Predicting NBA Game Outcomes for Informed Sports Betting

Gabriel Okner William Pham Jacob Quisumbing Davita Verma Joy Zhang

June 30, 2025

Contents

1 Introduction

1.1 Problem Statement

In 2018, the Supreme Court's landmark decision to overturn the federal ban on sports betting unleashed the multi-billion-dollar gaming industry, forever redefining the economic potential of professional sports. With sports betting now legal in dozens of states and a projected market size surpassing $200 billion globally by 2030, the possibility of capitalizing on data-driven insights is at an all-time high.

This project aims to develop a predictive model that leverages a team's in-game performance metrics to forecast the likelihood of winning its next game. Focusing on a single game and predicting the next ensures the model remains precise and manageable, allowing it to account for immediate performance trends and contextual factors. This approach would minimize data complexity while maximizing predictive accuracy for short-term outcomes.

1.2 Motivation

One natural way to apply these predictions is through the moneyline, a type of sports bet that focuses solely on picking the outright winner of a game. Unlike point spreads, which factor in the margin of victory, moneyline bets simplify the process by requiring only a correct prediction of which team will win.

Moneyline betting strategies are often complemented by hedge bets that offer a practical approach to managing risk. In sports betting, hedging involves strategically placing counter-bets to secure profits or mitigate potential losses as circumstances evolve, such as during live gameplay. For example, if a value bet is placed on Team A to win based on the model's predictions, and the dynamics of the game shift—say, Team B starts gaining momentum—a hedge bet can be placed on Team B to offset potential losses. This tactic ensures that a favorable position is maintained, regardless of the final outcome.

The model output, predictions of which team is going to win, serves as the foundation for identifying value bets—situations where the sportsbook's implied probabilities undervalue a team's chances of success. By consistently placing bets with positive expected value and hedging to minimize potential losses, the bettor leverages the model's predictive advantage to exploit inefficiencies in sportsbook odds. Even modest improvements in accuracy above the baseline threshold can lead to significant long-term profits, as these small edges compound over multiple wagers.

This paper will explore how different models, ranging from simple classifiers like logistic regression to more advanced machine learning techniques like XG boost, can be employed to predict game outcomes. By analyzing their accuracy and robustness, we aim to identify which approaches are most effective in providing actionable insights for betting decisions.

2 Data Collection

2.1 Data Source

The dataset, sourced from Kaggle, provides comprehensive game-by-game statistics for NBA matches, including team performance, opponent metrics, and game outcomes. Kaggle is a reputable platform for high-quality, curated datasets, ensuring reliability and consistency in data collection and structure.

This dataset includes features such as team identifiers, game location (home or away), seasonal context, and various performance metrics. The primary target variable we created from the dataset is *Won Next*, indicating whether the team will win their next game. Features have been numerically encoded for ease of modeling, and the data captures a wide range of performance metrics for both teams and opponents, enabling robust analysis of game outcomes.

2.2 Dataset Analysis

To address our research problem, we used the NBA Games dataset, which consists of 17,772 rows and 151 columns. This dataset captures comprehensive details about team performance, game statistics, and outcomes, providing an excellent foundation for predictive modeling and trend analysis in basketball.

The dataset includes a wide range of features:

- **Game-Level Statistics:** Field goals, turnovers, and shooting percentages.
- **Team Metrics:** Offensive/Defensive ratings and usage percentages.
- **Game Context:** Home/away indicators, opponent stats, and game date.
- **Outcome Metrics:** Win/loss indicator and scoring data.
- **Seasons:** Timeline of years used to split the data into test and training sets.

We selected this dataset for its ability to provide actionable insights into the factors influencing game outcomes, team strategies, and performance trends. Thus, this dataset aligns perfectly with our project objectives by offering:

- **Predictive Power:** Rich features enable the development of robust models for game outcome prediction.
- **Versatility:** Diverse metrics allow for both statistical analyses and advanced machine learning applications.
- **Scalability:** The dataset's size supports generalizable insights across multiple seasons.

3 Models and Procedures

3.1 Data Preprocessing

Initially, numerical features were filtered using a low variance threshold of 0.01, eliminating features with minimal variability that provide limited predictive value. To further enhance the model, highly correlated features (correlation ¿ 0.9) were removed, reducing redundancy and mitigating multicollinearity issues.

The resulting dataset was then standardized, ensuring all features had a mean of 0 and a standard deviation of 1, an essential step for logistic regression's sensitivity to feature magnitudes.

The target variable, "Won Next," was encoded into binary labels (0 and 1) using a label encoder, preparing it for supervised learning. Instead of random splitting, the dataset was divided into training and testing sets based on seasons, with data from 2016–2021 used for training and 2022 reserved for testing, mimicking real-world prediction scenarios.

3.2 Baseline Model

The baseline model predicts NBA game outcomes by comparing the cumulative wins of the team and its opponent. For each game, if a team's total wins exceed the opponent's cumulative wins, the model predicts a win for that team. This simple approach assumes that a team's past performance is a strong indicator of future success, serving as a basic benchmark for more complex models.

The baseline model achieved an accuracy of 0.51, slightly better than random guessing. The precision was 0.51, meaning that when the model predicted a win, it was correct about half of the time. While the model performed better than random chance, its simple approach highlights the potential for more sophisticated models to capture more nuanced factors influencing game outcomes.

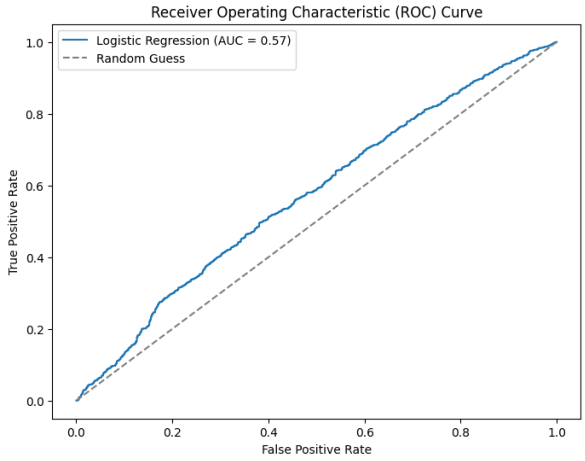
3.3 Logistic Regression

The logistic regression model developed for predicting the "Won Next" outcome incorporates a robust feature selection and preprocessing pipeline to ensure optimal performance.

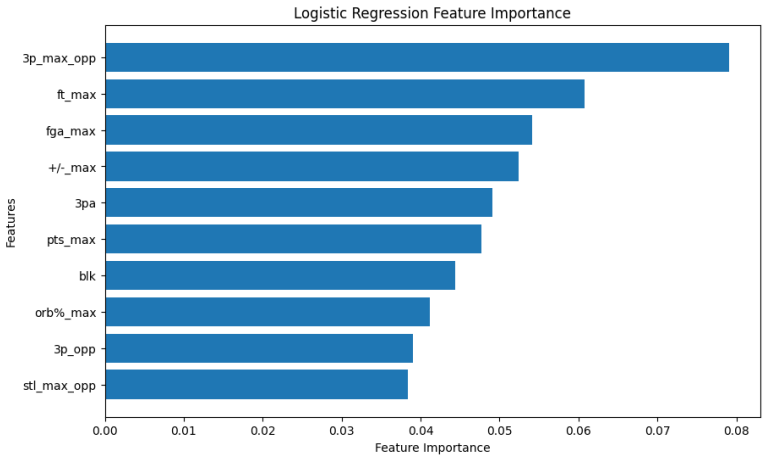
The logistic regression model was trained with a maximum of 2000 iterations to ensure convergence. Evaluation on the test set demonstrated the model's accuracy and provided detailed metrics, including precision, recall, and F1-score, through a classification report. This pipeline effectively balances simplicity and interpretability while addressing common challenges like feature redundancy, scale sensitivity, and realistic data splitting.

The results of the logistic regression model reveal moderate predictive performance, with an overall accuracy of 53.65% on the test set. The classification report provides further insight into the model's ability to distinguish between the two classes (0 and 1). For class 0, the model achieved a precision of 54%, recall of 46%, and an F1-score of 50%, indicating some difficulty in correctly identifying this class. Conversely, for class 1, the model showed slightly better performance with a precision of 53%, recall of 61%, and an F1-score of 57%, demonstrating greater success in identifying positive outcomes. The macro-average F1-score, which treats both classes equally, was 53%, while the weighted average F1-score, which accounts for class imbalance, was also 53%.

The probabilities generated by the model from *Won Next* can be directly compared to the implied probabilities from sportsbook odds. If the model predicts a higher likelihood of a team winning than the sportsbook odds suggest, it identifies a potential value bet, where the bettor has a statistical advantage to profit. This focused approach minimizes data complexity, ensuring the model remains manageable while capturing critical performance trends that are important for the bettor to keep in mind when placing a bet.



(a) Logistic Regression ROC Curve



(b) Logistic Regression Feature Importance

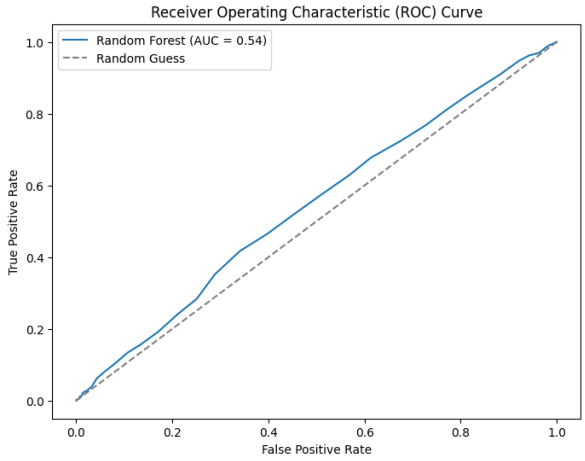
3.4 Random Forest

A Random Forest Classifier was chosen for its robustness in handling categorical relationships and modeling complex patterns. During data preparation, features (X) were derived by excluding the target variable (Won Next), which was then encoded into binary values (0 and 1) using LabelEncoder.

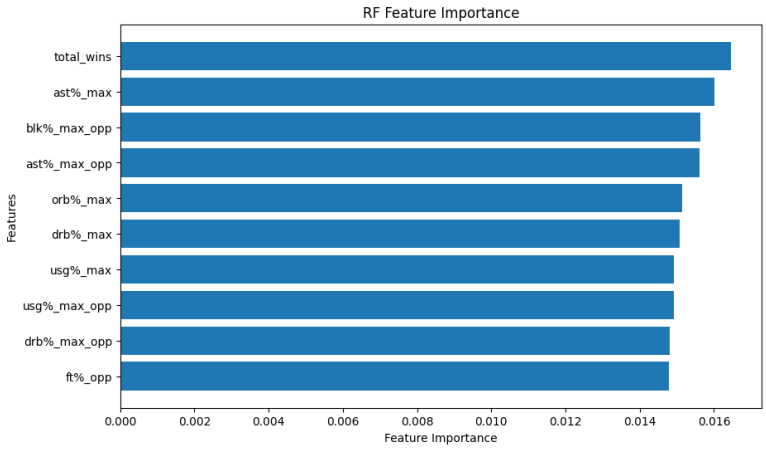
The model's performance was assessed using various evaluation metrics. It achieved an accuracy of 53.21%, slightly above random guessing (50%), indicating its ability to capture trends such as team strength and home advantage. Precision was 52.96%, meaning that when the model predicted a win, it was correct approximately half the time. With a recall of 57.49%, the model successfully identified a majority of actual wins. The F1-score, which balances precision and recall, was 55.13%, reflecting the model's overall consistency in predicting positive outcomes. The Random Forest algorithm's ability to model non-linear relationships contributed to the relatively strong recall performance. However, the modest accuracy and slightly lower precision suggest that the model's features do not fully capture the complexity of game outcomes, occasionally leading to false positives.

To enhance the model, feature engineering could be utilized. Adding metrics such as average win rates, point differentials, or historical performance trends would provide more context. Interaction terms between features like team and *home_opp* might also reveal deeper dynamics. Hyperparameter tuning, through methods like GridSearchCV, could optimize parameters such as *max_depth* and *min_samples_split* for improved performance. Comparing the Random Forest model with other approaches, such as Gradient Boosting or Logistic Regression, could identify a better-suited technique. Visualizations, including confusion matrix heatmaps and feature importance plots, would offer further insights into the model's strengths and limitations, guiding future improvements.

Random Forest is easier to implement than XGBoost, making it a practical choice for projects where simplicity and interpretability are priorities. However, it does not handle imbalanced data as effectively and can be less efficient at fine-tuning performance. XGBoost, on the other hand, uses advanced features like weighted loss functions and regularization, which often result in better accuracy and precision. It's also more computationally efficient for large datasets, though it requires more effort to set up and optimize. Compared to logistic regression, which is straightforward and provides clear probability estimates, Random Forest is better at capturing complex, non-linear relationships but can be harder to interpret. While Random Forest is a strong option for predicting game outcomes, XGBoost can outperform it in situations where accuracy and precision are critical, as long as the added complexity is manageable.



(a) Random Forest ROC Curve



(b) Random Forest Feature Importance

3.5 XG Boost

The third and final model explored was the Gradient Boosting Classifier, chosen for its ability to capture complex patterns in the data through iteratively improving accuracy by introducing small, weaker models that perform well on the residual errors of previous iterations. This boosting approach allows the model to focus on correcting the hardest-to-predict cases, gradually refining its overall predictive capability to a test set accuracy of 0.5596.

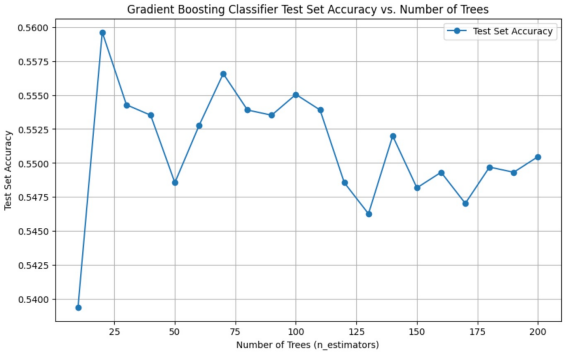
The test set accuracy decreases slightly as the number of trees, or boosting iterations, increases, as shown in the first figure, highlighting the danger of overfitting through excessive boosting. The feature importance plot in the second figure highlights the key variables contributing to the model's performance. Notably, being the Golden State Warriors emerged as the most influential feature, reflecting the dominance of specific teams in determining game outcomes. On the other hand, the model places significant weight on recent performance through heavily focusing on greatest lead and whether or not the team won their last game.

While these results suggest that the Gradient Boosting Classifier effectively leverages the provided features to make predictions, further refinement could enhance its performance. Feature engineering, such as incorporating additional metrics like recent win streaks or player-specific statistics, might reveal deeper insights. Having player-specific data would allow Gradient Boosting to capture unique interactions between players, such as the impact of specific matchups, player synergies, or defensive strengths, which could lead to more accurate and nuanced predictions.

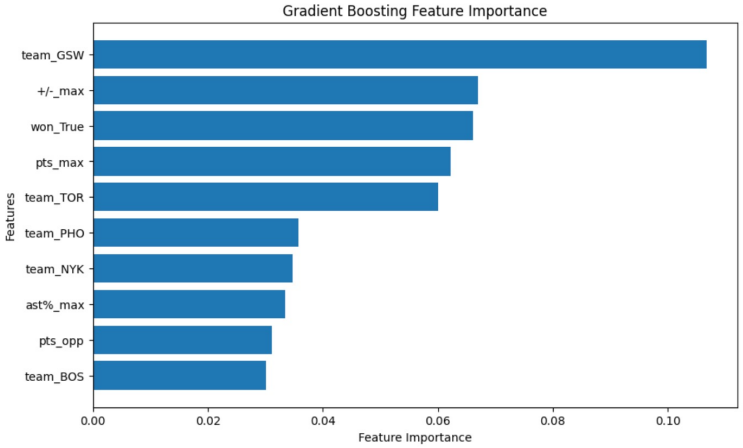
To optimize the model, hyperparameter tuning was conducted, including fine-tuning parameters like the learning rate and maximum tree depth using tools such as GridSearchCV. However, the cross-validation process for max depth proved computationally expensive, taking over two hours and 22 minutes to complete, with no meaningful improvement in test accuracy. This lack of improvement suggests that the model's current performance may already be close to optimal for the given features, and further increases in complexity do not provide additional value.

Compared to Logistic Regression, Gradient Boosting does the best job of predicting future games, but it carries a higher risk of overfitting and lacks natural probability outputs, making it less straightforward for calculating value bets.

Compared to Random Forest, gradient boosting offers less interpretability regarding which features to prioritize. In sports betting, this can be a disadvantage when trying to identify stable and highly predictable characteristics that consistently influence results, such as recent performance trends of a team. Random Forest's clearer ranking of important features can provide better guidance on which metrics to watch, helping bettors focus on reliable predictors. However, Gradient Boosting's superior accuracy gives it an edge in capturing subtle and complex patterns, which can result in more accurate predictions for less predictable or nuanced aspects of the game.



(a) Gradient Boosting Accuracy Chart



(b) Gradient Boosting Feature Importance

3.6 Model Comparisons

Based on our research and analyses, the performance and utility of the three models—Baseline, Logistic Regression, and Random Forest—exhibit notable differences in predictive accuracy, complexity, and interpretability. The Baseline model, which relies solely on cumulative team wins as a predictor, represents the simplest approach. While it achieved an accuracy of 51%, marginally better than random guessing, its reliance on a single metric limits its ability to capture nuanced game dynamics. This simplicity, however, makes it an effective benchmark for evaluating the incremental value added by more sophisticated models.

The Logistic Regression model, enhanced through robust preprocessing and feature standardization, achieved a slightly higher accuracy of 53.65%. Its interpretability is a key strength, as it provides insights into the relationships between features and game outcomes. However, the model displayed moderate precision and recall, with an F1-score of 53% on the test set, highlighting challenges in correctly identifying both winning and losing outcomes. Logistic Regression's linear decision boundary limits its capacity to model non-linear interactions, which could explain its modest improvement over the Baseline model.

The Random Forest model demonstrated a similar accuracy of 53.21% but stood out in terms of recall (57.49%), indicating a stronger ability to identify actual wins. The robustness of this model in handling nonlinear relationships and interactions between features allowed it to outperform Logistic Regression in identifying positive outcomes. However, its precision (52.96%) was slightly lower, suggesting a trade-off between identifying true positives and minimizing false positives. The Random Forest model benefits from its ability to capture complex patterns, but its results indicate that the features of the data set may not fully explain the game results, leading to occasional misclassifications.

Lastly, the Gradient Boosting model achieved the highest accuracy of 55.96%, leveraging iterative improvements to refine its predictions. This model excelled in capturing subtle patterns in the data, emphasizing influential features like recent performance and specific team identifiers. However, its susceptibility to overfitting, as evidenced by diminishing returns with excessive boosting, underscores the importance of careful hyperparameter tuning. Despite its higher complexity and computational cost, Gradient Boosting's precision and recall were comparable to Random Forest, indicating that while it may slightly improve accuracy, it shares similar limitations regarding feature sufficiency.

Overall, the Baseline model serves as a simple benchmark, while Logistic Regression offers interpretability and modest improvements. Random Forest balances complexity with better recall, capturing non-linear patterns more effectively. Gradient Boosting slightly outperforms the others in accuracy but at the cost of higher computational demands and diminishing returns without enhanced features. These comparisons highlight the importance of feature engineering and the trade-offs between complexity and interpretability in predictive modeling.

4 Conclusion

4.1 Summary

To summarize our predictive models, we can compare their performance in forecasting NBA game outcomes. The Baseline model, relying solely on cumulative wins, achieved an accuracy of 51%, serving as a simple benchmark as this was our most simple model. Even by employing this simple model, we still see a 1% increase in accuracy compared to random guessing which would yield 50% accuracy. Our Logistic Regression model showed improved accuracy to 53.65%, providing interpretable insights but limited capacity for modeling complex interactions. This model was used to capture how in game statistics can be used to predict game victors. Random Forest offered a comparable accuracy of 53.21%, excelling

in recall (57.49%) by effectively capturing non-linear relationships of game statistics on game outcome, such as amount of 3-pointers made. Gradient Boosting achieved the highest accuracy of 55.96%, leveraging iterative refinement to uncover subtle patterns, though it faced diminishing returns without further feature enhancement.

Model	Accuracy	TPR	FPR	Precision
Baseline	0.51	0.93	0.91	0.51
Random Forest	0.53	0.57	0.51	0.53
Logistic Regression	0.54	0.60	0.51	0.54
XGBoost	0.56	0.55	0.43	0.56

Table 1: Model Performance Comparison

4.2 Impact

Our results demonstrate the potential of these predictive models to enhance decision-making in contexts such as sports betting. The incremental improvements in accuracy, particularly the highest accuracy of 55.96% achieved by Gradient Boosting, highlight the ability of machine learning techniques to identify nuanced patterns within game statistics that traditional analytical methods might overlook. In a head-to-head betting scenario, even a slight edge above 50% accuracy can translate to long-term profitability, especially when applied systematically and at scale. However, most large-scale sports betting platforms place odds heavily in favor of the house and the odds of return are usually around 10% to 30%. This means that model accuracies would need to lean upwards of 70% to 90% to be remotely feasible in long-term profitability. If we wanted to improve our model accuracy, we can incorporate additional statistics such as player injuries or team dynamics to further refine predictions. Using an improved model to sports bet would require careful integration with risk management strategies and an understanding of betting market behaviors to maximize potential advantages while minimizing losses.