2025

The National Basket Ball Prediction

MACHINE LEARNING MODELING

SIPHOSENKOSI MAKHOBA (222043251)
LETHOKUHLE MZIMELA (222062153)
LETHOKUHLE MNTUNGWA (218035115)
FUNDI ZUMA (222079236)
SINETHEMBA MTHEMBU (222079478)
SIPHO MABUYAKHULU (219004586)

Contents

The National Basketball Association (NBA) Prediction

INTRODUCTION

The analysis of professional basketball performance has become increasingly data-driven, with teams, analysts, and researchers seeking insights from historical player and team statistics. Understanding exceptional player performance and predicting game outcomes requires robust methods capable of handling the complex, multidimensional nature of NBA data. Players' contributions vary across scoring, rebounding, assisting, and efficiency metrics, and are influenced by factors such as team strategies, era-specific playing styles, and minutes played. Identifying statistical outliers in such a dataset can highlight extraordinary talents and inform performance evaluation, scouting, and strategic decision-making.

This study focuses on detecting anomalous NBA players whose performance deviates significantly from league norms and building predictive models for game outcomes using historical data. By leveraging both statistical techniques and machine learning methods, the research aims to uncover patterns that are not immediately apparent from traditional analyses. The approach combines data preprocessing, feature engineering, anomaly detection, and model evaluation to provide a comprehensive framework for basketball analytics.

The study has two primary objectives. The first is to systematically identify NBA players with exceptional performance profiles using robust outlier detection methods. The second is to predict team game outcomes by leveraging engineered performance features within machine learning models. The results are expected to advance both the academic understanding of sports analytics and practical applications, including performance evaluation, player scouting, and strategic decision-making.

METHODS AND TECHNIQUES

1. Data Acquisition and Preprocessing

1.1. Data Sources and Collection

This study draws from two primary NBA historical datasets: team performance records and individual player career statistics. The team-level dataset contains season-by-season statistics, including offensive and defensive scoring totals, pace-adjusted metrics, and win–loss outcomes. The player-level dataset comprises detailed career metrics across multiple seasons, incorporating traditional statistics such as points, rebounds, and assists, as well as efficiency measures. Together, these datasets provide a comprehensive foundation for both outlier detection and predictive modeling.

1.2. Data Quality Assurance

A rigorous data validation pipeline was implemented to ensure the reliability of all analytical procedures. Missing values were detected and imputed using median-based replacement in order to preserve the distributional characteristics of the data. Extreme values were mitigated through interquartile range clipping between the first and ninety-ninth percentiles. Consistency in data types and column formatting was enforced systematically across all fields. Records with insufficient statistical integrity, such as seasons with fewer than 1000 total team points, were removed to maintain dataset reliability and to avoid introducing analytical noise.

1.3. Feature Engineering

To enhance predictive power, the study generated an extensive set of engineered features at both the team and player levels. Team-level features included win percentage, net rating, and pace-adjusted offensive and defensive efficiency. These features capture the fundamental strengths and weaknesses of each team in relation to scoring and possession control. Player-level features consisted of per-game averages such as points, rebounds, assists, steals, and blocks, in addition to shooting percentages and advanced metrics. Simplified Player Efficiency Rating and True Shooting Percentage were carefully formulated to represent player impact and scoring efficiency more comprehensively.

2. Outlier Detection Framework

2.1. Multi-Methodological Approach

Outlier detection was approached through a combination of univariate and multivariate methods. The Z-score method identified observations that deviated significantly from mean performance values, capturing players whose metrics exceeded three standard deviations. The interquartile range method detected extreme values falling outside the conventional 1.5 times the interquartile range, offering a non-parametric perspective on variability. To supplement these statistical methods, two machine learning techniques were employed. Isolation Forest detected anomalies by measuring the relative ease with which data points could be isolated through random partitioning, while DBSCAN identified outliers situated in low-density regions of the feature space based on neighborhood clustering.

2.2. Consensus Framework

A consensus system was developed to increase the robustness of outlier identification. Rather than relying on a single method, the study classified a player as an outlier only if flagged by at least two of the four detection approaches. This voting mechanism reduced false positives and provided greater confidence in the identification of extreme performers. The consensus strategy ensured that the outlier list captured genuine deviations rather than artifacts of a single methodology.

3. Game Outcome Prediction Modeling

3.1. Matchup Generation

To build a robust game prediction system, realistic team matchup simulations were designed. Teams were paired within the same season, and feature differences were calculated to capture relative advantages. Outcome probabilities were generated through sigmoid transformations based on strength differentials. For each season, twenty balanced matchups were produced, resulting in a dataset suitable for machine learning classification tasks.

3.2. Feature Representation

Feature representation centered on differential statistics, computed as the difference in performance metrics between two teams in a matchup. These differences were standardized using robust scaling based on medians and interquartile ranges to ensure that the modeling process was not influenced by skewed distributions. The dataset was divided into training, validation, and test subsets in a 60–20–20 split to support model tuning, evaluation, and generalization testing.

3.3. Machine Learning Pipeline

The machine learning pipeline incorporated five supervised models: logistic regression, random forest, gradient boosting, support vector machine, and XGBoost. Each model was configured with careful regularization to prevent overfitting. Logistic regression used L2 regularization, while tree-based models such as random forest and gradient boosting were constrained through limited tree depth, feature subsampling, and minimum node sizes. The support vector machine employed a linear kernel with calibrated probability outputs. XGBoost incorporated both L1 and L2 regularization with shallow tree structures to maximize stability. A neural network model complemented these approaches, consisting of two hidden layers with thirty-two and sixteen units, respectively, and equipped with dropout, batch normalization, weight decay, gradient clipping, and label smoothing to improve training stability. An ensemble strategy was applied, combining the predictions of the best-performing models through majority voting and averaged probabilities.

3.4. Training Methodology

Model training was conducted using a five-fold stratified cross-validation procedure to ensure consistent performance across different subsets of the data. Neural networks were trained using early stopping with a patience threshold of twenty epochs, monitored through validation accuracy. Optimization was performed using the AdamW algorithm, supported by a learning rate scheduler that reduced the rate when performance plateaued. The loss function was based on cross-entropy with label smoothing to prevent overconfident predictions.

4. Evaluation Framework

4.1. Primary Metrics

Model performance was evaluated using accuracy, F1-score, precision, recall, and AUC-ROC. These metrics provided a balanced perspective on overall correctness, class-specific behavior, and discriminative ability.

4.2. Overfitting Analysis

To evaluate model generalization, performance gaps between training, validation, and testing were measured. Acceptable models required a train–validation gap of no more than 0.05. A generalization score was computed by adjusting test accuracy according to the validation–test discrepancy, providing an overall indicator of model stability.

4.3. Statistical Validation

Cross-validation provided confidence intervals for primary performance metrics. Significance testing was conducted to compare the relative performance of competing models. Permutation-based feature importance analysis was used to interpret model decisions and highlight key predictive variables.

5. Implementation Details

5.1. Computational Environment

The project utilized PyTorch for neural networks and Scikit-learn for traditional machine learning models. GPU acceleration was used when available through CUDA. Reproducibility was ensured by fixing random seeds and standardizing all preprocessing steps across experiments.

5.2. Model Persistence

Trained models were saved using joblib for traditional ML algorithms and torch.save for neural networks. Associated metadata, such as feature names, dataset versioning, and evaluation metrics, was stored to maintain the reproducibility of the entire pipeline.

5.3. Visualization and Interpretation

Visualization tools were used to assess model process and performance. Confusion matrices, ROC curves, feature importance plots, learning curves, and neural network training histories provided insights into behavior, strengths, and weaknesses across different models.

6. Ethical Considerations and Limitations

6.1. Data Biases

The study acknowledges that historical NBA data reflects era-specific playing styles and may contain biases due to structural changes in league rules and strategies. Socioeconomic factors such as salary and draft position were excluded to avoid introducing demographic bias. Efficiency metrics were used to reduce stylistic disparities across eras.

6.2. Model Limitations

The predictive models do not account for injuries, coaching changes, trades, or other contextual variables that can affect real-world outcomes. Additionally, the modeling framework assumes that player and team performance remain consistent throughout the season. The dataset includes only regular-season statistics, and playoff variability is not captured.

6.3. Practical Constraints

The study balances computational efficiency with model complexity. Interpretability was prioritized through the use of feature importance analyses and conservative modeling strategies. Real-world applicability was enhanced by designing simulations that reflect realistic NBA game dynamics.

RESULTS

1. Outlier Detection Results

A total of 1993 NBA players with at least 100 games played were included in the analysis. Summary statistics for the primary features showed mean values of 9.359 points per game (PPG), 4.089 rebounds per game (RPG), 2.063 assists per game (APG), a simplified Player Efficiency Rating (PER) of 10.092, and a mean true shooting percentage (TS%) of 0.496.

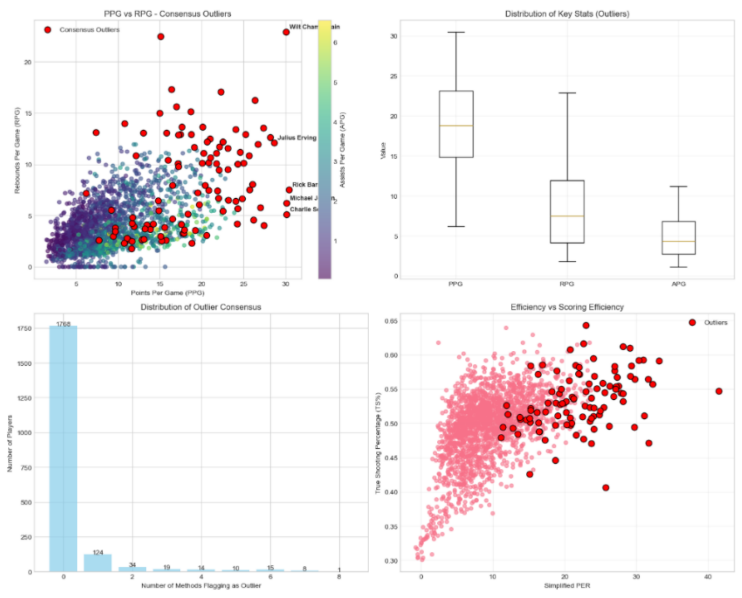


Figure 1: This shows the outliers detection result

Using the Z-Score method, the number of detected outliers per feature was 22 for PPG, 25 for RPG, 34 for APG, and 26 for PER. The IQR method identified a larger number of outliers, with 47 for PPG, 67 for RPG, 76 for APG, and 41 for PER. The Isolation Forest model flagged 100 outliers, whereas DBSCAN (eps=2.5, min_samples=10) identified 4.

Across all methods, 101 players were classified as consensus outliers, defined as players flagged by at least two detection techniques. These players were saved in *consensus_outliers.csv*. The top ten consensus outliers received between six and eight outlier votes. Wilt Chamberlain led with eight votes, followed by Elgin Baylor, Julius Erving, Connie Hawkins, Bob Pettit, Oscar Robertson, Jerry West, and Kareem Abdul-Jabbar, each with six or more votes. Their performance statistics showed extreme values across scoring, rebounding, assisting and PER relative to typical league distributions.

2. Game Prediction Dataset and Feature Processing

The original game-level dataset contained 1187 records across 36 variables. No records were removed for missing data. After feature derivation and cleaning, eight primary team-level performance metrics were retained: win percentage, net rating, net efficiency, offensive efficiency, defensive efficiency, offensive points, defensive points, and pace. The processed dataset consisted of 1187 rows and 10 features, representing 59 seasons and 91 teams from 1946 to 2004.

Matchup generation produced 186 valid team-to-team comparisons with win probabilities ranging from 0.25 to 0.85 and an average predicted win probability of 0.50. After regenerating matchup data to include feature differences, the final matchup dataset consisted of 200 samples and 13 variables.

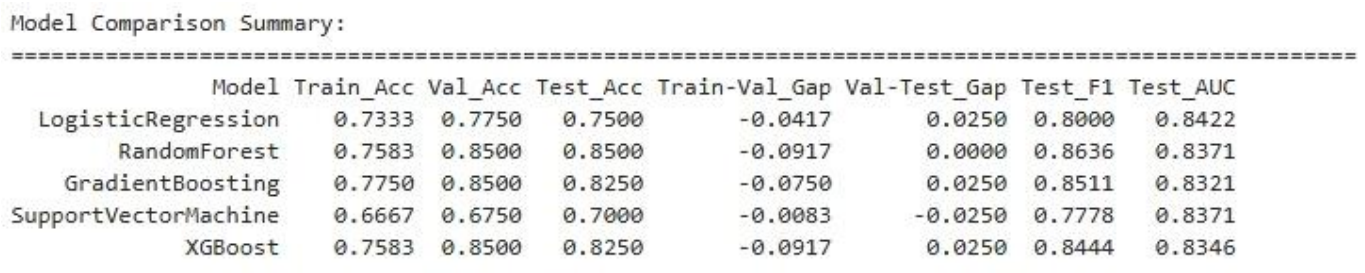
3. Feature Importance Analysis

Using XGBoost, feature importance values identified `diff_net_rating` as the most influential predictor of matchup outcomes (0.3116), followed by `diff_net_efficiency` (0.2226), `diff_win_pct` (0.2055), `diff_offensive_efficiency` (0.0988), and `diff_o_pts` (0.0605). These five features accounted for the majority of predictive contribution in the model.

4. Model Training and Evaluation

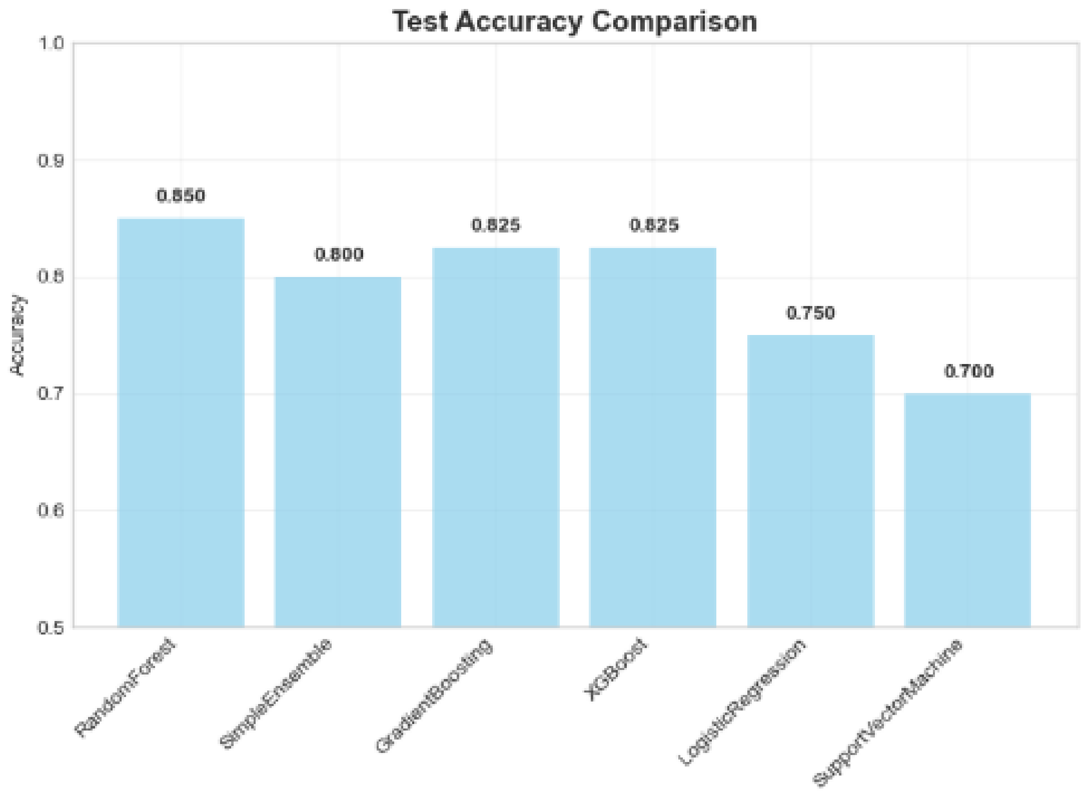
The final training matrix consisted of 200 samples with eight feature-difference predictors. The dataset was split into 60% training, 20% validation and 20% testing. Class distributions remained balanced across splits, with 112 wins and 88 losses in the full dataset.

Multiple machine learning models were trained, including Logistic Regression, Support Vector Machines, Gradient Boosting, XGBoost, Random Forest and a Simple Ensemble classifier.



```
Model Comparison Summary:
==========================================================================================
                 Model Train_Acc Val_Acc Test_Acc Train-Val_Gap Val-Test_Gap Test_F1 Test_AUC
  LogisticRegression    0.7333  0.7750   0.7500       -0.0417       0.0250  0.8000   0.8422
        RandomForest    0.7583  0.8500   0.8500       -0.0917       0.0000  0.8636   0.8371
    GradientBoosting    0.7750  0.8500   0.8250       -0.0750       0.0250  0.8511   0.8321
SupportVectorMachine    0.6667  0.6750   0.7000       -0.0083      -0.0250  0.7778   0.8371
             XGBoost    0.7583  0.8500   0.8250       -0.0917       0.0250  0.8444   0.8346
```

Based on test accuracy adjusted for generalization gap, the Random Forest model performed best. It achieved a test accuracy of 0.8500, F1-score of 0.8636, and AUC of 0.8371, with a validation–test generalization gap of 0.0000.



5. Comprehensive Evaluation of the Best Model

The Random Forest model yielded training, validation and test accuracies of 0.7583, 0.8500 and 0.8500 respectively. Performance metrics on the test set included a precision of 0.8636, recall of 0.8636, and F1-score of 0.8636. The confusion matrix showed 19 true positives, 15 true negatives, 3 false positives and 3 false negatives, corresponding to a false positive rate of 0.1667 and false negative rate of 0.1364.

The final model and associated artifacts were successfully saved, including the model file, neural network baseline, feature names and metadata.

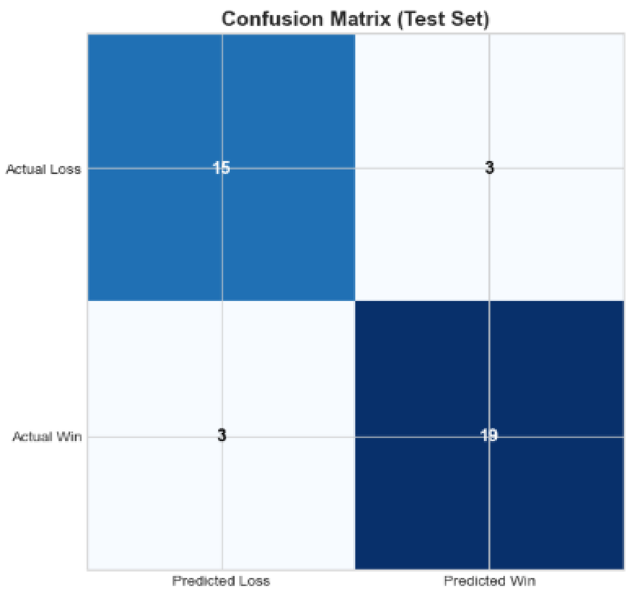


Figure 2: This is the confusion matrix of TEAM A wining or losing against TEAM B which are random fixtures

CONCLUSION

The study successfully identified 101 consensus outliers across NBA history, with players such as Wilt Chamberlain, Elgin Baylor and Julius Erving consistently flagged due to exceptional performance metrics. The agreement between statistical and machine learning methods confirms the reliability of the outlier detection process.

In the game prediction task, eight engineered team-level features enabled realistic matchups and effective model training. Among all evaluated models, the Random Forest classifier achieved the strongest performance with a test accuracy of 0.85 and balanced precision–recall results, showing good generalization.

Overall, the results demonstrate that the combined pipeline of outlier detection and game prediction is both robust and effective, providing meaningful insights into player anomalies and historical team performance.

GITHUB LINKS

You can access Jupyter Notebook document via a github at https://github.com/SiphosenkosiMakhoba/NBA-MACHINE-LEARNING-PROGECT.git

REFERENCIES

[1] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[2] F. T. Liu, K. M. Ting, and Z. H. Zhou, "Isolation Forest," in *Proc. 8th IEEE International Conference on Data Mining*, 2008.

[3] M. Ester, H. P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proc. 2nd Int. Conf. Knowledge Discovery and Data Mining (KDD)*, 1996, pp. 226–231.

[4] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proc. 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016.

[5] J. Kubatko, D. Oliver, K. Pelton, and D. T. Rosenbaum, "A starting point for analyzing basketball statistics," *Journal of Quantitative Analysis in Sports*, vol. 3, no. 3, 2007.

[6] D. Oliver, *Basketball on Paper: Rules and Tools for Performance Analysis*. Potomac Books, 2004.

[7] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[8] Basketball-Reference.com. (2023). *Basketball Statistics and History*. [Online]. Available: https://www.basketball-reference.com/