678 Final Project

Hsueh-Pin Liu

2022-12-08

I. Abstract

League of Legends is a MOBA video game developed by Riot Games in 2009 and is one of the most popular games in the world. The game's enduring appeal gives the credit to its frequent new versions as well as new champions. To attract new players and increase income, those new champions should be stronger than other champions and the win rate should be a little bit more than 50%. Also, for some champions that are too weak, Riot will remake or strengthen those champions. Thus, here comes the problem that how to balance the strength of all champions to ensure that there are not some champions that are too op(overpowered). It may not affect normal players like us a lot in the games, but it may cause a huge change for the pros. This report will be divided into four parts: Introduction, Method, Result, and Discussion.

II. Introduction

Since this year is S12(Season 12) of the game, I will mainly focus on the data of the versions in S12. There are 11 different versions of season 12(v12.7 is missing because the pros play v12.6 in their spring playoffs and finals, then they play v12.8 in MSI(Mid-Season Invitational International), which means v12.7 is skipped), and for the original data, there are 16 variables for all of the data including picks and bans, wins and losses, KDA, etc. First I will make some EDAs to better understand the data and have a basic knowledge of it, then I will try to fit the model using the analysis, and finally, I will output the result and discuss the change of League of Legends in S12 from MSI till the end. I will mainly focus on the importance of each position and how to balance the strength of all champions in the future by comparing my result and what Riot does in the coming version v13.1.

III. Method

Preprocessing

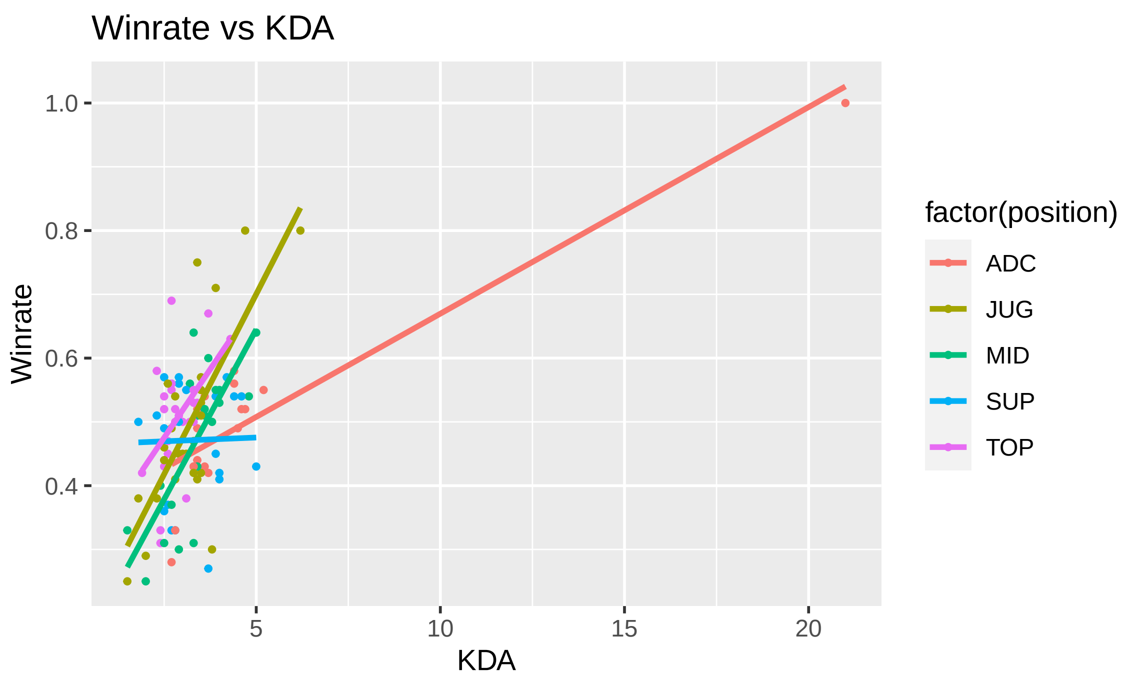
The data is from https://gol.gg/champion/list/season-S12/split-ALL/tournament-ALL/, and all of them are data in S12, but some champions have never been banned or picked, which means there are NAs in data, so I tidy the data by cleaning it. Also, to better analyze, the data, using 161 champions is too hard to group, so I add a tag for all the champions to better make visualization. Let's take a brief look at the variables.

column names	explanation
Champion	Name of the champion
Picks	Times the champion is picked
Bans	Times the champion is banned
Presence	Percentage of the champion is picked or banned in all games
Wins	Times the champion wins
Losses	Times the champion loses
Winrate	The oppurtunity the champion wins divided by total games
KDA	(Kills+Assists)/Deaths, symbols how well the champion is played
AVG.BT	Average ban turn
GT	Average game time
CSM	CS(Creep Score) per minute
DPM	Damage to champions per minute
GPM	Gold per minute
CSD.15	CS differential at 15 min
GD.15	Gold differential at 15 min
XPD.15	XP differential at 15 min

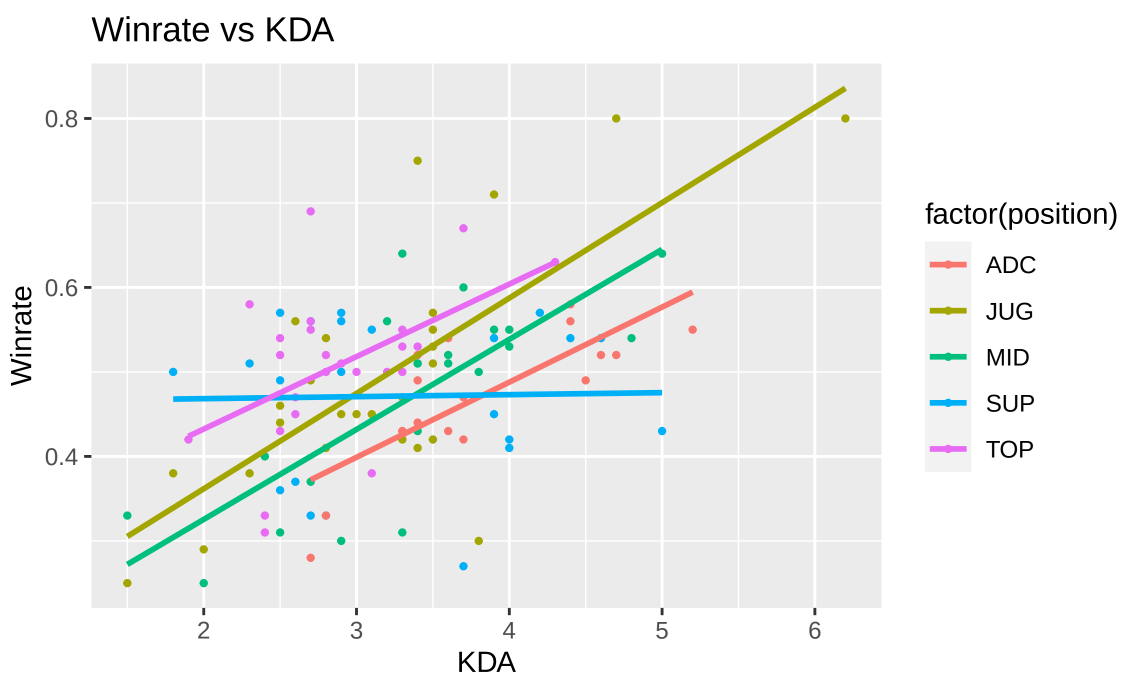
Then I tidy the data by changing those character variables to numeric ones, for example by deleting the character"%" and making the variables show decimal numbers, and I deleted those champions that presence is too low. I use picks+bans>10 as a condition.

EDA

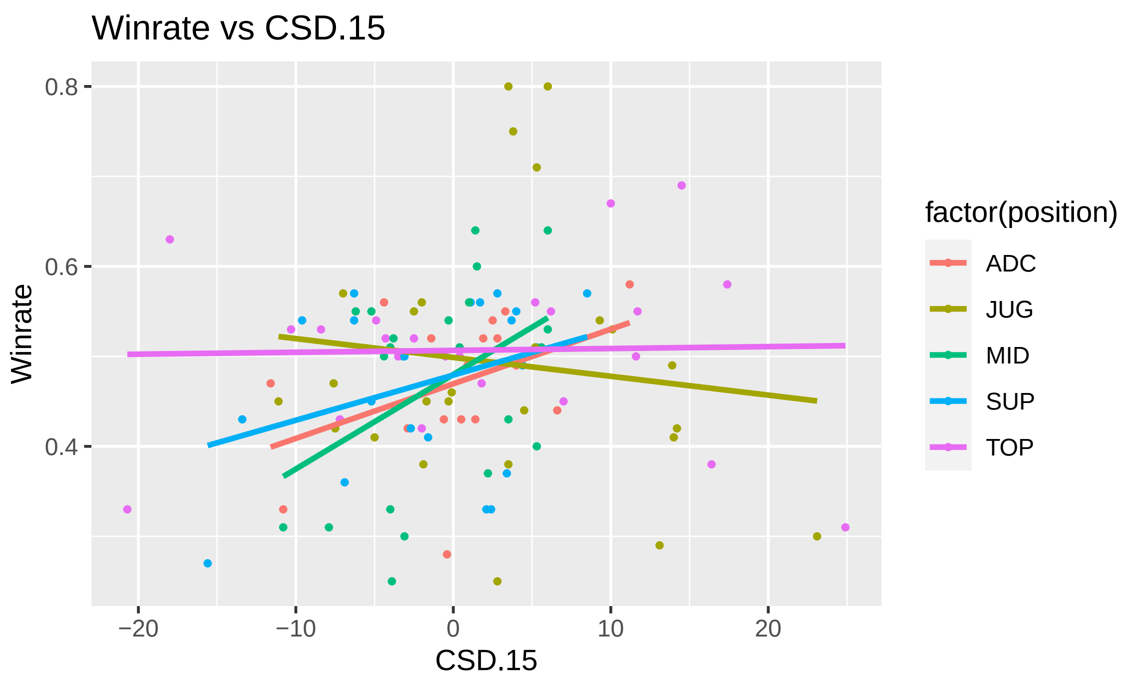
There are 16 variables but obviously, not all of them should be considered in fitting the model. So we are going to analyze the data, especially focusing on v12.8 and v12.12 because these three versions are the beginning, middle, and end of S12. After making the EDAs of v12.12, I can make a model and try to fit v12.8 in it.



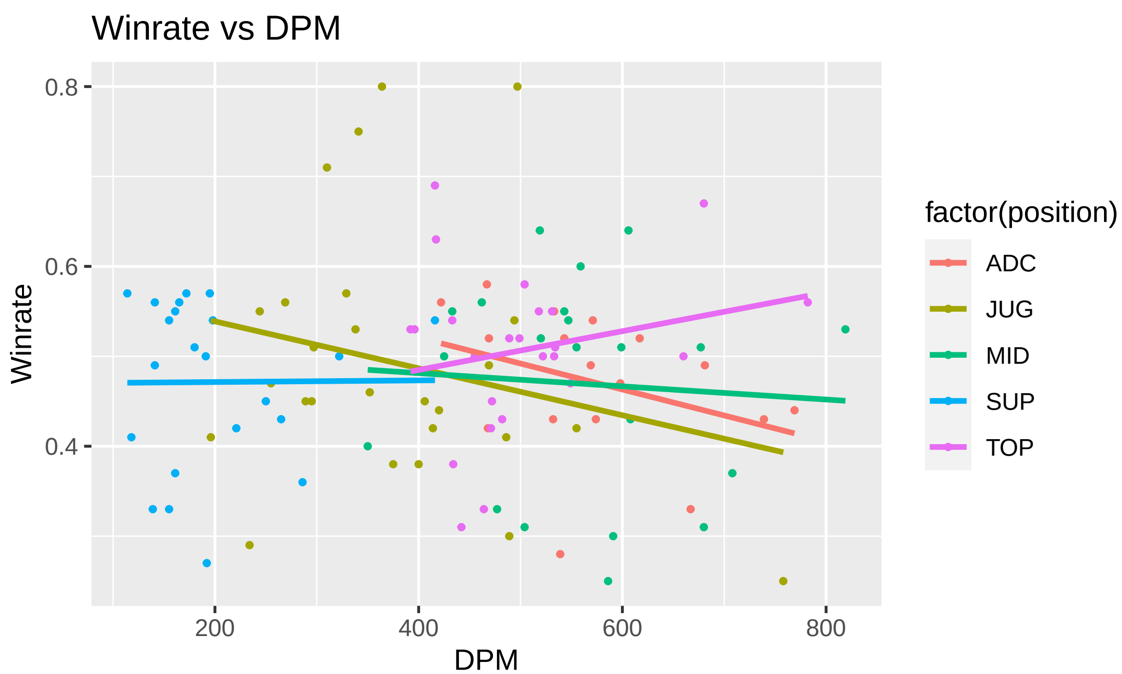
By looking at this figure, there is a strange point on the ADC line that has a 100% win rate and a relatively high KDA. By checking the data, it's a champion called "Jhin", which is picked 9 times and banned 2 times. To better visualize the data, I try to remove the data of "Jhin" and make the plot again.



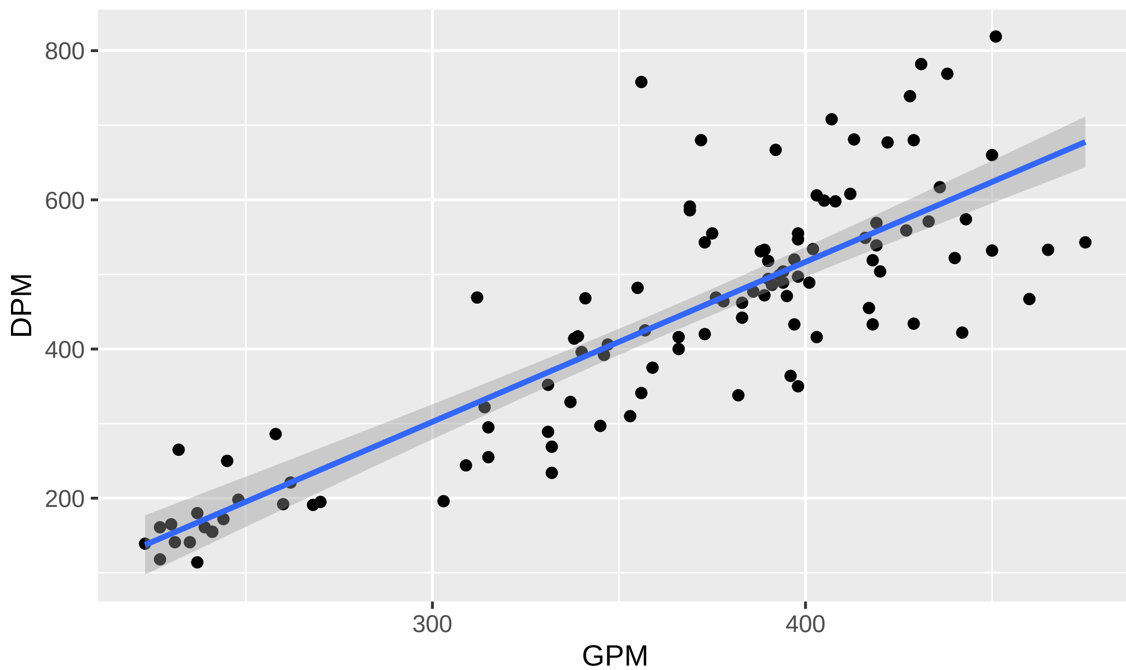
From the plot, we can see that except for SUP, the trend of the other positions is mostly the same. That's probably SUPin a game usually dies the most for the team's victory but they also have much assistance. So the KDA doesn't do much to its win rate. And for all the other positions, the lines are ordered by"TOP, JUG, MID, ADC", it's because an ADC in a team usually has the highest KDA so it doesn't reflect as much as the TOP position on win rate.

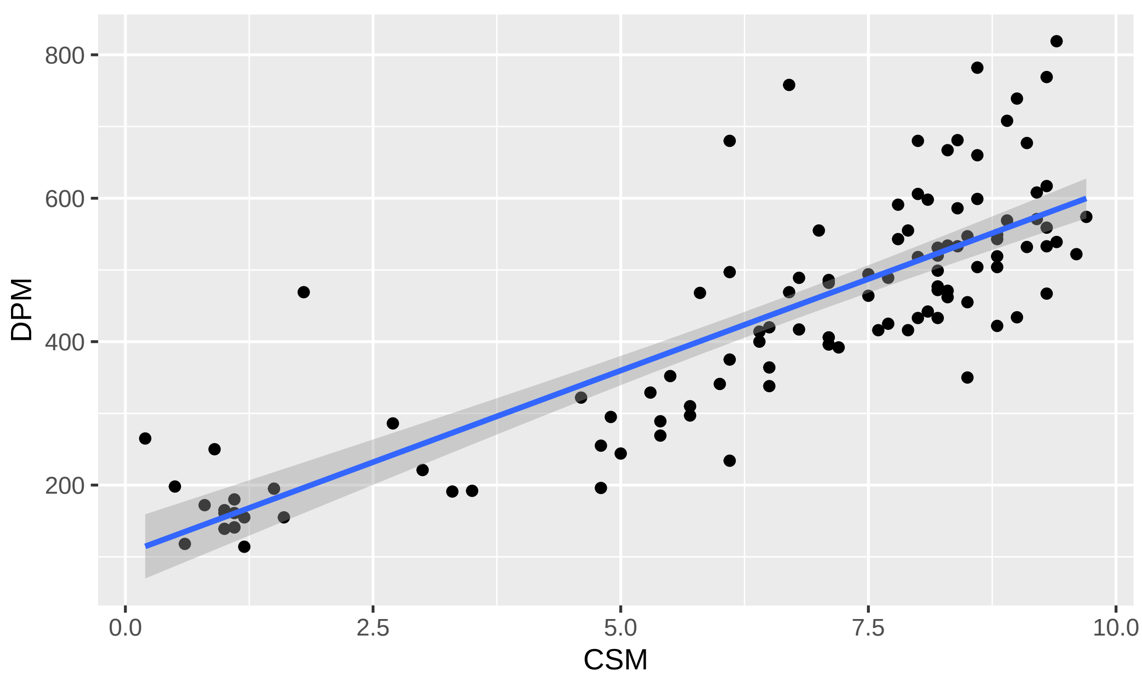


It’s an interesting graph for the position TOP and JUG, for TOP, no matter how much cs you lead your opponent, it doesn’t influence much of the game’s win or loss. It’s true because mostly the game is over 30 minutes and at that time how much your MID and ADC lead is more important. And for JUG, when you have few CS, which means you have more ganks in the first 15 minutes, your win rate is higher.



From the plot, we can know that the relation between Winrate and DPM is not linear, so we should consider more. So I decide to discover the relationship between DPM and GPM, DPM and CSM.

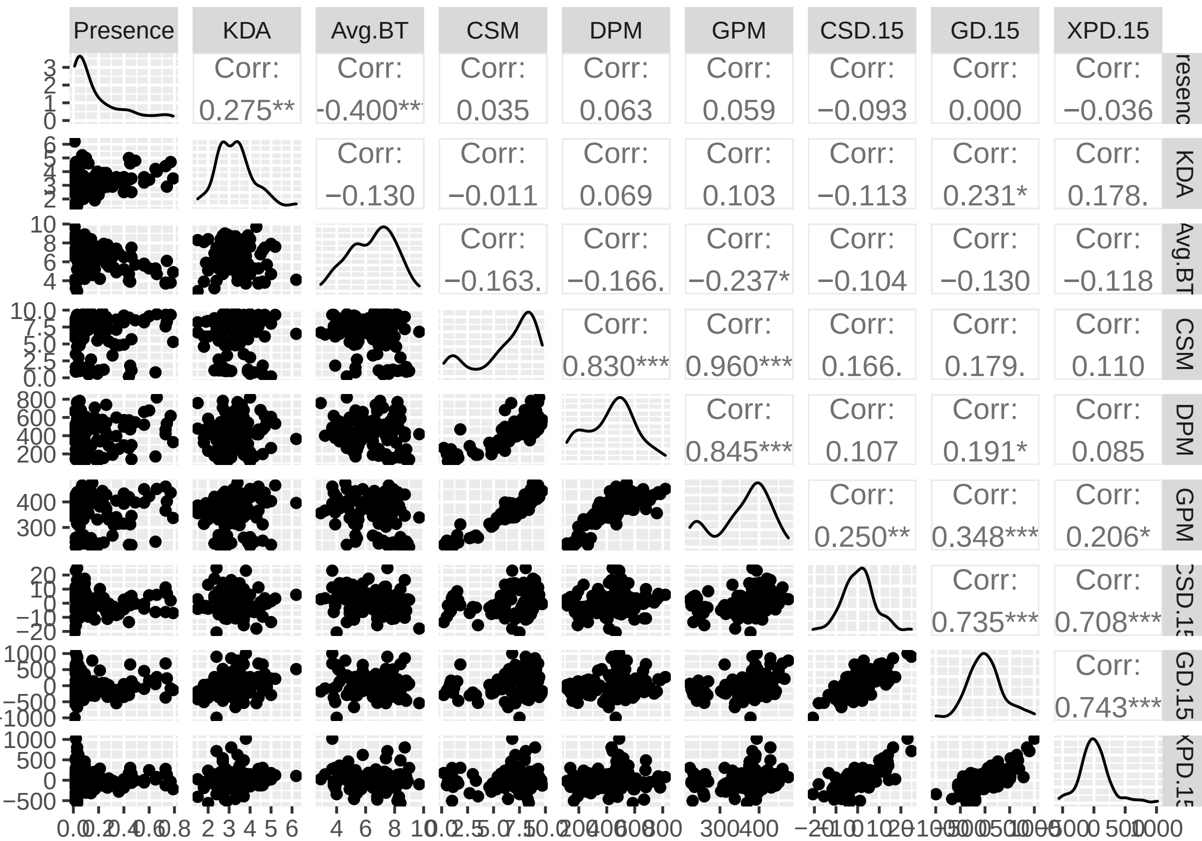




These two plots show DPM, CSM, and GPM are related and by calculating the two Pearson correlations are 0.85 and 0.83. It is reasonable since the more CS you get, the more gold you can earn and the more damage you can deal, maybe some of the support champions even have more money but all they do in fights is give shields so they don't have much improvement in damage, but those champions are just a few from all the champions so it doesn't matter much.

Model Fitting

While finding the correlation matrix, the picks and bans can be substituted by presence, and game length is not related to the winning rate so it is omitted. I reserve all the other variables and make the matrix.



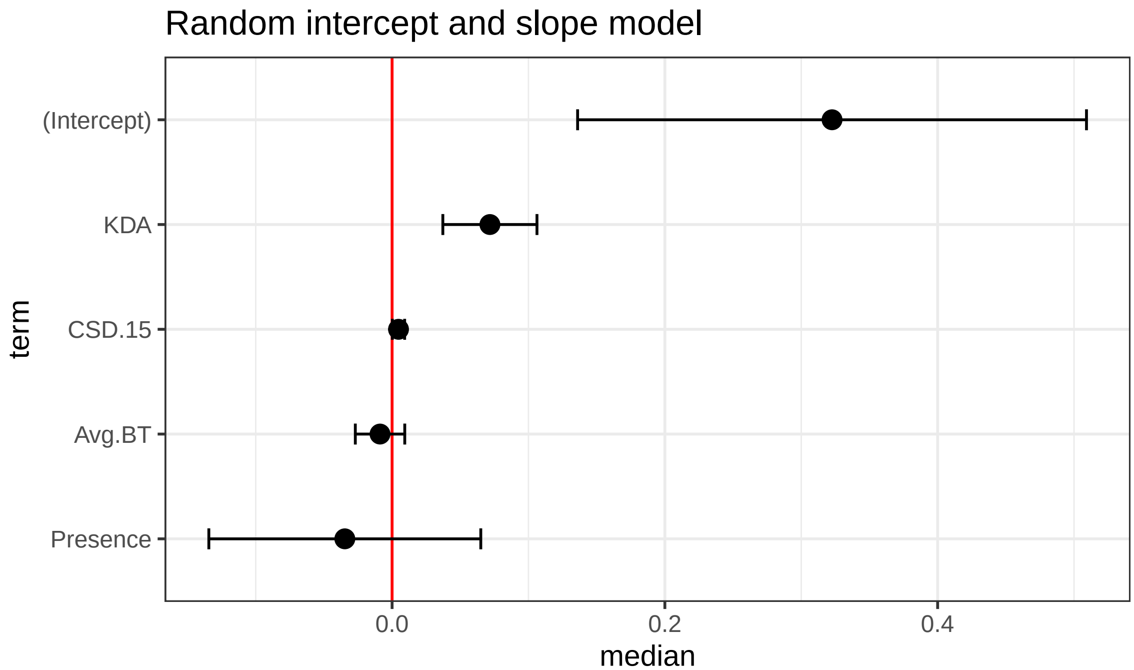
Using 0.7 as a limit, we can see that the conclusion that GPM, CSM, and DPM are related is correct. Also, CSD.15, GD.15, and XPD.15 are related. So I choose KDA, presence, and CSD.15 as the variables used in the model. Here's the model I choose:

```
model <- lmer(Winrate ~ KDA+Presence+CSD.15+Avg.BT+(1+KDA+Presence+CSD.15+Avg.BT|position),data = d12)
model
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: Winrate ~ KDA + Presence + CSD.15 + Avg.BT + (1 + KDA + Presence +
##     CSD.15 + Avg.BT | position)
##    Data: d12
## REML criterion at convergence: -198.5592
## Random effects:
##  Groups   Name        Std.Dev. Corr
##  position (Intercept) 0.167473
##           KDA         0.029133 -0.97
##           Presence    0.058557  0.17 -0.42
##           CSD.15      0.005085  0.99 -0.93  0.06
##           Avg.BT      0.017398 -0.90  0.76  0.27 -0.94
##  Residual             0.077687
## Number of obs: 110, groups:  position, 5
```

```
## Fixed Effects:
## (Intercept)          KDA     Presence       CSD.15       Avg.BT
##    0.302060     0.077013    -0.040844     0.004029    -0.008436
## optimizer (nloptwrap) convergence code: 0 (OK) ; 0 optimizer warnings; 1 lme4 warnings
```

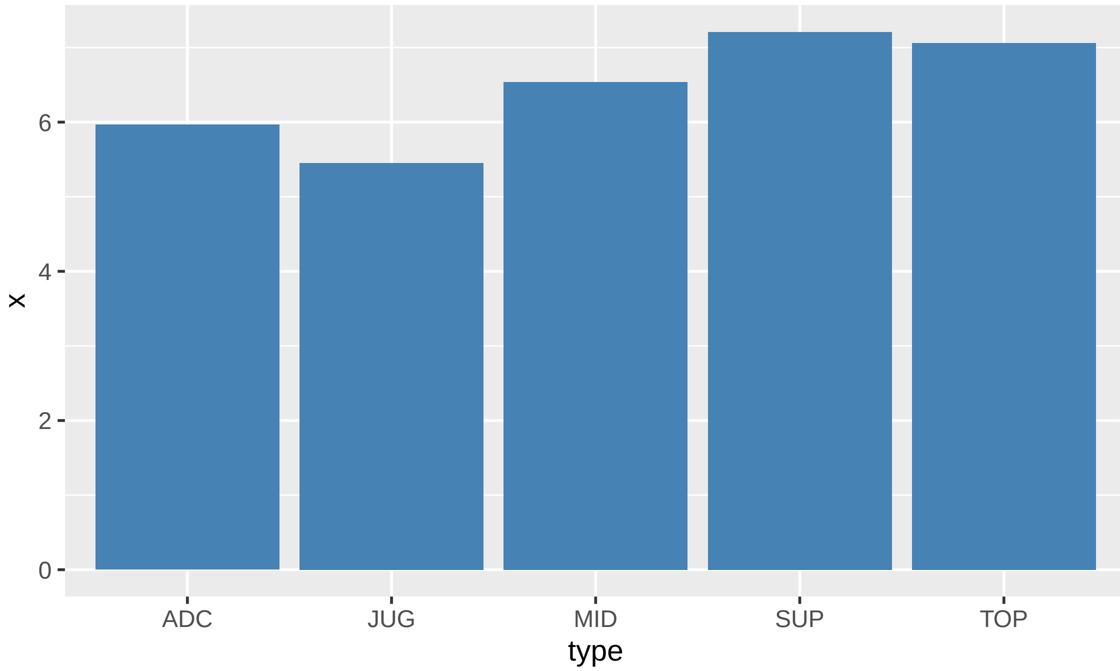
After making the model, let's take a look at its fixed effect.



We can see that CSD.15 seems to have less effect on the win rate, then we give out the random effects below, and we can see that it still has little impact on the win rate.

```
##     (Intercept)   KDA Presence CSD.15 Avg.BT
## ADC        0.09 -0.01    -0.07   0.00  -0.02
## JUG       -0.23  0.04    -0.01  -0.01   0.02
## MID        0.03  0.00    -0.03   0.00  -0.01
## SUP        0.18 -0.04     0.06   0.00  -0.01
## TOP       -0.07  0.01     0.04   0.00   0.01
```

We should know that the game's champion pick is as follows: ban 6 champions, pick 6 champions, ban 4 more and pick 4 more. So maybe there are some differences between the first 6 bans, but they are mainly for 5 and 6 because the first 4 bans are for the ops in that version, from my perspective, the first 6 bans can be considered of the same importance. We can divide ban turns into two parts, the first 6 as one group and 7-10 as another.

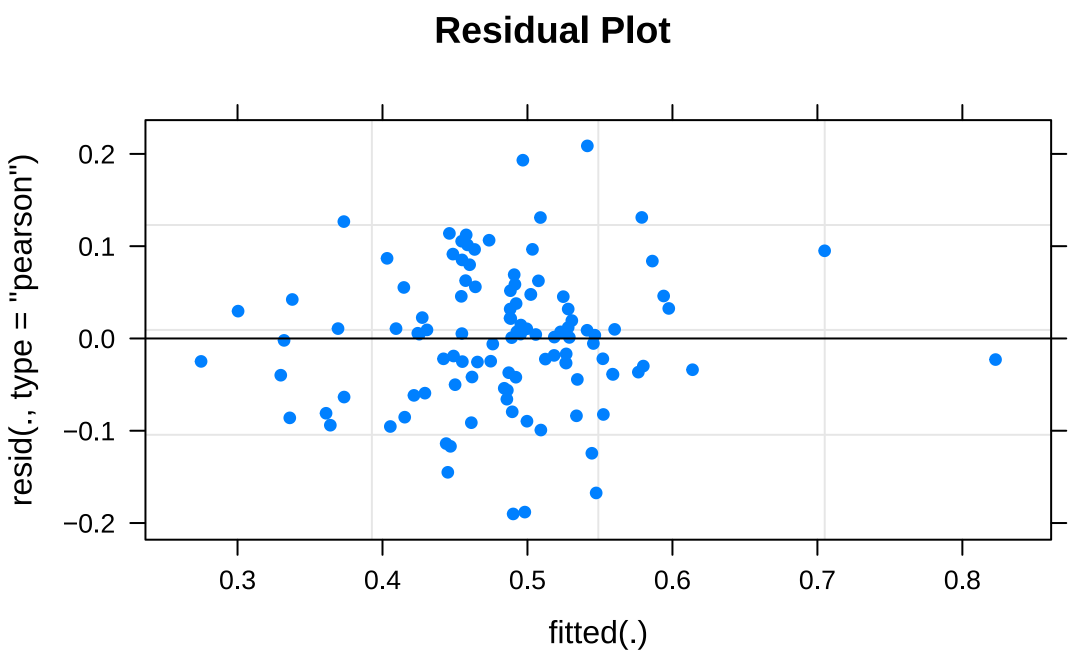


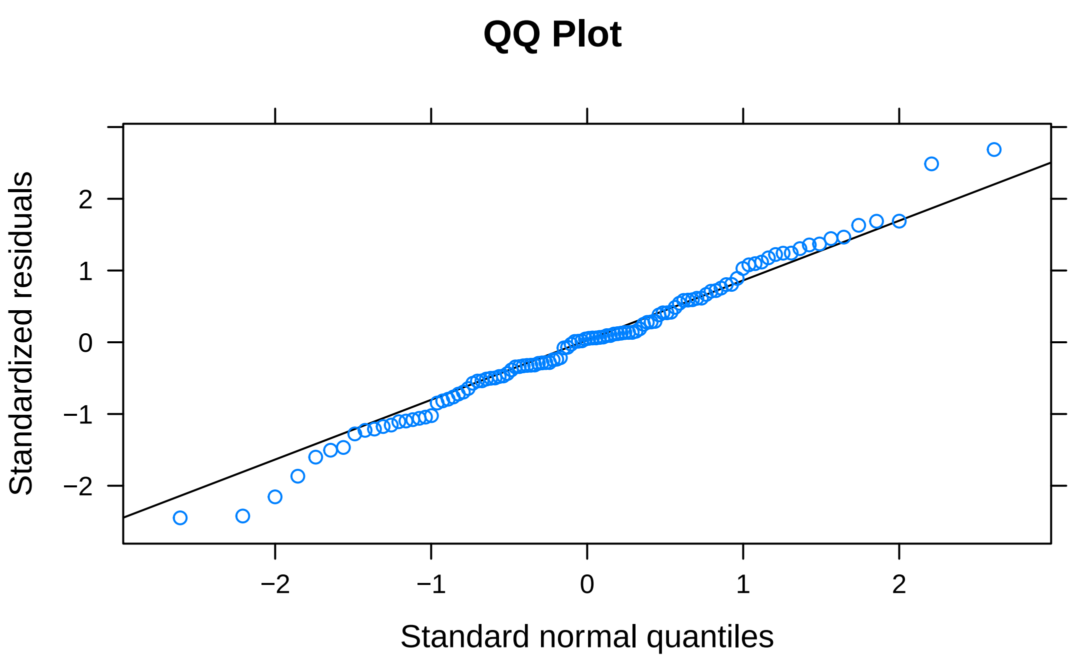
So from the data and plots, we can see that JUG is kind of important, just as mentioned that if a JUG can gank more in the first 15 minutes, the win rate will be much higher, that's probably why some op JUG is banned in very early time.

IV. Result

Checking the Model

First we can check whether the model is fit using a Residual Plot and a QQ Plot.





We can see that for the residual plot, the mean of the residuals is close to 0 and for the QQ plot, most of the points are on the line, which demonstrates that the model fits and the model checking is complete.

Comparing Models

By using the result above and trying to fit the same model by using the data v12.8, we have the results below and we can compare the results.

```
##     (Intercept)   KDA Presence CSD.15 Avg.BT
```

```
## ADC       -0.35 -0.01     0.53  -0.01   0.02
## JUG        0.13 -0.03    -0.03   0.00  -0.01
## MID        0.62 -0.07    -0.48   0.01  -0.03
## SUP       -0.22  0.02     0.22   0.00   0.01
## TOP       -0.18  0.09    -0.23   0.00   0.01
```

We can see that presence matters much more than it did in the newest version. I think there are two main reasons. First, pro plays in version 12.8 only include MSI games which mean there are only 11 teams instead of the world's teams. Also, to get the champion, most of the teams have no time to practice new champions and instead, they copy others' bans and picks leading to much higher influence in presence. The second reason is that after the MSI, Riot weaken those champions which are chosen often and make the new version more balanced, so more champions can be chosen for the teams.

V. Discussion

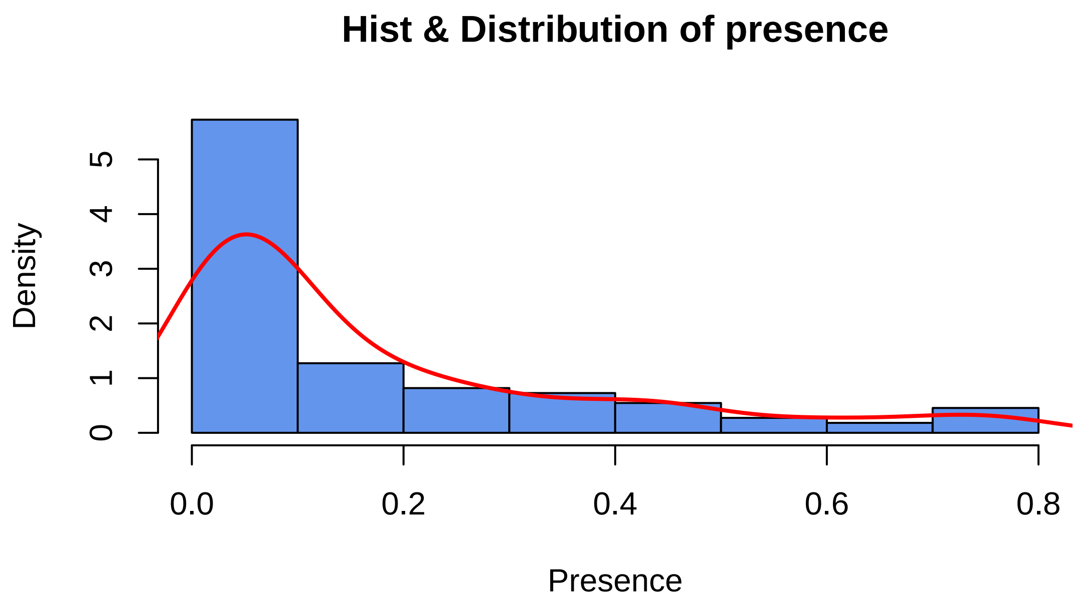
Future Prospects

In this project, I used the multilevel model to fit the data of pro play in v12.12 and compare the result with v12.8 and get the result of exactly what Riot do in the past half year. So in the future, according to the model of v12.12, Riot should decrease the impact of JUG and find a way to make TOP join in fights in the first 15 minutes. v13.1 will come out in January 2023, but the information comes out that they decrease the impact of JUG, and they make TOP tanker, I don't know whether it can successfully make TOP more powerful in the games, but we will see.

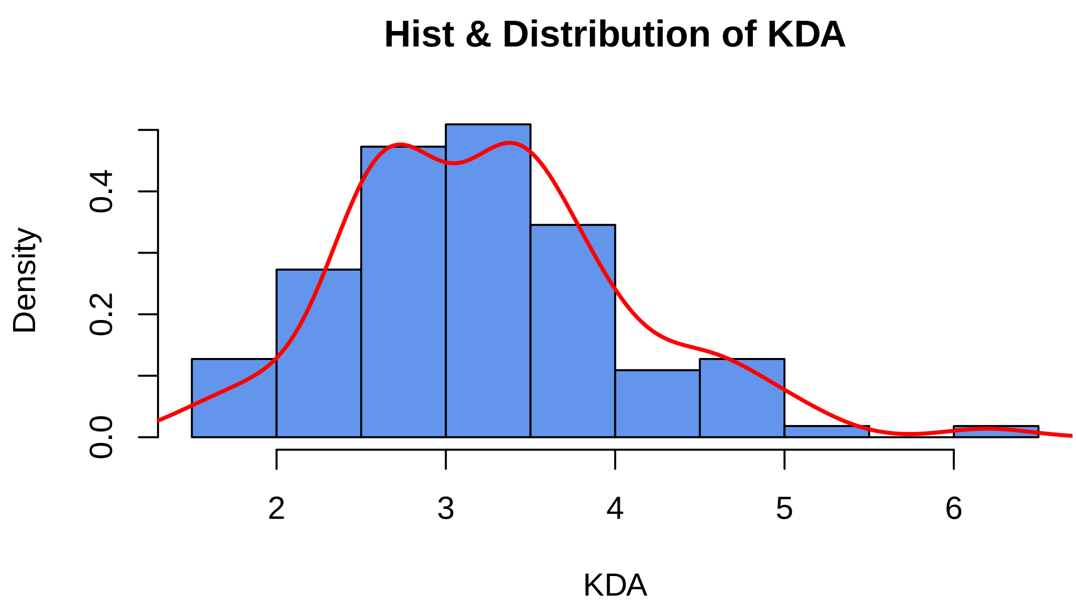
Shortage

First, some champions can be played at two positions but to the group, I always use the position that the champion appears most frequently, which may cause the data not so accurate. Second, the variables considered can choose more, for example, win rate between champions, win rate by time, etc. Then we can better analyze the data and choose the better formation. Last but not least, it's a game not only for the pros but also for ordinary people like me. By checking the data, I found that some champions are really strong in low rank and can't even appear in the pro play. So Riot not only has to find a balance in matches but also for all the players and it's a tough task.

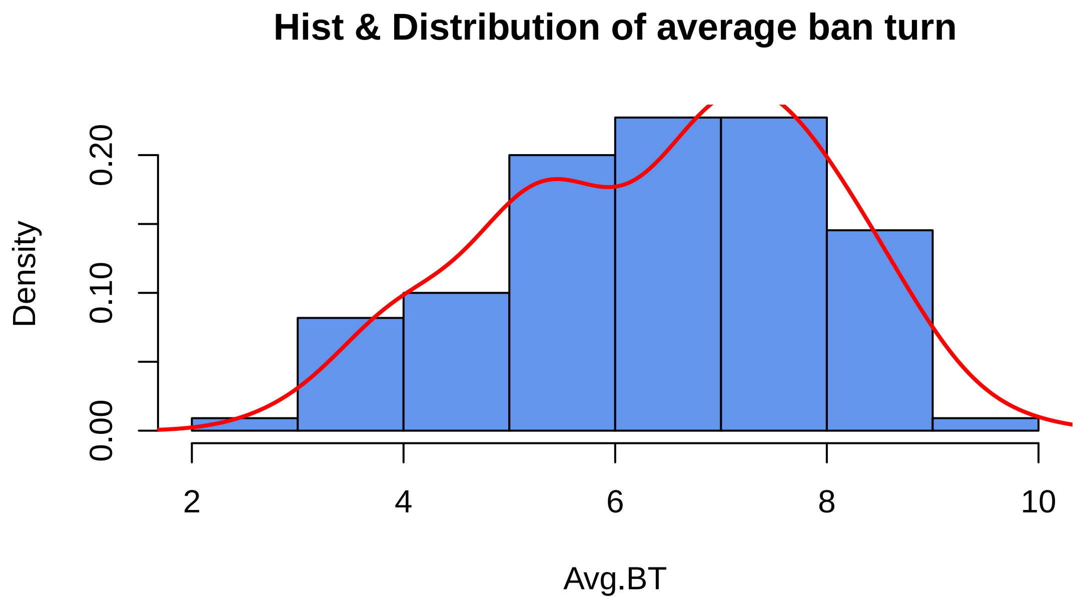
Appendix

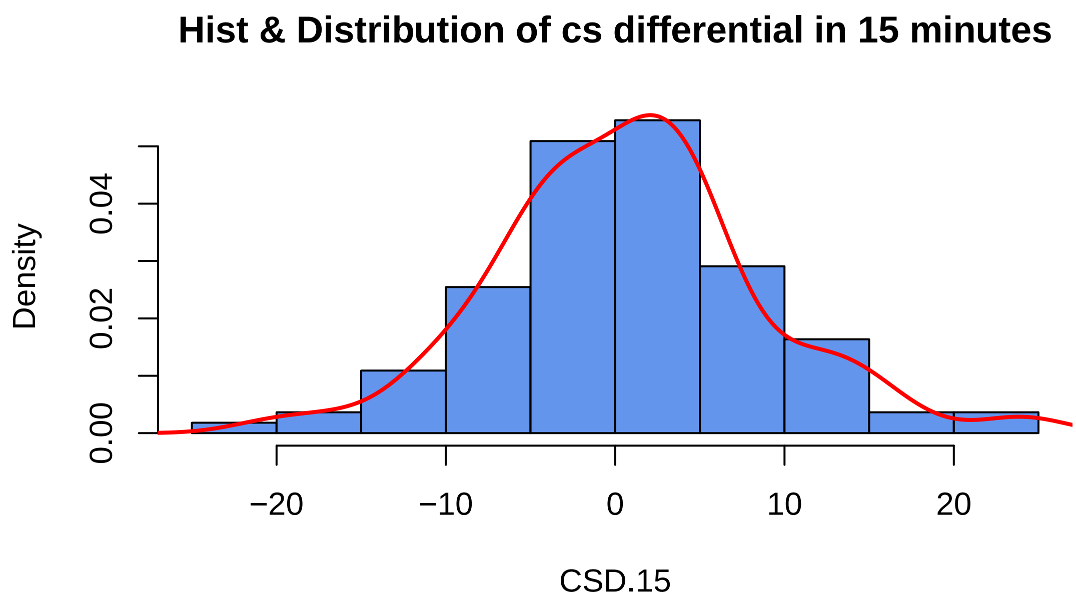


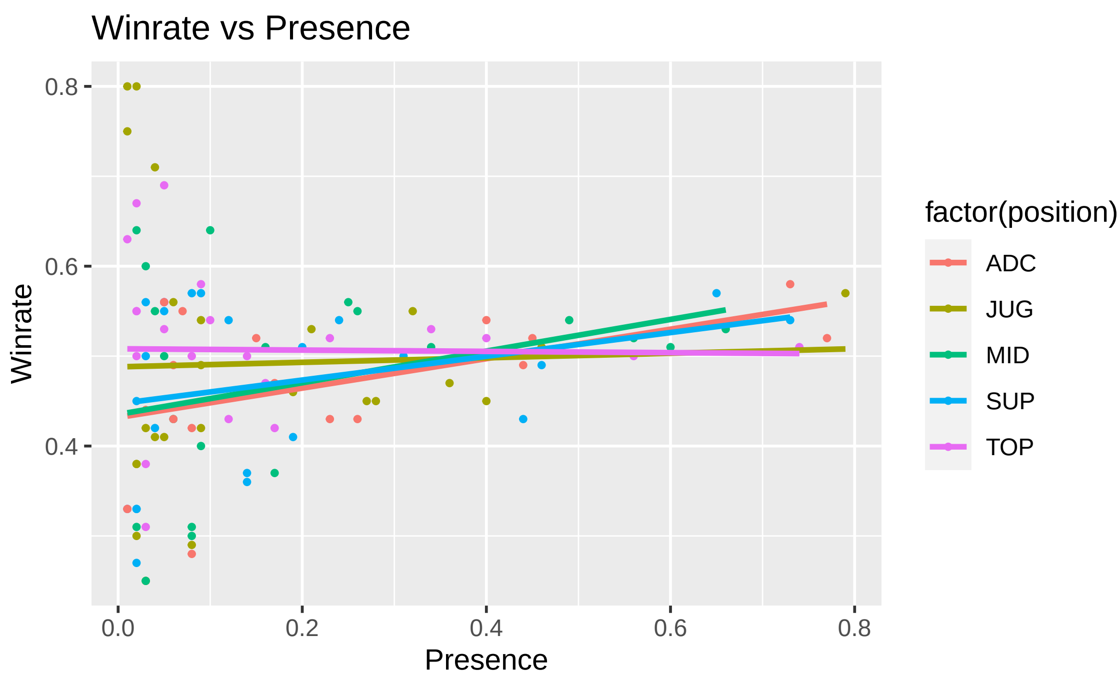
Hist & Distribution of presence
Density
0
1
2
3
4
5
0.0
0.2
0.4
0.6
0.8
Presence

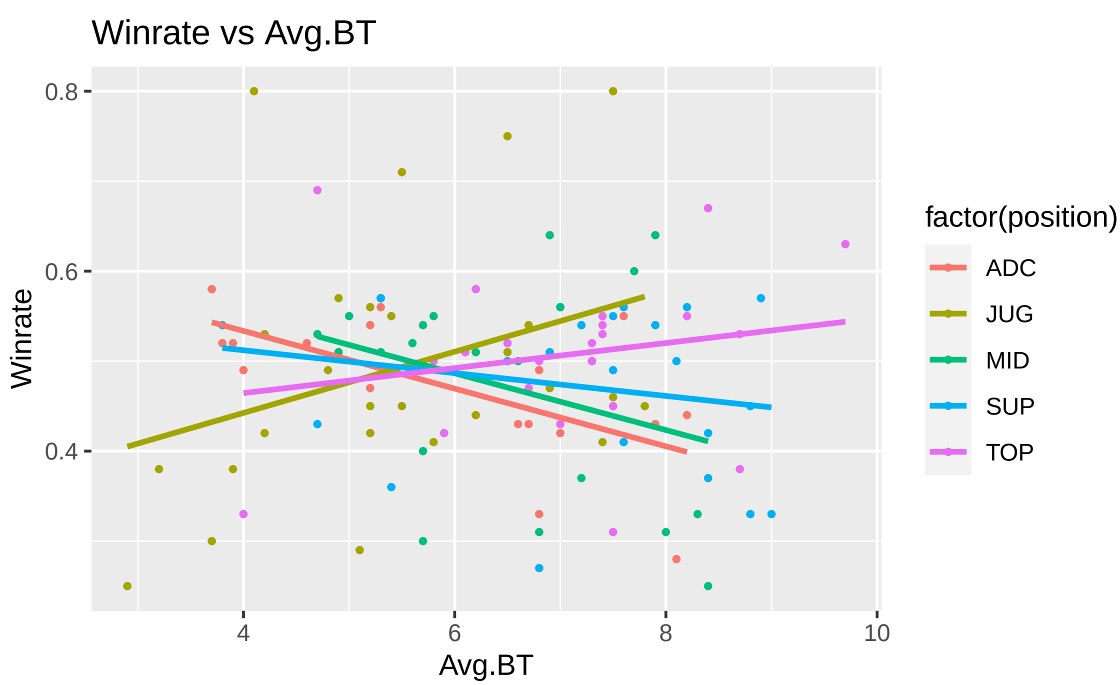


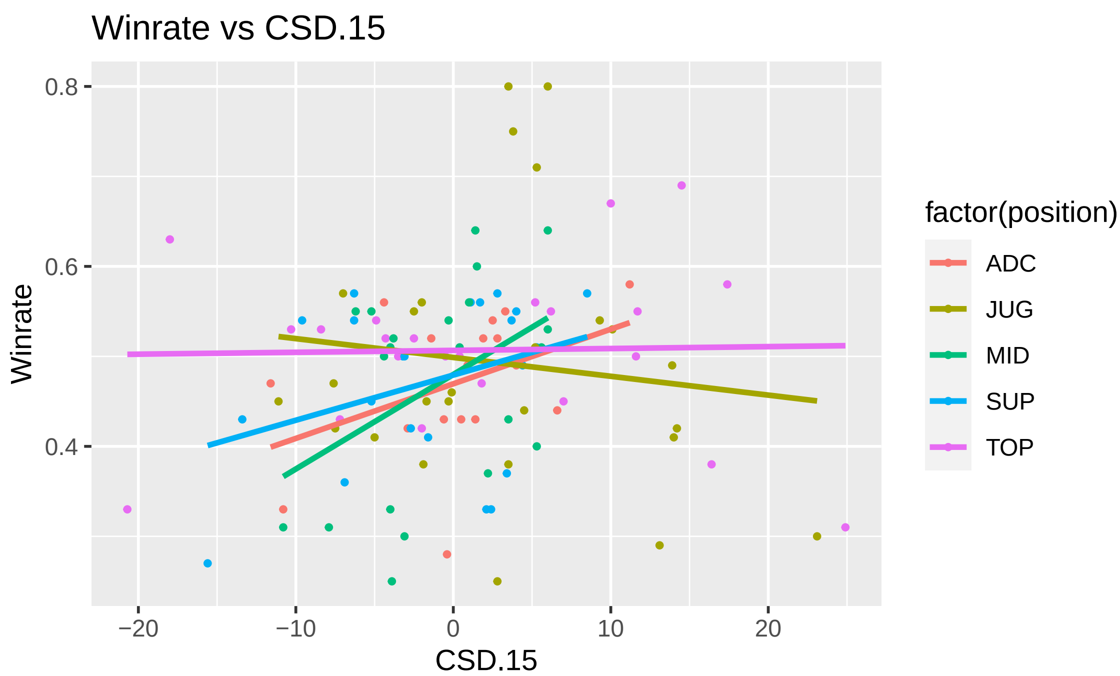
Hist & Distribution of KDA
Density
0.0
0.2
0.4
2
3
4
5
6
KDA


Hist & Distribution of average ban turn
Density
0.00
0.10
0.20
2
4
6
8
10
Avg.BT


Hist & Distribution of cs differential in 15 minutes
Density
0.00
0.02
0.04
−20
−10
0
10
20
CSD.15


Winrate vs Presence
Winrate
0.8
0.6
0.4
0.0
0.2
0.4
0.6
0.8
Presence
factor(position)
ADC
JUG
MID
SUP
TOP


Winrate vs Avg.BT
Winrate
0.8
0.6
0.4
4
6
8
10
Avg.BT
factor(position)
ADC
JUG
MID
SUP
TOP


Winrate vs CSD.15
Winrate
0.8
0.6
0.4
−20
−10
0
10
20
CSD.15
factor(position)
ADC
JUG
MID
SUP
TOP