Статистики върху героя Евелин в известната онлайн игра “League of Legends”

Изготвил: Стефан Николаев Ангелов, ИС, курс 2, група 3, 71961, ФМИ на СУ

Увод в изследвания проблем. Контекст на проблема.

В днешни времена компютърните игри са се превърнали в ежедневие. Хора на всякакви възрасти имат нужда от разтоварване и начин да забравят тревогите от всекидневието си. Съответно все повече от хората започват да играят онлайн игри, чрез които могат да се превъплатят в други същества и да общуват с други хора, които също искат да избягат от ежедневието си. В последните години една от най-известните онлайн игри става “League of Legends”. В нея играят два отбора по пет играча един срещу друг на карта, която предразполага разделянето на 5 основни позиции - горна пътека, средна пътека, долна пътека (тук играят съответно “носещият” и неговият support (поддръжник)) и джунгла. Последната позиция е може би най-интересната и трудна за играене. В нея играчът се движи по по-голямата част от картата и трябва да убива различни чудовища (дракони и други), чрез които дава различни бонуси на своите съотборници. Негова работа е също да помага на съотборниците си в атакуването на играчи от другия отбор, като когато го прави, винаги се получава числово превъзходство в полза на неговия отбор (феноменът е известен като “gank”). Разбира се, той трябва да се бие и с играча на същата позиция от противниковия отбор. Поради изискващата си позиция, шампионите (друга дума за героите) в играта на тази позиция трябва да имат интересни механики и начини да се придвижват бързо. Съответно един от най-играните герои като джунгла е Евелин (Evelynn). Поради тази причина за мен беше интересно да разбера какви са зависимостите между начина на игра, важността на героя в отбора и победите, когато някой от играчите играе с Евелин.

Избор на променливите

Данните са взети динамично от Riot API. Реших да съкратя малко колоните и така обособих 5 променливи - 3 категорийни и 2 числови. Числовите променливи са сътоветно killRatio и deathRatio. И двете са непрекъснати и представляват проценти спрямо другите 4 играча в отбора, съответно

първата променлива показва какъв процент от убиванията на противниците са направени от текущия играч, а втората - какъв процент от умиранията принадлежат на текущия играч. Категорийните променливи са съответно tier, blue_team и win. Първата показва в коя дивизия се бие играчът. Колкото по-голяма дивизия, толкова по-опитни са играчите в нея. Във възходящ ред стойностите ѝ са BRONZE < SILVER < GOLD < DIAMOND < PLATINUM. Другите две приемат само стойности TRUE и FALSE. Първата показва дали се намираме в единия отбор (blue team) или в другия (red team). Втората показва дали сме спечелили, или не.

Едномерен анализ на числовите променливи

Когато анализираме числови данни, важно е да видим дали няма някои стойности, които значително се отличават от останалите. За целта можем да използваме няколко метода. За изкарването на самите екстремуми, реших да използвам out вектора, намиращ се в boxplot.stats .

```
outliersKillRatio <- boxplot.stats(data$killRatio)$out #find outliers using IQR
```

Всички стойности, които се получават тук, се намират посредством IQR (interquartile range). За да разберем какво е това, е хубаво да уточним, че 5-номерната статистика е един полезен анализ на числова променлива - това са найните минимум, максимум, медиана, 1-ви и 3-ти квартил. По дефиниция квартилите разцепват непрекъснатата числова променлива на 4 части. Съответно първи квартил показва стойностите до 25-тия процент. Медианата е втори квартил, тоест показва 50-тия процент. Аналогично трети квартил показва 75-тия процент. Оттук можем да дефинираме коефициентът IQR = Q3 - Q1. Съответно всяка стойност, по-малка от Q1 - 1.5*IQR и по-голяма от Q3 + 1.5*IQR е екстремум за нас. За killRatio това са стойностите 0.75, 0.78. Неговата петорна характеристика е съответно min = 0, Q1 = 0.18, Q2 = 0.29, Q3 = 0.38, max = 0.78 . Можем да видим също и математическото очакване - 0.29, което е близко до медианата. Тези характеристики виждаме с този код:

```
summary(data$killRatio)
```

Освен локацията на разпределението, искаме да видим и начина на разсейване на данните. Първият начин е с коефициента MAD (median absolute deviation) = median(|Xi - X|), където Xi е всяка стойност, която взима променливата, а X - медианата. Освен него, искаме да сметнем и вариацията (дисперсията). Формулата е DX = VarX = E(X - X^2) = sd^2. Кодът, който изкарва тези коефициенти + IQR е:

```
cat("MAD = ", mad(data$killRatio), "\n")
cat("Var(x) = ", var(data$killRatio), "\n")
cat("IQR = ", IQR(data$killRatio), "\n")
```

Техните стойности са съответно MAD = 0.15, Var(x) = 0.02, IQR = 0.2 . Много важна стъпка за нас е да разберем дали нашата променлива има разпределние, близко до нормално. За целта съм използвал теста на Шапиро-Уилк. Нека с H0 да отбележим нулевата хипотеза на теста на Шапиро-Уилк, която е, че разпределението е нормално. Съответно с H1 отбелязваме обратната (алтернативна хипотеза) - разпределението не е нормално. Дефинираме стойност p value, която ни казва каква е вероятността H0 да е вярно. Ако минем някаква зададена от нас граница, alpha, то не можем да отхвърлим H0. Това в кода изглежда така:

```
if(shapiro.test(data$killRatio)$p.value >= 0.05) {
    print("Kill ratio is close to a normal distribution")
} else {
    print("Kill ratio is not close to a normal distribution")
}
```

Тук сме задали alpha = 0.05 . С моите данни получавам съобщението “Kill ratio is not close to a normal distribution”, тоест променливата не е близка до нормално разпределение. Аналогичното правим за променливата deathRatio:

```
summary(data$deathRatio)
cat("MAD = ", mad(data$deathRatio), "\n")
cat("Var(x) = ", var(data$deathRatio), "\n")
cat("IQR = ", IQR(data$deathRatio), "\n")

if(shapiro.test(data$deathRatio)$p.value >= 0.05) {
    print("Death ratio is close to a normal distribution")
} else {
    print("Death ratio is not close to a normal distribution")
}
```

Стойностите са:

```
   Min. 1st Qu.  Median Mean 3rd Qu.      Max.
 0.0000  0.1404  0.2000  0.1959  0.2500  0.7500
MAD =  0.08472
Var(x) =  0.009346794
IQR =  0.1096346
[1] "Death ratio is not close to a normal distribution"
```

<u>Едномерен анализ на категорийните променливи</u>

Единственият анализ, който можем да направим за този тип променливи, е честотен анализ в проценти. Това става като направим данните на таблица и после използваме prop.table, за да ги превърнем в проценти. Като код:

```
tableTier <- table(data$tier)
round(prop.table(tableTier) * 100, 2)

tableBlueTeam <- table(data$blue_team)
round(prop.table(tableBlueTeam) * 100, 2)

tableWin <- table(data$win)
round(prop.table(tableWin) * 100, 2)
```

Като резултати, имаме, че играчите по дивизии се разпределят така:

BRONZE	DIAMOND	GOLD	PLATINUM	SILVER
18.58	19.67	21.31	20.77	19.67

По разпределение на син и червен отбор:

FALSE	TRUE
48.63	51.37

По това дали побеждават, или не:

FALSE	TRUE
50.82	49.18

Много известни платформи като Blitz и Op.gg показват статистиката win-rate. В случая можем да заключим, че само 49.18% от играещите тази героиня печелят, но и освен това можем да заключим, че тя е статистически балансирана.

<u>Многомерен анализ на зависимостта между категорийните променливи</u>

Има ли зависимости между различните категорийни променливи? За да видим това, ще образуваме двумерни таблици от процентите на някоя стойност на едната променлива при застопорена стойност от другата променлива. Нека да разгледаме първо зависимостта между tier и blue_team. Попринцип тази статистика е безсмислена, защото дали ще играем в син или червен отбор се определя на псевдослучаен принцип преди всяка игра. Кодът за това:

```
tierBlueTeam <- table(data$tier, data$blue_team)

# Blue team selection based on the different tiers (in percentages)
prop.table(tierBlueTeam, margin = 1)

# Tiers based on blue team selection (in percentages)
prop.table(tierBlueTeam, margin = 2)
```

Резултатите:

	FALSE	TRUE
BRONZE	0.4411765	0.5588235
DIAMOND	0.5000000	0.5000000
GOLD	0.5384615	0.4615385
PLATINUM	0.3815789	0.6184211
SILVER	0.5694444	0.4305556

Виждаме, че голяма част от хората в бронзова дивизия попадат в син отбор. За хората от сребърна дивизия случаят е различен - повечето са били в червения отбор. Интересното е, че в диамантената дивизия разпределението е било равно. В платинената дивизия преобладава доминантно синия отбор. Да видим и обратната статистика:

	FALSE	TRUE
BRONZE	0.1685393	0.2021277
DIAMOND	0.2022472	0.1914894
GOLD	0.2359551	0.1914894
PLATINUM	0.1629213	0.2500000
SILVER	0.2303371	0.1648936

Тя не е чак толкова интересна. Тук се показва, че ако разглеждаме играчите от червения отбор в извадката, най-голям шанс има да си изберем играч от златна дивизия, а от синия - платинена. Аналогично, връзката между дивизия и победа:

```
tierWinRate <- table(data$tier, data$win)

# Win rate based on tier
prop.table(tierWinRate, margin = 1)

# Tier based on win rate
prop.table(tierWinRate, margin = 2)
```

	FALSE	TRUE
BRONZE	0.5588235	0.4411765
DIAMOND	0.4444444	0.5555556
GOLD	0.4871795	0.5128205
PLATINUM	0.5000000	0.5000000
SILVER	0.5555556	0.4444444

	FALSE	TRUE
BRONZE	0.2043011	0.1666667
DIAMOND	0.1720430	0.2222222

GOLD 0.2043011 0.2222222
PLATINUM 0.2043011 0.2111111
SILVER 0.2150538 0.1777778

Ако играем с Евелин в бронзова или в сребърна дивизия, има голям шанс да загубим. В златна и диамантена дивизия нещата се обръщат, а в платинена имаме равнопоставеност. Оттук следва, че Евелин е сравнително техничен герой и трябва да се играе от доста по-опитни играчи. От втората таблица следва, че ако кажем на някого, че сме спечелили с Евелин, най-малко вероятно е той да си помисли, че сме бронзова дивизия например. Това подкрепя тезата, че Евелин е доста по-трудна за неопитните играчи. Нека да видим и връзката между победи и отбори. Синият отбор започва най-отдолу на картата, докато червеният - най-отгоре. Ето защо може да има някаква корелация между двете позиции и победата с този играч. Код:

```
blueTeamWinRate <- table(data$blue_team, data$win)

# Win rate based on blue team
prop.table(blueTeamWinRate, margin = 1)

# Blue team based on win rate
prop.table(blueTeamWinRate, margin = 2)
```

Резултати:

	FALSE	TRUE
FALSE	0.5000000	0.5000000
TRUE	0.5159574	0.4840426

	FALSE	TRUE
FALSE	0.4784946	0.4944444
TRUE	0.5215054	0.5055556

Оттук следва, че в червените отбори печалбата и загубата са равно вероятни, докато сините губят малко по-често. Съответно ако сме загубили, по-вероятно е да сме били в синия отбор.

<u>Многомерен анализ на зависимостта между числовите променливи</u>

Нека да започнем с корелационен анализ. За да видим дали има някаква връзка между двете числови променливи, ще пресметнем коефициент на корелация Cor(X, Y) = Cov(X) / (sqrt(Var(X)) * sqrt(Var(Y))). В случая ще видим дали има зависимост между killRatio и deathRatio. Интерпретираме коефициента на корелация както следва:

- 0.9 <= x <= 1 - данните имат изключително силна корелация

- 0.75 <= x < 0.9 - данните имат силна корелация
- 0.5 <= x < 0.75 - данните имат слаба корелация
- 0 <= x < 0.5 - данните имат изключително слаба корелация или въобще нямат такава.

Като доста от тестовете, така и коефициентът на корелация на Пиърсън изисква нормалност. От едномерния анализ излезна, че и двете данни не спазват това свойство, затова трябва да използваме коефициента на корелация на Спиърмън. Код:

```
# Correlation analysis between kill ratio and death ratio
# Data is not normal. Therefore, using Spearman's correlation index
correl_index <- abs(cor(data$killRatio, data$deathRatio, method = "spearman"))

if(correl_index >= 0.9 && correl_index <= 1.0) {
    print("Kill ratio and death ratio are extremely correlated.")
} else if(correl_index >= 0.75 && correl_index < 0.9) {
    print("Kill ratio and death ratio are quite correlated.")
} else if(correl_index >= 0.5 && correl_index < 0.75) {
    print("Kill ratio and death ratio are correlated.")
} else {
    print("Kill ratio and death ratio are vaguely correlated or not correlated at all.")
}
```

Резултати:
[1] "Kill ratio and death ratio are vaguely correlated or not correlated at all."
Двете променливи нямат корелация. Нека да направим и тест за ковариация - дали може да направим линеен модел на връзката между тези две променливи. Затова ще съставим линейна регресия между двете променливи. Трябва да проверим дали коефициентите са статистически значими. Код:

```
test <- lm(data$killRatio ~ data$deathRatio)
test2 <- lm(data$deathRatio ~ data$killRatio)

# Source : https://stackoverflow.com/questions/5587676/pull-out-p-values-and-r-squared-from-a-linear-regression

areParametersRelevant <- function(regr) {
    fStatLinearReg <- summary(regr)$fstatistic
    fDist <- pf(fStatLinearReg[1],fStatLinearReg[2],fStatLinearReg[3],lower.tail=F)
    attributes(fDist) <- NULL

    if(fDist > 0.05) {
        print("All parameters are statistically irrelevant")
    }
}

areParametersRelevant(test)
areParametersRelevant(test2)
```

Резултат:

[1] "All parameters are statistically irrelevant"
[1] "All parameters are statistically irrelevant"
Тук имаме, че коефициентите не са статистически значими.

Многомерен анализ на категорийни (обясняващи) и числови (зависими) и обратното

От проведения тест на Шапиро-Уилк видяхме, че числовите променливи не приличат на нормално разпределение. Оттук следва, че ще използваме непараметрични тестове за локация на разпределението. Тъй като ще проведем тест с две променливи, тогава дефинираме нулевата хипотеза H0 като E(x) - E(y) = 0, тоест двата вектора от данни си приличат статистически. Нека в случая p value = E(x) - E(y). Тогава отново ще сложим граница alpha = 0.05 . Чрез операцията ~ в R показваме, че лявата променлива е зависима от дясната и това формира n на брой вектора, за които да се сравни дали имат еднакво математическо очакване, където n е броят на стойностите, които взима независимата променлива. За променливите win и blue_team знаем, че приемат само две стойности, тоест, когато те са обясняващи, ще използваме тест на Уилкоксън. Когато ползваме променливите killRatio, deathRatio и tier, ще имаме нужда от теста на Крускал. Съответно при p value <= 0.05 имаме, че лявата променлива е зависима от дясната. Код:

```r
tests <- function(vector, name) {
    if(wilcox.test(vector ~ data$win)$p.value < 0.05) {
        cat("Whether we win or not influences", name, "\n")
    } else {
        cat("Whether we win or not does not influence", name, "\n")
    }

    if(wilcox.test(vector ~ data$blue_team)$p.value < 0.05) {
        cat("Whether we are in blue team or not influences", name, "\n")
    } else {
        cat("Whether we are in blue team or not does not influence", name, "\n")
    }

    if(kruskal.test(vector ~ data$tier)$p.value < 0.05) {
        cat("Tier influences", name, "\n")
    } else {
        cat("Tier does not influence", name, "\n")
    }
}

tests2 <- function(vector, name) {
    if(kruskal.test(vector ~ data$killRatio)$p.value < 0.05) {
        cat("Kill ratio influences", name, "\n")
    } else {
        cat("Kill ratio does not influence", name, "\n")
    }

    if(kruskal.test(vector ~ data$deathRatio)$p.value < 0.05) {
        cat("Death ratio influences", name, "\n")
    } else {
        cat("Death ratio does not influence", name, "\n")
    }
}

tests(data$killRatio, "kill ratio")
tests(data$deathRatio, "death ratio")
tests2(data$win, "win rate")
tests2(data$blue_team, "being in blue team")
tests2(data$tier, "tier")
```

Резултати:
Whether we win or not does not influence kill ratio
Whether we are in blue team or not does not influence kill ratio
Tier influences kill ratio
Whether we win or not influences death ratio
Whether we are in blue team or not does not influence death ratio
Tier does not influence death ratio

Kill ratio does not influence win rate
Death ratio influences win rate
Kill ratio does not influence being in blue team
Death ratio does not influence being in blue team
Kill ratio does not influence tier
Death ratio does not influence tier

Оттук виждаме, че дивизията повлиява процента на убиване; победите повлияват процента на умиране; процента на умиране повлиява на победата. Останалите нямат връзка.

Забележка: Извадката е доста голяма, затова можем да използваме централната гранична теорема и да твърдим, че тестовете за нормалност са излишни и че всяка голяма извадка клони към нормална. Съответно използването на параметрични тестове не би трябвало да промени резултатите.

Извод

Статистиките показват, че Евелин е труден герой за играене. Само по-опитните могат да извлекат полза от него. Все пак при най-опитните, опонентите знаят как да се възползват от слабостите на Евелин, което обяснява по-ниския процент на печалба в платинена дивизия. Колкото по-голяма дивизия, толкова повече играчите убиват другите. От проведената статистика може да се извади много важен извод за играта - това, че убиваш повече противници, не значи, че ще спечелиш, но ако умираш, това води по-често до загуба. Това се проявява много при Евелин поради нейната позиция в играта.