Personal Project: Winner Prediction for League of Legends

Decision 561F: Foundations of Data Analytics
Coco (Kewei) Jiang

BUSINESS UNDERSTANDING

The ESports Gaming industry has seen tremendous growth over the years, both in terms of viewership and revenue. In this project, I will focus on analyzing the most popular esports game League of Legends by Riot Games. League of Legends is a multiplayer online battleground arena-style game. In the game there are two teams with 5 players on each team, competing to destroy the enemy base (Nexus). 3.8 million viewers cheered at its 10th World Championship competition in October 2020. The game generated 1.5 billion dollars in revenue in 2019. Over 32 brands have partnered with League of Legends teams. It is important to their brands, as well as the team itself to win both for revenue and reputation. How can we predict the winning team based on the two teams' performance in game?

To win the game, it is not only important to win fights, but also critical to make strategic plans to control map resources including dragons, barons and other jungle monsters to gain "buffs". Thus, it is valuable and important for gaming clubs to understand what the most important factors are that lead to the win. With that understanding, gaming clubs can better allocate time and resources to make more efficient strategy. Winning more often will also lead to more positive association for the sponsor brands as well as time on major stages if their teams advance in tournaments. For the largest tournaments, the prize pool can be up to 6.4 million dollars (LoL Worlds 2018), which is significant for the players as well.

DATA UNDERSTANDING

The source of my data is the dataset titled "League of Legends Ranked Games" by Mitchell J, as found on Kaggle (https://www.kaggle.com/datasnaek/league-of-legends?select=games.csv). The original data comes from League of Legend's public data platform Riot Games API (https://developer.riotgames.com/apis).

The dataset is a collection of 51,490 ranked LOL games with 61 variables. For each game, there are fields for:

- Game ID: Game ID
- Creation Time: Creation time
- Game Duration (in seconds): Game duration (in seconds)
- Season ID: Season ID

- Winner (1=team1, 2=team2): the winning team
- First Baron, dragon, tower, blood, inhibitor and Rift Herald (1 = team1, 2 = team2, 0 = none): The first Baron Nash, dragon, tower, blood, crystal, canyon pioneer
- Champions and summoner spells for each team (Stored as Riot's champion and summoner spell IDs): Heroes and summoning spells selected by each team
- The number of towers, inhibitors, Baron, dragon and Rift Herald kills each team has: Tower, Crystal, Baron, Dragon and Canyon Vanguard kills
- The 5 bans of each team (Again, champion IDs are used): The banned heroes of each team

The target variable of the classification model is **Winner.**

EXPLORATORY ANALYSIS

A. Distribution of winner and game duration

Firstly, I checked the distribution of the target variable Winner using the pie chart (Exhibit 1). There are a total of 51490 records in the data set, among which team 1 won 26077 times, accounting for 50.6%, and team 2 won 25413 times, accounting for 49.4%. There is no sample imbalance.

Then, I looked at the distribution of the game duration (Exhibit 2). It can be seen from the histogram that the game duration roughly obeys a normal distribution. The shortest game duration is 3 minutes, and the longest game duration is 79 minutes. The middle 50% is between 26 and 36 minutes. We can observe that there is a peak before 3 minutes, and that’s because in LOL, players have the right to vote to remake the game by the beginning 3 minutes. Another peak is at 15 mins. That’s because Players are allowed to vote to surrender at 15 minutes if they believe there is a high chance that they will lose in the end. Thus, I removed rows with a duration time less than 15 minutes.

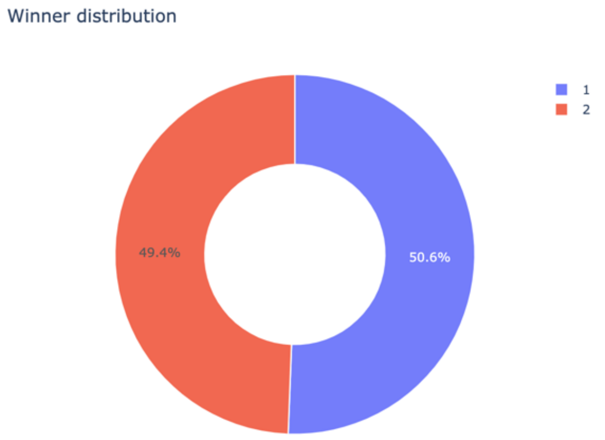


Exhibit 1.

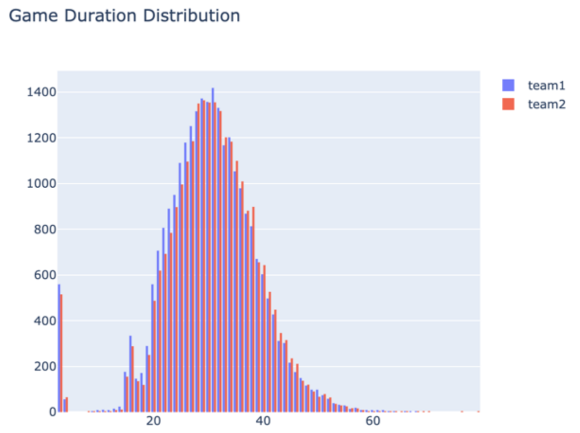


Exhibit 2.

B. Impact of first objectives on target variable

First blood (Exhibit 3): the winning rate of the team with the first blood is relatively high. In the first team's match, the winning rate when the first blood is first won is 59.48%, which is 18% higher than the game without the first blood. In the second team's game, the winning percentage when they got the first blood was 58.72%, which was 18% higher than the game without the first blood.

First tower (Exhibit 4): from the data point of view, the first defensive tower seems to be a more convincing indicator. In the first team's game, the team's winning rate when the first tower was destroyed was as high as 70.84%, which was 41.64% higher than the game without a tower. In the second team game, there are similar data performance.

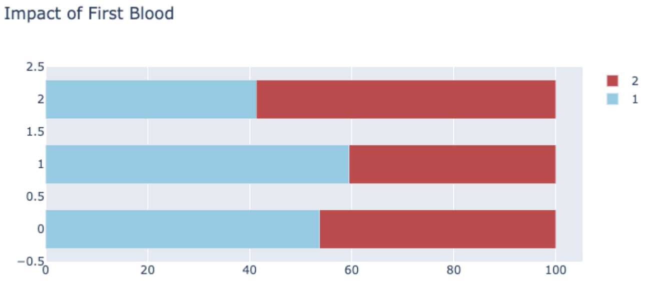


Exhibit 3.

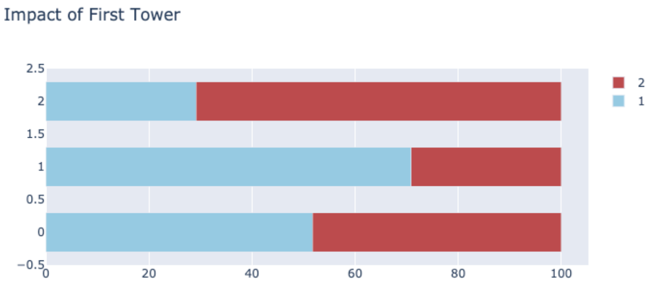


Exhibit 4.

First inhibitor (Exhibit 5): the team that gets the first inhibitor in the game can win 91% of the time. This is predictable to a certain extent, because destroying the inhibitor first represents the sufficient advantage that the team has accumulated, and the inhibitor is very powerful and valuable in the game.

First baron (Exhibit 6): the graph shows that the team that killed the first baron in the game has an 80% winning rate.

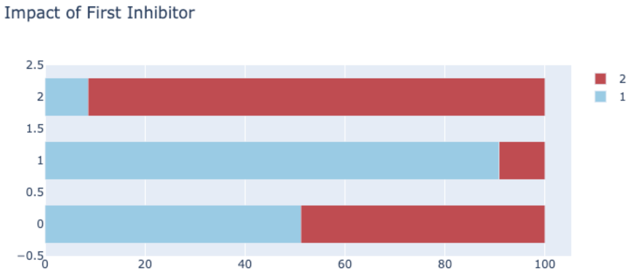


Exhibit 5.

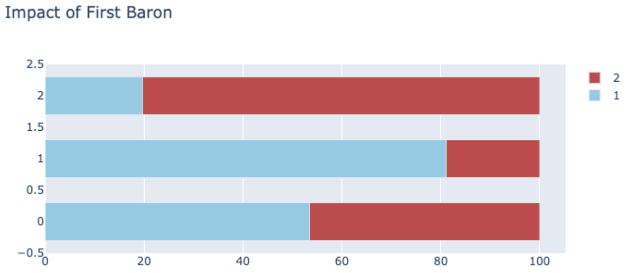


Exhibit 6.

First Dragon (Exhibit 7): the winning rate of the team that killed the first dragon is 68.6%, which is 36% higher than the winning rate of the games without killing the first dragon.

First Rift Herald (Exhibit 8): the winning rate of the team that killed the first rift herald is 69.45%, which is 38.92% higher than the winning rate of the games without killing the first rift herald.

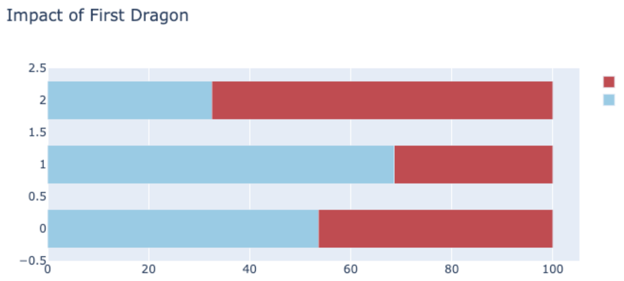


Exhibit 7.

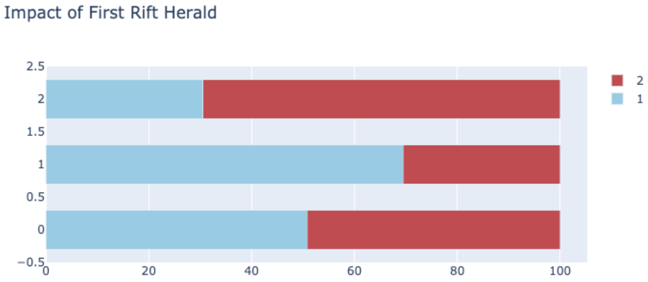


Exhibit 8.

C. Impact of kills of objectives on target variable

Tower Kills (Exhibit 9): by looking at the impact of tower kills by team 1, we can observe that the more towers destroyed, the greater the probability of winning. When the number is greater than 8, the winning rate is greater than 85%. When all 11 towers are destroyed, the winning rate is 99.16%.

Inhibitor Kills (Exhibit 10): the more inhibitors destroyed, the greater the probability of winning. The probability of winning without destroying the crystal is 12.55%, the probability of winning one is 81.11%, and the probability of two is 92.38%.

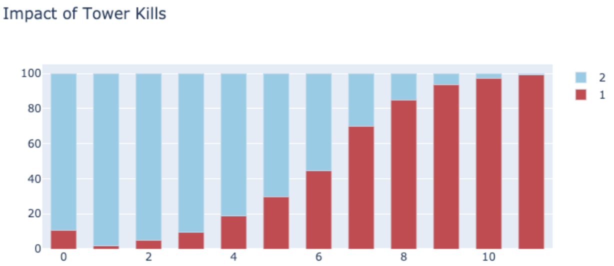


Exhibit 9.

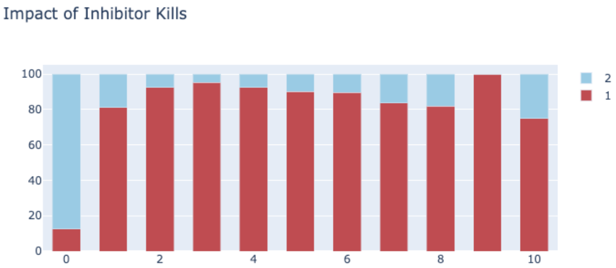


Exhibit 10.

Baron Kills (Exhibit 11): the more barons killed, the greater the probability of winning. There is only one observation in the data set with 5 baron kills, which needs to be removed later.

Dragon Kills (Exhibit 12): The probability of winning without destroying the crystal is 16.45%. The probability of winning with one dragon kill is 47.5%, and the probability of wining with two dragon kills is 72.18%.

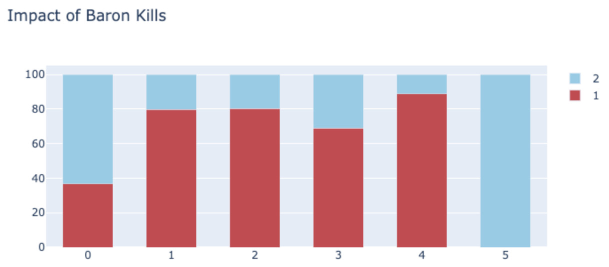


Exhibit 11.

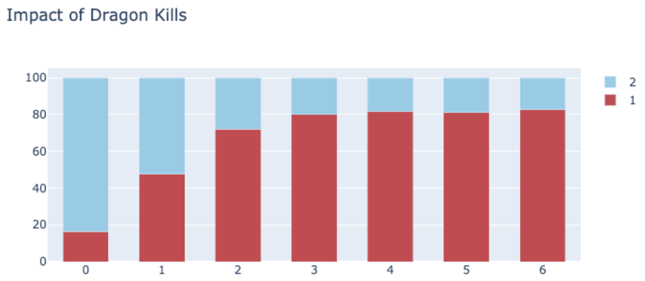


Exhibit 12.

D. Correlation

After visualizing the impact of game objectives on target variable “Winner”, I checked the impact of numerical variables on Winner using the correlation map (Exhibit 13). Among these variables, number of tower kills has the largest impact on Winner (0.77), followed by inhibitor kills (0.65).

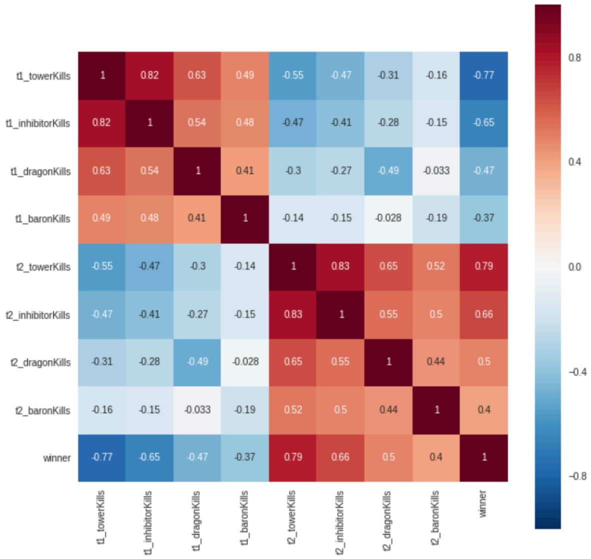


Exhibit 13.

Data Cleaning for Modeling

A. Check missing values

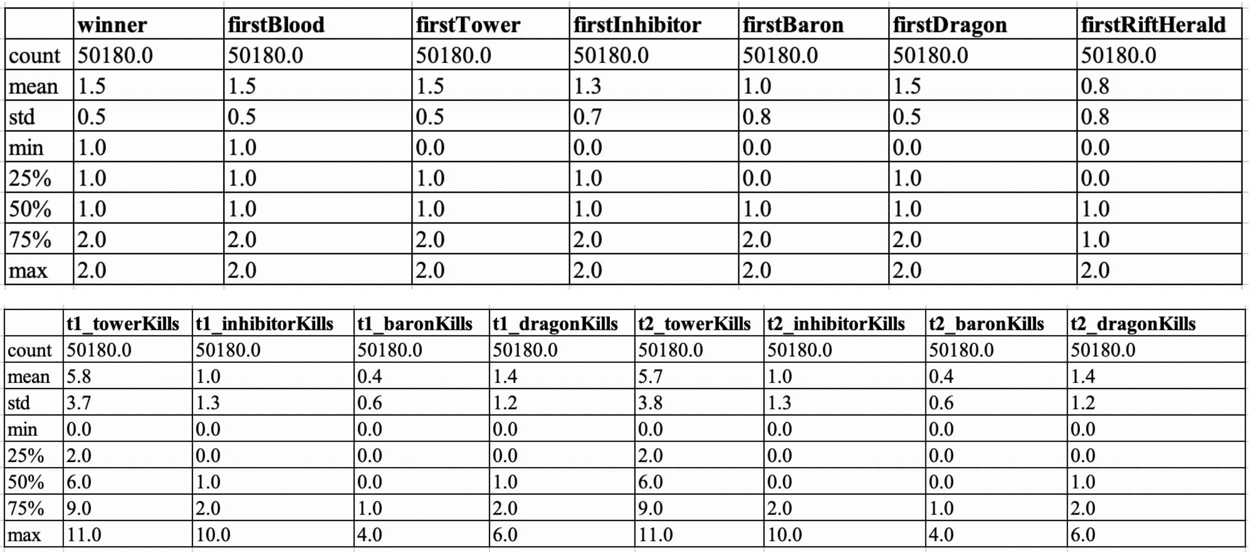
In the data set, there is no missing value for the observations. Thus, no action was needed for dealing with missing value.

B. Remove outliners

As mentioned in the conclusions of EDA, game with a duration under 15 minutes needed to be removed. Besides, the one game with 5 baron kills also needed to be removed.

C. Delete unrelated columns

Among the 61 variables, I deleted unrelated columns including game ID, season ID, champion selection and bans. After removing those variables, I selected 15 variables (Exhibit 14) to build the classification model.



	winner	firstBlood	firstTower	firstInhibitor	firstBaron	firstDragon	firstRiftHerald
count	50180.0	50180.0	50180.0	50180.0	50180.0	50180.0	50180.0
mean	1.5	1.5	1.5	1.3	1.0	1.5	0.8
std	0.5	0.5	0.5	0.7	0.8	0.5	0.8
min	1.0	1.0	0.0	0.0	0.0	0.0	0.0
25%	1.0	1.0	1.0	1.0	0.0	1.0	0.0
50%	1.0	1.0	1.0	1.0	1.0	1.0	1.0
75%	2.0	2.0	2.0	2.0	2.0	2.0	1.0
max	2.0	2.0	2.0	2.0	2.0	2.0	2.0

	t1_towerKills	t1_inhibitorKills	t1_baronKills	t1_dragonKills	t2_towerKills	t2_inhibitorKills	t2_baronKills	t2_dragonKills
count	50180.0	50180.0	50180.0	50180.0	50180.0	50180.0	50180.0	50180.0
mean	5.8	1.0	0.4	1.4	5.7	1.0	0.4	1.4
std	3.7	1.3	0.6	1.2	3.8	1.3	0.6	1.2
min	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
25%	2.0	0.0	0.0	0.0	2.0	0.0	0.0	0.0
50%	6.0	1.0	0.0	1.0	6.0	0.0	0.0	1.0
75%	9.0	2.0	1.0	2.0	9.0	2.0	1.0	2.0
max	11.0	10.0	4.0	6.0	11.0	10.0	4.0	6.0

Exhibit 14.

D. Splitting data

I split the data into train set and test set at a 4:1 ratio using stratified sampling method.

Modeling

Given that the goal of this project is to predict which team can win the game, this model is a supervised task. Thus, I applied decision tree algorithm to train set.

To select the best tree depth, I used GridSearchCV to decide the best parameters. The best tree depth is 7. As is shown in the tree (Exhibit 15.), the most important factors deciding the winner are t2_towerkills, t1_towerkills, firstinhibitor, t1_inhibitorKills and t2_inhibitorKills.

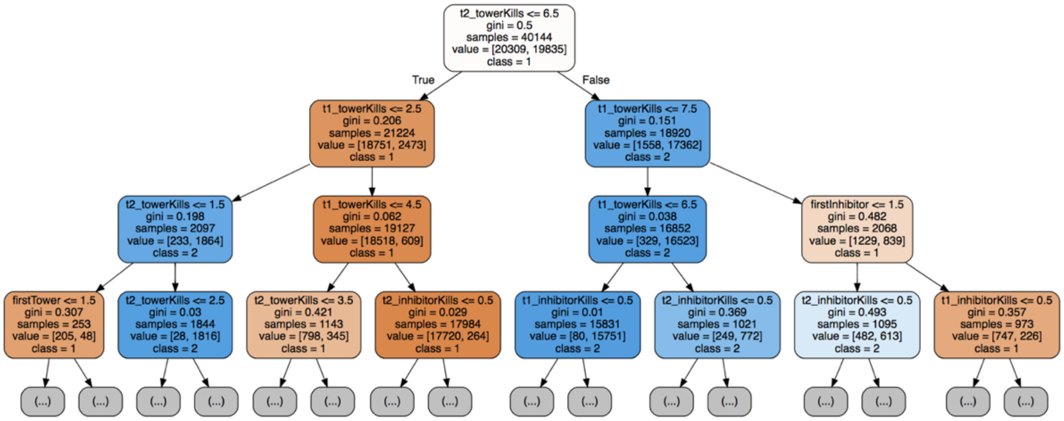
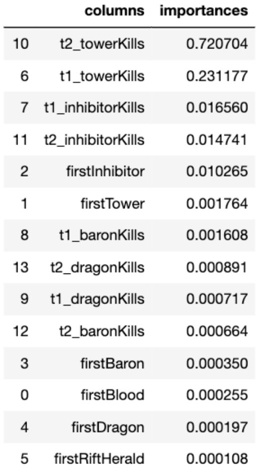


Exhibit 15.

To further looking at the most important variables influencing the winner, I looked at the importance level of each variable of the tree (Exhibit 16.)



	columns	importances
10	t2_towerKills	0.720704
6	t1_towerKills	0.231177
7	t1_inhibitorKills	0.016560
11	t2_inhibitorKills	0.014741
2	firstInhibitor	0.010265
1	firstTower	0.001764
8	t1_baronKills	0.001608
13	t2_dragonKills	0.000891
9	t1_dragonKills	0.000717
12	t2_baronKills	0.000664
3	firstBaron	0.000350
0	firstBlood	0.000255
4	firstDragon	0.000197
5	firstRiftHerald	0.000108

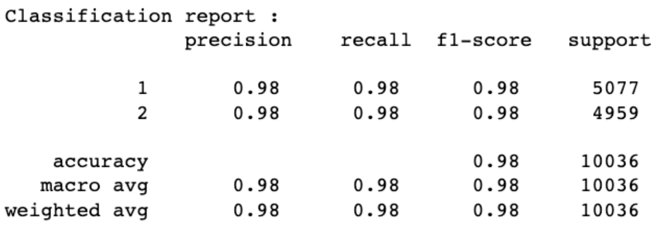
Exhibit 16.

I elucidate the following insights by examining the magnitude of the variables' importance:

- To win the game, the most important objective is to destroy as many enemy towers as possible. The more tower kills a team have, the more likely the team will win the game.
- Inhibitor kill is the second most important objective in the game in order to win. The team which gets the first inhibitor is also more likely to win the game.
- For map resources, the most import one to fight for is the barons. The team with more baron kills is more likely to win the game.
- According to the tree, for team1 which is the blue team, barons have higher importance. For team2 which is the red team, dragons are more important.
- Rift herald kills are not important for winning the game. Thus, it's fair to give up on rift herald for more baron and dragon kills.

Evaluation

To evaluate the decision tree model, I split the data into training set and testing set at a 4:1 ratio to calculate the out-of-sample accuracy. There are 10,036 observations in the testing set. The accuracy scores are summarized in the table (Exhibit 17.). As is shown, the out-of-sample accuracy of the decision tree is 98%.



```
Classification report :
               precision    recall  f1-score   support

           1       0.98      0.98      0.98      5077
           2       0.98      0.98      0.98      4959

    accuracy                           0.98     10036
   macro avg       0.98      0.98      0.98     10036
weighted avg       0.98      0.98      0.98     10036
```

Exhibit 17.

Deployment

The goal of this project is to provide insights from the models to gaming clubs seeking to improve their strategic planning in game and to Esports business to predict the winner of the games.

As was discussed in the evaluation section, this model has an out-of-sample accuracy of 98%. This model allows gaming to understand which map resources and game objectives are significantly important to determine the final winner. Thus, they will be able to allocate time more efficiently and make better strategic plan based on the importance generated by the model. For LOL and Esports fans, this model is useful for predicting the winner of games.