Leagues of Legends Logistic Regression

Haneul (Sky) Kim

Introduction

League Of Legend is an MOBA (multiplayer online battle arena), where the objective is to destroy enemy's nexus. In this project, I try to predict the win probability in binomial parameter pi, using logistic regression

Dataset Sourse:
https://www.kaggle.com/datasets/bobbyscience/league-of-legends-diamond-ranked-games-10-min?resource=download

Model 1

Blue team Logistic Regression model with highly correlated variables removed. Regression model using the blue team features of the game selected. The features with high correlation were removed since it can lead to unstable estimation of the weights(b's) because the nearly singular matrix has a large condition number, meaning it is sensitive to errors or changes in data.

The blue team features removed are the following:
blueCSPerMin, **blueGoldPerMin** : correlation 1.00 with blueTotal CS, blueTotalGold respectively
blueTotalExperience: correlated with blueAverageLevel 0.898
blueEliteMonsters: sum of dragons and heralds, so correlated highly with blueDragons **blueHeralds** (0.738, 0.652 respectively)

```
> summary(logistic_1)

Call:
glm(formula = blueWins ~ ., family = binomial, data = blue_team_stats_dropped)

Deviance Residuals: 
    Min       1Q   Median       3Q      Max  
-2.6957  -0.9107  -0.1475   0.9094   2.7990  

Coefficients:
                              Estimate Std. Error z value Pr(>|z|)    
(Intercept)                 -5.8479066  0.7185324  -8.139 4.00e-16 ***
blueWardsPlaced             -0.0012139  0.0013045  -0.931 0.352081    
blueWardsDestroyed           0.0027483  0.0109503   0.251 0.801827    
blueFirstBlood               0.1040768  0.0492419   2.114 0.034551 *  
blueKills                    0.2767271  0.0165206  16.750  < 2e-16 ***
blueDeaths                  -0.2336076  0.0105411 -22.162  < 2e-16 ***
blueAssists                  0.0001596  0.0103496   0.015 0.987700    
blueDragons                  0.4559493  0.0500251   9.114  < 2e-16 ***
blueHeralds                  0.0957700  0.0622184   1.539 0.123742    
blueTowersDestroyed          0.4290125  0.1213827   3.534 0.000409 ***
blueAvgLevel                 0.3525046  0.1243095   2.836 0.004573 ** 
blueTotalMinionsKilled       0.0090741  0.0014092   6.439 1.20e-10 ***
blueTotalJungleMinionsKilled 0.0182962  0.0028087   6.514 7.31e-11 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 13695  on 9878  degrees of freedom
Residual deviance: 10783  on 9866  degrees of freedom
AIC: 10809

Number of Fisher Scoring iterations: 4
```

Figure 1: Model 1

The first model shows the effect of the dragon and concluded that the dragons affect the game result a lot. Also, the gold difference and the experience difference contributes a lot to the higher winning possibilities.

This seems to agree with what players experience in League of Legends. The more gold you have(compared to the enemy team), the more items you can purchase. This leads to the advantage in 1 to 1 or teamfights. Also, the more experience(level up) you have, the higher level you are, which means you are stronger by the per level growth, and more mastery of the skillsets, and also higher level summoner spells (as of now, the smith deals the same but it dealt differently by player level).

However, the interesting insight from the data is that the effect of the dragons, especially when compared to the herald, is massive. In summary of the model 3, it shows that having killed a dragon is related to around 0.5 more win possibility (herald only accounting for 0.05 approximately)

It is assumed that the reason why dragons have more impact on the team winning possibility is that the dragon gives cumulative, team-wide buff effects whereas heralds are more like one time buff to a solo player when returning home.

However, this data only shows the consumption of the dragon in binary variables, eigher have had dragon, or not. If the exact number of dragons killed and the elder dragon(very powerful effect) is available, more accurate analysis will be possible.

Moreover, the summary of the data says 6303 blue teams finished the game without a dragon. And it is quite against what happens in the real game. So this data might contain some errors about the dragon statistics.

Other Models

Other models were used to find out if the difference statistics, for example differences in number of kills, is a good predictor variable for winning possibilities. Via model 2, 3, and 4, I could come to the conclusion that the difference in the number of kills is not a predictor variable that affects the winning possibility because gold gained by a kill is not proportional to the number of kills. In other words, in League of Legends there is a penalty to the well grown champion by giving more gold when a champion kills cumulatively without dying, meaning killing does not necessarily leads to a significant growth.

```
> summary(logistic_2)

Call:
glm(formula = blueWins ~ ., family = binomial, data = blue_team_stats_dropped_diff)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.7630  -0.8740  -0.1396   0.8732   2.7268

Coefficients:
                            Estimate Std. Error z value Pr(>|z|)
(Intercept)               -1.203e-01  8.117e-01  -0.148   0.8822
blueWardsPlaced           -1.941e-03  1.326e-03  -1.464   0.1433
blueWardsDestroyed         5.659e-04  1.121e-02   0.050   0.9597
blueFirstBlood             8.653e-02  5.210e-02   1.661   0.0967 .
blueKills                  6.094e-03  2.259e-02   0.270   0.7873
blueDeaths                -7.663e-03  1.702e-02  -0.450   0.6526
blueAssists               -4.231e-03  1.070e-02  -0.395   0.6925
blueDragons                5.240e-01  5.124e-02  10.227  < 2e-16 ***
blueHeralds                4.798e-02  6.327e-02   0.758   0.4483
blueTowersDestroyed       -1.199e-01  1.302e-01  -0.921   0.3568
blueAvgLevel               2.430e-02  1.320e-01   0.184   0.8539
blueTotalMinionsKilled    -2.054e-03  1.585e-03  -1.295   0.1952
blueTotalJungleMinionsKilled 4.715e-03  2.959e-03   1.594   0.1110
blueGoldDiff               3.973e-04  3.858e-05  10.299  < 2e-16 ***
blueExperienceDiff         2.394e-04  3.290e-05   7.277 3.42e-13 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 13695  on 9878  degrees of freedom
Residual deviance: 10449  on 9864  degrees of freedom
AIC: 10479

Number of Fisher Scoring iterations: 4
```

Figure 2: Model 2

```
> summary(logistic_3)

Call:
glm(formula = blueWins ~ ., family = binomial, data = blue_team_stats_killDiff_dropped)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.7634  -0.8741  -0.1398   0.8736   2.7262

Coefficients:
                             Estimate Std. Error z value Pr(>|z|)
(Intercept)                -1.184e-01  8.114e-01  -0.146   0.8839
blueWardsPlaced            -1.936e-03  1.325e-03  -1.461   0.1440
blueWardsDestroyed          6.103e-04  1.120e-02   0.054   0.9565
blueFirstBlood              8.633e-02  5.205e-02   1.659   0.0972 .
blueAssists                -4.834e-03  8.334e-03  -0.580   0.5619
blueDragons                 5.241e-01  5.124e-02  10.228  < 2e-16 ***
blueHeralds                 4.787e-02  6.326e-02   0.757   0.4492
blueTowersDestroyed        -1.205e-01  1.300e-01  -0.926   0.3543
blueAvgLevel                2.122e-02  1.275e-01   0.166   0.8678
blueTotalMinionsKilled     -2.006e-03  1.493e-03  -1.344   0.1791
blueTotalJungleMinionsKilled 4.783e-03  2.863e-03   1.671   0.0948 .
blueGoldDiff                3.972e-04  3.857e-05  10.299  < 2e-16 ***
blueExperienceDiff          2.390e-04  3.264e-05   7.322 2.44e-13 ***
killDiff                    7.447e-03  1.685e-02   0.442   0.6586
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 13695  on 9878  degrees of freedom
Residual deviance: 10449  on 9865  degrees of freedom
AIC: 10477

Number of Fisher Scoring iterations: 4
```

Figure 3: Model 3

```
> summary(logistic_4)

Call:
glm(formula = blueWins ~ ., family = binomial, data = blue_team_stats_killDiff_dropped_2)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.7105  -0.9123  -0.1456   0.9092   2.8241

Coefficients:
                              Estimate Std. Error z value Pr(>|z|)
(Intercept)                  -6.043141   0.714491  -8.458  < 2e-16 ***
blueWardsPlaced              -0.001343   0.001303  -1.031 0.302715
blueWardsDestroyed            0.001651   0.010934   0.151 0.879943
blueFirstBlood                0.107687   0.049205   2.189 0.028632 *
blueAssists                   0.016907   0.007994   2.115 0.034441 *
blueDragons                   0.454400   0.050007   9.087  < 2e-16 ***
blueHeralds                   0.099707   0.062151   1.604 0.108655
blueTowersDestroyed           0.448930   0.121028   3.709 0.000208 ***
blueAvgLevel                  0.451342   0.118103   3.822 0.000133 ***
blueTotalMinionsKilled        0.007904   0.001331   5.938 2.88e-09 ***
blueTotalJungleMinionsKilled  0.016712   0.002737   6.107 1.02e-09 ***
killDiff                      0.243225   0.009861  24.664  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 13695  on 9878  degrees of freedom
Residual deviance: 10789  on 9867  degrees of freedom
AIC: 10813

Number of Fisher Scoring iterations: 4
```

Figure 4: Model 4

Figure 5: Assessing Models

Models	AIC	Deviance
1	10809	10783
2	10479	10449
3	10477	10449
4	10813	10789

Conclusion

In conclusion, dragons have a great effect on winning rate, but the form of the data is limited(binary), so would only provide partial information about the real effect of the dragons. Also, difference in the number of kills between two teams does not have a great effect on the winning rate, and is thought to be because of the league system.