Project Report : STAT 303-3 Spring 2022

Aarushi Seth
aarushiseth2022@u.northwestern.edu

Vedant Ambani
vedantambani2023@u.northwestern.edu

Anushka Chaturvedi
anushkachaturvedi2024@u.northwestern.edu

Ahan Sahu
ahansahu2022@u.northwestern.edu

Abstract

League of Legends is an extremely popular team-based multiplayer online battle arena (MOBA) game where two teams, blue and red, of five powerful champions battle to destroy the other's base. As a part of the game, teams make plays, secure kills, and take down enemy towers to move towards victory. Due to the learning curve to the game and its strategic nature, many players struggle to decide the most suitable strategy to implement to win at League of Legends. For example, should a team focus on pursuing objectives such as turrets, killing opponent players or killing monsters such as dragons. In order to tackle this problem, our project focuses on determining the ideal style of team play and decision making during the first 10 minutes of a League of Legends game that will maximize the chances of victory. We hope to suggest the ideal early surface-level strategies that teams should focus upon to increase their chances of eventually winning the game. Our project is a classification problem as we are attempting to classify different strategies as winning or losing strategies.

1. Background / Motivation / Problem statement / Data sources

For our project, we are using a Kaggle data source on League of Legends that can be found at https://www.kaggle.com/datasets/bobbyscience/league-of-legends-diamond-ranked- games-10-min. This dataset contains the statistics for the first 10 minutes of 9879 League of Legends games. There are 19 features collected per team including kills, deaths, gold, experience and level that could be predictors in our model. Additionally, the blueWins column is the target variable, as a value of 1 means the blue team has won, while a value of 0 means a win for the red team. We will use these features as well as the given target variable data for which team eventually won the game to conduct our analysis.

The primary stakeholders of our project are active players. Through this model, we hope to advise them on how to improve their game and how to use these strategies to maximize their chances at winning. Therefore, we believe that they would benefit the most from this analysis.

Professional League of Legends betters are secondary stakeholders as they could use this information to decide which team to bet on based on their strategies. Riot Games, the developer, is also a secondary stakeholder as through our findings and model they can prevent the dominance of a single or few strategies. They could also use this information to develop new features to keep the players invested in the game.

2. Metric of Interest

To evaluate the goodness of our model, we will be relying on accuracy and ROC-AUC metrics. The ROC curve plots TPR (True positive rate) or recall against FPR (False positive rate), for all possible thresholds. We have a balanced dataset and given the nature of our problem, we prioritize both positive and negative values. As a result, both accuracy and ROC-AUC are well-suited for our problem. As we do not have to worry about large costs or hazards of giving a false negative or positive, we do not consider metrics such as false negative rate (FNR), precision or recall. Additionally, ROC-AUC is also scale-invariant and classification-threshold-invariant. ROC-AUC therefore does not rely on a single threshold probability, adding further robustness to the goodness of our model.

Our desired values for accuracy score and ROC-AUC would probably be around 70-80% since we are focusing on the first 10 minutes of the game and there can be plays made after that point that would lead to a change in the momentum of the game etc. An accuracy score 70-80% is still useful since it gives us a good idea of what may give a team an advantage in the early phases of the game.

2. Approach

As we attempted to build an accurate and powerful classification model to help our stakeholders ascertain the ideal type of team play and decision making during the first 10 minutes of a league of legends game, we used a stacking ensemble model as we had seen in class.

The first step involved cleaning the League of Legends dataset. Initially, the dataset contained 38 predictors. However, since some of them either interconnected or unimportant we shortlisted 8 predictors that we would run our machine learning models with.

We used a number of Level 0 models such as the Random Forest Classifier, Naive Bayes Classifier, Decision Tree Classifier, XG Boost Classifier and the SVM Classifier. We also used logistic regression as a Level 1 model for our purpose.

The first level 0 model that we investigated was the

Random Forest Classifier. We tuned it for max_features and n_estimators which had a 5-fold cross validation accuracy of 71.9% and 5-fold cross validation ROC-AUC score of 79.5%. The next level 0 model was the Naive Bayes Classifier. We tuned it for var_smoothing which had a 5-fold cross validation accuracy of 72.4% and 5-fold cross validation ROC-AUC score of 80.3%. The Decision Tree Classifier was tuned for max_depth, max_features, and max_leaf_nodes which had a 5-fold cross validation accuracy of 72.2% and 5-fold cross validation ROC-AUC score of 79.9%.

3. Developing the model(s)

We began developing the model by first cleaning the data. Initially the dataset had over 38 predictors, but after analyzing them, we realized that some were interconnected or unrelated. Therefore through exploratory data analysis, we were able to narrow it down to 8 predictors to run our machine learning models with. After data cleaning, we decided to explore the stacking ensemble model and considering the 5-fold accuracy and 5-fold ROC-AUC score to verify the success of our model. The first level 0 model we considered was Random Forest Classifier, tuning for max_features and n_estimators. Here, the 5-fold accuracy was 71.9% and the 5-fold ROC-AUC score 79.5%. The second level 0 model we considered was Naive Bayes Classifier, tuning for var_smoothing. Here, the 5-fold accuracy was 72.4% and the 5-fold ROC-AUC score 80.3%. The third level 0 model we considered was Decision Tree Classifier, tuning for max_depth, max_features and max_leaf_nodes. Here, the 5-fold accuracy was 72.2% and the 5-fold ROC-AUC score 79.9%. The fourth level 0 model we considered was XGBoost Classifier, tuning for n_estimators, max_depth, learning_rate, gamma and reg_lambda. Since our dataset was balanced, scale_pos_weight was chosen to be 1. Here, the 5-fold accuracy was 72.4% and the 5-fold ROC-AUC score 81.0%. The fifth level 0 model we considered was SVM Classifier, tuning for C, Gamma and Kernel. Here, the 5-fold accuracy was 73.1% and the 5-fold ROC-AUC score 80.7%. With our five base models ready, we began running the stacking ensemble model. To further increase the model accuracy, we implemented logistic regression as it was providing higher values in accuracy and ROC-AUC. This model increased the 5-fold accuracy was 74.0% and the 5-fold ROC-AUC score 80.8%. The final stacking ensemble model provided us with the highest combination of accuracy and ROC-AUC score.

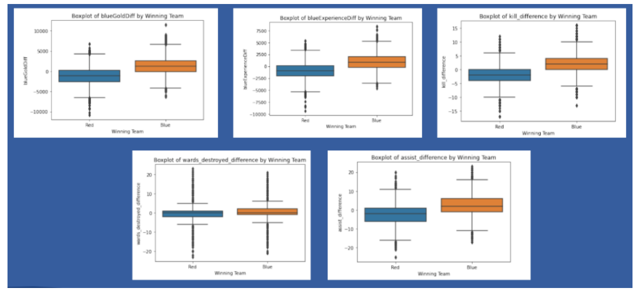

Figure 1: Data Cleaning

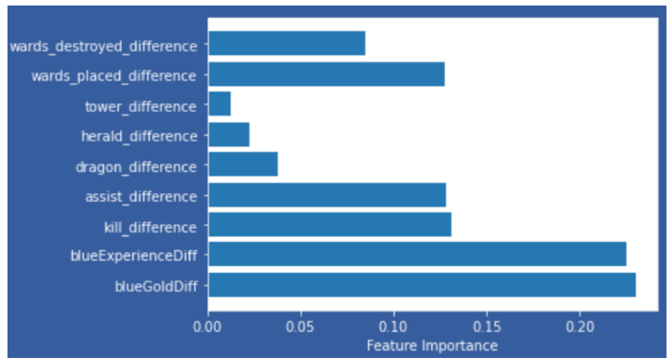

Figure 2: L0 Model 1

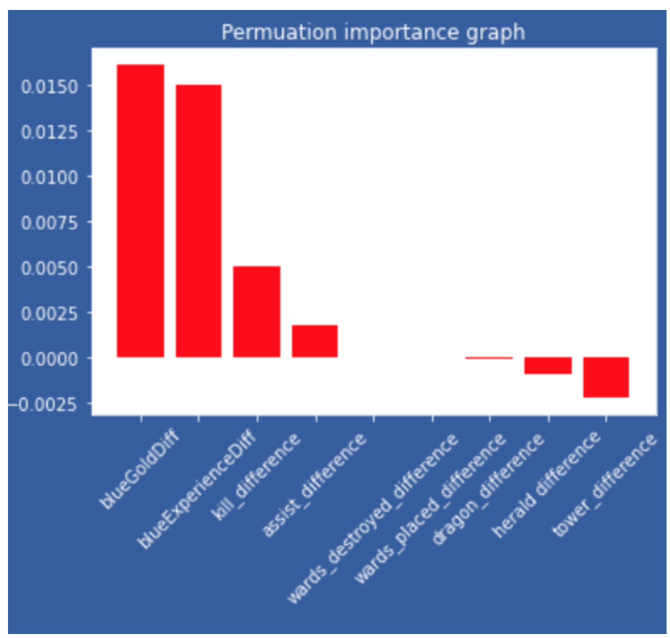

Figure 3: L0 Model 2

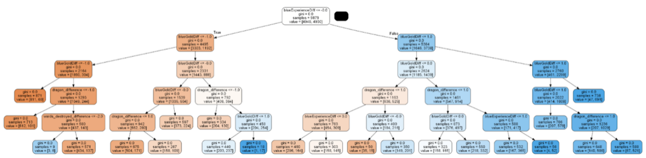

Figure 4: L0 Model 3

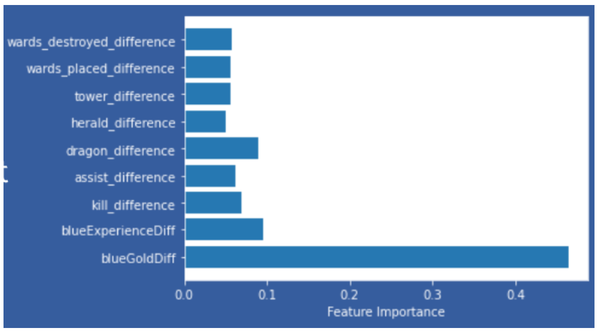

Figure 5: L0 Model 4

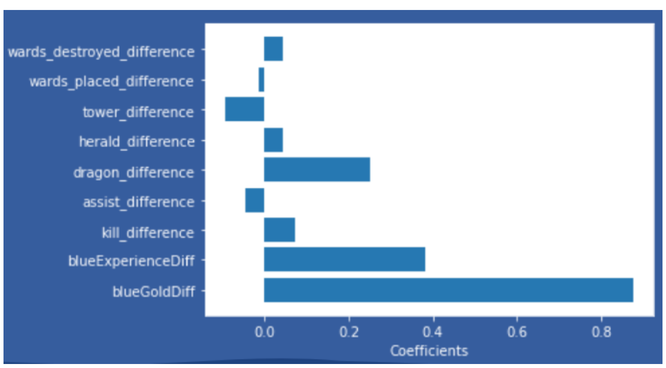

Figure 6: L0 Model 5

4. Conclusions and recommendations to the stakeholders

All of our models agree on the gold difference between the two teams within the first 10 minutes being the leading feature when considering whether the team wins the game or not. Surprisingly, the models do not give as much importance to gaining control of objectives such as dragons, heralds or towers. Destroying and placing wards also do not contribute too significantly towards the eventual result of the game. From our models, we are able to interpret that teams should be prioritizing gaining gold advantages over the other team, even if that results in the loss of objectives. Going out of your way to improve your team's vision or worsen the opponents (through wards) may not be worth it if it results in a gold deficit for you and your team.

For players, an important takeaway from our project is to prioritize gaining a gold advantage. For example, teams may want to give up objectives early in the game such as dragons, heralds and towers if it allows them to gain a lead in the gold they have accumulated. Moreover, during the laning phase of the game (early in the game), players should employ tactics that lead to loss of gold for the opponent players, such as freezing the wave to not allow the opponent to get gold as easily or shoving the wave if their opponent is not in lane. For bettors, they may want to use the insights our model provides to choose who to bet on - they may want to bet on teams that employ the strategies mentioned above. For coaches, they may want to use the insights from our model to create strategies that emphasize gaining a gold lead in the early game (even if it may mean sacrificing another aspect of the game, such as objectives).

5. Work Division

Student Name	Constributed Aspects	Details
Ahan Sahu	Model Building, Assumptions	Wrote code for the various models and addressed model assumptions.
Arushi Seth	Variable Selection, Interactions	Performed variable selection on an exhaustive list of predictors and identify relevant interactions
Anushka Chaturvedi	Data Cleaning	Performed data cleaning to ensure that dataset was usable
Vedant Ambani	Exploratory Data Analysis	Performed exploratory data analysis and visualizations to determine correlations between predictors.

References

"Ensemble Modeling." Ensemble Modeling - an overview | ScienceDirect Topics. Accessed June 7, 2022. https://www.sciencedirect.com/topics/computer-science/ensemble-modeling.

Kurama, Vihar. "A Guide to Understanding AdaBoost." Paperspace Blog. Paperspace Blog, April 9, 2021. https://blog.paperspace.com/adaboost-optimizer/#:~:text=AdaBoost%20is%20an%20ensemble%20learning,turn%20them%20into%20strong%20ones.

Morde, Vishal. "XGBoost Algorithm: Long May She Reign!" Medium. Towards Data Science, April 8, 2019. https://towardsdatascience.com/https-medium-com-vishalmorde-xgboost-algorithm-long-she-may-rein-edd9f99be63d.

Necati Demir, Ph.D. "Ensemble Methods: Elegant Techniques to Produce Improved Machine Learning Results." Toptal Engineering Blog. Toptal, February 4, 2016. https://www.toptal.com/machine-learning/ensemble-methods-machine-learning.

"Stacking Models for Improved Predictions." KDnuggets. Accessed June 7, 2022. https://www.kdnuggets.com/2017/02/stacking-models-imropved-predictions.html.