Machine Learning Models for Nba Game Prediction

Weichen Peng

Computer Science and Technology, Changsha University of Science & Technology, Changsha, China

Abstract. In the realm of sports analytics, the application of machine learning in predicting match outcomes has attracted considerable academic and practical interest. This study specifically examines NBA match data, encompassing key game indicators, team statistics, and individual player performance metrics. This work intends to investigate how many machine learning approaches—such as Random Forest, Decision Trees, K-Nearest Neighbors (KNN), and Linear Regression—contribute to determine game results by means of their respective strengths and algorithmic characteristics. Several performance evaluations will help to assess the prediction power of these models, and numerous measurements are used to allow a comprehensive comparison of their dependability, robustness, and computational efficiency. The results show that since ensemble learning approaches can capture complicated interactions among variables, they show higher prediction accuracy and generalizability. Linear models, meantime, do well with organized statistical datasets and require less training time. For sports bettors, team strategists, and sports analysts, these revelations are priceless. By using deep learning models, sophisticated statistical approaches, and more extensive outside data sources—such as player biometrics, psychological profiles, and real-time game conditions—future research might further hone predicted accuracy and practical applicability.

1 Introduction

Sports analytics has been somewhat well-known in recent years because to developments in machine learning methods and the availability of massive data. Predicting game results in professional sports—especially the National Basketball Association (NBA)—is a difficult chore given their complicated and often changing character. Still, accurate projections may be rather important for improving strategic decision-making, game strategy optimization, and audience involvement enhancement. Teams, coaches, sports broadcasters, and the betting business among other stakeholders gain from these realizations.

Using individual data, team performance measures, and game-specific variables, several research have investigated many machine learning techniques to forecast sports results. For example, used artificial neural networks and decision trees to forecast NBA game results,

weichenpeng@stu.csust.edu.cn

so proving that defensive rebounds, three-point percentage, and free throws produced much impact match results [1]. Another research using many machine learning approaches examined past game data in order to find important performance variables influencing game outcomes [2]. Furthermore, stressing the efficiency of modern algorithms in collecting complex patterns in basketball data [3], comparing logistic regression, support vector machines, deep neural networks, and random forests. Recent research utilizing combined XGBoost and SHAP models shows that field goal %, defensive rebounds, and turnovers are regularly correlated to game outcomes [4].

By means of player data, team records, and game indications, this paper fills in holes in NBA match prediction by pointing out important elements. To identify the best successful method it contrasts machine learning models including linear regression, K-Nearest Neighbors, decision trees, random forests, XGBoost, LightGBM, and Support Vector Regression. Key performance measures are examined using feature selection and statistical evaluation—including the R^2 coefficient—using NBA STATS data. This research advances predictive analytics, contributing to real-time modeling and broader applications in competitive sports.

2 Methodology

2.1 Data collection

The dataset used in this study was obtained through web scraping from the NBA STATS official website.A custom web scraper was built to gather player statistics, such as per-game performance, shooting efficiency, and defensive contributions. The collected data was then cleaned and organized into a CSV format, making it ready for further preprocessing and model training.

2.2 Dataset specifics

The dataset includes a variety of performance indicators for both individual players and teams, organized into different feature groups. 错误!未找到引用源。 shows the composition of the dataset. It encompasses multiple NBA player statistics, such as individual performance metrics like points scored (PTS), assists (AST), rebounds (REB), shooting accuracy (FG%), and player efficiency metrics (FP, +/-). Prior to being used for training predictive models, the dataset underwent preprocessing.

Table 1 Example table

Statistic Type	Number of Features	Numerical Features
Basic Information	4	3
Game Statistics	4	4
Shooting Performance	4	4
Three-Point & Free Throws	6	6
Defense & Fouls	8	8

2.3 Prediction models

Certain prediction models were used in order to assess the performance of certain machine learning techniques.

2.3.1 Linear Regression (LR)

Assuming a linear connection between features and the target variable, linear regression is a basic yet effective prediction model. Its simplicity and interpretability provide it a baseline from which to contrast more intricate models 5. Notwithstanding its limitations, in machine learning it is fundamental as it clarifies models and expands them to nonlinear methods [6]. Other techniques including regularization help to increase its predictive power.

2.3.2 K-Nearest Neighbors (KNN)

Calibrated using hyperparameters including neighbors, distance metric, and weighting technique, this distance-based model projects total points based on related player data. Recent research highlight the need of improving model accuracy by utilizing a suitable distance measure, such Cosine or Mahalanobis [7]. Moreover, graph neural networks have been studied to improve the robustness of kNN by incorporating the weighting mechanism inside a learned graph representation [8]. Particle Swarm Optimization and other optimization techniques have shown somewhat significant adjustment of kNN hyperparameters across different datasets [9].

2.3.3 Decision Tree (DT)

To forecast results, a tree-based model—adjusted for factors including max_depth and min_samples_split—partitions data depending on feature thresholds. Changing hyperparameters such max_depth and min_samples_split shows that Grid-SearchCV and cross-valuation greatly improve model generalization, thereby improving classification accuracy [10]. Moreover, ensemble techniques such Random Forest improve prediction stability by reducing overfitting with optimal parameter calibration [11].To project outcomes, a tree-based model—modified for variables including max_depth and min_samples_split—divides data according on feature thresholds By changing hyperparameters like max_depth and min_samples_split, research shows that Grid-SearchCV and cross-valuation considerably improve model generalization, thereby increasing classification accuracy. Furthermore, by reducing overfitting with optimal parameter adjustment, ensemble techniques like Random Forest increase prediction stability.

2.3.4 Random Forest (RF)

By aggregating various models to reduce variance and bias, an ensemble learning approach containing several decision trees can considerably improve prediction accuracy. Many studies—especially in sports analytics and challenging classification issues—have demonstrated the effectiveness of such methods. Yeung noted, for instance, that Random Forest—a widely used ensemble learning method—achieved the greatest accuracy in projecting NBA playoff qualifications by increasing the number of estimators (n_estimators) and tree depth [12]. Another work proposed a hybrid ensemble learning framework combining bagging and subspace approaches, demonstrating that carefully changing the tree depth may significantly enhance the prediction performance of the model [13]. These results underline the need of hyperparameter adjustment in enhancing the accuracy and dependability of ensemble decision tree models.

2.3.5 Gradient Boosting Methods (XGBoost & LightGBM)

Advanced boosting techniques are used by gradient boosting methods like XGBoost and LightGBM to repeatedly improve predictions by decreasing residual errors at each stage. Highly successful for predictive modeling, these models shine in managing structured data and catching complex feature relationships. Using histogram-based learning, LightGBM optimizes training speed; XGBoost, well-known for its speed and efficiency, reduces overfitting by means of regularization methods. These models together provide strong performance in prediction tasks by balancing interpretability with accuracy.

2.3.6 Support Vector Regression (SVR)

Assistance Applied kernel-based regression methods, Vector Regression (SVR) models intricate nonlinear connections in data. SVR finds best hyperplanes for prediction by mapping input properties onto higher-dimensional environments. However, when working with huge, high-dimensional datasets that have strong feature interactions, its performance is frequently less effective than tree-based models. Despite this limitation, SVR remains valuable for capturing subtle patterns in smaller datasets with well-defined structures. Each model underwent cross-validation and hyperparameter tuning to maximize predictive power and generalization performance.

3 Results and Discussion

3.1 Model performance evaluation

The author implemented various machine learning models to predict NBA game scores, including Linear Regression (LR), K-Nearest Neighbors (KNN), Decision Tree Regression (DTR), Random Forest Regression (RFR), XGBoost, LightGBM, and Support Vector Regression (SVR). Model performance was assessed using the R index, which gauges how well the independent variables account for the dependent variable.

Table **2** presents the R^2 indices for each model.

Table 2 R^2 indices for each model

Model	LR	KNN	DT	RF	XGBoost	LightGBM	SVR
R^2 Index	0.975	0.872	0.940	0.942	0.h910	0.912	0.816

3.2 Model comparison and analysis

3.2.1 Linear Regression

Linear Regression achieved the highest R^2 index (0.975), indicating a strong linear structure in the dataset, where features such as points, assists, and rebounds are highly correlated with the total score [14]. Linear regression can be categorized into simple linear regression, which models the relationship between a single independent variable and the response variable, and multiple linear regression, which considers multiple independent variables as inputs to describe a plane in three-dimensional space [2-4]. The Least Squares Method is typically applied to determine the regression coefficients for both simple and multiple linear regression models. The model offers several advantages. A suitably large training set

helps maintain stable training while lowering the danger of overfitting, and the dataset is well-structured with low collinearity across variables. Additionally, feature selection was applied to remove highly correlated variables, such as field goal percentage, improving overall model stability. Nevertheless, even while linear regression is good at capturing linear correlations, it could have trouble simulating intricate nonlinear interactions in the dataset.

3.2.2 Decision Tree and Random Forest

Logic-based decision tree algorithms use a sequence of hierarchical whether statement comparisons to classify a target variable Each node in the tree represents a decision point, and the leaves serve as class designations or predictions. In regression tasks, decision tree-based models like Decision Tree Regression (0.940) and Random Forest Regression (0.942) have demonstrated strong performance. The superior results of Random Forest can be attributed to several factors. Its ensemble learning approach helps reduce overfitting, while its ability to capture nonlinear feature relationships enhances predictive accuracy. Additionally, it can rank feature importance, allowing for more optimized predictions.

3.2.3 K-Nearest Neighbors (KNN)

The KNN regressor classifies unknown samples by identifying the K closest known samples and assigning the most frequent category among them[15]. By choosing K comparable data points, comparing characteristics across the training and test sets, and identifying the dominant category, KNN performs well. However, KNN achieved an R^2 index of 0.872, lower than tree-based models. This performance gap may be due to the choice of the optimal hyperparameter (K = 9) affecting model accuracy and the inefficiency of Euclidean distance calculations in high-dimensional data spaces.

3.2.4 XGBoost and LightGBM

XGBoost (0.9104) and LightGBM (0.9123) also performed well, capturing complex data relationships but slightly underperforming compared to Random Forest. The need for significant hyperparameter adjustment and susceptibility to outliers, which can impact model performance and stability, are two possible explanations.

3.2.5 Support Vector Regression (SVR)

Strong supervised learning models called Support Vector Machines (SVM) can handle nonlinear classification and regression problems as well as high-dimensional spaces [16]. However, in this study, SVR exhibited the lowest R^2 index (0.8162), which may be attributed to the need for further kernel function optimization, as well as its high computational cost and relatively low training efficiency.

3.3 Data visualization and model analysis

The author visualized the distribution of predicted versus actual values to analyze model errors. Fig 1 illustrates the performance of key models. It is evident from the figure that Linear Regression aligns well with actual values, indicating high model fit. Decision Tree and Random Forest show some fluctuations, while KNN and SVR exhibit larger prediction errors.

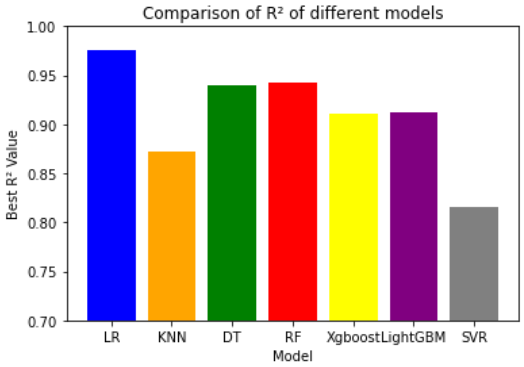

Fig 1 Comparison of R2 of different models

错误!未找到引用源。 A scatter plot of fitted values under Linear Regression compares predicted and actual values. The regression line shows the model's trend, while the ideal line (y = x) represents perfect predictions. The close alignment of anticipated values (orange dots) with the ideal line indicates high accuracy. The model's strong fit and low bias are evident, though slight deviations at larger values suggest potential errors due to data noise or model limitations.

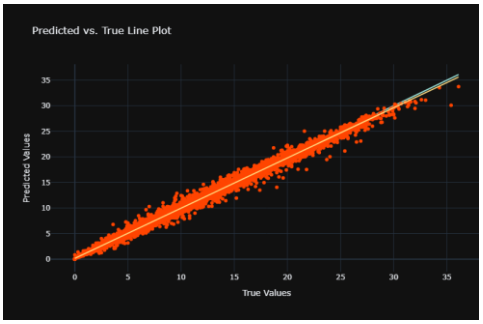

Fig 2 Predicted vs True Line Plot

3.4 Future improvement directions

3.4.1 Feature Engineering Optimization

Other sophisticated elements, such Win Shares (WS), Player Efficiency Rating (PER), True Shooting Percentage (TS%), and past player performance data, can be added to improve model accuracy.

3.4.2 Adoption of More Advanced Models

Future work could explore deep learning approaches, such as Recurrent Neural Networks (RNNs) or Transformer-based models, to capture the temporal characteristics of player data.

3.4.3 Dataset Expansion

Expanding the dataset with additional seasons, incorporating player injury reports, team strategy information, and social media data (e.g., fan votes, news trends) could improve prediction performance.

4 Conclusion

This study explored the application of machine learning models in predicting NBA game results based on various statistical features. With the highest R^2 values reaching 0.9425, the author discovered via rigorous experimentation with models including Linear Regression, K-Nearest Neighbors (KNN), Decision Trees, Random Forests, XGBoost, LightGBM, and Support Vector Regression (SVR) ensemble-based approaches achieved superior predictive performance. Linear Regression also demonstrated strong results due to the inherent linear relationships in the dataset.

The author's findings indicate that incorporating various statistical features, such as shooting performance, defensive metrics, and team statistics, can significantly improve prediction accuracy. This research provides data-driven approaches to evaluate team and player effectiveness, offering valuable insights for sports analysts, betting markets, coaching staff, and strategic decision-making.

To increase forecast accuracy, future research can concentrate on include further elements such player injuries, tiredness degrees, and game environment. Moreover, using deep learning methods—especially recurrent and transformer-based models—may enable one to capture intricate temporal patterns in player performance and game results. Further improving model resilience would be to extend the dataset to incorporate multi-season analysis and outside variables such fan participation and coaching techniques. In sports analytics, this study eventually provides a basis for more complex prediction models, hence bridging the gap between data science and competitive sports decision-making.

References

1. Thabtah, F., Zhang, L., Abdelhamid, N.: 'NBA game result prediction using feature analysis and machine learning', Ann. Data Sci., 2019, 6, (1), pp. 103–116.
2. Patrot, A., Harish, H., Shambbavi, B., et al.: 'NBA Game Prediction Using Machine Learning Algorithm', Proc. 2023 Int. Conf. Recent Trends Electronics Commun. (ICRTEC), IEEE, 2023, pp. 1–6.
3. Wang, J.: 'Predictive Analysis of NBA Game Outcomes through Machine Learning', Proc. 6th Int. Conf. Machine Learning Machine Intell., 2023, pp. 46–55.
4. Ouyang, Y., Li, X., Zhou, W., et al.: 'Integration of machine learning XGBoost and SHAP models for NBA game outcome prediction and quantitative analysis methodology', Plos One, 2024, 19, (7), e0307478.
5. Qu, K.: 'Research on linear regression algorithm', MATEC Web Conf., EDP Sciences, 2024, 395, 01046.
6. Radhakrishnan, A.: 'Lecture 2: Linear Regression', (2022-01-21), [EB/OL], accessed 27 Nov 2024.
7. Kalra, V., Kashyap, I., Kaur, H.: 'Effect of distance measures on K-nearest neighbour classifier', Proc. 2022 2nd Int. Conf. Comput. Sci. Eng. Appl. (ICCSEA), IEEE, 2022, pp. 1–7.
8. Kang, S.: 'K-nearest neighbor learning with graph neural networks', Mathematics, 2021, 9, (8), 830.
9. Rizki, M., Hermawan, A., Avianto, D.: 'Optimization of Hyperparameter K in K-Nearest Neighbor Using Particle Swarm Optimization', J. JUITA: Jurnal Informatika, 2024, 12, (1), pp. 71–79.

10. Shen Z, Lei J, You Y.:'Tennis Player Winning Prediction Based on Grid-Search Random Forest Algorithm', 2024 International Conference on Telecommunications and Power Electronics (TELEPE). IEEE, 2024: 560-563.
11. Xiong, A.: 'Analysis of NBA player salary based on multiple linear regression model', Theor. Nat. Sci., 2024, 51, pp. 206–213.
12. Yeung, M.: 'Multiple Machine Learning Algorithms-based NBA Team Playoffs Prediction', ITM Web Conf., EDP Sciences, 2025, 70, 04024.
13. Cai, W., Yu, D., Wu, Z., et al.: 'A hybrid ensemble learning framework for basketball outcomes prediction', Physica A: Stat. Mech. Appl., 2019, 528, 121461.
14. Goh, Y. L., Goh, Y. H., Bin, R. L. L., et al.: 'Predicting the performance of the players in NBA Players by divided regression analysis', Malays. J. Fundam. Appl. Sci., 2019, 15, (3), pp. 441–446.
15. Pan, S., Yang, B., Song, Q.: 'A Short-term Time Series Predictive Algorithm Based on Rolling Prediction and PSO-SVR', Proc. 2024 IEEE 2nd Int. Conf. Control, Electron. Comput. Technol. (ICCECT), Jilin, China, 2024, pp. 1430–1434.
16. Qi, X., Gao, Y., Li, Y., Li, M.: 'K-nearest Neighbors Regressor for Traffic Prediction of Rental Bikes', Proc. 2022 14th Int. Conf. Comput. Res. Dev. (ICCRD), Shenzhen, China, 2022, pp. 152 – 156, doi: 10.1109/ICCRD544 09.2022.9730527.