League of Legends Classification

1. **Introduction**

League of Legends (LOL) is a famous online MOBA game that has been last for 10 years. 10 players are having a 5 vs 5 competition during every game until one team has destroy another team’s base. During every game, each player chooses a champion, killing minions, monsters, and enemies’ champions to earn golds. Then purchase items to assistant to destroy enemy’s base. Not only the champion user selected are significant, golds, towers, but do the cooperation between teammates contributes significantly to the final result.

1. **Motivation and introduction of the problem**

Every year, LOL worldwide professional competition will be held around the world. Numbers of professional teams fight for their dreams. As the audience, people are significantly expected their favorite team to win. Consequently, many people are willing to develop a method to predict if one team can win during each match. At the same time, since there are many aspect such as gold and kills that may influence during the game, athletes and other LOL players are also interested in how those aspects affect the result of each match.

Consequently, as the main goal of this project, we will focus on the relationship between the amount of golds earned by each five positions (Top, Jungle, Middle, Attack Damage Carry (ADC) and Support) for both blue team (consider as ally) and red team (consider as enemy) and the result of the match. More specifically, we will use different statistical classification models to classify if the blue team in a match will win or will lose according to the amount of golds earned by all ten players at the 10th minute, the 20th minute, and finally the 30th minute. And analysis how amounts of golds during each match influence the result of that match.

1. **Data**

The original data are collected from Kaggle (url:<https://www.kaggle.com/chuckephron/leagueoflegends#LeagueofLegends.csv>), which is distributed by Chuck Ephron.

This data has total 57 columns and 7620 observations that including all professional competitions from 2015 to 2018. The data has the following aspects:

* 1. Address: website address the data is scraped from.
  2. League: League or Tournament the match took place in.
  3. Year: Year the match took place in.
  4. Season: Spring or Summer depending on which half of the year the match took place in.
  5. Type: Season, Playoffs, Regional, or International match.
  6. Team Tag for both side: (blueTeamTag, redTeamTag).
  7. gamelength: the length of the match.
  8. Golds: the golds earn by all ten players in each minute, the total amount of gold in both team, and the gold difference.
  9. Champions: The champions that are banned before the game and the champions used by players.
  10. Players: the names (ID) of all players.
  11. Kills: the killer, victim, assistants, time, and location for each kill.
  12. Dragons/Barons/Heralds: the time and the object time slayed by each side, for dragon the types of dragon are also included.
  13. Towers/Inhibitors: the time the construction destroyed and the position of it.

This data set almost include all the information that can be collected during each match. However, there are some problems that prevent to be used for classification. For this reason, we clean this data set to make sure it works for all classification models.

1. **Data preprocessing**

The script used is “DataClean.py”

* 1. **Data cleaning: string**

Since R can easily handle string easily by applying as.numeric or as.factor as needed, string features do not need to be deleted or extremely modified here.

For all string features, the script shows that they are proper enough. Thus, we do not modify them.

* 1. **Data cleaning: array**

Array values in this data do not have a fixed length. Thus the data we remained is the length of those arrays, and the first element of those arrays

Golds:

Calculate the golds for all two teams for all five positions for 10min, 20min, and 30 min.

If the games ends before 30min, we will set the gold to be the amount at the end of gold.

Counting Legendary Monsters: Dragon

Count the first dragon slayed for both sides, and the total number of dragons for both sides.

If one side does not slay any dragon, record the time as the end of game.

There are elements dragon in 2017 and 2018, we ignore them here for now.

Counting Legendary Monsters: Baron/Herald

Count the first baron slayed for both sides.

Counting Towers: Tower/Inhibitor

Count the first tower destroyed for both sides, and the total number of towers/Inhibitors for both sides.

If one side does not destroy any tower/Inhibitor, record the time as the end of game.

Counting Kills

Count the first kill for both sides, and the total number of kills for both sides.

If one side does not kill, record the time as the end of game.

Finally, we drop all address, champions banned, rResult, Year, Season, (b/r)TeamTag, League, Type and name of players since they are not important or hard to clean. The result data named “LOL.csv”, contains only checked string, floating points, and integers. For response, integer “0” indicates that blue team (Ally) loses that match, and “1” indicates that ally wins that match. See “DataClean.pdf” for more details.

There is a significant characteristic in both the original dataset and the new dataset, which is that data is highly correlated. The reason is related the actual property of LOL, for example, team who get a Baron will be given a positive buff that they can easily destroy towers, which can also influence their golds. If we only focus on gold only, team which gain more golds at the beginning means all members in that time have more gold to purchase items, which means they can get other benefits easier than the enemy to earn more gold later. Consequently, a champion who gain more golds may be easier to gain more rapidly later. There are also some potential dependencies between different positions, which will be discussed and tested later.

1. **Methodology**

First, we will using Linear Discriminal Analysis (LDA) to classify the whole dataset, we get the following result:

图片包含 屏幕截图

描述已自动生成

Figure 1: LDA Visualization for whole data set

Error rate:

|  |  |
| --- | --- |
|  | Error Rate |
| Training Set Error | 0.01181102 |
| Validation Set Error | 0.01207349 |
| Testing Set Error | 0.009973753 |

With all the explanatory variables, LDA has already been a perfect split with 1% test error rate. Look into the importance of explanatory variable:

图片包含 文字, 室内

描述已自动生成

Figure 2: Significance for explanatory variables

Examining the top 20 explanatory variables, we can see that the explanatory variables named "number of \*\*\*" are especially important. It's reasonable, since these explanatory variables are too strong as they are the data of game when the whole game is over. For example, if a team has more kills when the game is end, we can make an intuitive guess that that team win the game.

And in terms of prediction, these explanatory variables do not help the analysis. We actually cannot achieve them until the end of the game. To achieve the model for prediction, we should drop these explanatory variables. (also the "game length")

For the rest of the project, we will drop all explanatory except golds (30 columns total). To achieve the goal of this project, which is to discover the relationship between golds at some time and the result of the match. The data sets we used later on are:

Response and the amount of golds at the 10th minute for all ten players.

Response and the amount of golds at the 10th minute and 20th minute for all ten players.

Response and the amount of golds at the 10th minute, 20th minute, and 30th minute for all ten players.

* 1. **Linear Discriminant Analysis**

图片包含 屏幕截图

描述已自动生成

Figure 3: LDA Visualization for 10min, 20min, and 30min

Error rate:

|  |  |  |  |
| --- | --- | --- | --- |
|  | 10th minute | 20th minute | 30th minute |
| Training Error | 0.3149606 | 0.2233596 | 0.1244094 |
| Validation Error | 0.3070866 | 0.2215223 | 0.1312336 |
| Test Error | 0.3275591 | 0.2283465 | 0.1270341 |