Data 1030 Midterm Project Report

Yunxuan Zeng

10/13/2020

Introduction

a. League of Legends

League of Legends (LoL) is a popular multiplayer online battle arena video game for its high competitivity and balance, which are developed and published by Riot Games. It has become one of the most played games around the world with a total of 115 million monthly players. With more than 140 champions, each player is allowed to choose one of them and group with the other four people as a team to battle against the enemy team. To attain victory in the game, the team has to find a way to destroy the enemy's base which is called Nexus.

b. Why Interesting

League of Legends has a large diversity of champions, players are able to find the perfect match to master one of them using their own unique playstyle. Ranked is LoL's most competitive mode where players could keep advancing up the ranked ladder and get ranked reward by continuously winning games. Based on the matching algorithm, players are competed with those with the same level of skills. Thus, a god team composition and early development in the game play an significant role on winning in the ranked mode.

c. Goals

There are two main objectives for this project:

1. How accurately could I predict the outcome of the ranked game given the composition of two teams in the early stage of the game?
2. How does each feature plays a role in the outcome of the game? What features are the most important ones in the first 10-minute ranked games?

d. Data Collection

The dataset has been acquired from Kaggle which contains the first 10mins. Stats of 9,879 ranked games. The skill level of players ranges from DIAMOND I to Master which means that players are roughly from the same level. There are two teams (red and blue) in the game. Each game shares 19 features (38 in total). And the target variable in this dataset is "blueWins": 1indicates the blue team wins and 0 indicates red team wins. Therefore, this problem is a type of classification.

e. Public Projects or Publications

From the Kaggle, one of the authors used this dataset to cluster player behavior and learn the optimal team composition for multiplayer online games. By applying the cluster and classification models, the author is able to determine wins and losses with around 70% accuracy for their target game. Another author used this dataset to predict a class of blueWins. By applying logistic regression, this author is able to make a prediction about an outcome of a game with an accuracy of 74.646%.

Exploratory Data Analysis

Since there are 38 features in total, they are categorized as categorical column and continuous column. Five interesting figures are shown as follows:

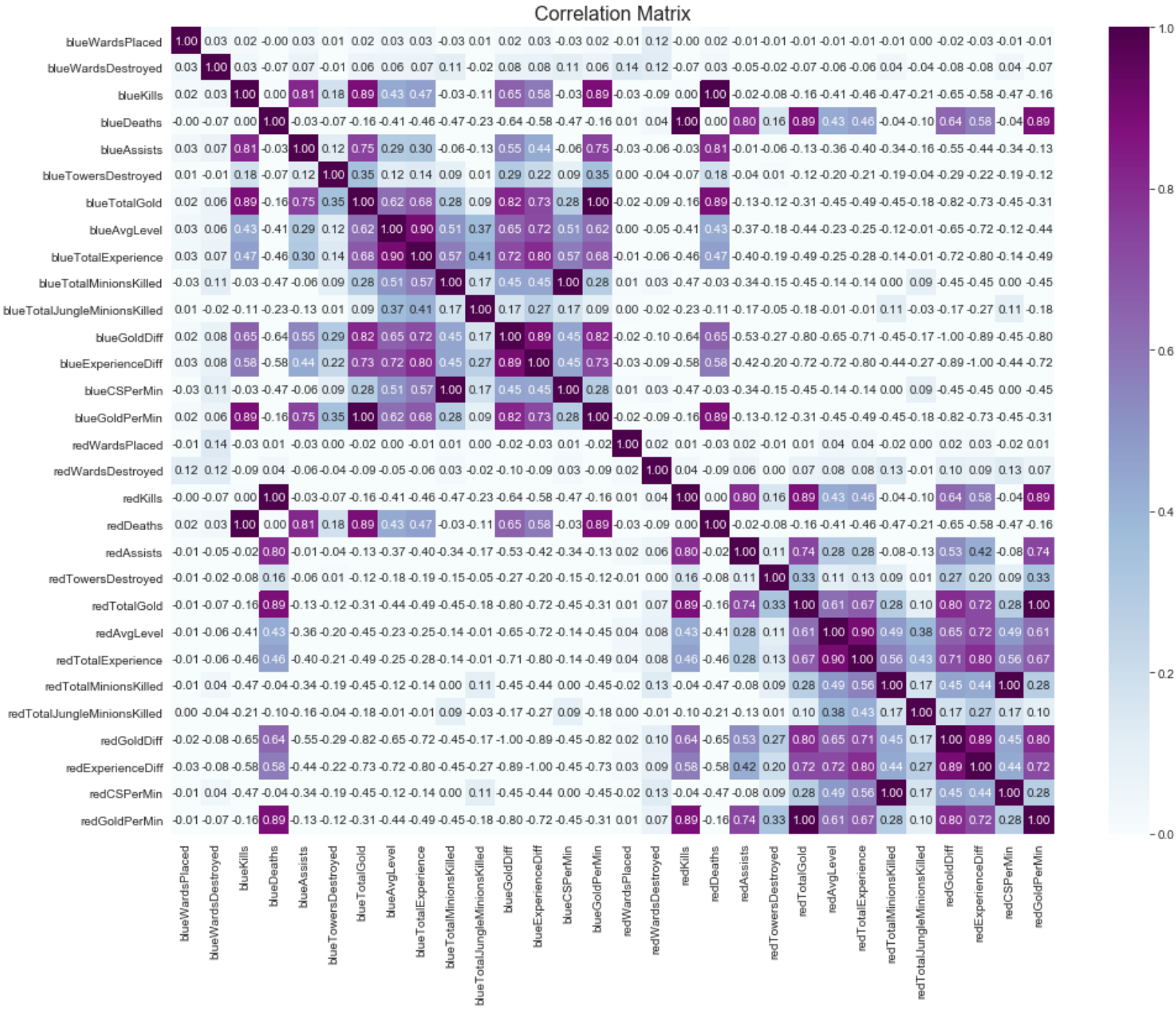

Figure 1: From the figure above, we can see correlations between two features. For instance, "Blue Total Experience" has a strong correlation with "Blue Average Level" with a value of 0.90 but has a weak correlation with "Blue Wards Placed".

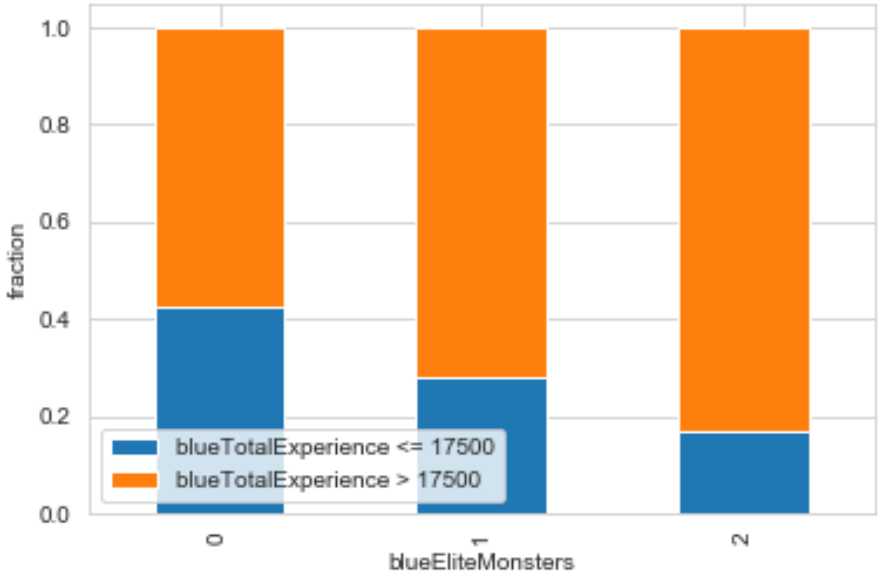

Figure 2: From the figure above, we can see the x-axis is the blue Elite Monsters(categorical) and y-axis is fraction of players in group. The orange color represents "blue total experience > 17500" while the blue color represents "blue total experience <= 17500". We can see that as the the number of blue Elite Monsters increases, the fraction of "blue total experience <= 17500" decreases..

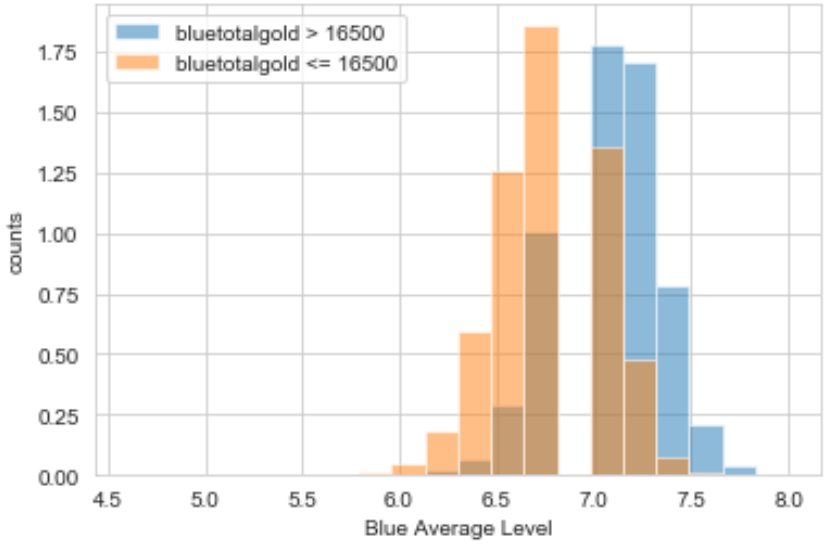

Figure 3: From the figure above, the x-axis is blue average level, and the y-axis is the counts. The light orange color represents "blue total gold <= 16500" while the light blue color represents "blue total gold > 16500". We can see that as the blue total gold > 16500, the blue average level is likely higher.

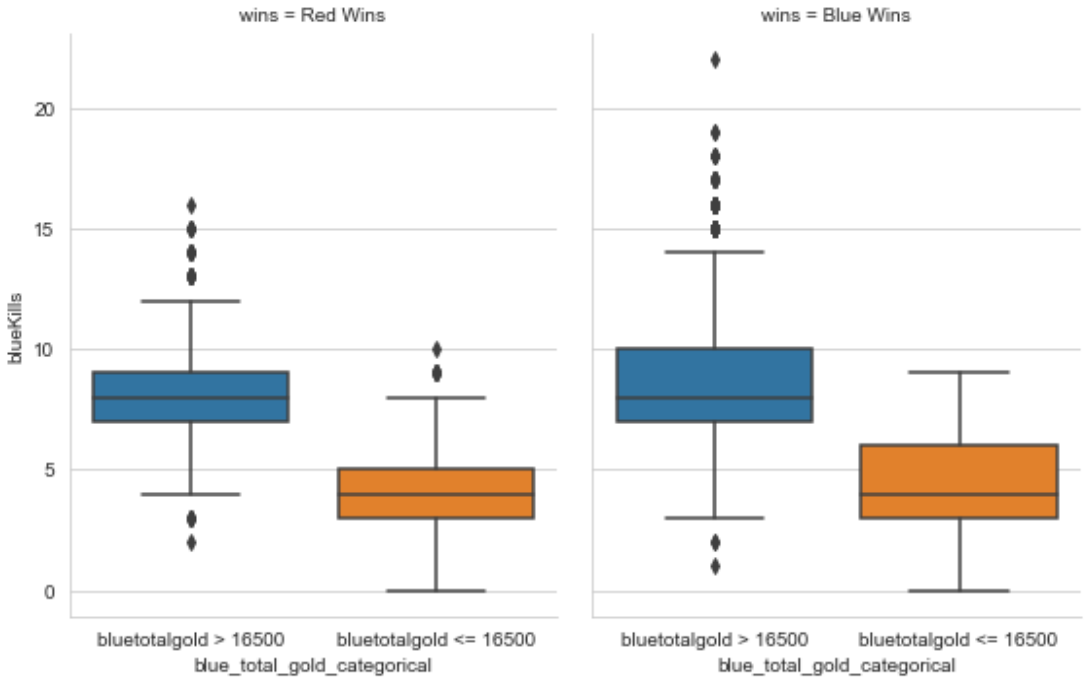

Figure 4: From the figures above, the y axis is "blue kills" and the x axis is the blue total gold by category. The left indicates red wins while the right figure indicates blue wins. We can see that there are more outliers and larger range if blue wins compared to red wins. The interquartile range is smaller in red wins compared to the blue wins. However, their medians are similar.

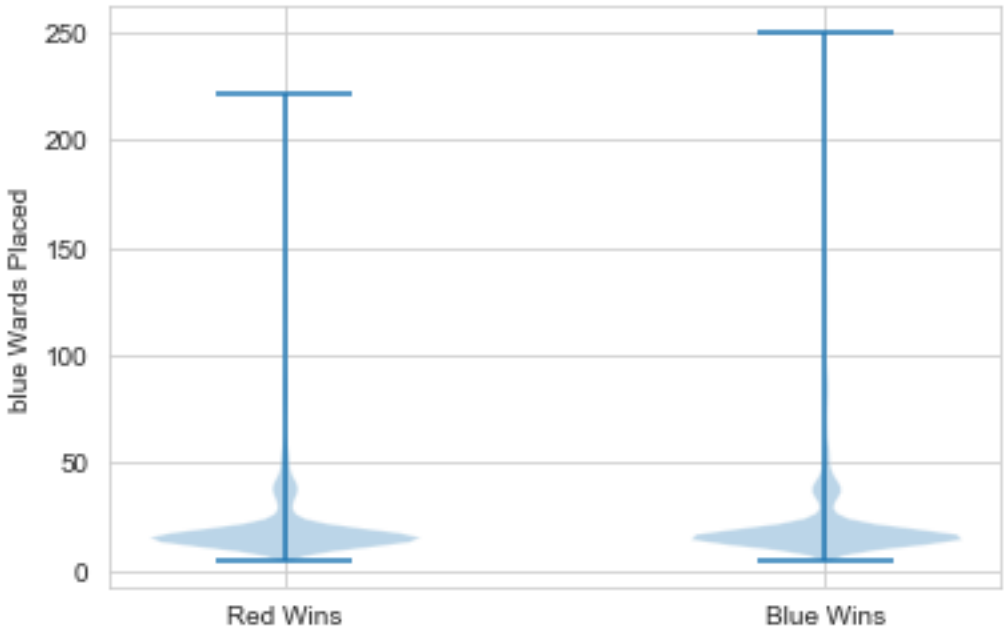

Figure 5: From the figure above, the x-axis is blue wins(target) and y-axis is blue_wards placed. We can see that the range of the target variable for red wins is a little bit smaller than that for blue wins. And the median for both are similar.

Data Preprocessing

This dataset is independent and identically distributed since each random variable has the same probability distribution as the others and all are mutually independent. In addition, all samples stem from the same generative process and share the same features. And the generative process is assumed to have no memory of past generated samples. And by observation, this dataset does not have a group structure and it is not a time series data.

Since this dataset only has 9879 samples which means large datasets. This dataset will be at first split into other dataset and test dataset with the ratio of 9: 1 and then split other dataset into train and validation datasets with the ratio of 9:1. Therefore, 81% of points are in the training dataset, 9% of points are in the validation dataset, and 10% of points are in the testing dataset.

On the other hand, three different preprocessors and 38 features have been used in the preprocessed data. The first one is the OneHotEncoder which converts categorical features into dummy arrays. From the dataset, “firstblood”, ‘Herald” and “Dragons” in both red and blue teams are applied. The reason is that in the first 10 minutes of the game, there are only 2 categories for each feature which 0 or 1. Therefore, OneHotEncoder is a good way to make conversion.

The second one is MinMaxScaler which scales continuous variables be between 0 and 1. From the dataset, “AvgLevel”, “TowersDestroyed”, and “EliteMonsters” are applied. The reasons are that AvgLevel is expected to be within the range of 1 and 18, TowersDestroyed is expected to be within the range of 0 to 9, and Elite Monsters is expected to be within the range of 0 and 3.

The last one is StandardScaler. From the dataset, 26 features from both teams are applied to this preprocessor. The reason is that, each feature is continuous and follows a tailed distribution. Therefore, StandardScaler is a good way to be used here. And there is no need to do with the target variable since it look good with 0 standing for red wins and 1 for blue wins.

Reference

1. Fanboi, Michel's. “League of Legends Diamond Ranked Games (10 Min).” *Kaggle*, 13 Apr. 2020, www.kaggle.com/bobbyscience/league-of-legends-diamond-ranked-games-10-min.
2. Juras, Marta. “League of Legends' Ranking System Explained: How It Works.” *Dot Esports*, 16 Nov. 2019, dotesports.com/league-of-legends/news/league-of-legends-ranking-system-explained-17171.

Github Link: https://github.com/YunxuanZeng/1030Project