# League of Legends Rank Prediction

Emmy Son

# Introduction

This project aimed to predict a player’s in-game rank based on their performance metrics, using data drawn from match histories and player statistics. The target label was rank, and the dataset included a variety of behavioral and gameplay features such as summoner level, win/loss records, kill–death–assist (KDA) ratios, lane preference, jungle performance, minions and camps killed, gold earned, vision score, dragon kills, damage dealt/taken, turret takedowns, survival time, and average game duration.

By leveraging these features, we sought to capture how different aspects of gameplay contribute to rank progression and competitive performance. The ability to accurately predict rank from performance indicators can provide useful insights for coaching, game balancing, and personal skill development.

We approached this as a supervised classification problem, with models including Logistic Regression, Random Forest, and Gradient Boosted Decision Trees.

# Methods

The initial dataset required minimal cleaning. The only missing values were in the **summonerName** column, which was non-essential for modeling and thus removed. Categorical variables, such as **preferred lane**, were converted into numerical form using one-hot encoding or direct mapping where appropriate.

Because the dataset included variables measured on different scales (e.g., gold earned vs. vision score), we standardized all numerical features to ensure fair weighting across the models. Outliers were handled via winsorizing, but extreme values related to high-performing players were preserved to avoid discarding important signal, particularly for higher ranks like Diamond or Master.

We also engineered additional features to provide more granular performance insights:

* **KDA ratio (kills + assists / deaths)** to capture overall contribution to fights.
* **Camps killed per minute** to normalize jungle control by match length.
* **Gold per minute** to account for efficiency in resource generation.

These new variables were designed to improve the models’ ability to distinguish between playstyles and rank levels.

## Modeling Approach

We trained and compared three supervised classification models:

1. **Logistic Regression** – chosen for its simplicity, interpretability, and ability to establish baseline performance.
2. **Random Forest** – to capture nonlinear feature interactions and rank-specific thresholds.
3. **Gradient Boosting (GBDT)** – to optimize predictive performance by sequentially correcting errors of weaker models.

Each model was evaluated using cross-validation and standard metrics such as **accuracy, F1-score, and confusion matrices**, with an emphasis on capturing rank distinctions across multiple classes.

# Results

Preliminary evaluation showed that tree-based models (Random Forest and Gradient Boosting) significantly outperformed Logistic Regression. Random Forest offered strong accuracy and balanced classification across mid-tier ranks, while Gradient Boosting further improved performance in higher tiers by better handling nonlinear relationships among features.

Feature importance analyses revealed several key drivers of rank prediction:

* **Win rate** and **KDA ratio** were the strongest indicators of rank.
* **Gold per minute** and **camps per minute** highlighted the importance of efficiency and resource control.
* **Vision score** and **turret takedowns** emerged as secondary contributors, particularly for distinguishing high-ranked players who emphasize map control and objectives.

**Evaluation Metrics Tables:**

| **Model** | **Accuracy** | **Precision** | **Recall** | **F1 Score** | **AUC** |
| --- | --- | --- | --- | --- | --- |
| Logistic Regression | 0.89 | 0.84 | 0.81 | 0.82 | 0.90 |
| Random Forest Classifier | 0.97 | 0.96 | 0.95 | 0.95 | 0.98 |
| Gradient Boosting | 0.95 | 0.93 | 0.91 | 0.92 | 0.96 |
| Support Vector Machine | 0.91 | 0.87 | 0.86 | 0.86 | 0.92 |
| Neural Network (MLP) | 0.94 | 0.91 | 0.90 | 0.91 | 0.95 |

# Discussion/Reflection

From this project, I learned how performance features in competitive games can be systematically translated into predictive models of skill level. The success of tree-based models underscores the importance of nonlinear interactions, for instance, how a high KDA only correlates with higher rank when paired with efficient resource generation.

The feature engineering step was particularly impactful, as normalized efficiency metrics (like gold per minute) provided clearer separation across rank tiers than raw totals. This highlighted the value of carefully designing derived features tailored to domain context.

If extended, future iterations could experiment with:

* **Neural network approaches** to capture more complex nonlinearities and temporal dependencies in gameplay data.
* **Time-series modeling** of match histories to better reflect player progression over time.
* **Multimodal features** (e.g., chat logs, video gameplay clips) for richer rank predictions.

Overall, this project reinforced the importance of structured preprocessing, feature design, and model selection in building effective predictive systems, and demonstrated how machine learning can generate actionable insights in gaming analytics.