting a Validation Method in Machine Learning on Predicting Basketball Game Outcomes. Symmetry. SCITEPRESS.
7. S. Ji, J. Li, T. Du, 2019. A Review of Machine Learning Model Interpretability Methods, Applications and Security Research. Computer Research and Development. SCITEPRESS.
8. R. Khanmohammadi, et al., 2022. MambaNet: A Hybrid Neural Network for Predicting the NBA Playoffs. arXiv preprint arXiv:2210.17060. SCITEPRESS.
9. V. Madhavan, 2016. Predicting NBA Game Outcomes with Hidden Markov Models. Berkeley University. SCITEPRESS.
10. D. Miljković, et al., 2010. The Use of Data Mining for Basketball Matches Outcomes Prediction. In Proceedings of the IEEE 8th International Symposium on Intelligent Systems and Informatics. IEEE. SCITEPRESS.
11. B. Taylor, 2017. https://fansided.com/2017/08/11/nylon-calculus-measuring-creation-box-score/. SCITEPRESS.
12. F. Thabtah, L. Zhang, N. Abdelhamid, 2019. NBA Game Result Prediction Using Feature Analysis and Machine Learning. Annals of Data Science. SCITEPRESS.