Predicting Match Outcomes in League of Legends

Yannick Broich

January 2025

Abstract

This project predicts match outcomes in League of Legends (LoL) based on early game performance metrics. The analysis uses data from approximately 10,000 matches, focusing on the first 10 minutes of gameplay. The dataset includes features such as gold earned, kills, deaths, and experience. After accounting for multicollinearity and employing feature engineering, our models identify critical predictors of match outcomes, such as gold per minute, experience gap, and first blood. We evaluated several models, including Logistic Regression, LASSO, k-Nearest Neighbors, and Random Forest, on the basis of accuracy and interpretability. The results show that resource management and early game objectives are crucial in predicting wins. While Logistic Regression and Random Forest performed similarly in terms of prediction accuracy, Random Forest provided deeper insights into feature importance. The findings highlight the importance of early-game strategy and efficient resource management in achieving success. Finally, omitted factors, such as team composition and player behavior, could improve prediction accuracy and model robustness.

1 Introduction

In the world of competitive gaming, split-second decisions and strategic foresight can determine victory or defeat. League of Legends (LoL), perhaps the most popular (citation needed) multiplayer online battle arena (MOBA) game, is no exception. Each match is a complex web of player actions, map control, and team coordination. But what if we could predict the outcome of a game based on early-game performance?

In this project, we dive into the data from LoL matches, focusing on key metrics collected after the first 10 minutes of gameplay. By analyzing these variables—such as gold earned, kills, turret damage, and other strategic elements—we aim to uncover which factors are most influential in shaping the trajectory of the game. The goal is to build a predictive model that not only helps to understand the underlying dynamics of a match but also sheds light on the critical decisions that can swing the game in favor of one team or the other.

2 League of Legends in a Nutshell

2.1 The Map

League of Legends is played on a map called Summoner's Rift, featuring three lanes—Top, Mid, and Bot—and a jungle that lies between them. Each lane has two outer turrets a inhibitor and a inhibitor. The inhibitors, when destroyed, allow your team to spawn stronger minions in that lane. Inside each team's base are two Nexus towers that protect the Nexus, the core structure that must be destroyed to win the game. Minions spawn from each team's Nexus starting at 1:05 and continue spawning every 30 seconds. Players gain gold by killing these minions, which can be used to buy items that strengthen their champions. The jungle is filled with neutral monsters that the Jungler can defeat for gold and experience. Additionally, the jungle houses powerful objectives like the Dragon, which provides permanent team-wide buffs, and Baron Nashor, whose buff is temporary and empowers minions which aid your team pushing towards the enemy base. The entire map is covered in fog of war excluding the bases. Towers, players and wards create vision.

Figure 1: Summoner's Rift

2.2 The Players

Each team consists of five players: the Top laner, who is usually a tanky fighter; the Mid laner, often a high-damage mage or assassin; the Bot laner (or Marksman), who focuses on ranged damage; the Support, who aids the team by protecting allies and setting up kills; and the Jungler, who roams the map to secure objectives and assist the other lanes.

2.3 Road to Victory

In LoL, to win the game, you need to destroy the nexus of the opposing team. To achieve this, teams must take down enemy turrets, destroy inhibitors to strengthen their minions, and secure key objectives like the Dragon and Baron Nashor to gain strategic advantages that help in pushing through to the enemy's Nexus.

3 Data Description

The dataset provides statistics from the first 10 minutes of approximately 10,000 ranked games played at a high ELO (top 0.5%) level.

For each game, 19 features per team (38 in total) are recorded after the first 10 minutes, covering key metrics such as kills, deaths, gold, experience, and player levels.

To get a rough overview of the relationship between the variables and the target variable blueWins please see figures 3 to 9 in the appendix.

4 Feature Engineering

Due to the nature of the data, we anticipated a significant degree of multicollinearity among the features.

Feature	**VIF**
blueWardsPlaced	2.556
blueWardsDestroyed	2.756
blueFirstBlood	2.513
blueKills	60.395
blueDeaths	28.838
blueAssists	13.263
blueEliteMonsters	∞
blueDragons	∞
blueHeralds	∞
blueTowersDestroyed	1.568
blueTotalGold	∞
blueAvgLevel	2079.0
blueTotalExperience	2150.4
blueTotalMinionsKilled	46.684
blueTotalJungleMinionsKilled	18.456
blueGoldDiff	18.456
blueExperienceDiff	∞
blueCSPerMin	∞
blueGoldPerMin	∞

Table 1: VIF before feature engineering

Based on the VIF analysis, several variables were removed or transformed to mitigate multicollinearity:

Deleted Variables

- **blueTotalGold** was removed in favor of **blueGoldPerMin**.
- **blueTotalExperience** was removed in favor of **blueAvgLevel**.
- **blueTotalMinionsKilled** was excluded, as **blueCSPerMin** offers a normalized measure of minion efficiency.

Newly Created Variables

- **blueKDA** was introduced by combining **blueKills**, **blueDeaths**, and **blueAssists**.
- **blueVisionScore** was created by aggregating **blueWardsPlaced** and **blueWardsDestroyed**.
- **blueTotalJungleUnits** was created by combining **blueDragons** and **blueTotalJungleMinionsKilled**, excluding the Rift Herald.

After these transformations, the remaining features were re-evaluated to ensure that multicollinearity was sufficiently reduced while retaining critical predictive features.

Feature	VIF
blueWins	2.719936
blueFirstBlood	2.327524
blueDragons	1.749220
blueHeralds	1.328353
blueTowersDestroyed	1.257660
blueAvgLevel	571.682257
blueExperienceDiff	2.016036
blueCSPerMin	139.240888
blueGoldPerMin	303.138607
blueKDA	3.194518
blueVisionScore	2.912515
blueTotalJungleUnits	34.696691

Table 2: VIF after transformations

5 Methodology

To predict game outcomes (blueWins), we utilized both linear and non-linear models. Linear models, such as Logistic Regression (LR) and LASSO Logistic Regression, were selected due to their simplicity and suitability for exploring relationships in datasets with low-dimensional structures. These models allow for clear interpretability of the effects of individual features on game outcomes and provide a baseline for comparison with more complex models.

Non-linear models, including k-Nearest Neighbors (kNN) and Random Forests, were chosen to capture potential non-linear relationships between features and the target variable. League of Legends is a highly dynamic game, where complex interactions between variables (e.g., gold income, experience differences, and skirmish success) might play a significant role in determining outcomes. Non-linear models are well-suited to account for these complexities, particularly in scenarios where linear relationships may fail to fully explain the variance in the data.

Our hypothesis is that while the dataset may exhibit an underlying low-dimensional structure favoring linear models, non-linear models will provide additional flexibility to capture subtle interactions and patterns.

6 Results

The results align well with the hypothesis presented at the beginning of the analysis. Here are the key points for discussion:

6.1 Model Performance

- **k-Nearest Neighbors (kNN):** The kNN model achieved an accuracy of 0.705 with an optimal k value of 31. While it highlights clear separations in the dataset, kNN is less interpretable than other models as it does not provide feature importance.

- **Random Forest:** The random forest model achieved the highest accuracy of 0.731. Its top predictors, such as *blueExperienceDiff*, *blueKDA*, and *blueGoldPerMin*, align with domain knowledge about the game.
- **Logistic Regression:** The baseline logistic regression achieved an accuracy of 0.721, supporting the hypothesis that the data has a low-dimensional structure where simple linear models can perform well.
- **LASSO Logistic Regression:** LASSO shrunk the coefficients on all polynomial and interaction terms to zero, resulting in performance identical to the baseline logistic regression.
- **XGB Logistic Regression:** Achieved the lowest accuracy of 0.696, further supporting the hypothesis that the underlying relationships are simple.

6.2 Hypothesis Validation

Feature Importance Across Models:

- LASSO and Logistic Regression highlighted *blueDragons*, *blueAvgLevel*, and *blueFirstBlood* as key features, aligning with game mechanics where early objectives often dictate the game's momentum.
- Random Forest emphasized *blueExperienceDiff*, *blueKDA*, and *blueGoldPerMin*, reflecting the importance of resource control and successful skirmishes.

Low-Dimensionality Hypothesis: To validate the hypothesis, we ran simple logistic regressions on *blueDragons* (the most important feature in linear models) and *blueExperienceDiff* (the most important feature in non-linear models). While the *blueDragons* model performed poorly, the *blueExperienceDiff* model achieved comparable accuracy and AUC to kNN, and only slightly lower than Logistic Regression and Random Forest. This supports the idea of an underlying low-dimensional structure.

6.3 Insights from ROC Curves

The ROC curves for all models—Logistic Regression, LASSO Logistic Regression, Random Forest, and kNN—showed similar AUC values ranging from 0.79 to 0.82. This consistency further supports the reliability of the predictions across model types.

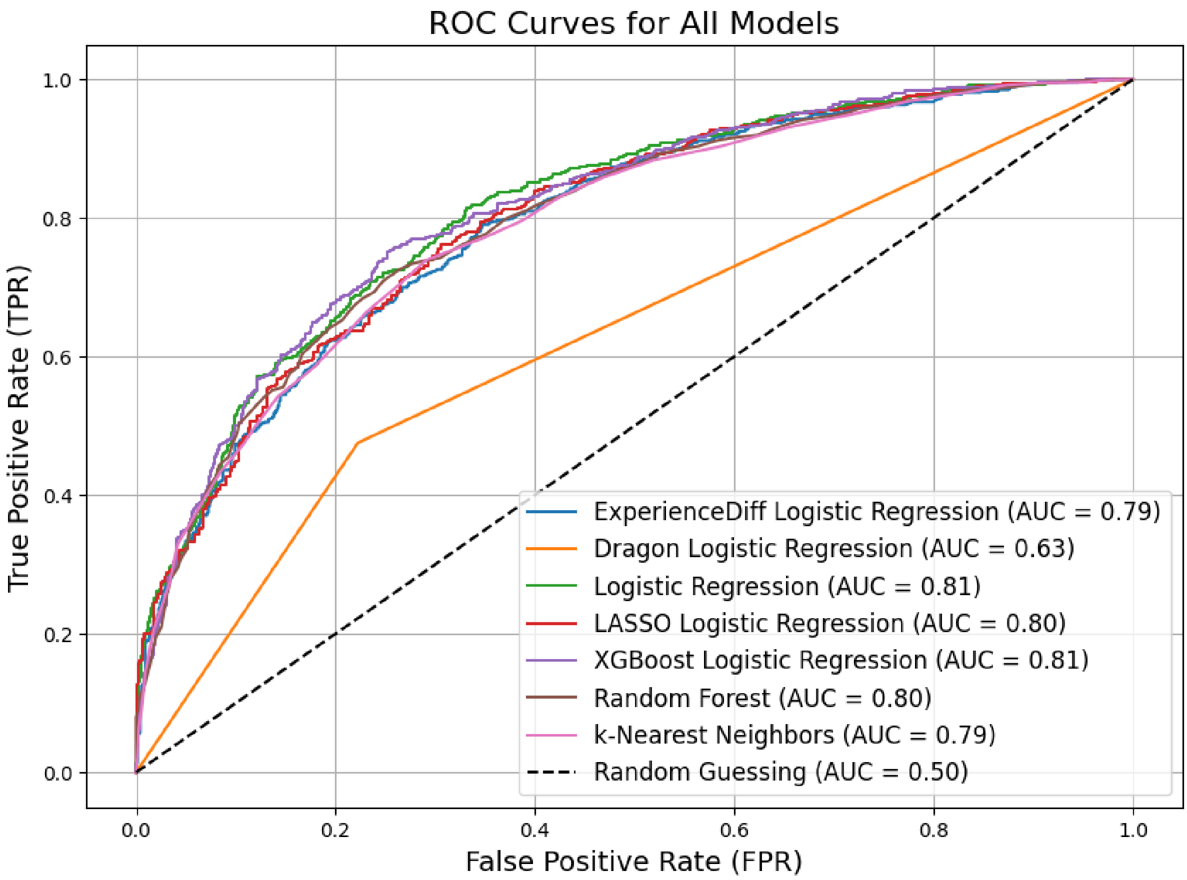

Figure 2: ROC Curve for All Models

6.4 Interpretation of Results

The models consistently highlight the importance of resource-based features (*blueGoldPerMin*, *blueExperienceDiff*) and early-game advantages (*blueFirstBlood*, *blueDragons*). These results validate the hypothesis that strategic resource management and early skirmishes are critical for predicting outcomes.

Interestingly, features like *blueVisionScore* and *blueTowersDestroyed* showed lower importance. This may be due to the dataset cutoff at 10 minutes, as both variables tend to have greater relevance in later stages of the game. While kNN demonstrated clear patterns in the data, its lack of interpretability makes it less suitable for actionable insights.

To see the variable importance of each model please see figures 10 to 13 in the appendix.

6.5 Potential Omitted Variable Bias

While the models incorporate a range of gameplay features, there are factors not included in the dataset that could influence game outcomes.

1. Champion Picks and Game Phases: The champions selected by players greatly affect performance. For example, teams with late-game scaling champions may underperform in the first 10 minutes but excel later. Without champion-specific data, the model might overemphasize early-game metrics, such as kills or gold difference, and

Rank	Model	Accuracy
1	Random Forest	0.730769
2	Logistic Regression	0.720648
3	LASSO Logistic Regression	0.720648
4	k-Nearest Neighbors	0.705466
5	ExperienceDiff Logistic Regression	0.701417
6	XGBoost Logistic Regression	0.696356
7	Dragon Logistic Regression	0.633603

Table 3: Model comparison ranked by accuracy.

misinterpret intentional scaling strategies.

2. Player Behavior and Toxicity: Player behavior, such as toxic chat usage or surrender votes, can significantly impact gameplay. For instance:

- Excessive chat usage may indicate distractions or internal team conflicts.
- Early surrender votes can lower team morale.
- Conversely, constructive communication might reflect better coordination and execution.

The omission of these social and psychological dynamics limits the model's ability to fully explain game outcomes.

—

6.6 Feature Insights

Resource management metrics (e.g., gold per minute, experience gap) and early-game events (e.g., first blood) emerged as dominant predictors of match outcomes. Vision control and late-game objectives showed limited impact within the 10-minute scope.

7 Conclusion

This project set out to predict match outcomes in League of Legends using data from the first 10 minutes of gameplay. Our analysis revealed that early-game metrics, such as *blueGoldPerMin*, *blueExperienceDiff*, *blueDragons*, and *blueFirstBlood*, play a crucial role in determining success. These metrics highlight the importance of resource management, early skirmishes, and securing objectives to gain a competitive advantage.

Linear models like Logistic Regression and LASSO performed well, confirming that the data has a low-dimensional structure. However, Random Forest offered a deeper understanding of the interactions between features, reinforcing the importance of team coordination and resource control during the early game.

At the same time, we recognize the limitations of our analysis. Factors like champion selection, player behavior, and team dynamics were not included, even though they influence match outcomes. Incorporating these elements in future research could improve the accuracy and robustness of the models.

Overall, this study shows that early-game decisions matter significantly in League of Legends. Whether it's efficiently managing resources, securing first blood, or focusing on early objectives like dragons, these strategies can greatly improve a team's chances of winning. For players and teams looking to improve, these findings emphasize the value of a strong early game in setting the stage for victory.

Appendix

Variable Definitions

- **blueWardsPlaced:** Number of wards placed by the blue team (proxy for vision control).
- **blueWardsDestroyed:** Number of red team wards destroyed by the blue team (proxy for vision denial).
- **blueFirstBlood:** Indicates which team secured the first kill in the game (early-game momentum indicator).
- **blueKills:** Total number of kills by the blue team.
- **blueDeaths** Total number of deaths of the blue team.
- **blueAssists:** Total number of assists by the blue team (high assists indicate teamwork, while low assists suggest solo kills).
- **blueEliteMonsters:** Number of elite jungle monsters slain by the blue team.
- **blueHeralds:** umber of Rift Heralds slain by the blue team. (Note: Rift Herald spawns at 9:50; securing it often reflects a significant lead).
- **blueTowersDestroyed:** Total number of towers destroyed by the blue team.
- **blueTotalGold:** Total amount of gold accumulated by the blue team.
- **blueAvgLevel:** Average level of all players on the blue team.
- **blueTotalExperience:** Total experience points gained by the blue team.
- **blueTotalMinionsKilled:** Total number of lane minions killed by the blue team.
- **blueTotalJungleMinionsKilled:** WTotal number of jungle minions killed by the blue team (separate from elite monsters).
- **blueGoldDiff:** Difference in total gold between the blue and red teams (BlueTeamGold - RedTeamGold).
- **blueExperienceDiff:** Difference in total experience points between the blue and red teams (BlueTeamExp - RedTeamExp).
- **blueCSPerMin:** Creep Score (CS) per minute for the entire blue team (proxy for farming efficiency; 10 CS/min per player is an optimal benchmark).
- **blueGoldPerMin:** Gold earned per minute by the blue team.
- **blueFirstBlood:** Whether the blue team secured the first kill.
- **blueFirstBlood:** Whether the blue team secured the first kill.

Created Variables

- **blueKDA:** blueKills, blueDeaths, and blueAssists were combined into a single metric, blueKDA, which provides a more holistic measure of team performance in skirmishes.
- **blueVisionscore:** blueWardsPlaced and blueWardsDestroyed were aggregated into a blueVisionScore to represent the team's overall vision control, combining placed vision and denied enemy vision.
- **blueTotalJungleUnits:** blueEliteMonsters, which counts both Dragons and Heralds, was removed. Instead, we combined blueDragons with blueTotalJungleMinionsKilled to create a new feature, blueTotalJungleUnits, which represents all jungle units killed, excluding the Rift Herald. The Herald remains as a separate variable due to its unique strategic importance.

Figures

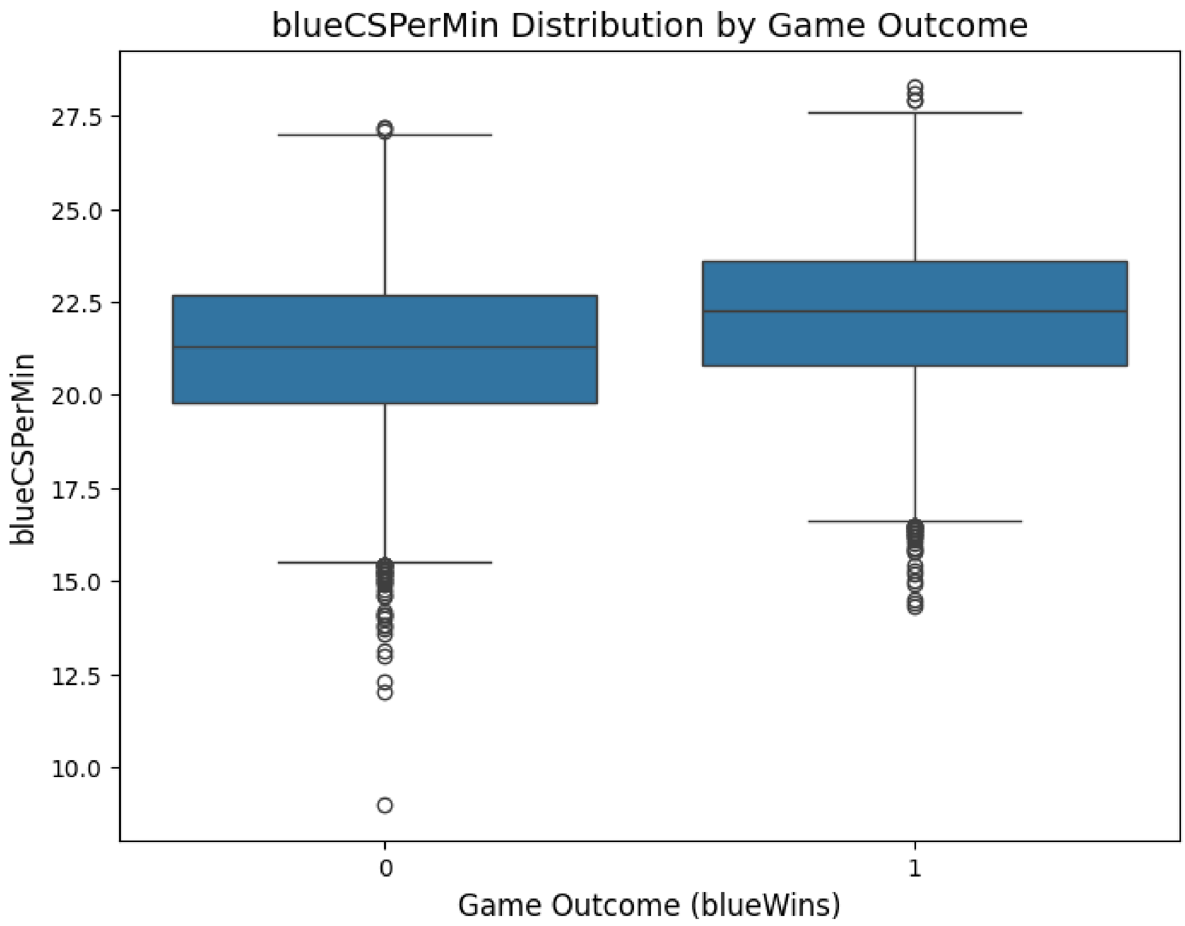

Figure 3: Box Plot CSPerMin

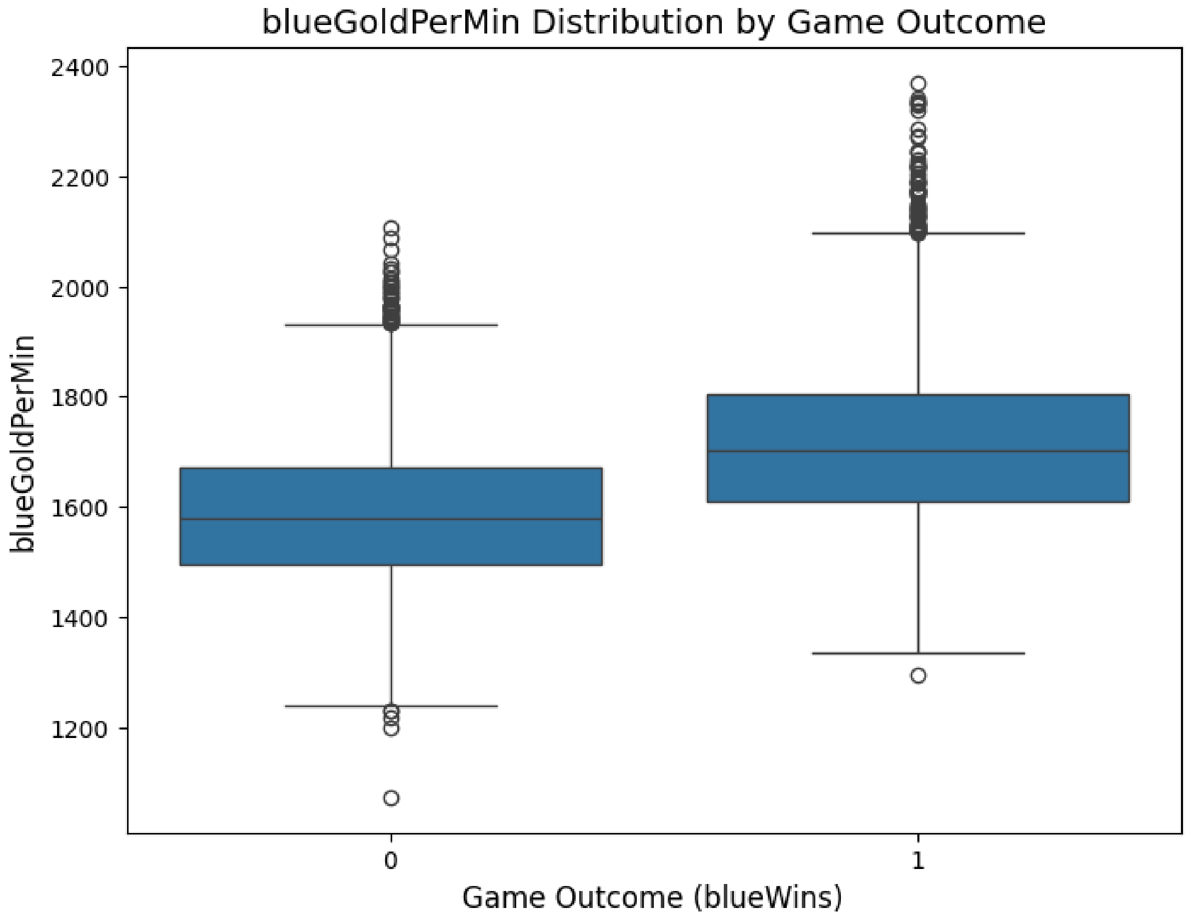

Figure 4: Box Plot GoldPerMin

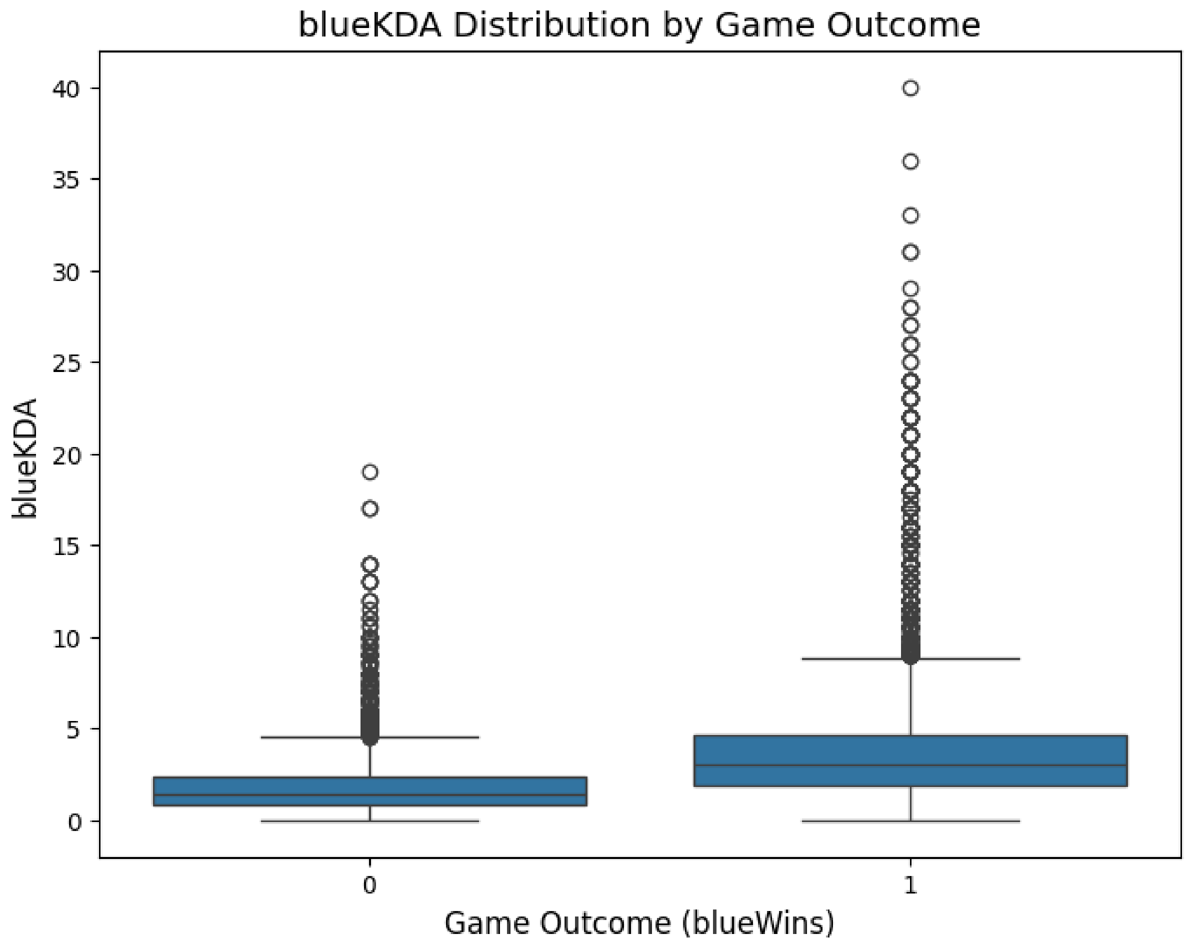

Figure 5: Box Plot KDA

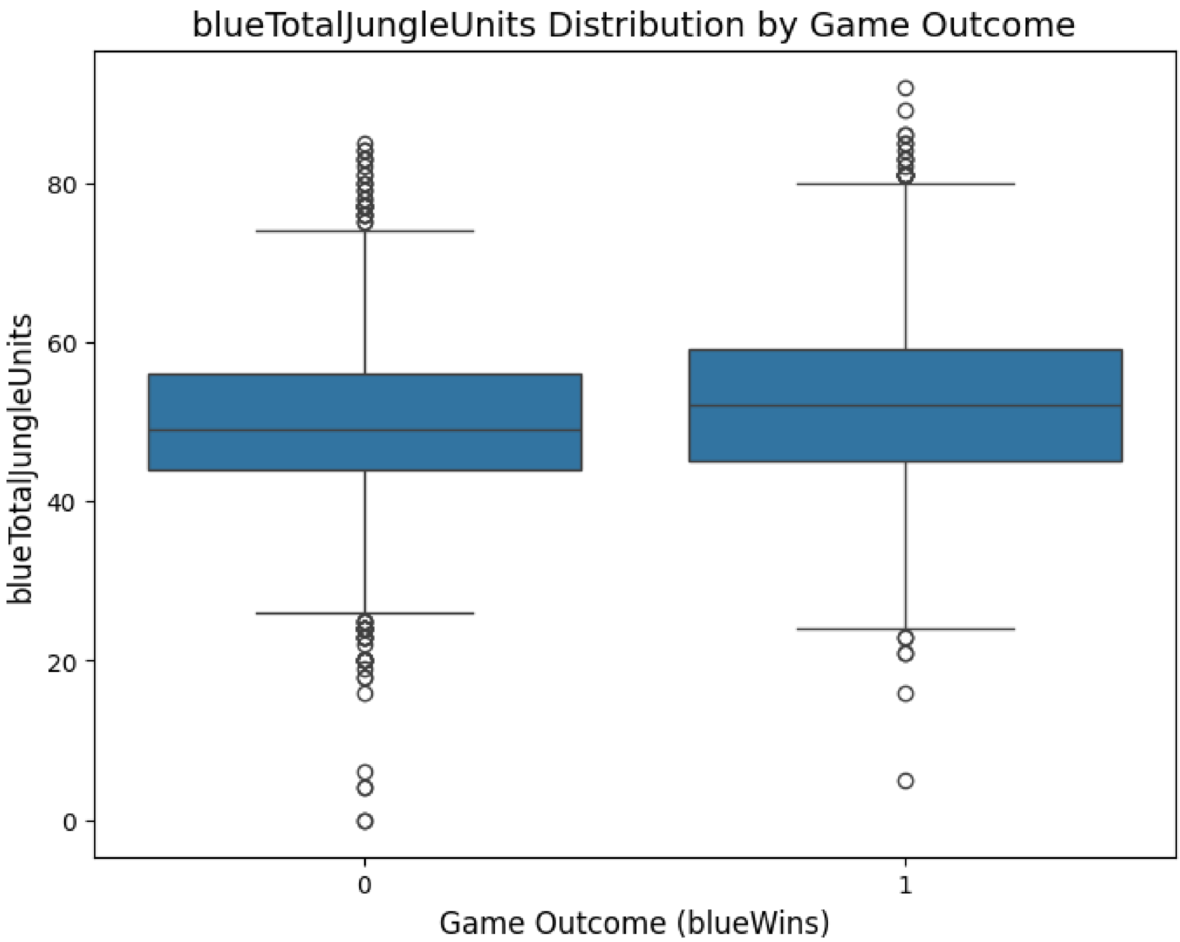

Figure 6: Box Plot TotalJungleUnits

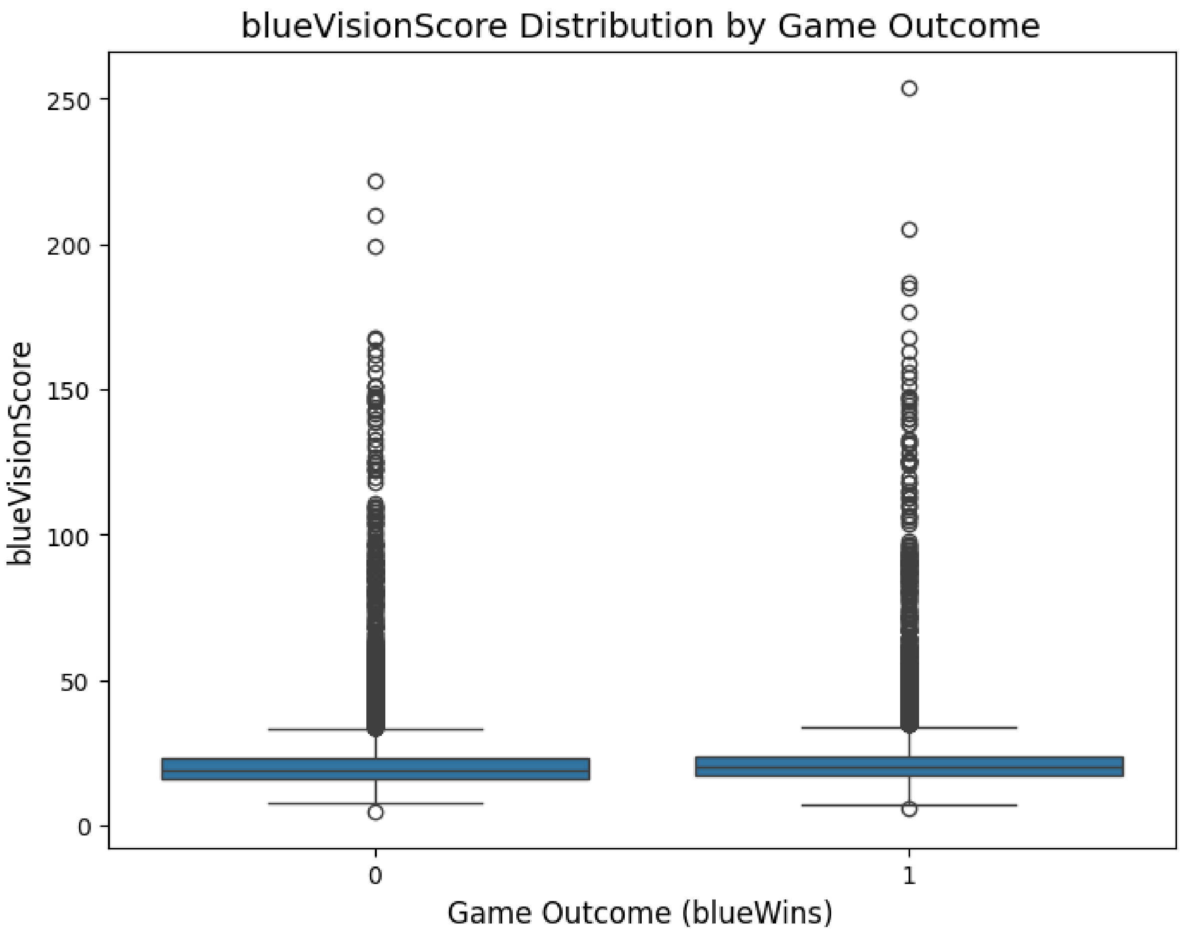

Figure 7: Box Plot Visionscore

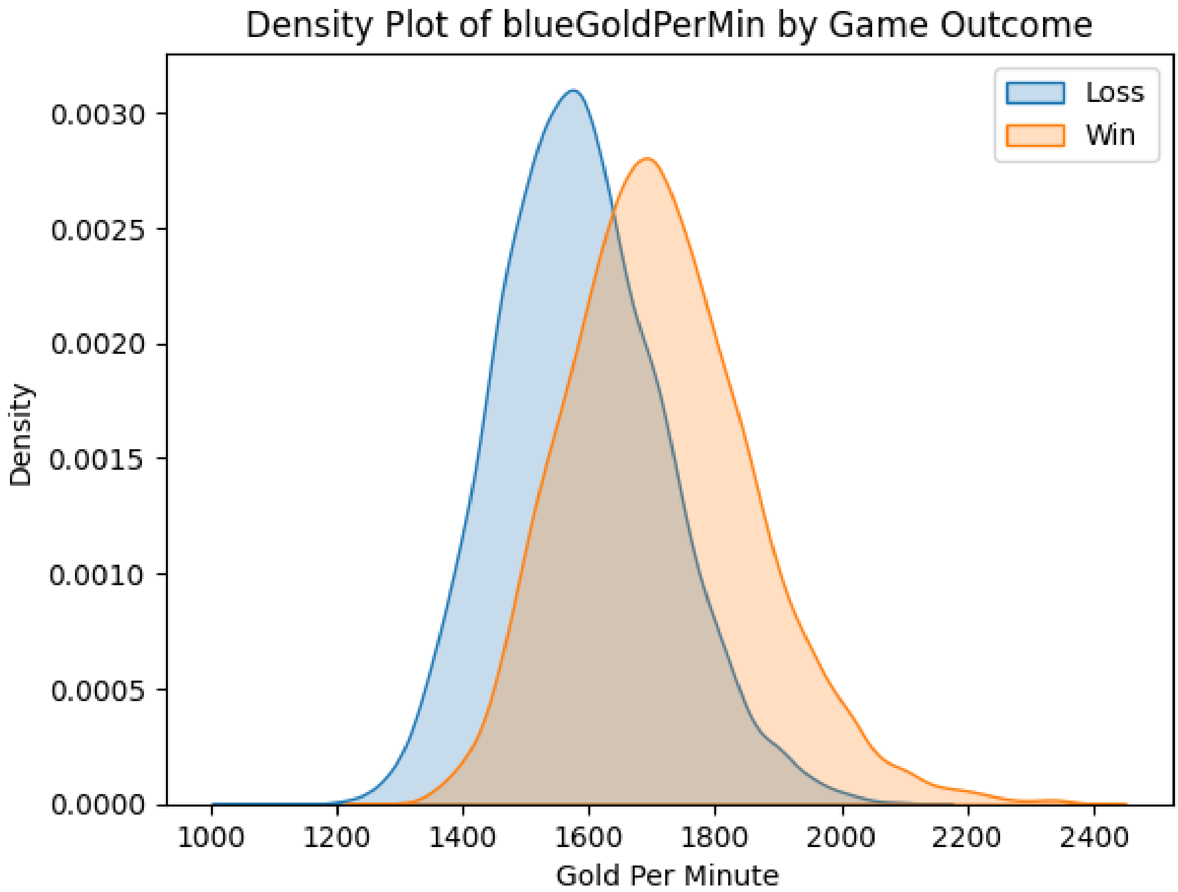

Figure 8: Density Plot GoldPerMin

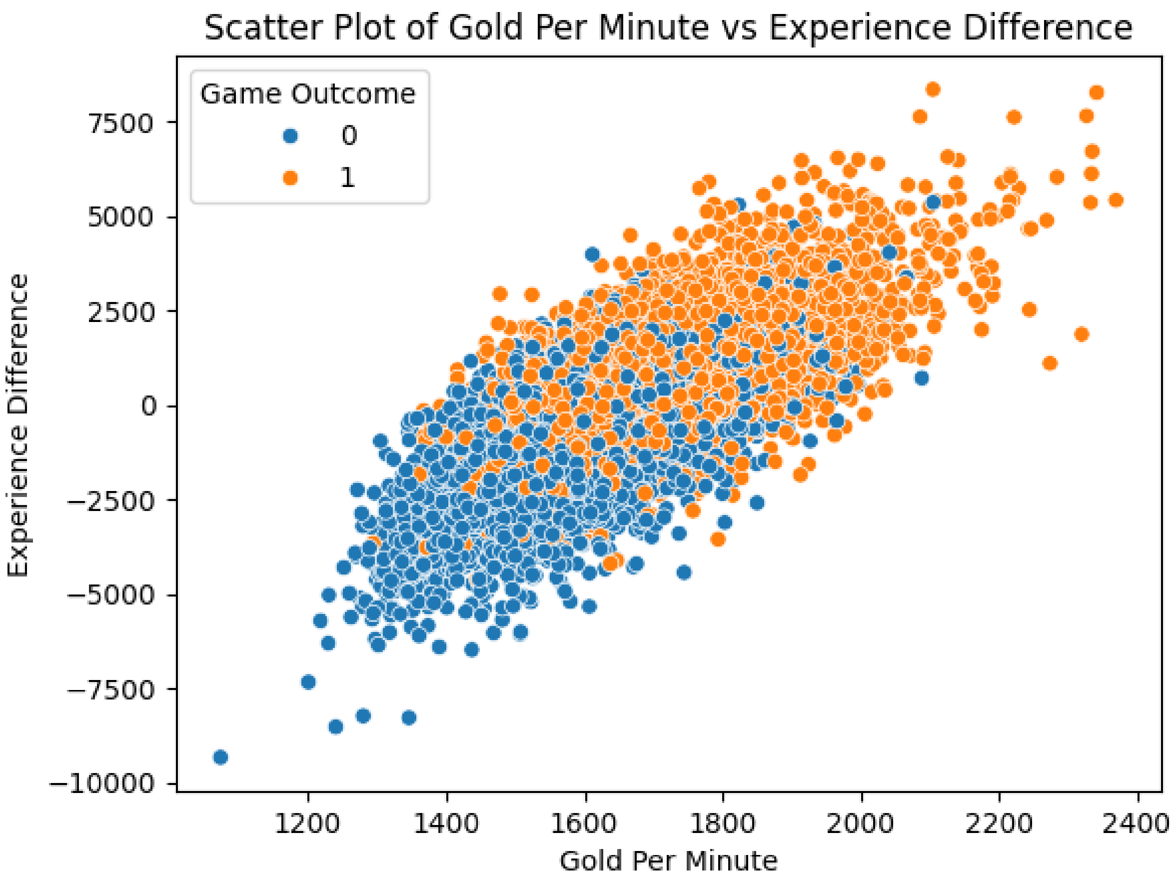

Figure 9: Scatter Plot GoldPerMin

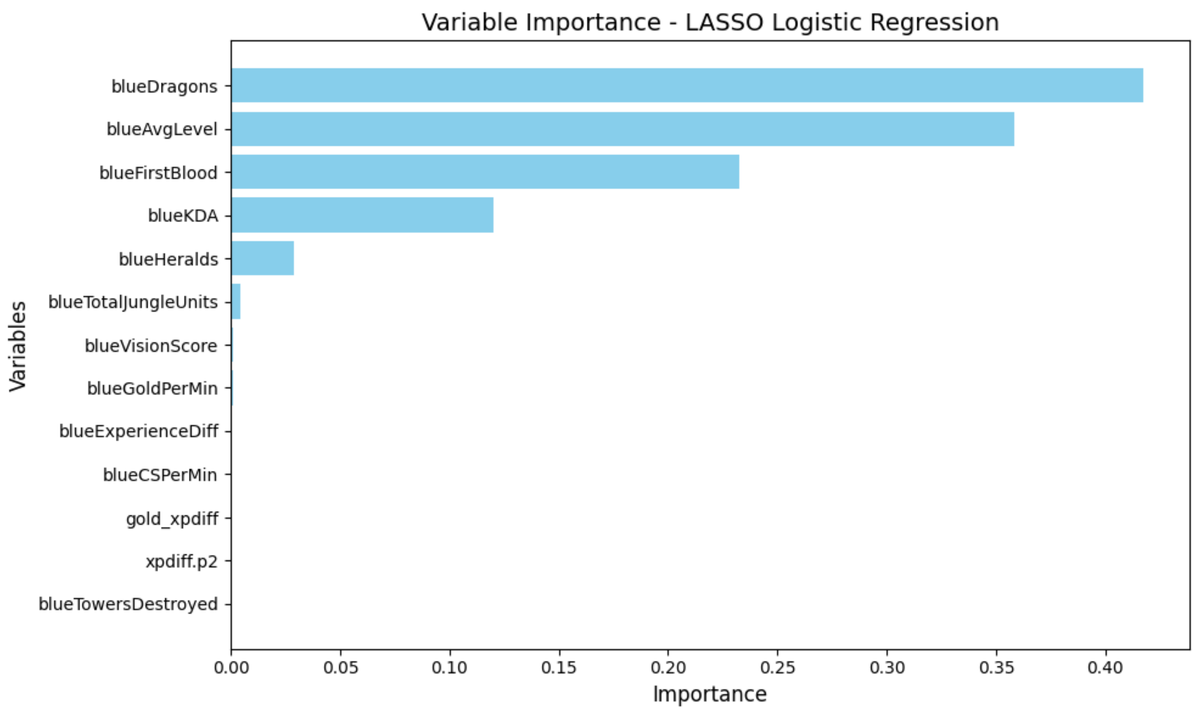

Figure 10: Variable Importance LASSO

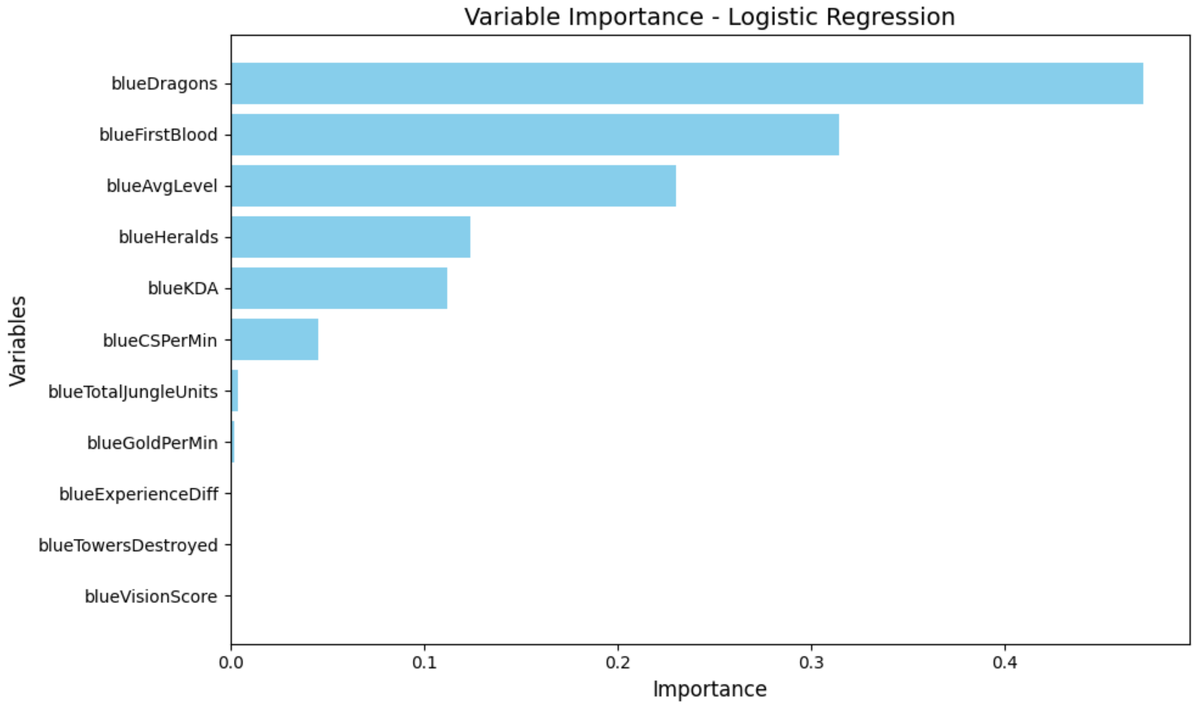

Figure 11: Variable Importance Logistic Regression

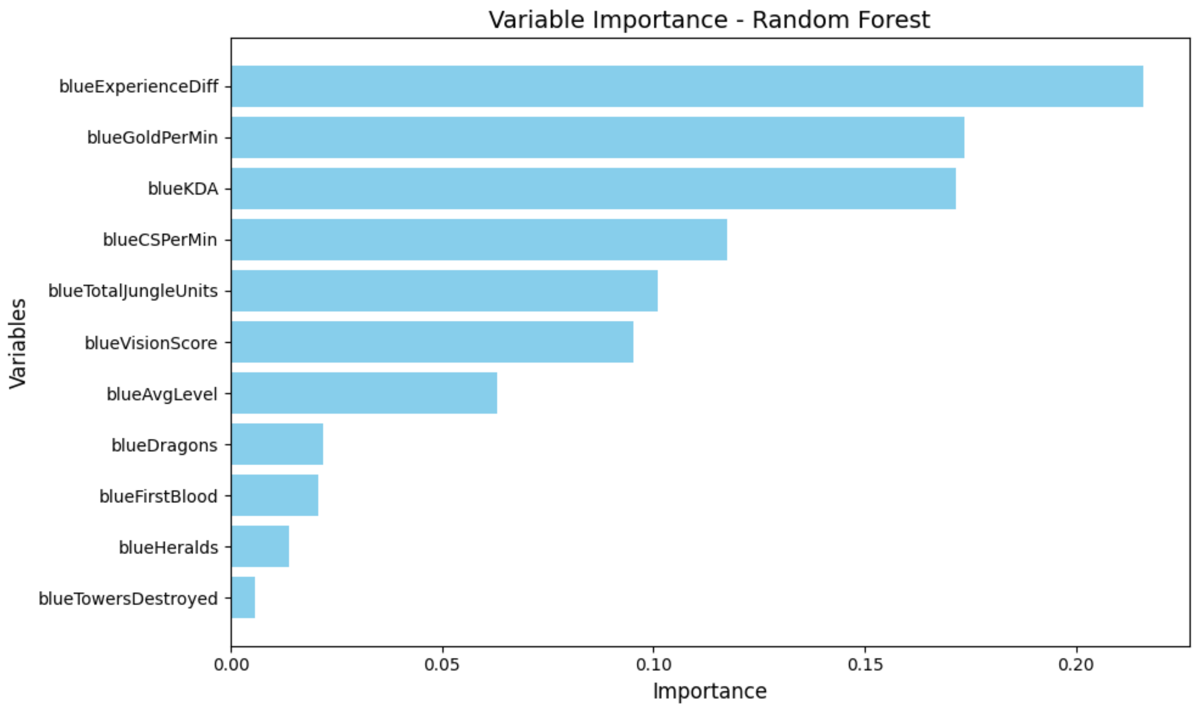

Figure 12: Variable Importance Random Forest

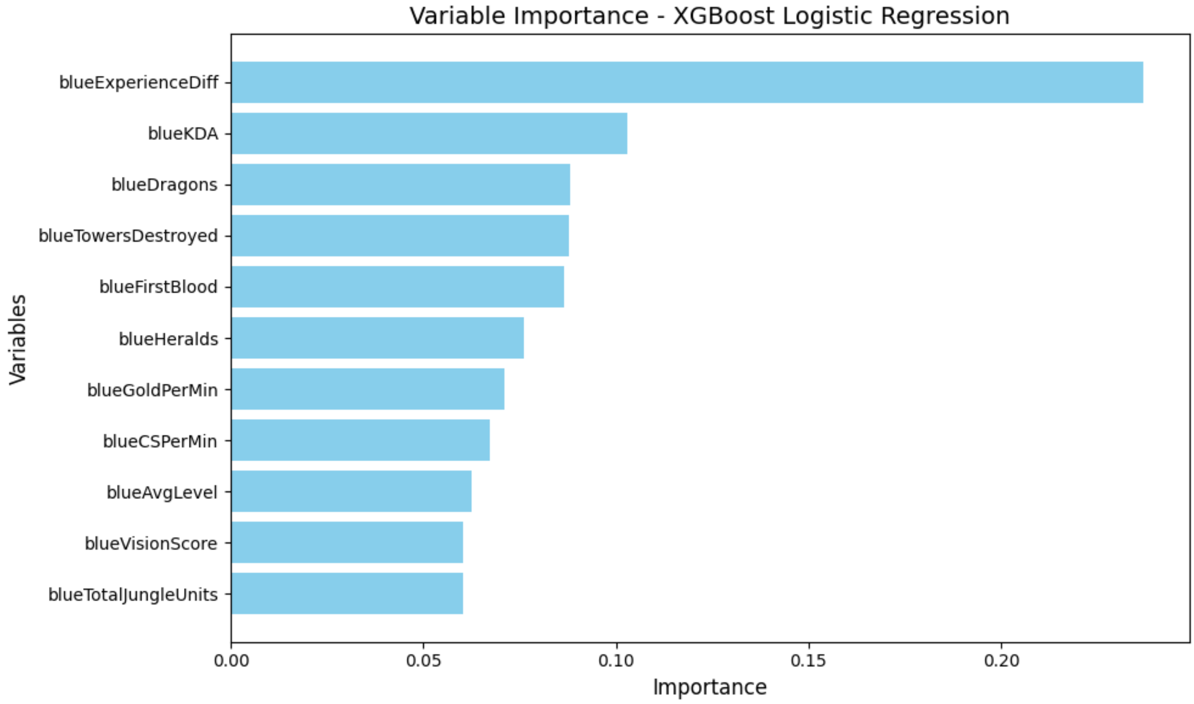

Figure 13: Variable Importance XGBoost Logistic Regression