Python-to-R-Apex

djamesw

2024-03-13

Overview

The data set used for this analysis is based on the game Apex Legends. I collected the data myself, over the course of 30+ weeks using the stats provided by the game at the end of every match. The same three players are the only three players in the data. Only games where we placed top 5 are included. Since I collected it myself, there should be no need for any pre-processing. The goal is get an intermediate analysis from our games to see what works best and what may not work the best.

Evaluation may be used using regression models or tree based methods.

Explanation of the variables in the dataset:

- djdmg, popdmg, spoondmg: The amount of damage each user accumulated during the match
- djkill, popkill, spoonkill: The amount of kills each user accumulated during the match
- location: The map the match was played on. There are three maps used in this rotation; World's Edge (WE), Olympus (OL), and Kings Canyon (KC)
- dlegend, plegend, slegend: The legend each user used during that match
- placing: The placement we received at the end of the match. This number is 1-5
- date: Calendar date the match was played on
- day: Calendar day the match was played on
- mode: The game mode the match was played on. This is either ranked or pub (public)

Load data into R

```
data <- read.csv("apex2022.csv")

# dropping columns
data <- data[, !(names(data) %in% c('Time.of.Day', 'Week', 'X'))]

# lowercase column names
names(data) <- tolower(names(data))

# standardize mode
data$mode <- tolower(trimws(data$mode))

head(data)
```

```
##   djdmg djkill popdmg popkill spoondmg spoonkill location    dlegend  plegend
## 1   738      2    670       0      269         1       WE Pathfinder Lifeline
```

```
## 2  1158       4   1701       4      833          4       WE Pathfinder Lifeline
## 3   896       0   1117       5     1014          3       WE     Wraith     Loba
## 4   367       1    931       4      680          1       WE     Wraith Lifeline
## 5   227       1    172       1      678          1       WE    Wattson Lifeline
## 6   938       3   1359       5     1054          6       WE    Wattson   Wraith
##   slegend placing   date       day   mode
## 1  Octane       1 23-Dec Wednesday ranked
## 2  Octane       2 23-Dec Wednesday ranked
## 3  Crypto       1 24-Dec  Thursday ranked
## 4  Crypto       2 24-Dec  Thursday ranked
## 5  Octane       5 24-Dec  Thursday ranked
## 6  Octane       1 24-Dec  Thursday ranked
```

After loading the data, I got rid of the columns 'Time of Day' and 'Week' and also lowercased the column names.

Check data

```r
# fix wrong capitalization
data$dlegend <- ifelse(data$dlegend == 'GIbraltar', 'Gibraltar', data$dlegend)

# count null values
print(colSums(is.na(data)))
```

```
##     djdmg    djkill    popdmg   popkill  spoondmg spoonkill  location   dlegend
##         0         0         0         0         0         0         0         0
##   plegend   slegend   placing      date       day      mode
##         0         0         0         0         0         0
```

```r
# describe data
summary(data)
```

```
##      djdmg            djkill           popdmg        popkill
##  Min.   :   9.0   Min.   : 0.000   Min.   :   0   Min.   :0.000
##  1st Qu.: 574.0   1st Qu.: 1.000   1st Qu.: 318   1st Qu.:0.000
##  Median : 866.0   Median : 2.000   Median : 549   Median :1.000
##  Mean   : 909.5   Mean   : 2.307   Mean   : 597   Mean   :1.775
##  3rd Qu.:1156.5   3rd Qu.: 3.000   3rd Qu.: 799   3rd Qu.:3.000
##  Max.   :3193.0   Max.   :11.000   Max.   :2100   Max.   :8.000
##     spoondmg        spoonkill       location           dlegend
##  Min.   :   0.0   Min.   : 0.00   Length:515         Length:515
##  1st Qu.: 542.0   1st Qu.: 1.00   Class :character   Class :character
##  Median : 828.0   Median : 2.00   Mode  :character   Mode  :character
##  Mean   : 890.1   Mean   : 2.61
##  3rd Qu.:1188.5   3rd Qu.: 4.00
##  Max.   :3091.0   Max.   :12.00
##    plegend            slegend             placing          date
##  Length:515         Length:515         Min.   :1.000   Length:515
##  Class :character   Class :character   1st Qu.:1.000   Class :character
##  Mode  :character   Mode  :character   Median :2.000   Mode  :character
##                                        Mean   :2.489
```

```
##                                        3rd Qu.:3.000
##                                        Max.   :5.000
##      day                 mode
##  Length:515         Length:515
##  Class :character   Class :character
##  Mode  :character   Mode  :character
##
##
##
```

There was a wrongly capitalized legend name in the data when I checked it, so that had to be corrected. Everything else looked normal. No null values either.

User-Legend data analysis

Usage per user, per legend

```r
legend_counts <- list()
legend_columns <- c('dlegend', 'plegend', 'slegend')

for (column in legend_columns) {
  counts <- table(data[[column]])
  legend_counts[[column]] <- counts
}

for (column_name in names(legend_counts)) {
  cat("Player:", column_name, "\n")
  print(legend_counts[[column_name]])
}
```

```
## Player: dlegend
##
## Bloodhound    Caustic     Crypto       Fuse  Gibraltar    Horizon   Lifeline
##         24          3          6          3         95         94         22
##     Mirage     Octane Pathfinder   Revenant   Valkyrie    Wattson     Wraith
##         11         10        112         54          2         21         58
## Player: plegend
##
##  Bangalore Bloodhound     Crypto  Gibraltar   Lifeline       Loba     Mirage
##          3        189          3          7        140         11          7
##     Octane Pathfinder    Rampart   Revenant   Valkyrie    Wattson     Wraith
##          7          4          2         10         71         38         23
## Player: slegend
##
##  Bangalore    Caustic     Crypto       Fuse  Gibraltar     Mirage     Octane
##         75         29         26          6          1          7        342
## Pathfinder   Revenant       Seer   Valkyrie    Wattson     Wraith
##          3         10         10          3          1          2
```

DJ used Pathfinder the most, Pop used Bloodhound the most, and Spoon used Octane a ton. This may cause outliers in some parts of the data.

Highest damage per user, per legend

```
##       dlegend djdmg
## 10 Pathfinder  3193
## 3      Crypto  2378
## 14     Wraith  2323
## 5   Gibraltar  2269
## 6     Horizon  2129
## 13    Wattson  2013
## 7    Lifeline  2007
## 11   Revenant  1848
## 8      Mirage  1710
## 1  Bloodhound  1616
## 9      Octane  1345
## 12   Valkyrie  1196
## 4        Fuse  1059
## 2     Caustic   711
```

```
##       plegend popdmg
## 12   Valkyrie   2100
## 5    Lifeline   1865
## 2  Bloodhound   1651
## 13    Wattson   1593
## 8      Octane   1439
## 14     Wraith   1428
## 4   Gibraltar   1415
## 6        Loba   1281
## 11   Revenant    974
## 9  Pathfinder    832
## 7      Mirage    747
## 3      Crypto    693
## 1   Bangalore    581
## 10    Rampart    544
```

```
##       slegend spoondmg
## 1   Bangalore     3091
## 7      Octane     2458
## 9    Revenant     2446
## 2     Caustic     2362
## 3      Crypto     2330
## 10       Seer     1940
## 11   Valkyrie     1704
## 6      Mirage     1686
## 8  Pathfinder     1431
## 5   Gibraltar     1394
## 4        Fuse     1075
## 13     Wraith      921
## 12    Wattson      506
```

Despite using Octane an extreme amount, Spoon actually has his highest damage with Bangalore. Pop's highest damage is Valkyrie, with Bloodhound ranking 3rd.

Highest kills per user, per legend

```
##       dlegend djkill
## 10 Pathfinder     11
## 14     Wraith      9
## 1  Bloodhound      8
## 5   Gibraltar      7
## 6     Horizon      7
## 3      Crypto      6
## 7    Lifeline      6
## 9      Octane      6
## 11   Revenant      6
## 12   Valkyrie      6
## 13    Wattson      6
## 8      Mirage      5
## 2     Caustic      3
## 4        Fuse      2
##       plegend popkill
## 12   Valkyrie       8
## 5    Lifeline       7
## 13    Wattson       7
## 2  Bloodhound       6
## 4   Gibraltar       6
## 6        Loba       5
## 8      Octane       5
## 11   Revenant       5
## 14     Wraith       5
## 1   Bangalore       4
## 3      Crypto       3
## 7      Mirage       3
## 9  Pathfinder       3
## 10    Rampart       1
##       slegend spoonkill
## 2     Caustic        12
## 7      Octane        11
## 1   Bangalore         9
## 10       Seer         7
## 8  Pathfinder         6
## 9    Revenant         6
## 3      Crypto         5
## 5   Gibraltar         5
## 11   Valkyrie         5
## 4        Fuse         4
## 6      Mirage         4
## 13     Wraith         3
## 12    Wattson         2
```

Again, despite being a popular character for Spoon, his Octane fell short from having the highest kills and damage stat from his pool of characters. DJ's Pathfinder remains consistent in highest damage and kill stat from his pool of characters.

Summation data per user, per legend

```
##       dlegend djdmg djkill
## 10 Pathfinder 96862    263
## 5   Gibraltar 92667    211
## 6     Horizon 85419    192
## 11   Revenant 54347    148
## 14     Wraith 49546    133
## 7    Lifeline 21627     60
## 1  Bloodhound 20240     56
## 13    Wattson 19240     48
## 8      Mirage  8387     30
## 9      Octane  8001     20
## 3      Crypto  6670     15
## 4        Fuse  2119      2
## 12   Valkyrie  1690      6
## 2     Caustic  1583      4
##       plegend popdmg popkill
## 2  Bloodhound 108984     332
## 5    Lifeline  86194     254
## 12   Valkyrie  47951     130
## 13    Wattson  21185      65
## 14     Wraith  11962      34
## 6        Loba   6870      20
## 4   Gibraltar   6161      22
## 11   Revenant   5620      18
## 8      Octane   4468      14
## 7      Mirage   2472      10
## 9  Pathfinder   2375       6
## 3      Crypto   1521       3
## 1   Bangalore    850       4
## 10    Rampart    833       2
##       slegend spoondmg spoonkill
## 7      Octane   305296       898
## 1   Bangalore    72213       202
## 2     Caustic    25151        76
## 3      Crypto    18228        47
## 9    Revenant    10181        30
## 10       Seer     8684        31
## 6      Mirage     5964        16
## 11   Valkyrie     3484        13
## 4        Fuse     3338        11
## 8  Pathfinder     2281         9
## 13     Wraith     1656         4
## 5   Gibraltar     1394         5
## 12    Wattson      506         2
```

What is shown here is each players summed kills and damage with each character they play. Spoon's Octane usage finally shows here, with almost double as much damage and kills on one character than Pop and DJ's top legend combined.

Each users best legend

```
##       dlegend djdmg djkill
## 10 Pathfinder  3193     11
## 3      Crypto  2378      6
## 14     Wraith  2323      9
## 5   Gibraltar  2269      7
## 6     Horizon  2129      7
## 13    Wattson  2013      6
## 7    Lifeline  2007      6
## 11   Revenant  1848      6
## 8      Mirage  1710      5
## 1  Bloodhound  1616      8
## 9      Octane  1345      6
## 12   Valkyrie  1196      6
## 4        Fuse  1059      2
## 2     Caustic   711      3
```

```
##       plegend popdmg popkill
## 12   Valkyrie   2100       8
## 5    Lifeline   1865       7
## 2  Bloodhound   1651       6
## 13    Wattson   1593       7
## 8      Octane   1439       5
## 14     Wraith   1428       5
## 4   Gibraltar   1415       6
## 6        Loba   1281       5
## 11   Revenant    974       5
## 9  Pathfinder    832       3
## 7      Mirage    747       3
## 3      Crypto    693       3
## 1   Bangalore    581       4
## 10    Rampart    544       1
```

```
##       slegend spoondmg spoonkill
## 1   Bangalore     3091         9
## 7      Octane     2458        11
## 9    Revenant     2446         6
## 2     Caustic     2362        12
## 3      Crypto     2330         5
## 10       Seer     1940         7
## 11   Valkyrie     1704         5
## 6      Mirage     1686         4
## 8  Pathfinder     1431         6
## 5   Gibraltar     1394         5
## 4        Fuse     1075         4
## 13     Wraith      921         3
## 12    Wattson      506         2
```

If you look at the MAX damage and kills to see the best legend for each user, it looks to be what you expect. Let's check what would happen if we took the average.

Each users best legend (by average)

```r
# calculate the averages and counts for each legend for dj
best_legend_dj_avg <- data %>%
  group_by(dlegend) %>%
  summarise(
    avg_dj_dmg = mean(djdmg),
    avg_dj_kill = mean(djkill),
    count = n()
  ) %>%
  filter(count > 20) %>% # filter out legends with less than 20 uses since we're taking average
  arrange(desc(avg_dj_dmg), desc(avg_dj_kill))

# calculate the averages and counts for each legend for pop
best_legend_pop_avg <- data %>%
  group_by(plegend) %>%
  summarise(
    avg_pop_dmg = mean(popdmg),
    avg_pop_kill = mean(popkill),
    count = n()
  ) %>%
  filter(count > 20) %>% # filter out legends with less than 20 uses since we're taking average
  arrange(desc(avg_pop_dmg), desc(avg_pop_kill))

# calculate the averages and counts for each legend for spoon
best_legend_spoon_avg <- data %>%
  group_by(slegend) %>%
  summarise(
    avg_spoon_dmg = mean(spoondmg),
    avg_spoon_kill = mean(spoonkill),
    count = n()
  ) %>%
  filter(count > 20) %>% # filter out legends with less than 20 uses since we're taking average
  arrange(desc(avg_spoon_dmg), desc(avg_spoon_kill))

print(best_legend_dj_avg)
```

```
## # A tibble: 8 x 4
##   dlegend    avg_dj_dmg avg_dj_kill count
##   <chr>           <dbl>       <dbl> <int>
## 1 Revenant        1006.        2.74    54
## 2 Lifeline         983.        2.73    22
## 3 Gibraltar        975.        2.22    95
## 4 Wattson          916.        2.29    21
## 5 Horizon          909.        2.04    94
## 6 Pathfinder       865.        2.35   112
## 7 Wraith           854.        2.29    58
## 8 Bloodhound       843.        2.33    24
```

```r
print(best_legend_pop_avg)
```

```
## # A tibble: 5 x 4
```

```
##   plegend    avg_pop_dmg avg_pop_kill count
##   <chr>            <dbl>        <dbl> <int>
## 1 Valkyrie          675.         1.83    71
## 2 Lifeline          616.         1.81   140
## 3 Bloodhound        577.         1.76   189
## 4 Wattson           558.         1.71    38
## 5 Wraith            520.         1.48    23
```

```r
print(best_legend_spoon_avg)
```

```
## # A tibble: 4 x 4
##   slegend   avg_spoon_dmg avg_spoon_kill count
##   <chr>             <dbl>          <dbl> <int>
## 1 Bangalore          963.           2.69    75
## 2 Octane             893.           2.63   342
## 3 Caustic            867.           2.62    29
## 4 Crypto             701.           1.81    26
```

Interestingly enough, the best legend didn't change for anybody except for DJ, with Revenant shooting up multiple placements vs his placement using the max damage and kills. It could be due to Pathfinder having double the game count than Revenant. Regardless, Pathfinder will still be counted as DJ's best legend.

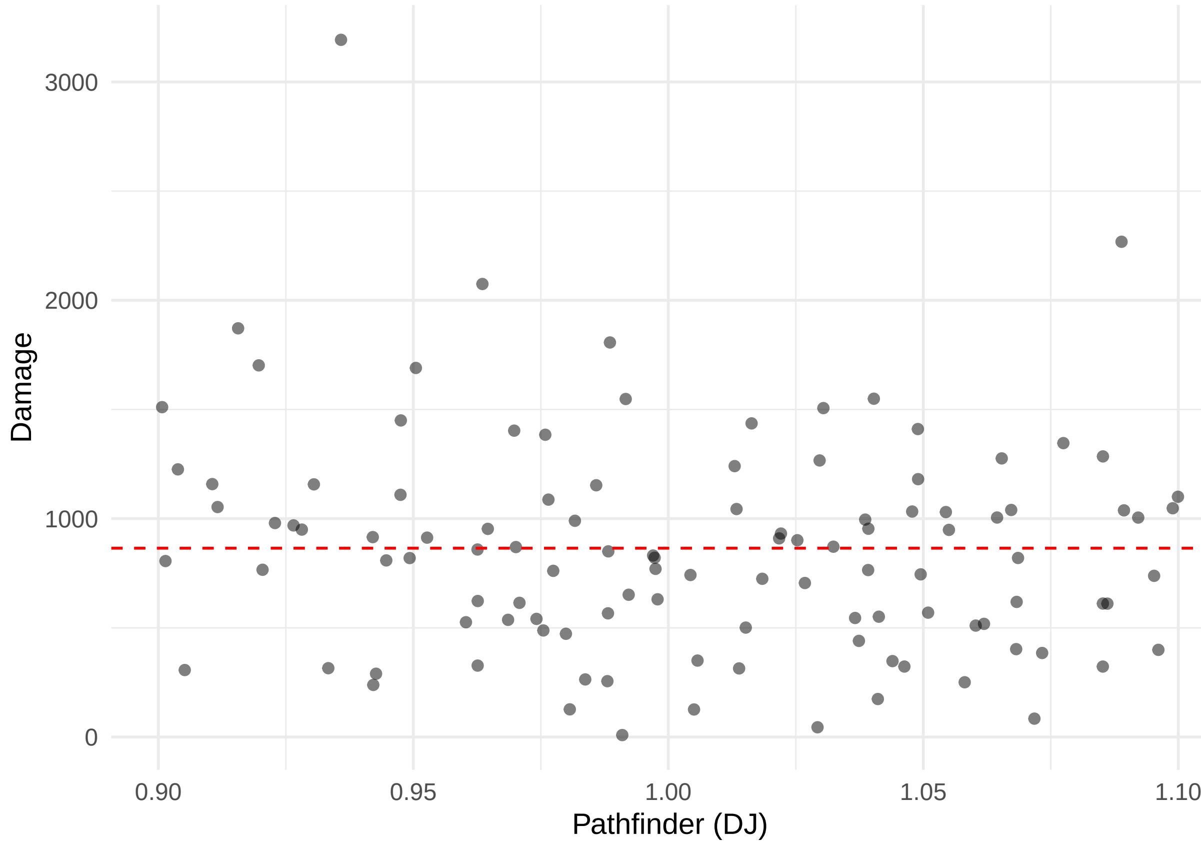
Best Legend (by Win Rate)

```
## Win rates for dlegend :
##
## Bloodhound    Caustic     Crypto       Fuse  Gibraltar    Horizon   Lifeline
## 0.16666667 0.00000000 0.33333333 0.00000000 0.28421053 0.27659574 0.36363636
##     Mirage     Octane Pathfinder   Revenant   Valkyrie    Wattson     Wraith
## 0.09090909 0.20000000 0.31250000 0.27777778 0.00000000 0.28571429 0.29310345
##
## Win rates for plegend :
##
##  Bangalore Bloodhound     Crypto  Gibraltar   Lifeline       Loba     Mirage
##  0.0000000  0.2910053  0.0000000  0.2857143  0.2785714  0.2727273  0.1428571
##     Octane Pathfinder    Rampart   Revenant   Valkyrie    Wattson     Wraith
##  0.1428571  0.0000000  0.0000000  0.2000000  0.2816901  0.4210526  0.1739130
##
## Win rates for slegend :
##
##  Bangalore    Caustic     Crypto       Fuse  Gibraltar     Mirage     Octane
##  0.2666667  0.3103448  0.2307692  0.1666667  1.0000000  0.1428571  0.3011696
## Pathfinder   Revenant       Seer   Valkyrie    Wattson     Wraith
##  0.0000000  0.0000000  0.1000000  0.3333333  0.0000000  0.0000000
```

Win rate for each legend was found by finding the number of wins a user has with that legend divided by the total number of times the legend was used by that user. Going by win rate alone, Lifeline is DJ's best, Wattson is Pop's best, and Gibraltar (he only had one game where he won on Gibraltar) or Valkyrie is Spoon's best.

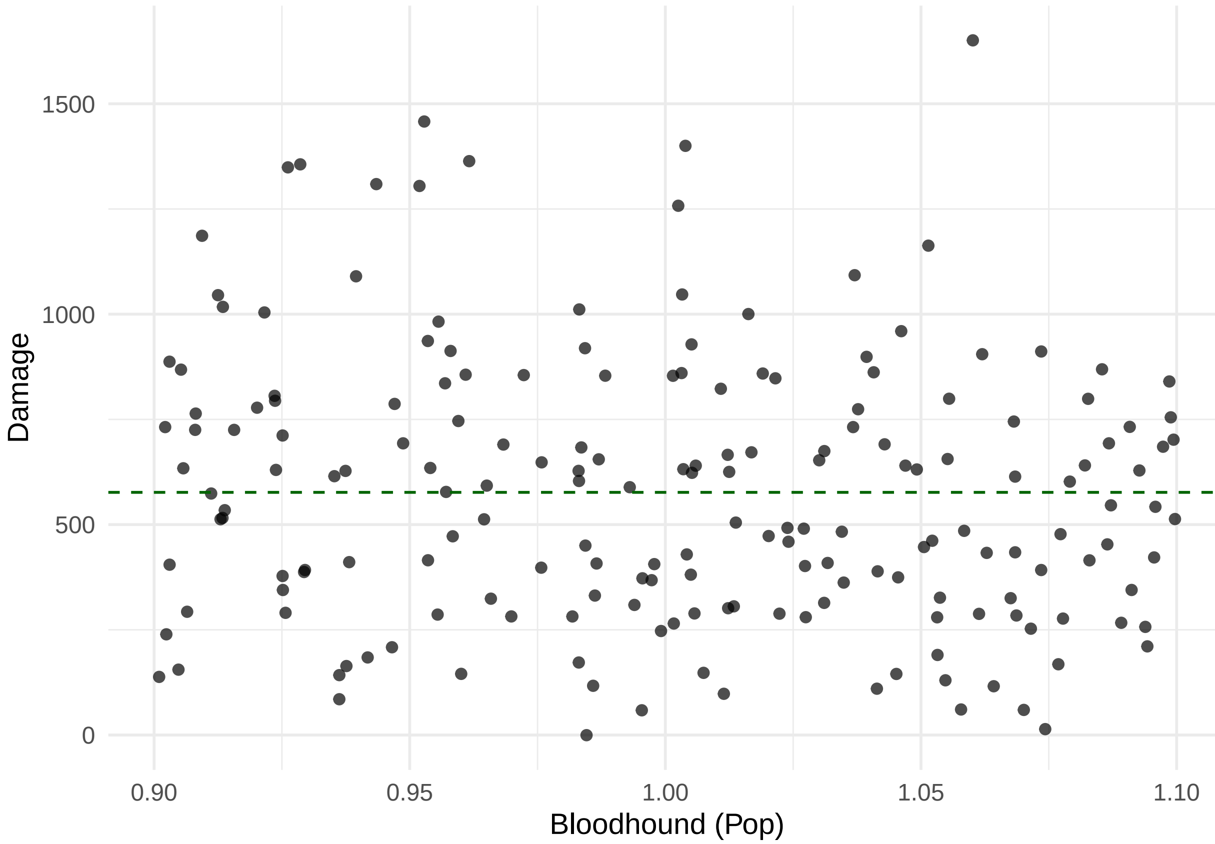
Plotting data

Plotting each players best legend



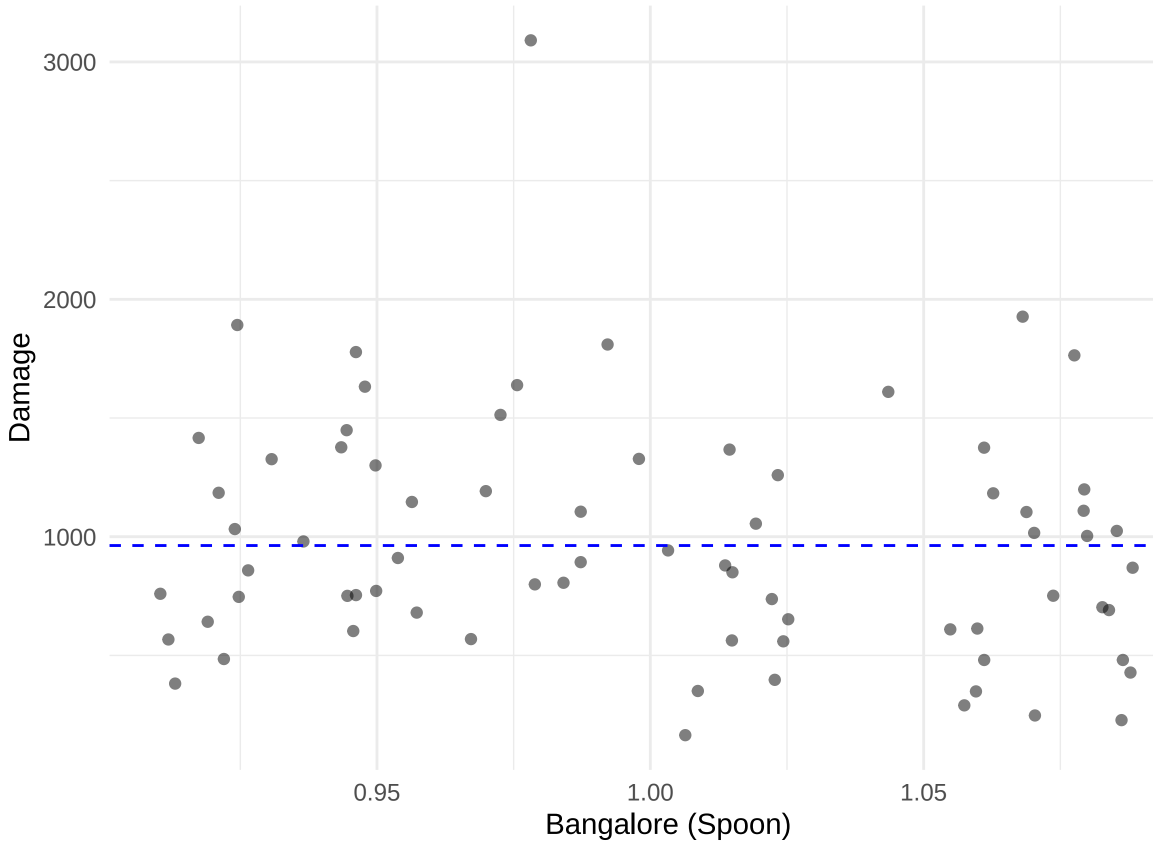
DJ's Pathfinder

Most of DJ's Pathfinder damage falls along the average, with a few outliers.



Pop's Valkyrie

Pop's damage with Bloodhound looks like it's more prone to having outliers, where less of Pop's games fall along the average.

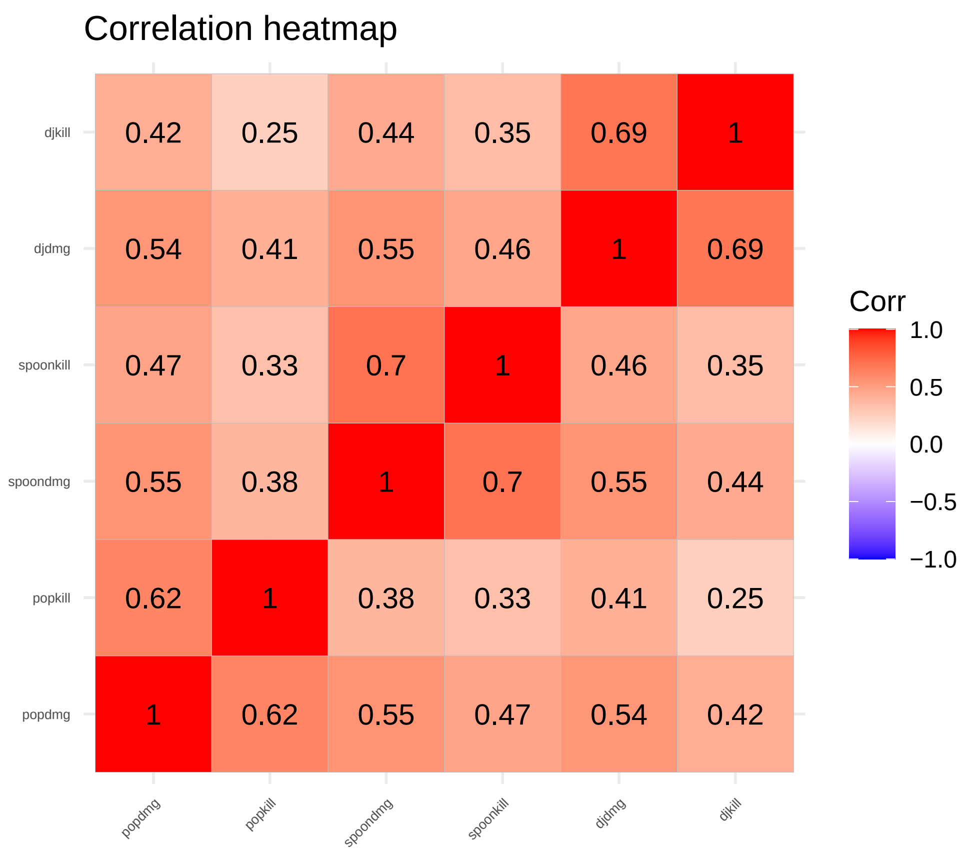


Spoon's Bangalore

Spoon's damage seems to be consistent with the average, not shooting too far above or below it.

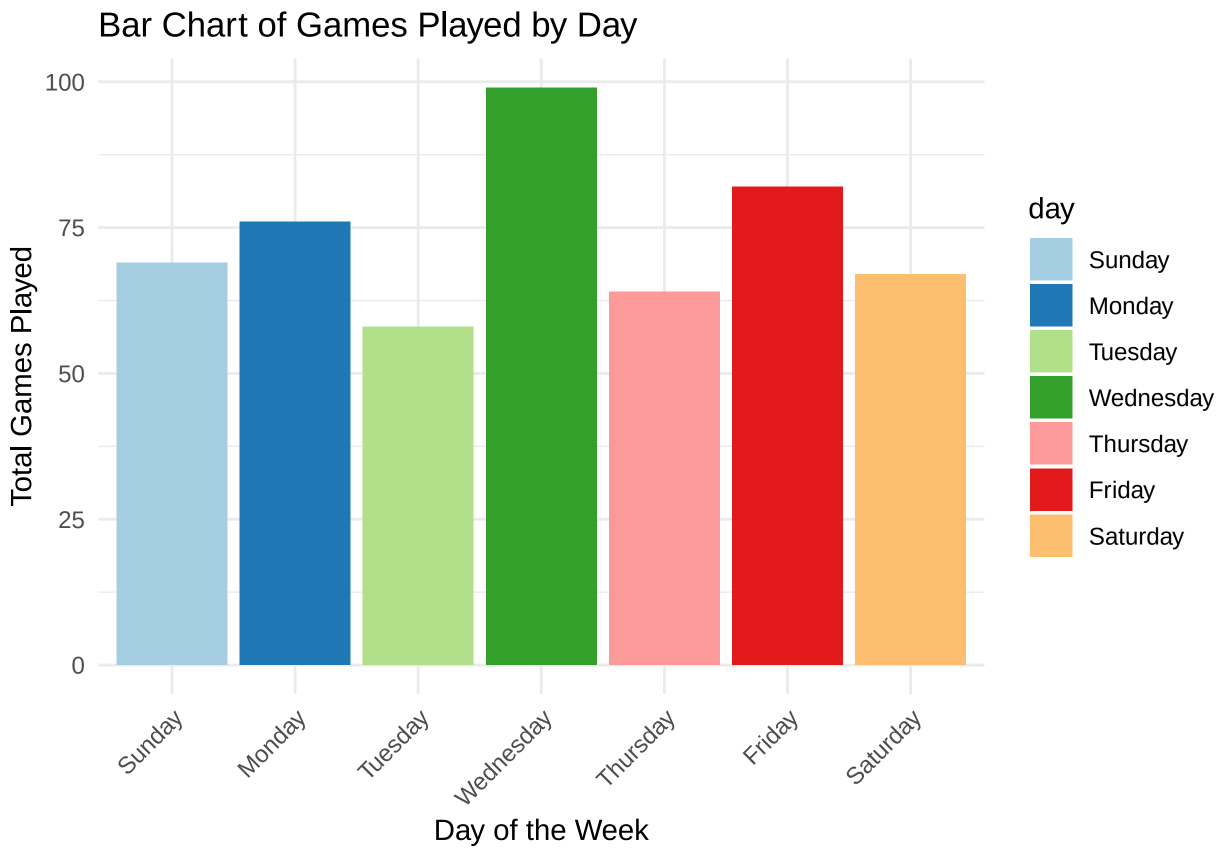
Correlation matrix plot

```
# get the numeric columns for the correlation matrix
numeric_data <- data[c('djdmg', 'djkill', 'popdmg', 'popkill', 'spoondmg', 'spoonkill')]

# calculate correlation matrix
corr_matrix <- cor(numeric_data)

# plot matrix
ggcorrplot(corr = corr_matrix, lab_size = 4, tl.cex = 5,
           lab = T, title = "Correlation heatmap", hc.order = T)
```



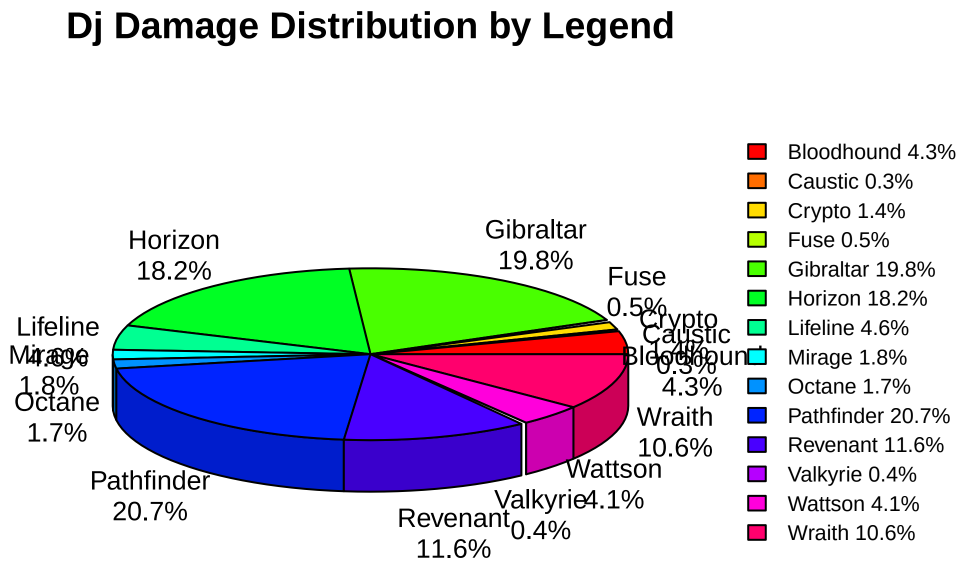
It looks like each players kills and damage alike is positively correlated with themselves, but there isn't much correlation between each players kills and damage with another player. Not much what I was expected, I figured if one player did good in one game, then the other 2 would also do somewhat good.

What days of the week did we play most?

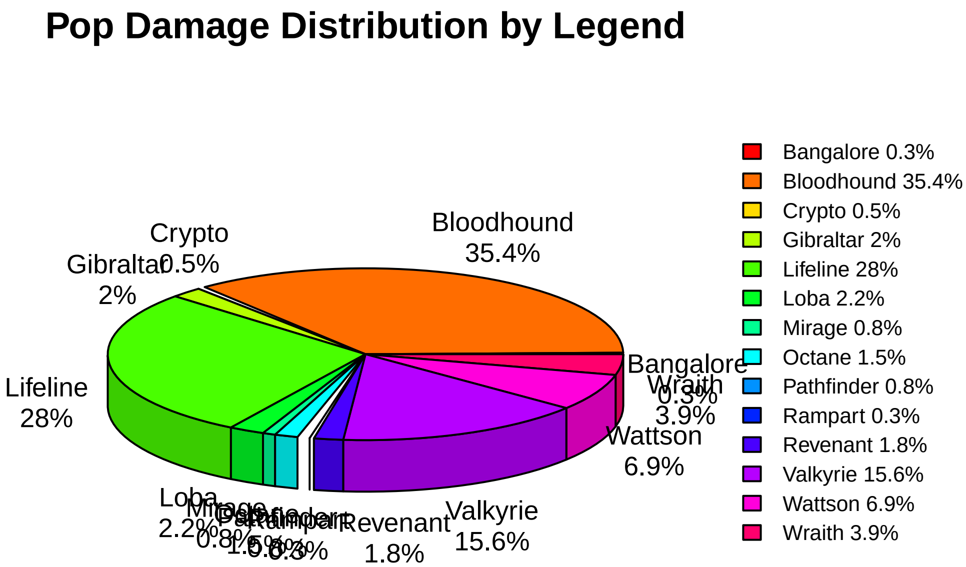



```
## $day
## [1] "Wednesday"
##
## $count
## [1] 30
```

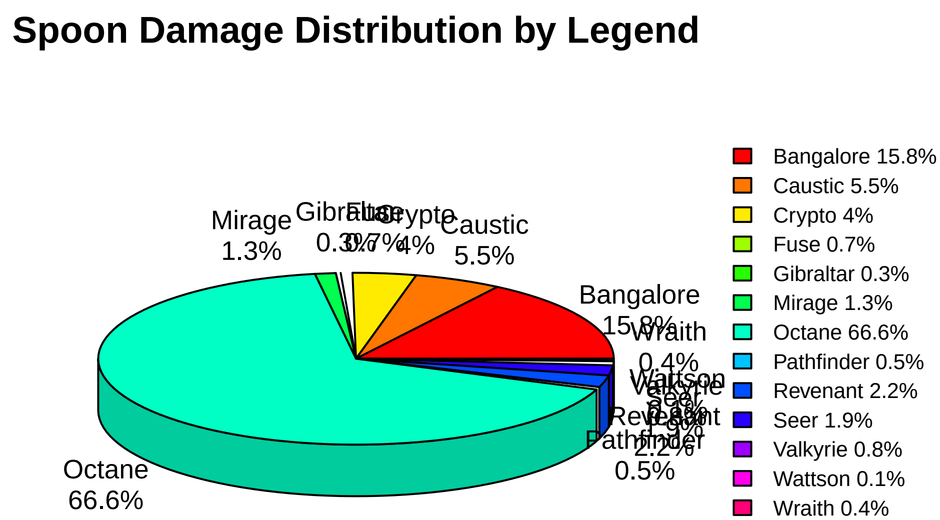
I guess Wednesday was our most commonly played day. We also got the most 1st places on Wednesday as well (30)



DJ has 3 characters that total about 60% of the damage distribution. Horizon, Pathfinder, and Gibraltar.



Popshot has 2 characters that account for about 60% of the damage distribution. Bloodhound and Lifeline.

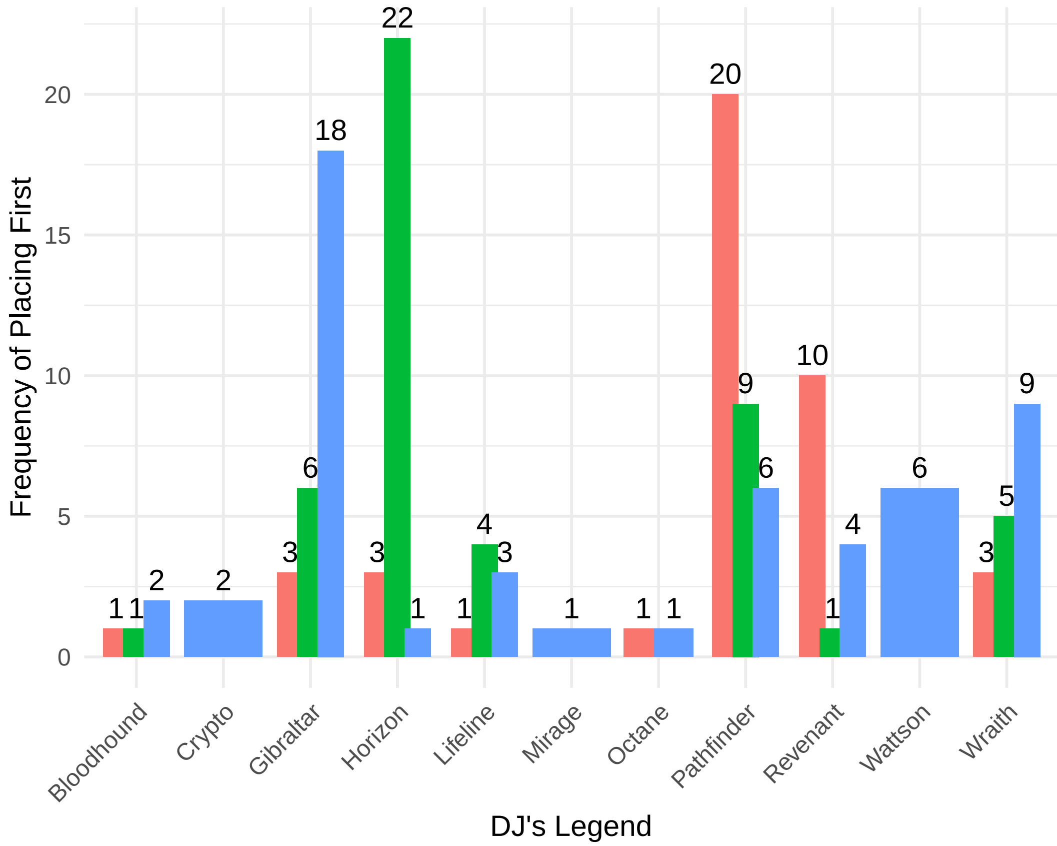


Spoon results were easy to spot from a mile away, seeing the number of games on Octane from earlier.

When do we win?

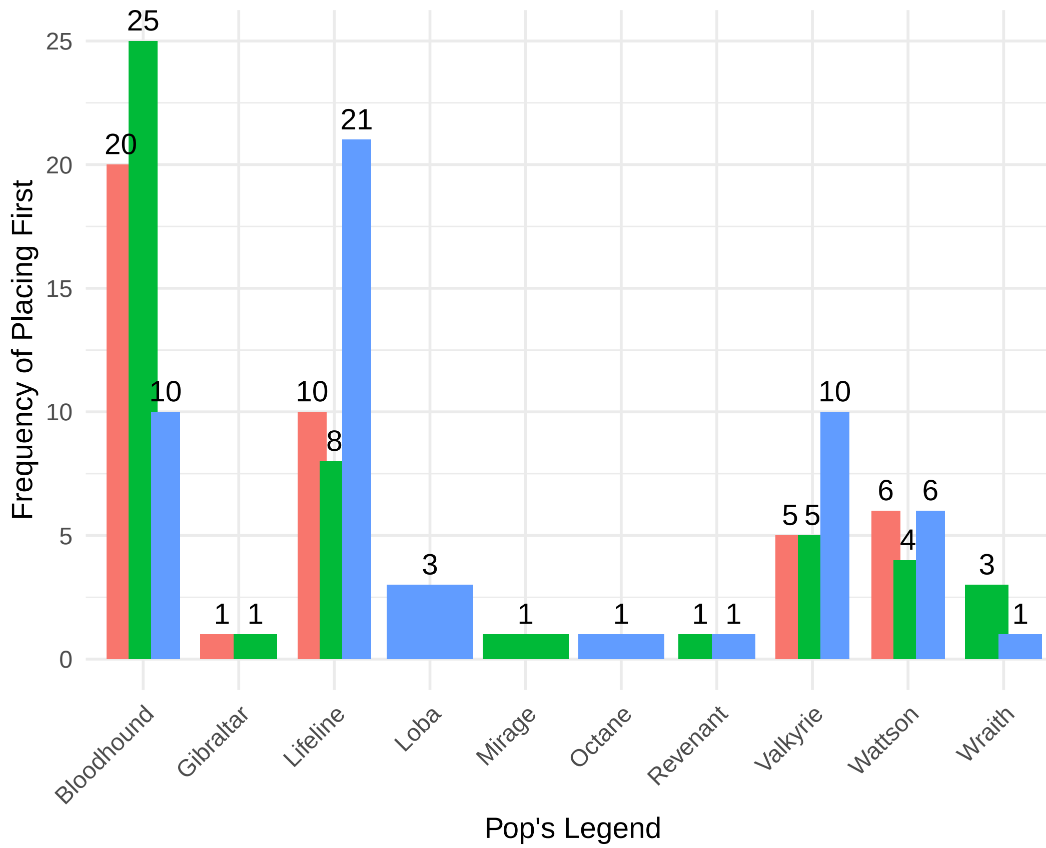
Here we will focus on what the scenario is when we win. Is it a specific character for each player, or maybe a team of 3 specific legends.

First place frequencies



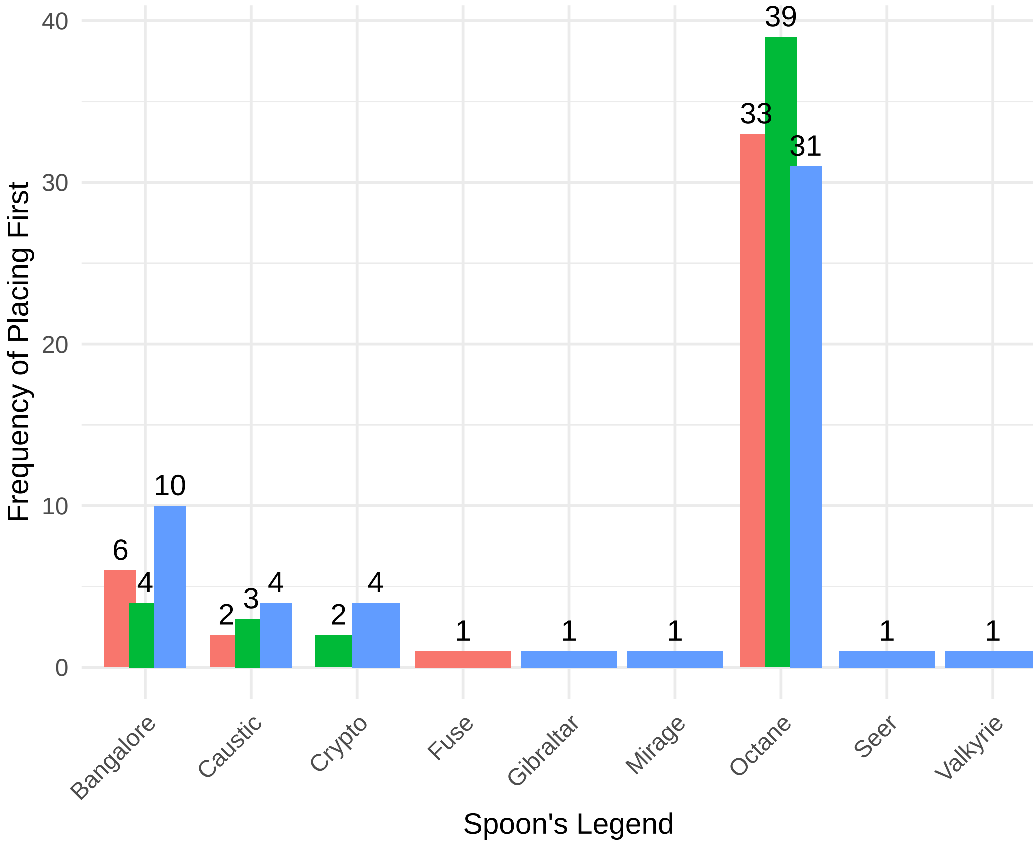
DJ First place frequencies

It's clear to see that for DJ had favorites for each respective map. Olympus was a favorite for Horizon, World's Edge was most Gibraltar, and on Kings Canyon we saw more Pathfinder.



Pop First place frequencies

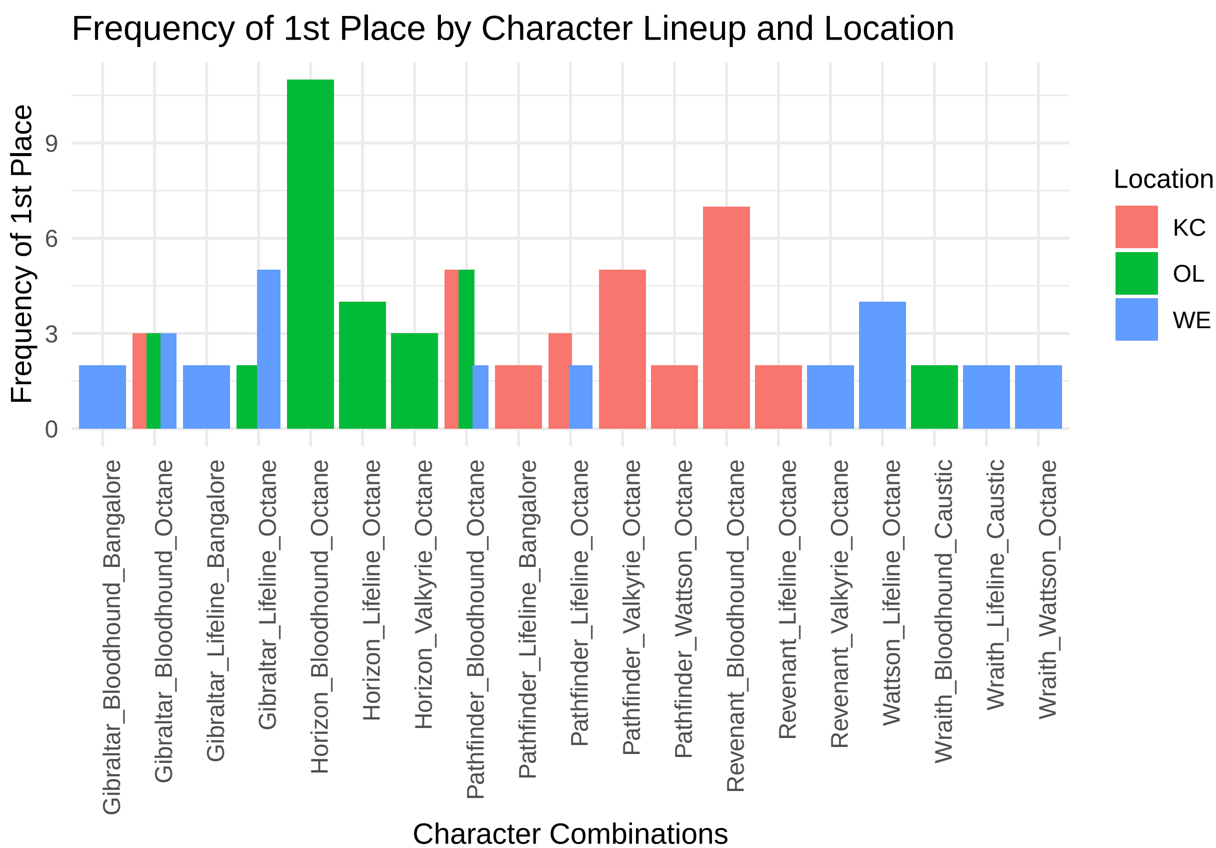
Clearly for Pop, Bloodhound was a fan favorite and a crucial character pick to getting a first place. Lifeline on World's Edge picked up Bloodhound's shortcomings on the map.



Spoon first place frequencies

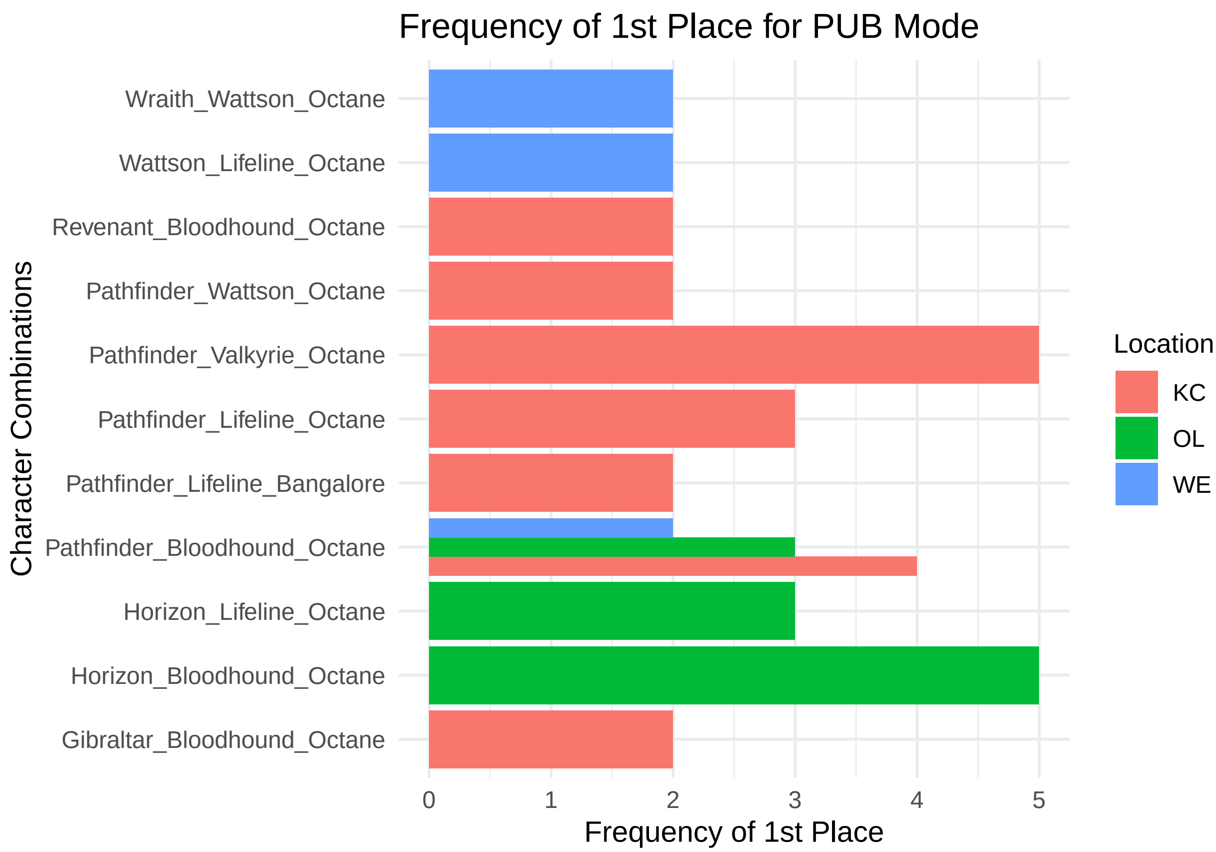
To no one's surprise, Octane was vital to getting a victory on all maps. This is probably due to sheer playtime on Octane though.

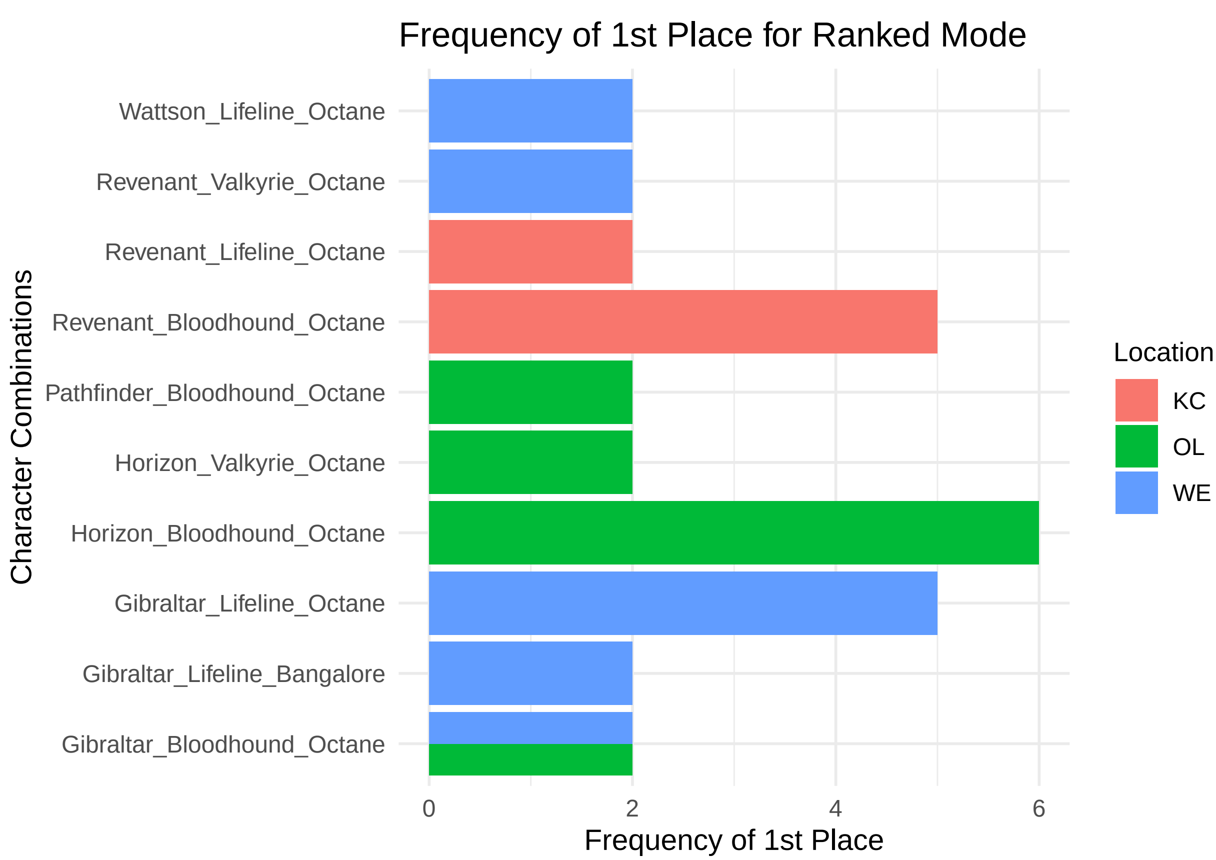
What about team combinations?



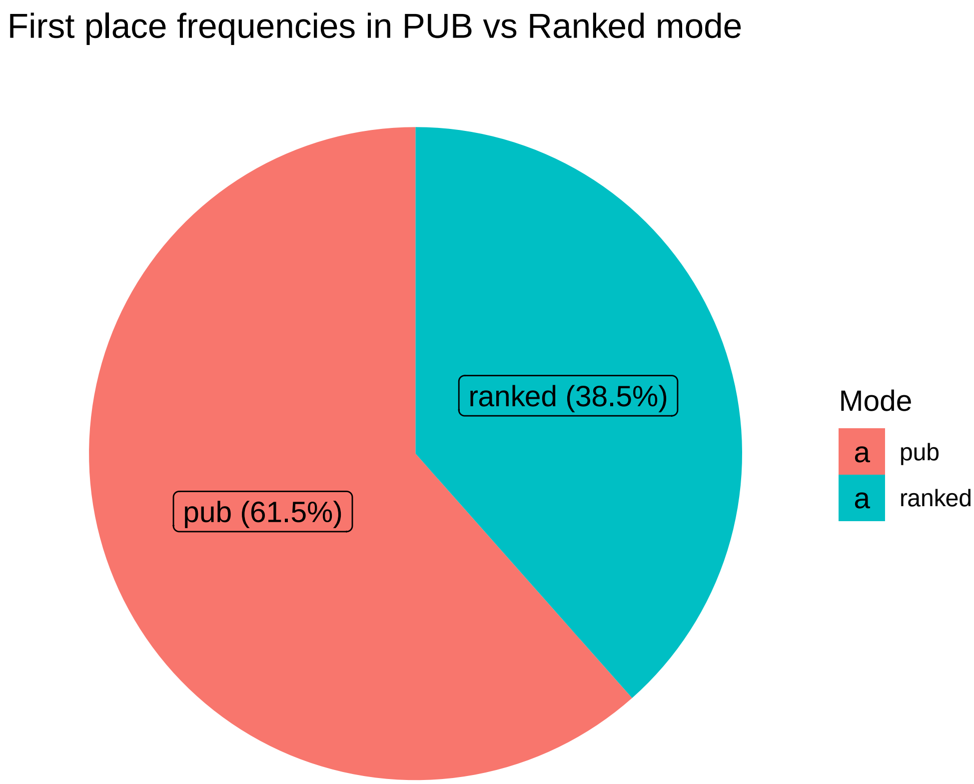
On Olympus the prime trio was Horizon from DJ, Bloodhound from Pop, and Octane from Spoon of course. This might be due to Horizon's strength on Olympus. Interestingly, on Kings Canyon, Revenant from DJ, Bloodhound from Pop, and Octane from Spoon was the best trio on that map. This is explained to due to RevTane meta, where Revenant and Octane on a team meant free kills.

Are we better in ranked or pub matches?





For both modes, it seems like Horizon + Bloodhound + Octane was the all star squad, on Olympus at least.




```
## # A tibble: 2 x 2
##   mode   count
##   <chr>  <int>
## 1 pub      346
## 2 ranked   169

## [1] 0.3254438

## [1] 0.2543353
```

Although the pie chart shows that around 60% of 1st places comes from pubs, and 40% comes from ranked, this could be brushed off as saying pub is easier, but really it is due to the high count of pub matches played. 346 pub matches played, and only 169 ranked matches played. It's arguable to say that we do better in ranked, since we win a higher percentage of ranked games out of all the ranked games played, a 32% win rate, while in pubs our win rate is only 25%.

Scoring

Scoring method - Kill and Damage weighted

```r
# define kill and damage weights
kill_weight <- 0.55
damage_weight <- 0.45

# define a list to store scores for each legend
legend_scores <- list()

# list of legend columns and their corresponding kill and damage columns
legend_columns <- c('dlegend', 'plegend', 'slegend')
kill_columns <- c('djkill', 'popkill', 'spoonkill')
damage_columns <- c('djdmg', 'popdmg', 'spoondmg')

# iterate over each legend
for (i in seq_along(legend_columns)) {
    # compute the score for the current legend
    legend_score <- (data[[kill_columns[i]]] * kill_weight) + (data[[damage_columns[i]]] * damage_weigh
    # store the mean of scores divided by 1000 in the list
    legend_scores[[legend_columns[i]]] <- mean(legend_score) / 1000
}

# print
for (legend in names(legend_scores)) {
    cat(paste("Score for", legend,":", legend_scores[[legend]], "\n"))
}
```

```
## Score for dlegend : 0.41054854368932
## Score for plegend : 0.269618252427184
## Score for slegend : 0.401958058252427
```

The scoring method is pretty simple, we just determined we value kills a bit more than damage, so I gave kills a little more significance when determining the 1st score. There's not much that goes into the score besides kills and damage.

Wrap Up

Wrapping up, I found:

- each users best character
- our best combinations as a team
- each users score based on their overall damage and kills
- our gameplay in ranked vs pub lobbies
- first place frequencies, individually and as a team
- what day of the week we play the most and win the most on
- damage distribution for each user

Next iteration, on top of what is already found here, there will be more variables to look at, such as assists, knocks, revives, respawns, etc as well as Machine Learning techniques like regression or decision trees.