

PROJET PROBABILITES STATISTIQUES : League of Legends

Bizot Mael, Chaudron Charles

5 janvier 2021

Projet de probabilité et de statistiques sur League of Legends

rendu : 05/01/2021 12:00

Pour cet analyse de données, nous nous sommes principalement penché sur les 1040 parties de leagues of legends et avons tout de meme utilisé le fichier Timeline d'une partie afin d'appuyer un peu plus nos analyses.

Lorsque nous avons découvert le sujet, nous avons eu un peu de mal à trouver une ligne directrice car nous avons mis du temps avant de bien saisir toutes les variables présentes dans la base de données. Puis finalement nous nous somme penché sur la variable qui selon nous est la plus importante dans une partie : la victoire ou la defaite. En effet nous avons voulu montrer grace a nos analyse, quels sont les éléments les plus décisive lors d'une partie. Et nous avons voulu savoir ce qui faisait la différence entre une victoire et une défaite.

Nos analyse portent principalement sur les Variables suivantes : -les golds -les kills -les monstre epiques tués -le nombre de mort -les champions

```
library(readr)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##   filter, lag

## The following objects are masked from 'package:base':
##
##   intersect, setdiff, setequal, union
```

```
library(questionr)
library(OneTwoSamples)
```

```
player_team_lol_data <- read_csv("C:/Users/maelb/Desktop/cours/L3/proba stat/cours lol/player_team_lol_
```

```
##
## -- Column specification -----
## cols(
```

```
## .default = col_double(),
## lane1 = col_character(),
## lane2 = col_character(),
## lane3 = col_character(),
## lane4 = col_character(),
## lane5 = col_character(),
## winTeam1 = col_character(),
## lane6 = col_character(),
## lane7 = col_character(),
## lane8 = col_character(),
## lane9 = col_character(),
## lane10 = col_character(),
## winTeam2 = col_character()
## )
## i Use 'spec()' for the full column specifications.
```

```
lol <- player_team_lol_data
```

Les Golds

Dans un premier temps, nous allons comparer les golds moyens d'une équipe qui perd, face aux golds moyen de l'équipe gagnante

```
les_wins <- filter(player_team_lol_data, winTeam1 == "Win")
les_loose <- filter(player_team_lol_data, winTeam1 == "Fail")

moyenne_gold_gagnant <- mean(les_wins$gold1)

print(moyenne_gold_gagnant)
```

```
## [1] 11166.33
```

```
moyenne_gold_perdant <- mean(les_loose$gold1)

print(moyenne_gold_perdant)
```

```
## [1] 9696.993
```

Ici on peut constater que l'équipe perdante a environ 2000 golds de moins que l'équipe gagnante. Cela nous donne donc un première indice les critères de victoire. Ici on constate que les golds sont important.

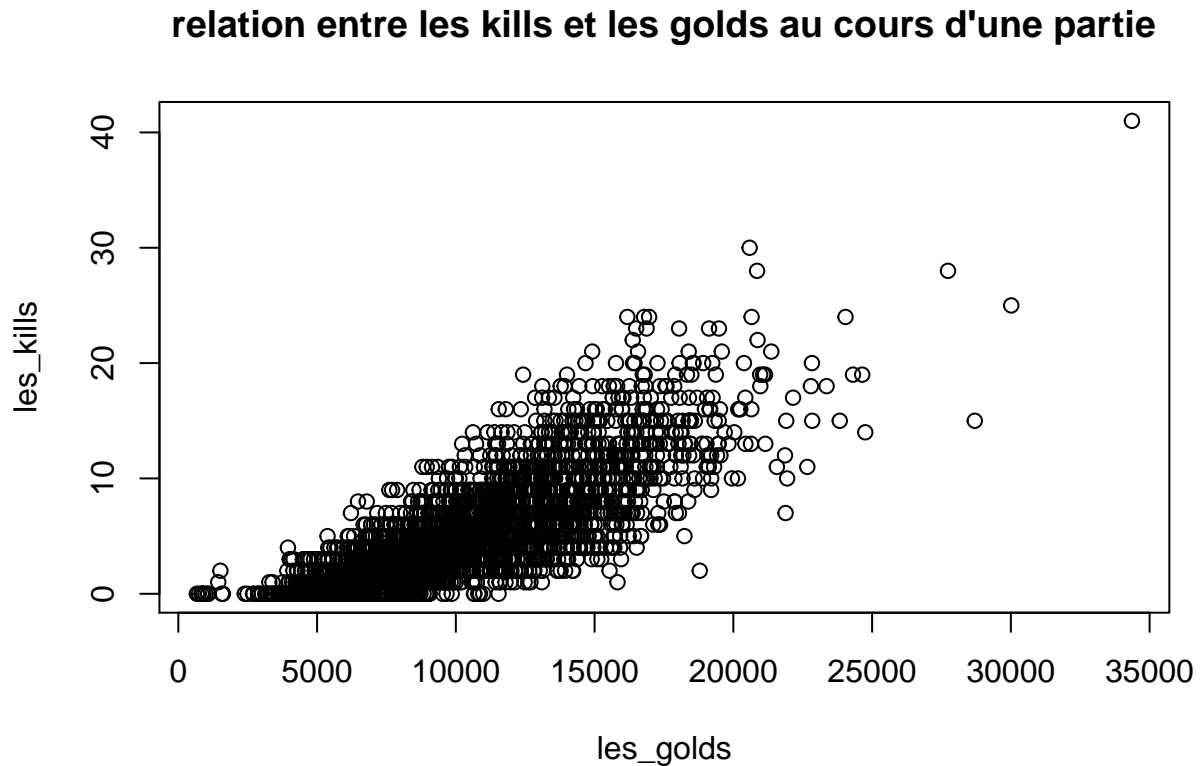
Corrélation entre les golds et les kills

Par la suite nous avons voulu savoir comment etait obtenu ces golds. Nous avons donc decider de comparer les golds avec les kills. Pour cela nous avons regarder la potentiel corrélation entre ces deux variables.

Ci-dessous, nous avons pris un échantillon de l'équipe 1. Nous avons mis les golds gagné de 3 des joueurs dans une variable et leur kills respectif afin de pouvoir calculer le coefficient de person et les afficher à l'aide d'un plot.

```
les_golds <- c(lol$gold1, lol$gold2, lol$gold3)
les_kills <- c(lol$kills1, lol$kills2, lol$kills3)
```

```
relation_gold_kills <- plot(les_golds, les_kills, main = "relation entre les kills et les golds au cours d'une partie")
```



```
print(relation_gold_kills)
```

```
## NULL
```

```
correlation_gold_kills <- cor(les_golds, les_kills)
print(correlation_gold_kills)
```

```
## [1] 0.7854359
```

Sur le graphique, nous constatons bien que le nombre de golds augmentent en fonction du nombre de kills. De plus, le coefficient de Pearson étant proche de 1 nous pouvons valider cette corrélation. Pour synthétiser nos deux analyses précédentes, nous pouvons dire que les kills sont assez décisifs dans le cours d'une partie car ils apportent des golds et les golds sont généralement plus élevés chez l'équipe gagnante.

Gold au sein d'une équipe et taux de victoire

afin de confirmer cette hypothèse et montrer que LoL est un jeu d'équipe, nous allons observer le taux de victoire en fonction du nombre de personnes ayant atteint un certain nombre de golds. Car en effet, à partir

d'un certain nombre de gold le champion que le joueurs joue devient assez puissant pour déterminer le cours d'une partie. Afin d'appuyer nos hypothèses, nous avons re-pris les golds des joueurs 1, 2 et 3.

```
bcp_gold <-(filter(lol, gold1 > 10000))  
freq(bcp_gold$winTeam1)
```

```
##          n    % val%  
## Fail 228 44    44  
## Win  290 56    56
```

```
bcp_gold2 <-(filter(lol, gold2 > 10000 & gold1 > 10000))  
freq(bcp_gold2$winTeam1)
```

```
##          n    % val%  
## Fail 151 41.9 41.9  
## Win  209 58.1 58.1
```

```
bcp_gold3 <-(filter(lol, gold2 > 10000 & gold1 > 10000 & gold3 > 10000))  
freq(bcp_gold3$winTeam1)
```

```
##          n    % val%  
## Fail 110 40.7 40.7  
## Win  160 59.3 59.3
```

```
bcp_gold4 <-(filter(lol, gold2 > 10000 & gold1 > 10000 & gold3 > 10000 & gold4 > 10000))  
freq(bcp_gold4$winTeam1)
```

```
##          n    % val%  
## Fail  89 40.8 40.8  
## Win  129 59.2 59.2
```

Grâce à ces fréquences nous pouvons donc affirmer que le taux de victoire augmente si le nombre de joueurs ayant atteint 10K golds au cours d'une partie augmente aussi. Mais ce taux de victoire tend vers 60%. Ce phénomène peut s'expliquer par le fait que si la plupart des joueurs de l'équipe 1 ont atteint ce montant, il est aussi probable que plusieurs membre de l'équipe adverse l'aient atteint aussi, ce qui "équilibre" un peu la balance.

Impact des Monstres épiques

Après nous être intéressé à ces deux variables, les golds et les kills, nous avons voulu voir l'impact qu'avait les monstres épiques sur le cours d'une partie. Les deux monstres épiques présent sur la faille de l'invocateur sont le Baron nashor et le Dragon. Il faut savoir que ces deux monstres réapparaissent quelque minutes après avoir été tué et c'est donc pour cela qu'une équipe peut en avoir tué plusieurs.

Encore une fois, nous nous sommes penché sur le taux de victoire en fonction du nombre de Baron Nashor tué par une équipe.

Nous avons donc mis dans une table les victoires de l'équipe 1 et leur nombre de baron tué.

```
tab_baron <- table(lol$winTeam1, lol$baronKillsTeam1)

cprop(tab_baron)
```

```
##
##      0      1      2      3      Ensemble
## Fail  66.8  21.0  25.0  33.3  52.3
## Win   33.2  79.0  75.0  66.7  47.7
## Total 100.0 100.0 100.0 100.0 100.0
```

Grâce aux pourcentages que nous observons, nous pouvons conclure que le Baron nashor est un atout majeur pour la victoire. Nous avons voulu savoir ce qui le rendait important. Nous avons donc décidé de voir si il donné des golds à l'équipe qui le tué.

Baron nashor et Golds

Afin de vérifier ces informations, nous allons utiliser le fichier qui détaille toutes les actions d'une partie (fichier Timeline). La partie que nous allons analyser a duré 27 minutes.

En observant les différents événements qui se sont déroulés au cours de cette partie, nous avons trouvé qu'un Baron nashor a été tué à 25 minutes de jeux. Nous avons par la suite cherché qui était le tueur du nashor.

```
timeline <- read_csv("C:/Users/maelb/Desktop/cours/L3/proba stat/cours lol/timeline.csv")
```

```
##
## -- Column specification -----
## cols(
##   .default = col_double(),
##   'events/0/type' = col_character(),
##   'events/1/type' = col_character(),
##   'events/2/type' = col_character(),
##   'events/3/type' = col_character(),
##   'events/4/type' = col_character(),
##   'events/5/type' = col_character(),
##   'events/6/type' = col_character(),
##   'events/7/type' = col_character(),
##   'events/8/type' = col_character(),
##   'events/9/type' = col_character(),
##   'events/10/type' = col_character(),
##   'events/11/type' = col_character(),
##   'events/11/levelUpType' = col_character(),
##   'events/12/type' = col_character(),
##   'events/13/type' = col_character(),
##   'events/14/type' = col_character(),
##   'events/15/type' = col_character(),
##   'events/15/levelUpType' = col_character(),
##   'events/16/type' = col_character(),
##   'events/17/type' = col_character()
##   # ... with 150 more columns
## )
## i Use 'spec()' for the full column specifications.
```

```
print(timeline$`events/20/monsterType`)
```

```
## [1] NA NA NA NA NA
## [6] NA NA NA NA NA
## [11] NA NA NA NA NA
## [16] NA NA NA NA NA
## [21] NA NA NA NA "BARON_NASHOR"
## [26] NA NA NA NA NA
```

```
print(timeline$`events/20/killerId`)
```

```
## [1] NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA 4
## [26] NA NA
```

```
print(timeline$`participantFrames/4/totalGold`)
```

```
## [1] 500 500 577 948 1281 1571 1960 2497 3426 3632 3950 4311
## [13] 4822 5145 5828 6097 6557 6926 7470 7778 8187 8489 8807 9611
## [25] 10519 11104 11464
```

Puis en second lieux, nous avons créés deux tables comportant les golds de l'équipe ayant tué le nash avant, puis une autre comptant les golds 1 min après le kills du nash afin de voir si le montant a augmenté ou non.

```
gold_avant_nash = c(9676,9206,12715,10519,6682)
```

```
gold_après_nash = c(10585,9562,13785,11104,6983)
```

Les deux échantillons sont appariés car ce sont les mêmes individus qui tuent le barons nashor

```
t.test(gold_avant_nash, gold_après_nash, paired = T, alternative = "less")$p.value
```

```
## [1] 0.006488612
```

La P-value étant très faible, on peut affirmer que le fait de tuer un baron nashor donne un avantage en gold.

Dragons et taux de victoire

Afin de continuer dans l'analyse de l'impact des monstres épiques, nous avons voulu nous tourner vers le dragon. En effet, le dragon donne des stats Bonus en % à l'équipe qui la tue et nous voulions donc vérifier si ces stats bonus sont essentiels pour gagner une partie.

Pour ce faire, nous avons décidé d'effectuer un test d'hypothèse : On veut affirmer avec un faible risque de se tromper que l'équipe qui a tué des dragons a plus de chance de gagner

```
var_wins = var(les_wins$dragonKillsTeam1)
var_loose = var(les_wins$dragonKillsTeam2)
```

```
mean_test2(les_wins$dragonKillsTeam1, les_wins$dragonKillsTeam2, sigma = c(var_wins, var_loose), side =
```

```
## [1] 1
```

Au vu de la valeur du P-Value, nous pouvons valider notre hypothèse. Et confirmer que l'équipe ayant tué le plus de dragon a un taux de victoire plus élevé.

On peut d'ailleurs le constater aussi grâce à cette table qui regroupe les pourcentage de victoire en fonction du nombre de dragons tués.

```
tab_2 <- table(lol$winTeam1, lol$dragonKillsTeam1)

cprop(tab_2)
```

```
##
##      0      1      2      3      4      5      6      Ensemble
## Fail  74.0  63.8  36.5  24.1  11.9   0.0   0.0  52.3
## Win   26.0  36.2  63.5  75.9  88.1 100.0 100.0  47.7
## Total 100.0 100.0 100.0 100.0 100.0 100.0 100.0 100.0
```

afin de conclure sur l'importance des dragons sur LoL nous avons voulu démontrer par une loi de probabilité que le dragon permet de d'en gagner une plus facilement en effet j'ai reuni tout les parties ou les 5 joueurs de l'équipe 1 avait plus de gold que l'équipe 2 si on prend parmi tout ces matchs, quel est la probabilité d'avoir au moins 2 dragons tué par l'équipe 1

```
test_loi <- filter(lol, gold1 > gold6, gold2 > gold7, gold3 > gold8, gold4 > gold9, gold5 > gold10)

table_dragon_tue_equipe_gagnante <- table(test_loi$dragonKillsTeam1)
freq(table_dragon_tue_equipe_gagnante)
```

```
##      n      % val%
## 0 35 32.1 32.1
## 1 19 17.4 17.4
## 2 31 28.4 28.4
## 3 21 19.3 19.3
## 4  3  2.8  2.8
```

ici on peut voir que la fréquence de 2 dragons tué est de 28.4% le pourcentage de chance est donc de 0.28 nous allons donc calculer la probabilité de trouvé un match ou l'équipe 1 a tué 2 dragons ici nous tirons au sort parmi les match ou l'équipe 1 a plus de gold que l'équipe 2 jusqu'à ce que nous obtenions un match.

```
1-dgeom(3, 0.28)
```

```
## [1] 0.8954906
```

ce test signifie que la probabilité de tirer au sort un match ou l'équipe 1 a tué deux dragons après 3 relance est deux 89%. autrement dit, la probabilité est très élevé car les équipes remportant les matchs ont tendance a plus tuer les drakes. De plus, on constate dans le jeux de données crée "test_loi", lorsque tout les joueurs d'une équipe on leur golds supérieur à leur opposants respectif, le taux de victoire est de 100 pour l'équipe ayant le plus de gold !

Choix du champion et taux de victoire

Pour la prochaine analyse, nous avons décidé de nous intéresser au taux de victoire en fonction de la présence de certain champions. En effet, certain champions apparaissent plus souvent que d'autre dans le jeux de données et nous voulions savoir si ces champions avaient un impact positif ou négatif. Nous avons donc regardé les ID de trois champions fréquemment joué grâce à cette commande

```
champions_joue <- table(lol$championID1)
```

nous avons ensuite classé par ordre croissant et répété l'action jusqu'au ChampionID5. Nous avons remarqué que les champions les plus fréquemment joué sont le numéro 236, 22, 202

nous avons donc réunis le nombre de win et de loose de ces trois champions dans trois tableau différents afin de pouvoir en extraire les chiffres et créer une matrice. Le champion et la victoire/defaite sont deux critère qualitatif, c'est pour cela que nous avons décider d'effectuer ce test d'indépendance

```
champion_1 <-(filter(lol, championID1 == 236 | championID2 == 236 | championID3 == 30 | championID4 == 236))
win_champions_1<- table(champion_1$winTeam1)

champion_2 <-(filter(lol, championID1 == 22 | championID2 == 22 | championID3 == 22 | championID4 == 22))
win_champions_2 <- table(champion_2$winTeam1)

champion_3 <-(filter(lol, championID1 == 202 | championID2 == 202 | championID3 == 202 | championID4 == 202))
win_champions_3 <- table(champion_3$winTeam1)
```

Une fois ces tableaux créés nous avons pu extraire les chiffres pour en créer une matrice. Puis nous allons effectuer

```
Independance_champion_victoire = matrix(c(64, 66, 76, 46, 50, 54), nrow = 2, byrow = T)
chisq.test(Independance_champion_victoire)$p.value
```

```
## [1] 0.9665111
```

ce résultat nous permet de pouvoir affirmer que les champions choisis et le taux de victoire/defaite sont dépendants. On peut affirmer que le choix du champions influe aussi sur le taux de victoire.

Nombre de mort et taux de victoire

Afin de compléter notre dernière analyse, nous avons décider de nous intéresser aux nombre de mort dans une équipe afin de voir si eux aussi ont un impact négatif pour une équipe.

Nous avons donc créé une table comportant les victoires et défaites d'une équipe ayant 5 joueurs avec au moins 5 morts chacun et au moins 1 des joueurs en face à - de 5 morts. Cela permet d'obtenir une table avec un taux de victoire très bas pour l'équipe avec beaucoup de deaths.


```
bcp_mort <- filter(lol, deaths1 > 5, deaths2 > 5, deaths3 > 5, deaths4 > 5, deaths5 > 5, deaths6 < 5)

tab_win_mort = table(bcp_mort$winTeam1)

print(tab_win_mort)
```

```
##
## Fail Win
## 47 2
```

On va donc chercher a savoir quel est la probabilité que l'équipe ayant beacoup de morts gagne pas plus de 5 match :

```
ppois(5, 2)
```

```
## [1] 0.9834364
```

On constate ici que la probabilité est hautement élevé, cela nous confire alors que le nombre de morts lors d'une partie est important, et que le fait de ne pas mourir beacoup l'est tout autant.

Conclusion

Pour conclure, nous avons pu constater après chaque analyse que chaque critère que nous avons traité a sont importances au cours d'une partie : que ce soit les dragons tuées, les nashor tuées, le nombre de morts, le nombre de kills etc... Malgré tout, nous avons pu relever que les golds sont un des critères majeur pour gagner une partie. Car, si un joueur a plus de gold, il pourra s'acheter plus d'item, il sera donc plus puissant dans la partie, il pourra donc tuer plus facilement ses adversaires et pourra donc aller plus facilement détruire les batiments ennemies !

Bilan (Charles)

Au départ, le sujet ne me parlait pas vraiment car je n'avait jamais joué à league of legends et j'avais du mal à saisir le fonctionnement du jeux. Cela a donc été un peu difficile de me lancer dans le projet. Cependant, au fur et a mesure de l'avancement je me suis mis à m'approprier les données et à comprendre de mieux en mieux le jeux. Le projet était assez intéressant mais j'ai trouvé le fichier timeline assez compliqué à comprendre ce qui est assez dommage surtout que nous n'étions pas en presentiel et que cela rend les explications plus compliqué...

Concernant la répartition des taches , j'ai effectué les analyse de stastiques descriptives. Mon but était de nous ouvrir quelque piste afin de répondre à notre problématique principal.

Bilan (Mael)

Pour ma part, lorsque j'ai appris que nous allions travailler sur league of legends j'ai été vraiment surpris et ravie à la fois car LoL est un de mes jeux préférés. Je connais très bien le jeux et il a été facile pour moi de comprendre les données etc car je connaissais déjà tout le vocabulaire anglais lié à ce jeux. Cependant, étant un joueur habitué et expérimenté je me suis parfois demandé si les analyses fournit était trop "simple" et evidente. Malgré tout, le projet était très intéressant et assez concret. De plus, les jeux de données étaient très complet et permettaient une assez grande liberté dans le choix de la problématique. Une difficulté aussi

que j'ai eu pas mal de fois au cours du projet, c'est la fonction filter qui ne fonctionnait pas, j'ai mis pas mal de temps à résoudre ce problème et ça m'a pas mal bloqué dans l'avancement car la fonction est très utile.

Pour ce qui est du travail fourni, je me suis occupé des test d'hypothèses et des analyse statistiques par l'intermédiaire de lois de probabilités.