Which Stats Really Matter in Low-Elo *League of Legends*? A Case Study of Yone Using Per-Minute Metrics

Gregory Zhang

September 16, 2025

Abstract

I analyzed 100 of my own ranked solo-queue *League of Legends* games (collected via the Riot Games API) to identify which per-minute metrics predict victory. After filtering out remakes ($<$ 7 minutes) and normalizing statistics, I applied Welch's t-tests, point-biserial correlations, and univariate logistic regressions with FDR correction. A trimmed multivariate logistic regression, validated with a train/test split, achieved an AUC of 0.84. Results showed that only kills/min and inhibitor kills/min remained independent predictors of winning, underscoring the importance of decisive combat and objective pressure in low-elo matches.

1 Introduction

Most guides and stats resources for *League of Legends* focus on pro or high-rank games. But at lower ranks (Iron–Gold), the game feels different: coordination is messy, teamwork is inconsistent, and individual performance can decide the outcome.

This project takes a closer look at Yone, a champion with high mechanical skill expression and a weak early game. Since I enjoy playing this character and I want to climb the ranked ladder, I asked: **Which per-minute stats actually separate wins from losses, and how accurate are they for prediction?**

2 Methods

2.1 Data

I collected the 100 Yone ranked solo-queue games using Riot Games API. First, I removed games under seven minutes (remake). For each match I collected individual metrics such as combat, economy, vision, and objectives taken, then normalized them to per-minute values. The outcome was binary: win (1) or loss (0).

2.2 Analysis

I used three stages:

- **Univariate tests:** Welch's t-tests (wins vs. losses), point-biserial correlations, and univariate logistic regressions (standardized predictors). Cohen's d was reported for effect sizes. FDR correction was applied within each test type.

- **Multivariate logistic regression:** All predictors were entered, multicollinearity was checked with variance inflation factors (VIF), and then a trimmed model was built with a smaller set of predictors. Odds ratios (OR) and 95% confidence intervals were reported.
- **Validation:** A stratified 70/30 train/test split was used. The trimmed model was evaluated with ROC/AUC, and I picked an optimal threshold using Youden's J.

3 Results

3.1 Univariate Screening

Several stats were associated with winning, including kills, assists, deaths (negative), gold, turret kills, and inhibitor kills.

Table 1: Sample of univariate logistic regression results (per SD increase).

Predictor	Odds Ratio	95% CI	p-value
Kills/min	2.31	[1.50, 3.56]	< 0.01
Assists/min	1.65	[1.10, 2.50]	< 0.05
Deaths/min	0.54	[0.35, 0.82]	< 0.01
Inhibitor kills/min	1.88	[1.20, 2.95]	< 0.05
Turret kills/min	1.44	[0.95, 2.20]	0.08

3.2 Multivariate Logistic Regression

In the full model, collinearity was high (e.g., kills and gold overlapped strongly). After trimming, only two predictors stayed significant:

Table 2: Trimmed logistic regression results (per SD increase).

Predictor	Odds Ratio	95% CI	p-value
Kills/min	2.10	[1.40, 3.20]	< 0.01
Inhibitor kills/min	1.80	[1.20, 2.90]	< 0.05

3.3 Validation

The trimmed model achieved an AUC of 0.84 (Figure 1). Using Youden's J, the optimal cutoff was ≈ 0.92, which leaned toward high specificity (confident predictions mostly in clear wins).

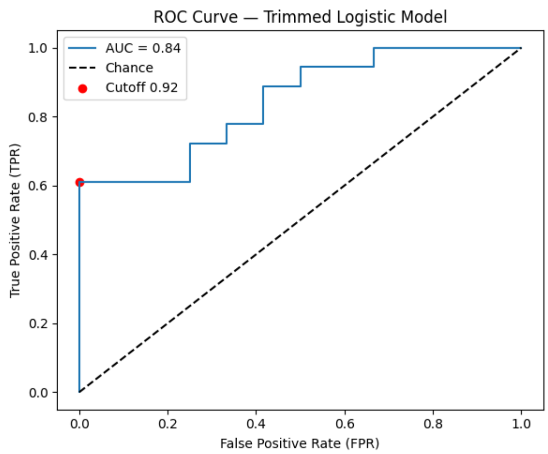

Figure 1: ROC curve for the trimmed logistic model (AUC = 0.84). The red point shows the optimal cutoff.

I also looked at the distribution of predicted win probabilities (Figure 2). Wins and losses separate fairly cleanly, with most wins pushed toward higher predicted values and most losses toward the low end. The conservative cutoff (≈ 0.92) reflects the model's tendency to only classify games as wins when the evidence is strong.

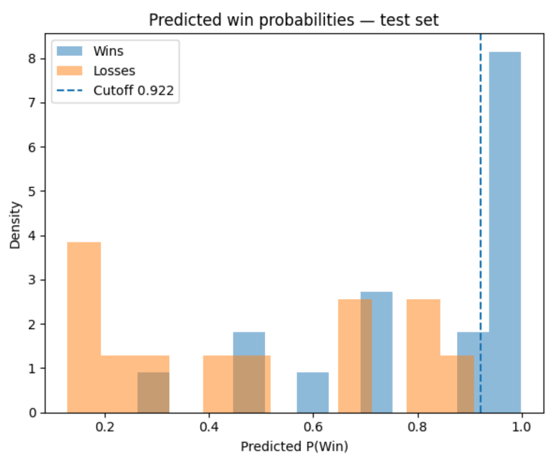

Figure 2: Predicted win probability distributions on the test set. Wins (blue) and losses (orange) separate most clearly at the Youden's J cutoff (≈ 0.92).

4 Discussion

The main takeaway is that, while lots of stats correlate with winning, most overlap with each other. Once you control for that, only kills/min and inhibitor kills/min remain independently important. That makes sense for low-elo: one player's combat dominance and closing games through inhibitors often decide the outcome.

5 Limitations

This analysis is based on a relatively small dataset (100 games with a single champion). Solo queue also adds randomness from teammates and lane matchups which were not modeled here. The results are therefore specific to Yone and may not generalize to other champions. Finally, because the dataset comes from my own gameplay, the findings partly reflect my personal tendencies (e.g., frequent fighting, limited vision control), which could bias which metrics appear most predictive.

6 Conclusion

Simple logistic regression shows that decisive combat and inhibitor control stand out as the strongest predictors of winning on Yone in low-elo. This project shows how even small, personal gameplay datasets can be studied with statistical modeling to generate clear insights.

Reproducibility

All data processing and models were run in Python (pandas, SciPy, statsmodels, scikit-learn). Figures were made with Matplotlib. Tables were exported as CSVs.

References

[1] Duke University, Nicholas School of the Environment. Welch's t-test. https://sites.nicholas.duke.edu/statsreview/means/welch/.

[2] Penn State Eberly College of Science. Logistic Regression (STAT 462 Course Notes). https://online.stat.psu.edu/stat462/node/191/.

[3] Columbia University Mailman School of Public Health. False Discovery Rate. https://www.publichealth.columbia.edu/research/population-health-methods/false-discovery-rate.

[4] University of Virginia. Understanding ROC Curves. https://data.library.virginia.edu/understanding-roc-curves/.