An analysis on some important hero stats of Reinhardt

Xuan Zhu

November 12, 2018

Introduction

Background Info For a good explanation of how overwatch is played, go to this website:

https://www.wired.com/2017/01/overwatch-guide/.

For those who are already overwatch players and are interested in Reinhardt, this analysis is focused on the effect of death rate(per game) and the number of Fire kills, Charge kills and ult kills on your win rate. In common sense, we know that the less death rate is, the higher the win rate; and the more kills we get, the higher the win rate. But numerically, how much the effect could be is what we still do not know.

*The "Fire", "Charge" and "Ultimate skill" are names of three unique hero skills belong to Reinhardt.

Method

i. Data Source

I used the public data from www.overbuff.com. For detailed procedure on how to web scrape statistics from the website, please refer to the Appendix: Data Mining Instruction. Several things need to be mentioned before you dive into the model:

a) The game records are taken in the time period of late-Nov,2018, corresponding to the game version of PATCH 1.30.0.1.The variables I choose to put in my model are usually not sensitive to new patches. But this is not 100% guaranteed. Please be careful.

b) The game records are from top 100 players rated by Overbuff instead of Blizzard. The difference is subtle, but Overbuff's rating system is able to filter out those who play a lot and get a relatively stable high skill rating score. For example, a top 500 support player in Blizzard's rating system is required to play at least 50 games to be shown on the top list, but we do not know how many games he was playing on Ana. Maybe he just played 2 or 3 games as Ana and played the rest 48 games as Zenyatta. But Overbuff's hero ranking system can ensure each player recorded in my dataset is particularly pro in his/her chosen heroes.

```
##   WIN_RATE WIN LOSS PLAYER_ID DEATH CHARGE   ULT  FIRE
## 1   0.5000   3    3 Arty-1346 10.00   2.00  4.83  7.00
## 2   1.0000   1    0 Arty-1346 21.00   6.00 10.00 16.00
## 3   0.2500   1    3 Arty-1346  8.50   2.75  4.00  8.00
## 4   0.6667   2    1 Arty-1346  6.67   1.67  1.33  6.00
## 5   0.5333   8    7 Arty-1346  8.93   2.93  3.93  7.13
## 6   1.0000   2    0 Arty-1346  9.00   4.50  3.50  8.50
```

Each row in the data set represents one record of each time one player opens Overwatch. For example, Arty-1346 logged into Overwatch last night, playing 6 games as Rein in total and the win-loss record is 3-3. The numbers following the PLAYER_ID columns are the average stats **per game** for the 6 games he played.

ii.Model Used

The model used for analysis is a multilevel logistic model, which takes the variablity of personal performance into account.

$$logit^{-1}(\beta_0 + \beta_{charge}Charge + \beta_{Ult}Ult + \beta_{Fire}Fire + \beta_{Death}Death) = P_{Win}$$

```
## [1] 9.296472

## [1] 2.847276

## [1] 4.855759

## [1] 8.849342

## Generalized linear mixed model fit by maximum likelihood (Laplace
##    Approximation) [glmerMod]
##  Family: binomial  ( logit )
## Formula: cbind(WIN, LOSS) ~ c.DEATHR + c.CHARGE + c.ULT + c.FIRE + (1 |
##      PLAYER_ID)
##    Data: GameRecordS13R
##
##      AIC      BIC   logLik deviance df.resid
##   1378.1   1403.9   -683.0   1366.1      541
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.74443 -0.65320  0.03589  0.69328  2.84133
##
## Random effects:
##  Groups    Name        Variance Std.Dev.
##  PLAYER_ID (Intercept) 0.00206  0.04539
## Number of obs: 547, groups:  PLAYER_ID, 94
##
## Fixed effects:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.31280    0.03923   7.974 1.53e-15 ***
## c.DEATHR    -0.16888    0.02211  -7.637 2.22e-14 ***
## c.CHARGE     0.11840    0.04677   2.532   0.0114 *
## c.ULT        0.08165    0.03494   2.337   0.0194 *
## c.FIRE       0.10872    0.02400   4.530 5.91e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##          (Intr) c.DEAT c.CHAR c.ULT
## c.DEATHR  0.001
## c.CHARGE -0.076 -0.192
## c.ULT     0.050 -0.068 -0.174
## c.FIRE    0.048 -0.214 -0.143 -0.487
```

Result

i. Model Choice

1.As we observe from model output, the standard error(σ_x^2) is very small, which means that All top Rein players play rein at a similar level. This makes sense because our dataset is focused on the performance of "best Reinhardts". Those players may reach the upper limit of imroving their Reinhardt stats.(And ** Numbers does not mean everything. **) This is something we cannot control. If we have the dataset of

players from lower tiers, the group variance is expected to be large and we will not be able to ignore it. Therefore, I tend to keep this model as my first choice rather than simply applying the logistic model.

All the predictors are significant at 95% confidence level, as we expected.

Are there any interaction terms? Good Fire kills leads to faster ult, and more ults in games may imply more ult kills. Let's try:

```
mTry <-glmer(cbind(WIN,LOSS)~c.DEATHR+c.FIRE*c.ULT+c.CHARGE +(1|PLAYER_ID), data=GameRecordS13R,family=
summary(mTry)
```

```
## Generalized linear mixed model fit by maximum likelihood (Laplace
##    Approximation) [glmerMod]
##  Family: binomial  ( logit )
## Formula: cbind(WIN, LOSS) ~ c.DEATHR + c.FIRE * c.ULT + c.CHARGE + (1 |
##      PLAYER_ID)
##    Data: GameRecordS13R
##
##      AIC      BIC   logLik deviance df.resid
##   1379.2   1409.4   -682.6   1365.2      540
##
## Scaled residuals:
##       Min       1Q   Median       3Q      Max
## -2.74656 -0.64120  0.03142  0.68856  2.79585
##
## Random effects:
##  Groups     Name        Variance Std.Dev.
##  PLAYER_ID (Intercept) 0.002294 0.0479
## Number of obs: 547, groups:  PLAYER_ID, 94
##
## Fixed effects:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)   0.326762   0.042189   7.745 9.55e-15 ***
## c.DEATHR     -0.167703   0.022150  -7.571 3.69e-14 ***
## c.FIRE        0.109590   0.023991   4.568 4.92e-06 ***
## c.ULT         0.084965   0.035135   2.418   0.0156 *
## c.CHARGE      0.115473   0.046827   2.466   0.0137 *
## c.FIRE:c.ULT -0.007362   0.008069  -0.912   0.3616
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##            (Intr) c.DEAT c.FIRE c.ULT  c.CHAR
## c.DEATHR    0.020
## c.FIRE      0.063 -0.212
## c.ULT       0.084 -0.062 -0.482
## c.CHARGE   -0.095 -0.195 -0.144 -0.180
## c.FIRE:.ULT -0.366 -0.053 -0.052 -0.108  0.067
```

It seems like the interaction term is not significant.

ii.Interpretation Average Death per game:9.296472 Average Charge Kills per game:2.847276 Average Ult Kills per game: 4.855759 Average Fire Kills per game: 8.849342

If you are a top Reinhardt player(SR>4400) and have an average death &Fire Kills& Ult kills, your odds of win is $e^{0.3267} = 1.3863$, which is the same as win rate = 57.98%. Usually, the average win rate for GrandMaster(SR>4000) Rein players is around 55%. So from our model, we can get a rough idea that the

win rate of Rein main players will increase 2% if you play hard to make your score rank up from 4000 to 4400.

If you make a mistake in game and this leads you to die once more, your odds of win will decrease by $e^{0.1677}-1=$15%. Similarly, we can calculate the odds change for other predictors.

iii.Model Checking

Binned residual plot
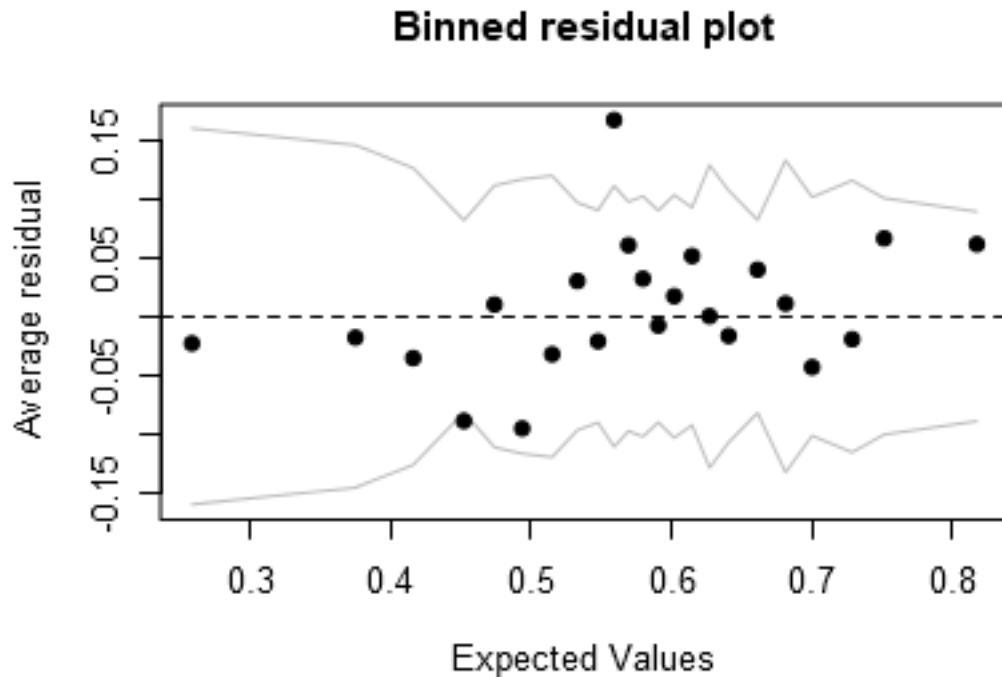
The binned residual looks good. They scattered equally above and below 0 average residual. One might notice that the expected values are kind of centered between 0.5-0.6, and this is because the current average win rate for Grandmaster-level Rein player is about 55%. In other words, The distribution of our original win rate is not normal.

Discussion

It's understandable why Fire, Ult and Charge appear in the model, but why death is important as well rather than picking KD(Killing/Death ratio) as the predictor? Because killing enemies is not the only job a tank is expected to do from a team perspective. Sometimes Rein's teammates need Rein to protect his teammate first. A good Rein can have a low KD. In other words, when converting the death rate to KD ratio, we lose some key information.

Appendix

Data Mining Instruction:https://github.com/xuz057/overwatch/blob/master/Data_Mining_Instruction.Rmd