香港中文大學
The Chinese University of Hong Kong

FINA6609 - Group11

Win-Loss Prediction in NBA Games

Jianbo XIAO
Sijing HUANG
Wai Sum AU
Wing Yan LAM

A report submitted in partial fulfillment of the requirements
for the degree of MSc in Actuarial Science and Insurance Analytics

Department of Finance, Business School

Contents

1 Introduction

Professional basketball has developed rapidly at commercial leagues in various countries, especially during the past two decades. The National Basketball Association (NBA) in the United States, undoubtedly the most successful professional basketball league, has witnessed steady increase in revenue in recent years[1]. According to Forbes Magazine, the top five NBA teams are worth 27.1 billion USD in total[2].

Given the obvious economic incentive, different parties are looking for metrics, or formulas to predict game outcomes and maximize their winning probabilities on or off the field. Both analytical models and expert systems have been proposed to conduct sports forecasts, in particular the win-loss result or point spread (typical in betting odds) of games (*Song et al.*, 2020). As for analytical efforts, machine learning techniques and statistical analysis are adopted to address the problem. We presume some basic knowledge of the rules and gameplay of basketball. For newcomers, a useful glossary of common terms[3] from the NBA is strongly recommended.

In light of the above, this project aims to harness statistical tools in win-loss prediction of future NBA game based on box score data. The box score includes summaries of discrete in-game events that are recognized by eye, such as shots attempted and made, points, assists, personal fouls, and time spent on the court. In our analysis, statistics are fed into several machine learning models, namely *Naïve Bayes, K-nearest Neighbors (K-NN), Logistic Regression, Decision Tree, and Random Forest* to examine which model can deliver the highest overall accuracy with reasonable interpretability. The primary research question is: can we predict the game outcome for a team based on previous box score data?

[1] www.statista.com/statistics

[2] https://www.forbes.com/nba-valuations/list/

[3] https://stats.nba.com/help/glossary/

The rest of the paper is organized as follows. In Section 2, we discuss background information of the data set and methods for modeling game outcomes. Section 3 illustrates how we determine the final model. The Results section interprets the final model fit. We conclude with a summary of our findings and a brief outlook on the future of basketball game forecasting.

2 Data and Methodology

Team-level data from the NBA's official statistics website[4] are extracted by utilizing the web-scrapping functions in R package **hoopR**. To be consistent, all data manipulation, visualization, modeling, and testing are carried out by R 4.2.3 under RStudio environment. The alpha values of all statistical tests were set as 5%.

There used to be a number of other variables in the data set, but we exclude them in the following research by domain knowledge. There were 37 variables in for data cleansing, including the three variables ($Season$, $Type$, and $Date$) for game details, the sixteen box statistics for the subject team and the exact same sixteen box statistics for the opponent, and the two variables for game result ($DIFF$, and WL). Using these in-game statistics, we create new variables to compare the differences in performance between "Team" and "Opponent" in each of the respective categories. These new variables for our analysis are listed in Table 1.

The primary research question is a binary classification problem as there are two potential outcomes, to win (marked as 1) or to lose (marked as 0). A 3-game simple moving average with one lag is applied to the 15 team-level box statistics respectively. In other words, the above mentioned predictors are the moving average of last three games' metrics. For

[4] https://www.nba.com/stats

Variable Abbreviation	Description
Season	Season (in year)
Type	Season type
Date	Game date
Team	Team's name
Opponent	Opponent's name
WL	Team's outcome: Win (1) or Lose (0)
DIFF	Score difference
AST	Difference in Assists
BLK	Difference in Blocks
STL	Difference in Steals
DRB	Difference in Defensive rebounds
ORB	Difference in Offensive rebounds
PIP	Difference in Points in the paint
FBP	Difference in Fast break points
FOUL	Difference in Fouls
TOV	Difference in Turnovers
TOVP	Difference in Turnover points
FTp	Difference in Free throw percentage
FGp	Difference in Field goal percentage
P2p	Difference in Two point field goal percentage
P3p	Difference in Three point field goal percentage

Table 1: Description of 21 variables used for analysis

instance, for Game 4, we would calculate the moving average using the metrics from games 1 to 3. For Game 5, we would use games 2 to 4, and so on.

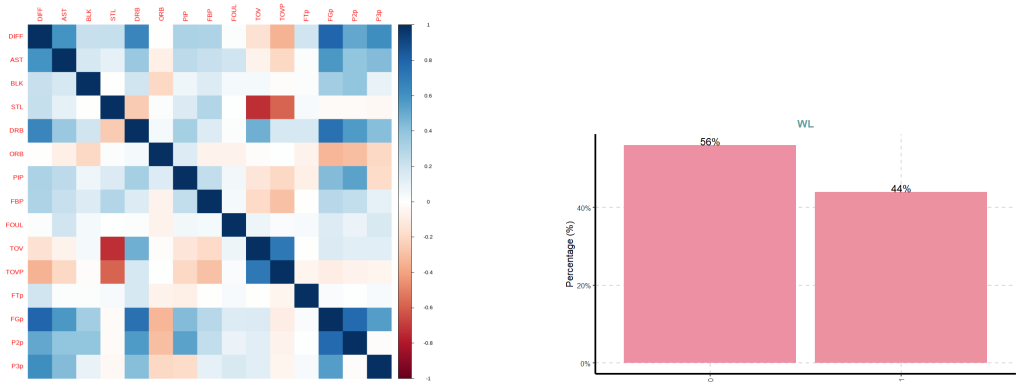
2.1 Exploratory Data Analysis

After outlier removal and standardizing numerical variables, game records from 16 October 2018 to 20 July 2021 are served as the training data while the game records from 19 October 2021 to 16 June 2022 are served as the test data. There are 3,524 game records in the training data

and 1,317 game records in the test data.

Some phenomena are discovered in exploratory data analysis, which shed light on follow-up research. Figure 1 (a) visualizes the Pearson correlation between numerical variables listed in Table 1. All correlations are below 0.8, suggesting no strong linear relationship or multicollinearity among predictors. However, it does not guarantee the absence of multicollinearity completely, and hence we should also check variance inflation factor (VIF) whenever appropriate in regression part (*Kumari*, 2008).

Classification imbalance problem arises as the response WL is slightly more likely to be 0 (56%) as shown in Figure 1 (b). Thus, the performance of some machine learning algorithms may degrade as they are designed to optimize accuracy while ignoring the effect of imbalanced distribution (*Zou et al.*, 2016).



(a) Heatmap of correlation

(b) Distribution of WL

Figure 1: Visualization

Therefore, different metrics should be used to evaluate the performance of a classifier apart from Accuracy (in Equation 1). Precision (in Equation 2) measures probability that a positive prediction is correct, which is useful when FP (False Positive) is a higher concern than FN (False Negative). Recall (in Equation 3) indicates how well a binary classifier correctly

identifies a conditional probability of correctly labeled members of the target class, and it is useful when FN trumps FP. Finally, the F1 score (in Equation 4) is a harmonic mean of precision and recall, it captures both trends in a single metric and is scored [0,1] (*Lipton et al.*, 2014). We emphasize F1 score more to access the performances of classifiers in the test data set. These statistics are calculated according to the confusion matrix in Table 2.

ACTUAL CLASS / PREDICTED CLASS	**Class=Yes**	**Class=No**
Class=Yes	TP	FN
Class=No	FP	TN

Table 2: Confusion Matrix for Classification

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{1}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{2}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{3}$$

$$\text{F1} = \frac{\text{2*Precision*Recall}}{\text{Precision} + \text{Recall}} \tag{4}$$

With the above insights, popular classification algorithms such as *Naïve Bayes, K-nearest Neighbors (K-NN), Logistic Regression, Decision Tree, and Random Forest* are employed in this context. The best model is selected based on its performance and interpretability on the validation data set.

2.2 Naïve Bayes

Naïve Bayes is a classification technique with a strong (and naive) assumption that all the predictors are independent to each other. The main interest is to find the posterior probabilities, or the probability of a class given some observed features. It helps to make computation simple and has better speed and accuracy for large data. In addition, it can calculate the most possible output based on input and add new data at run time (*Ahmad et al.*, 2015). We intend to use Naïve Bayes classifier as our benchmark for comparison.

2.3 K-nearest Neighbors

K-Nearest Neighbors (KNN) is a popular algorithm used for classification tasks for several reasons, such as simplicity and intuition. The K represents the number of closest neighbors to the new data point to classify by majority vote. It is a non-parametric algorithm, which means it does not assume a specific functional form for the data. This flexibility allows KNN to be effective in situations where the decision boundary is complex or non-linear (*Guo et al.*, 2003). In our approach, the optimal value of $K = 13$ was selected based on five-fold cross-validation on training set as seen in Figure 2.

2.4 Logistic Regression

Logistic regression is a generalized linear regression model commonly used in classification. Through logistic regression, weights of different predictors can be acquired, which means it is possible to identify some important influential elements among all others in the model (*Kirasich et al.*, 2018). There should be a decision boundary or threshold for the model which separates the two outcomes, which is often set to be 0.5

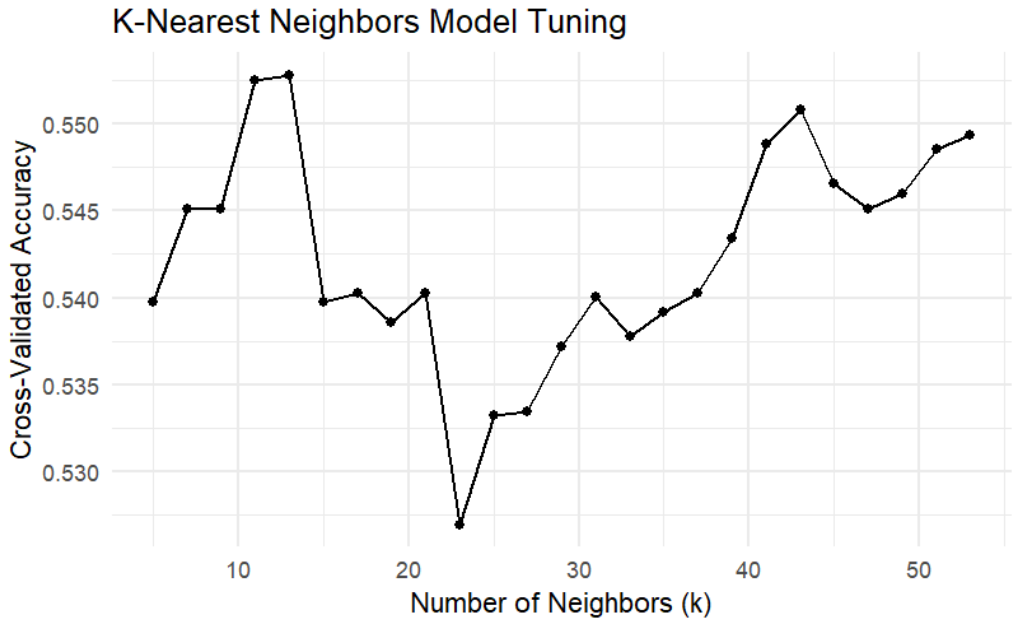


Figure 2: Tuning for Optimal K

in conventional analyses.

2.5 Decision Tree

Decision tree is often used to evaluate the risk of some projects or assess the probability of successful solutions to certain problems. It is actually equivalent to a mapping relationship between attributes and target values (*Patel and Prajapati*, 2018). We prune the tree (see Figure 3) corresponding to the optimal complexity parameter obtained using cross-validation. When a decision tree model is performing average, it is worth considering random forest algorithm to potentially improve the model's performance.

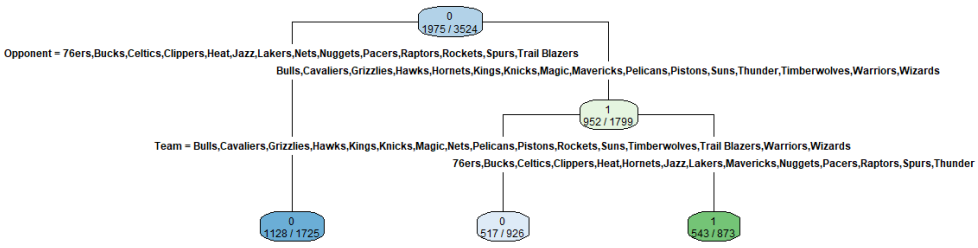


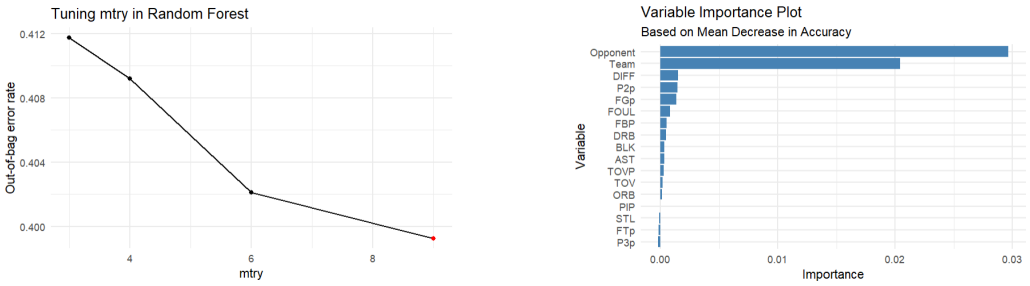
Figure 3: Decision Tree with Optimal Tree Size

2.6 Random Forest

Random forest is an ensemble learning method for classification. It builds multiple decision trees and aggregate them together to get a more accurate prediction. In a random forest, only a subset of the features are taken into consideration by the algorithm for splitting a node. The advantages include computing efficiency, robustness to outliers and noises, reducing the variance and overfitting, and applicable to high-dimensional data (*Li et al.*, 2015).

Random forests, are prone to overfitting, especially when allowed to grow very deep trees (*Cutler et al.*, 2012). The $mtry$ parameter does not directly control the depth or complexity of the trees. Hence, we restrict other parameters, such as $maxnodes$ (the maximum number of terminal nodes trees in the forest can have) to be 50, and $nodesize$ (the minimum size of terminal nodes) to be 3, to potentially reduce overfitting. For each bootstrapped replicate, $mtry$ variables are considered at each split at random. For tuning parameters, the increase of the number of trees will surely increase accuracy at the cost of computation time. And for the choice of the number of candidate variables at each split, the optimal choice is $mtry = 9$

according to out-of-bag error shown in Figure 4(a). Figure 4 (b) suggests that $Opponent$, $Team$, and $DIFF$ are the most important variables in the random forest model.



(a) Tuning for Optimal *mtry*

(b) Variable Importance Plot

Figure 4: Random Forest

3 Model Selection and Validation

Following the methodology, we apply these machine learning algorithms in an attempt to predict the dichotomous variable WL. The optimal model is to be determined based on its performance as well as interpretability, specifically on the validation data set. To assess the performance, we first develop the best-performing model under each algorithm in an iterative manner using the training set, and then evaluate these models on two important metrics, Accuracy and F1 score, using the test set. In particular, we attach greater importance to F1 score as explained in **Exploratory Data Analysis**.

The performance measures are provided in Figure 5. Although Random Forest outperforms all the other models in the training set, it does not generate satisfactory result on the test set. Whereas, the benchmark model Naïve Bayes exhibits the highest test accuracy of 56.87%.

Model	Training_Accuracy	Test_F1_score	Test_Accuracy
Naïve Bayes	59.68%	50.44%	56.87% (0.54, 0.60)
K-nearest Neighbors	55.28%	39.26%	52.54% (0.50, 0.55)
Logistic Regression	62.32%	55.52%	55.35% (0.53, 0.58)
Decision Tree	62.09%	32.62%	52.32% (0.50, 0.55)
Random Forest	72.47%	39.84%	53.23% (0.50, 0.56)

Figure 5: Performance Measures

However, an optimal model is not only the one that produces the highest accuracy, but also one that maintains a balance between precision (proportion of true positives over all positive predictions) and recall (proportion of true positives over actual positives). The F1 score serves as an effective measure of this balance, providing a single metric that combines both precision and recall. In our tests, Logistic Regression demonstrates the highest F1 score of 55.52%.

Moreover, interpretability is a significant factor in determining the final model. While black-box models like Random Forest might provide decent performance on training data set, they lack the transparency required for complete understanding and explanation. Models like Logistic Regression and Decision Trees, on the other hand, provide feasible interpretability. In particular, Logistic Regression provides the ease of interpretation with the coefficients indicating the impact and direction of the predictor variables on the outcome.

Taking all these considerations into account, and given its high F1 score and interpretability, we select Logistic Regression as the best model for the predictive task on WL. We apply backward selection to reduce model complexity by its advantage of considering interactions between variables from the start (*Andersen and Bro*, 2010). We determine the op-

timal subset of predictors by AIC criterion on training set. The final logistic regression model includes $Team$, $Opponen$, $DIFF$, and $TOVP$. Next, we seek the optimal threshold that maximizes the sum of sensitivity and specificity from the ROC curve (receiver operating characteristic curve) for this specific task as inspired by **Exploratory Data Analysis**. The ultimate threshold is found to be 0.4166 (Figure 6), which is consistent with the imbalanced distribution of WL. The largest VIF being 1.55 comes from $Team$, which further confirms our little multicollinearity assumption among the independent variables.

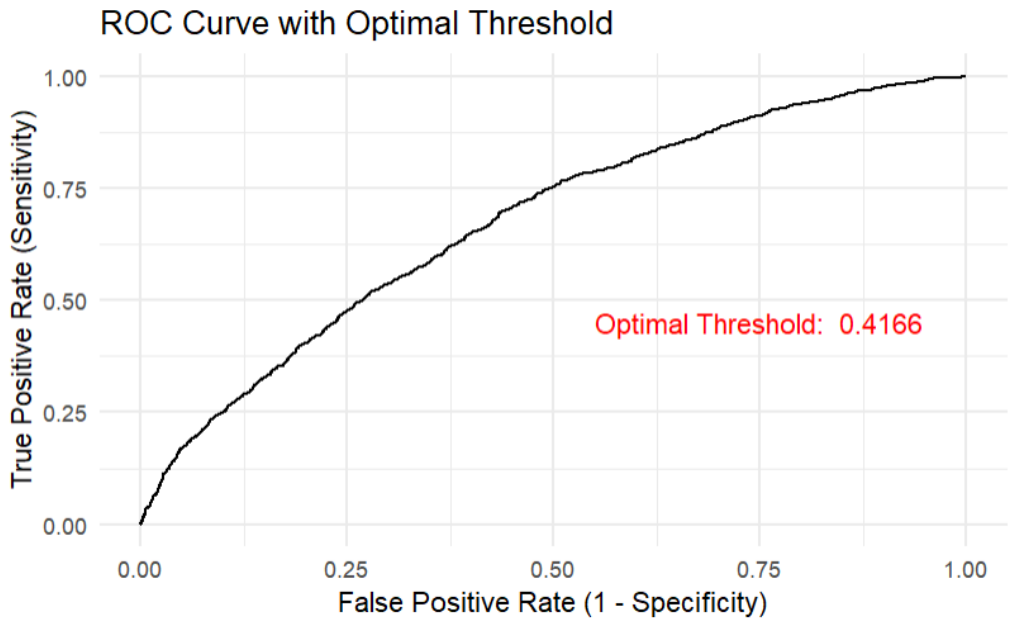


Figure 6: Optimal Threshold for Logistic Regression Model

4 Results

The coefficients of the logistic model are listed in Table 3. They indicate the change in the log odds of WL for a one-unit increase in the

Variable	Coefficient	Variable	Coefficient
Intercept	-1.151	TeamBucks	0.509
TeamBulls	-0.498	TeamCavaliers	-1.026
TeamCeltics	0.213	TeamClippers	0.521
TeamGrizzlies	-0.199	TeamHawks	-0.458
TeamHeat	0.173	TeamHornets	-0.377
TeamJazz	0.222	TeamKings	-0.245
TeamKnicks	-0.623	TeamLakers	0.491
TeamMagic	-0.395	TeamMavericks	0.131
TeamNets	0.059	TeamNuggets	0.267
TeamPacers	0.113	TeamPelicans	-0.320
TeamPistons	-0.822	TeamRaptors	0.484
TeamRockets	-0.213	TeamSpurs	-0.172
TeamSuns	0.064	TeamThunder	0.085
TeamTimberwolves	-0.659	TeamTrail Blazers	0.118
TeamWarriors	0.034	TeamWizards	-0.741
OpponentBucks	-0.017	OpponentBulls	1.934
OpponentCavaliers	1.971	OpponentCeltics	0.654
OpponentClippers	0.545	OpponentGrizzlies	1.202
OpponentHawks	1.248	OpponentHeat	0.835
OpponentHornets	1.353	OpponentJazz	0.135
OpponentKings	1.228	OpponentKnicks	1.650
OpponentLakers	0.710	OpponentMagic	1.403
OpponentMavericks	1.015	OpponentNets	0.714
OpponentNuggets	0.270	OpponentPacers	0.963
OpponentPelicans	1.371	OpponentPistons	1.596
OpponentRaptors	0.730	OpponentRockets	0.755
OpponentSpurs	0.841	OpponentSuns	1.104
OpponentThunder	1.054	OpponentTimberwolves	1.546
OpponentTrail Blazers	0.647	OpponentWarriors	0.980
OpponentWizards	1.232	DIFF	0.146
TOVP	-0.129		

Table 3: Coefficients of Logistic regression

corresponding predictor, assuming all other predictors are held constant.

For example, when the 'TeamBucks' predictor increases by one unit

(i.e., when the team playing is the Bucks), the log odds of WL increases by 0.509, given all other variables are held constant. This suggests that when the Bucks are playing, the odds of winning tend to be higher than the reference team - 76ers. Conversely, for 'TeamBulls', the coefficient is -0.498, which indicates that when the Bulls are playing, the log odds of WL decrease, again assuming all else constant.

The variables 'OpponentBulls', 'OpponentCavaliers', and etc., can be interpreted in a similar way but for the opponent's team. For example, when the opposing team is the Bulls, the log odds of WL increase significantly, which suggests that winning is more likely when playing against the Bulls.

These coefficients from $Team$ and $Opponent$ align well with the actual team rankings[5] in the NBA, as depicted in Figure 7.

For numeric variables, the positive $DIFF$ coefficient suggests that higher this value, the greater the log odds of WL. $DIFF$ can be interpreted as a comprehensive representation of both offensive and defensive prowess. It encapsulates the net performance outcome, factoring in the points scored by the team and those conceded to the opposition.

Conversely, $TOVP$ serves as an indicator of defensive vulnerabilities: the negative $TOVP$ coefficient suggests that a higher $TOVP$ value decreases the log odds of the outcome. In the context of NBA games, turnovers represent one of the most straightforward avenues for opposing teams to score points. As such, a high $TOVP$ value signifies lapses in the team's defensive strategy, thus indicating a weaker defense.

These two parameters together provide a robust assessment of team performance, where $DIFF$ signifies overall strength and $TOVP$ points towards potential areas of improvement in defensive tactics.

[5]https://www.basketball-reference.com/leagues/

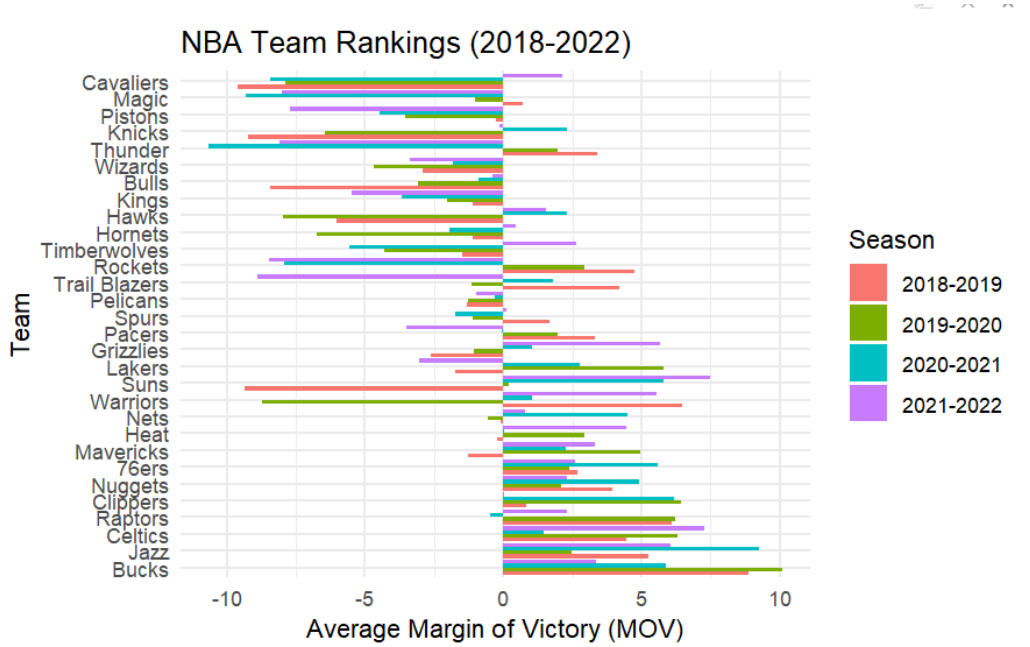


Figure 7: NBA Team Rankings

5 Conclusion

Our project focuses on prediction of game outcomes, which is a topic of considerable interest in the realm of sports analytics.

Our data set presents an inherent challenge due to class imbalance, which is a common issue in many real-world scenarios. This imbalance can lead to skewed predictions favoring the majority class. To tackle the problem, we use the best-tuned models developed from five popular machine learning algorithms (*Naïve Bayes, K-nearest Neighbors (K-NN), Logistic Regression, Decision Tree, and Random Forest*) to make predictions and evaluate their performance by two key metrics: accuracy and F1 score. This dual evaluation framework allows us to ensure that our final model is not only accurate but also balanced in their predictions.

Upon rigorous testing and comparison, the logistic regression model turns out to be the most effective. It not only excels in predicting game outcomes, but it also provides interpretability which is crucial for stakeholders to understand which factors are significant in determining game outcomes. Our findings provide quantitative evidence to the arguments put forth by NBA coaches and experts (*Baker and Farrow*, 2015). Specifically, the coefficients of our logistic regression model present a compelling narrative.

Future research in the domain of basketball analytics promises exciting advancements and opportunities. Current efforts can be extended in several directions, each with the potential to bring about considerable improvements in our understanding and prediction of game outcomes.

One promising avenue involves the exploration of alternative machine learning methodologies. Techniques such as support vector machines (SVM) or artificial neural networks (ANN) could offer fresh perspectives and potentially superior predictive power (*Kollár*, 2021). This exploration could also involve revisiting the feature selection phase. By incorporating additional factors into consideration, such as player-level metrics, and venue characteristics, we could add more complexity and nuance to the predictive process (*Zuccolotto and Manisera*, 2020).

Another compelling area of future research pertains to the shift from box-score metrics towards models that infer latent aspects of team and player performance from rich spatio-temporal data, where deep learning methods can play an instrumental role (*Sicilia et al.*, 2019), particularly in the context of tracking data.

In essence, the future of basketball analytics is bright and brimming with possibilities. By combining advanced machine learning techniques, and sophisticated feature selection, we can deepen our understanding of the game, inform strategic decision-making, and potentially transform the way basketball is played and analyzed.

6 References

Ahmad, P., Qamar, S., & Rizvi, S. Q. A. (2015). Techniques of data mining in healthcare: a review. *International Journal of Computer Applications*, 120(15).

Andersen, C. M., & Bro, R. (2010). Variable selection in regression—a tutorial. *Journal of chemometrics*, 24(11□12), 728-737.

Baker, J., & Farrow, D. (2015). Routledge handbook of sport expertise. *Routledge*.

Cutler, A., Cutler, D. R., & Stevens, J. R. (2012). Random forests. *Ensemble machine learning: Methods and applications*, 157-175.

Guo, G., Wang, H., Bell, D., Bi, Y., & Greer, K. (2003), KNN model-based approach in classification, *On The Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE: OTM Confederated International Conferences, CoopIS, DOA, and ODBASE 2003, Catania, Sicily, Italy, November 3-7, 2003. Proceedings*, 986-996, Springer Berlin Heidelberg.

Kirasich, K., Smith, T., & Sadler, B. (2018), Random forest vs logistic regression: binary classification for heterogeneous datasets, *SMU Data Science Review*, 1(3), 9.

Kollár, A. (2021). Betting models using AI: A review on ANN, SVM, and Markov Chain.

Kumari, S. S. (2008). Multicollinearity: Estimation and elimination. *Journal of Contemporary research in Management*, 3(1), 87-95.

Li, L., Wu, Y., & Ye, M. (2015). Experimental comparisons of multi-class classifiers. *Informatica*, 39(1).

Lipton, Z. C., Elkan, C., & Narayanaswamy, B. (2014). Thresholding classifiers to maximize F1 score. *arXiv preprint arXiv:1402.1892*.

Patel, H. H., & Prajapati, P. (2018), Study and analysis of decision tree based classification algorithms, *International Journal of Computer Sciences and Engineering*, 6(10), 74-78.

Sicilia, A., Pelechrinis, K., & Goldsberry, K. (2019, July). Deephoops: Evaluating micro-actions in basketball using deep feature representations of spatio-temporal data. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery* & *Data Mining* (pp. 2096-2104).

Song, K., Zou, Q., & Shi, J. (2020). Modelling the scores and performance statistics of NBA basketball games. *Communications in Statistics-Simulation and Computation*, 49(10), 2604-2616.

Zou, Q., Xie, S., Lin, Z., Wu, M., & Ju, Y. (2016), Finding the best classification threshold in imbalanced classification, *Big Data Research*, 5, 2-8.

Zuccolotto, P., & Manisera, M. (2020). Basketball data science: With applications in R. CRC Press.

Acknowledgements

This project did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors. All authors contributed in the following manner to the development of this paper. Jianbo XIAO and Wai Sum AU developed the design of the study, performed analyses, and discussed the results, and all authors contributed to the writing of the manuscript. Especially, these authors thank Prof. Ben Lim for the guidance and help at the early stage of this work.

Disclosure Statement

No potential conflict of interest was reported by the authors.