Trabalho Prático de Introdução a Ciência dos Dados

Felipe A. Melo, Luis Eduardo L. Brito, Natanael S. Júnior, Ulisses Rosa

[1]Departamento de Ciência da Computação - Universidade Federal de Minas Gerais (UFMG)
Minas Gerais, Brasil

1. Introdução

Nesse trabalho prático o nosso objeto de estudo foi o jogo multijogador Tom Clancy's Rainbow Six Siege (R6), um FPS (first person shoot game) extremamente tático.

O TP foi dividido nas seguintes etapas: caracterização e análise exploratória dos dados, construção dos testes de hipótese e desenvolvimento de modelos de machine learning. Vale dizer que ao longo do projeto estipulamos perguntas e buscamos responder elas utilizando os conceitos e técnicas vistos em sala de aula. A fim de cumprir as especificações do trabalho, foram elaboradas as seguintes perguntas de pesquisa:

1. Há mapas que favorecem uma equipe acima da outra? Se sim, quais ?
2. Jogadores de ranks altos usam o operador Jager em uma proporção maior do que os jogadores de ranks baixos ?

2. Contextualização e caracterização da base de dados

2.1. Funcionamento do jogo

Em cada partida de R6 duas equipes disputam uma sequência de rounds em um determinado mapa, alternando entre a situação de atacante e defensor. Existem três tipos de objetivo no jogo, a saber: área segura (os defensores protegem uma determinada área), bomba (os atacantes devem tentar desarmar uma bomba) e refém (os atacantes devel salvar um refém). Outro aspecto muito importante de Rainbow Six, é que os jogadores, a cada round, pode escolher um operador. É necessário ressaltar que os operadores são divididos entre atacantes e defensores. Os operadores atacantes só podem ser utilizados pela equipe atacante, e isso vale analogamente para a equipe defensora. Cada um dos operadores possui um gadget (análogo a uma habilidade especial) único pode ser utilizado para obter algum tipo de vantagem (informação, melhoria de atributos, destruição/proteção do cenário e entre outros) durante o round.

Cada round funciona da seguinte maneira. O time atacante começa em uma zona segura e parte para a área controlada pelos defensores com a missão de concluir o objetivo estabelecido na partida. É importante dizer que cada round possui um cornômetro, que quando zerado, os time defensor vence. Ademais, se um jogador é executado por outro, ele não pode renascer no mesmo round, ou seja, ele só volta no outro round.

Por se tratar de um jogo competitivo, os jogadores são classificados por ranks de acordo com a sua quantidade de vitórias e estatísticas individuais coletadas ao longo dos rounds.

2.2. Base de Dados

Para realizar o estudo sobre Rainbow Six, escolhemos um respositório no kaggle [1] que contém dados de todas as partidas ranqueadas da quinta temporada do jogo. O repo-

[1]Link para acessar os datasets

sitório possui 20 datasets salvos no formato csv e foi construido utilizando a API oficial da Ubisoft (detentora do R6). Os conjuntos de dados possuem apenas informações de partidas ranqueadas pois diversos jogadores não levam a sério partidas casuais. Portanto, para realizar análises mais profundas à respeito das habilidades de cada jogador é mais interessante utilizar os dados de apenas partidas que valem rank.

2.3. Composição da base de dados

Cada linha da base de dados contém os status (rank, operador utilizado, time, número de kills, entre outros) de um determinado jogador em um round de uma partida. Além disso outros dados que não são exclusivos do jogador (mapa, plataforma, tipo de jogo, data, roundID, matchID, duração do round, etc) também são armazenados em cada linha do dataset.

3. Fluxo de trabalho

3.1. Divisão de tarefas

Todos os membros participaram ativamente de todas as etapas do trabalho. Entretanto, Ulisses e Luiz ficaram encarregados majoritariamente da análise exploratória de dados e da escrita do relatório enquanto Felipe e Natanael ficaram focados com o desenvolvimento dos modelos de regressão e classificação. Todos os membros participaram igualmente do desenvolvimento das perguntas e construção dos testes de hipótese.

3.2. Ferramentas e Modelos utilizados

Para performar as análises de dados e treinar os modelos de regressão/classificação, foram utilizadas os seguintes pacotes python: numpy, scikit-learn, pandas e scipy.

Para responder a pergunta 1 utilizamos testes com permutação. A pergunta 2 foi respondida via bootstrap. Por fim, utilizamos um modelo de classificação e um de regressão (depois utilizamos classificação novamente).

4. Tratamento dos Dados

A primeira medida em relação ao tratamento dos dados tomada foi juntar todos os 20 datasets em um único dataset. Isso facilitou significativamente o workflow da execução dos análises em cima dos dados. Outro ponto importante, é que salvamos os dados no formato $parquet$, o que reduziu significativamente o espaço gasto em disco [2], consequentemente diminuindo o tempo de leitura/escrita dos dados em disco, o que deixou o nosso fluxo de trabalho mais prático.

Em rounds de R6, durante a fase de seleção de operadores, se um jogador não selecionar nenhum operador ele jogara com o recruta, que é análogo a um personagem sem habilidades especiais. Considerando que a ocorrência de recrutas era muito pequena em partidas ranqueadas, e que eles são operadores sem gadget, as ocorrências de recrutas não contribuiriam positivamente para as análises de dados que seriam performadas. Portanto, foram removidas todas as linhas do dataset em que o operador era recruta.

[2] O repositório no formato csv consumia 20GB de dados, salvando em $parquet$ apenas 1GB foi gasto

As colunas referentes ao gadget e as armas dos operadores [3] foram removidas do dataset. Essa escolha foi tomada pois, a arma utilizada pelo jogador é determinada pelo operador que ele está utilizando, e também pelo fato de que cada operador possui um gadget específico (não existem dois operadores diferentes com um mesmo gadget). Portanto, essas informações não agregariam positivamente em análises feitas em cima de dados dos operadores.

Por fim, a última medida de tratamento de dados tomada foi a criação de uma nova coluna. Em Rainbow Six Siege, existem classes de operadores (os operadores são classificados de acordo com o efeito que o seu gadget causa no jogo). Portanto, achamos prudente considerar a classe do operador em nossas análises. Dessa maneira, foi criada a coluna $SPECIALTY$ no dataset que indica a classe do operador que está sendo utilizado pelo jogador. Vale ressaltar que seguimos o padrão de classificação da Ubisoft [4].

5. Análise Exploratória de Dados

Para a Análise dos Dados, decidimos dar uma olhada nas colunas que mais nos interessavam, como a dos mapas e dos ranks. Conversando entre os membos, elaboramos métodos para explorá-los, além das interações entre elas e os outros dados disponíveis para nós.

As variáveis escolhidas para a Análise Exploratória são:

- **Operadores**;
- **Ranks dos Jogadores**;
- **Mapas**;
- **Plataforma**.
- **Duração da Rodada**.

5.1. Operadores

Nossa análise consiste em observar quais são os operadores mais escolhidos, em média. Olhar esa informação nos ajudaria a entender o comportamento da população do jogo, e a observamos tendências dentro da comunidade. Elaboramos então o seguinte gráfico, que revela a porcentagem média de escolha de cada operador:

[3] A saber: `primaryweapon, primaryweapontype, primarysight, primarygrip, primaryunderbarrel, primarybarrel, secondaryweapon, secondaryweapontype, secondarysight, secondarygrip, secondaryunderbarrel, secondarybarrel, secondarygadget`

[4] Link para acessar os dados oficiais dos operadores

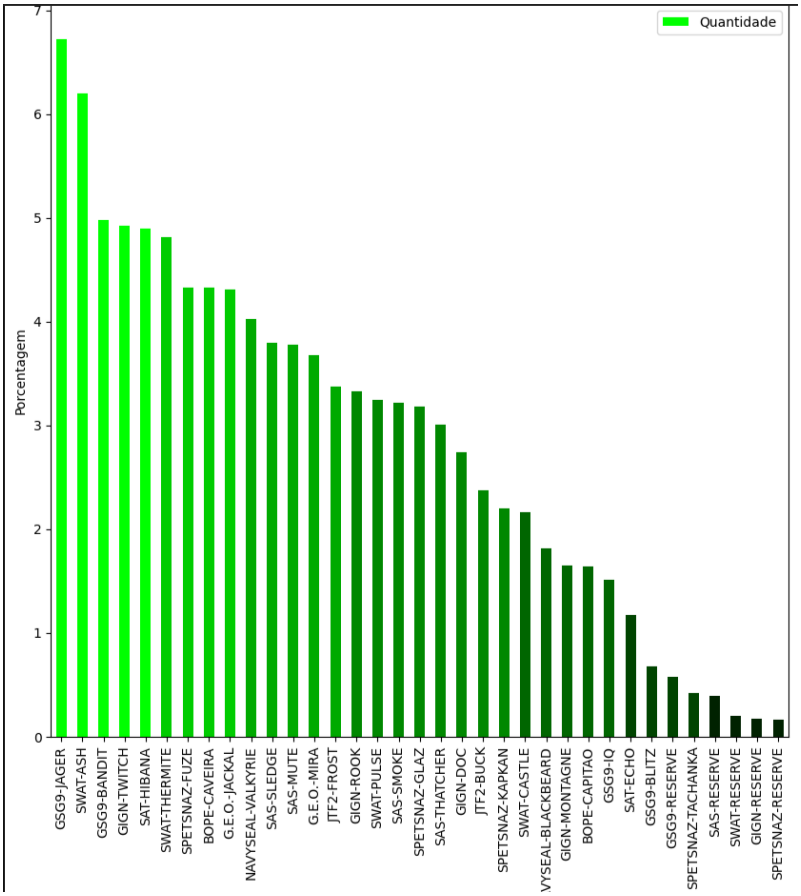

Figura 1. Porcentagem de escolhas dos Operadores

Pelo gráfico, podemos notar uma grande diferença entre as escolhas dos jogadores. Há uma preferência em escolher os operadores ”GSG9-Jager”e ”Swat-ASH”acima das outras opções. Além disso, os operadores que terminam em ”Reserve”são bem menos escolhidos, representando apenas aproximadamente 1.5% das escolha acumulada.

Há vários possíveis fatores para tais resultados, como os personagens mais escolhidos serm mais eficientes que os menos escolhidos, ou os operadores ”Reserve”não serem divertidos de se jogar com.

5.2. Ranks dos Jogadores

Nossa análise consiste em observar quais são os ranks com a maior quantidade de jogadores. Olhar esa informação nos ajuda a fazer escolhas sobre a importância de se separar jogadores em grupos, uma vez que estratégias e tendências são mais prevalentes nos ranks mais altos comparados aos mais baixos. Isso se deve ao fato que jogadores mais investidos tendem a subir nos ranks, devido a sua vontade de melhorar nos jogo, por isso eles elaboram novas estratégias e buscam melhorar suas habilidades.

Geramos um gráfico que separa os jogadores em cada um dos ranks:

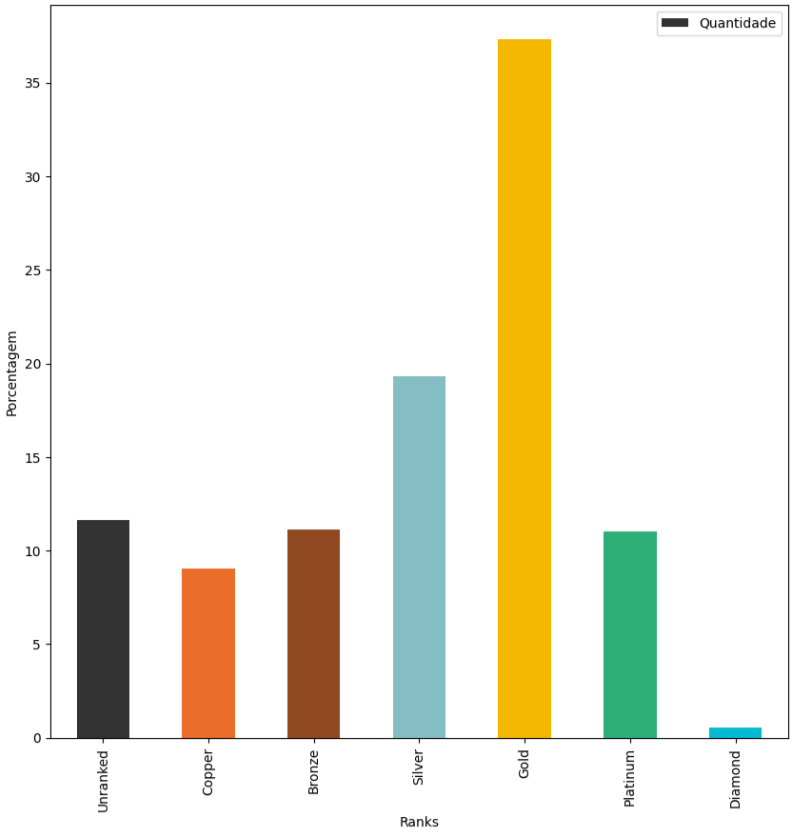

Figura 2. Porcentagem de Jogadores em cada Rank

Pelo gráfico, podemos notar uma enorme disparidade entre os ranks. O rank ”Gold”tem a maior quantidade de jogadores dentro de qualquer rank. Pelo outro lado, o rank ”Diamond”tem a menor quantidade, com menos de 1% dos jogadores.

Esse resultado é esperado, pois jogadores mais investidos no jogo tendem a chegar em ranks mais altos (”Silver”e ”Gold”). O rank ”Diamond”representa os jogadores de maior nível do jogo, logo é comum que há uma concentração pífia de jogadores.

5.3. Mapas

Nossa análise consiste em observar quais são os mapas com duração média de Round mais altos. Olhar esa informação serve de base para compreender como os mapas impactam na jogabilidade. Essa análise serviu de inspiração para uma das perguntas que respondemos no segmento de ”Teste de Hipóteses”.

Foi elaborado o seguinte gráfico, para ilustrar tal impacto:

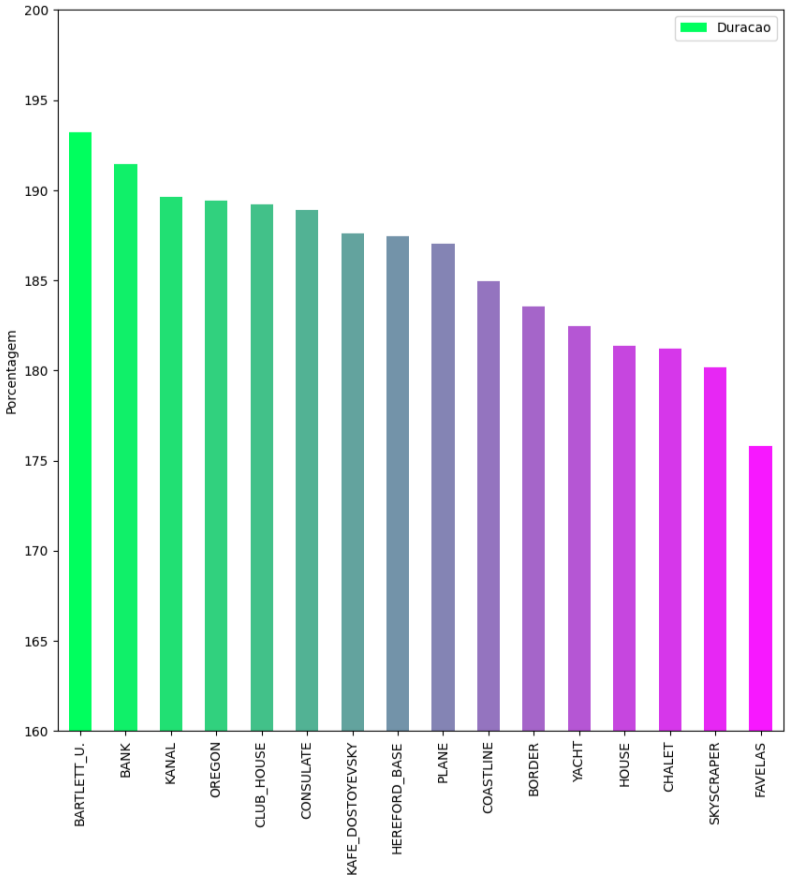

Figura 3. Duração média de Rounds por Mapa

Pelo gráfico, podemos notar que há uma disparidade na duração dos rounds. Mapas como ”Bartlett U.”e ”Bank”demoram mais, resultando em partidas mais longas. Há mapas, no entanto, como ”Favelas”que demoram consideravelmente menos que os outros mapas.

Analisando os resultados, e juntando nosso conhecimento sobre o jogo, pensamos em alguns possíveis fatores que explicam esses resultados. Talvez os mapas colocam as equipes mais distantes uma das outras.Ou até mesmo a disposição de cobertura e paredes nos mapas dificulta as batalhas entre os jogadores.

5.4. Plataforma

Nossa análise consiste em observar quais são as Plataformas mais populares. Olhar essa informação nos traria um percepção na população, pois plataformas diferentes possui comunidades diferentes. Além disso, há diferença em acessibilidade e habilidade permitida entre cada platafora. Logo descobrir onde a população se encontra principalmente é útil para ter um bom parâmetro sobre os jogadores que estamos analisando.

Para observar as diferenças na popularidade das plataformas, desenhamos o seguinte gráfico:

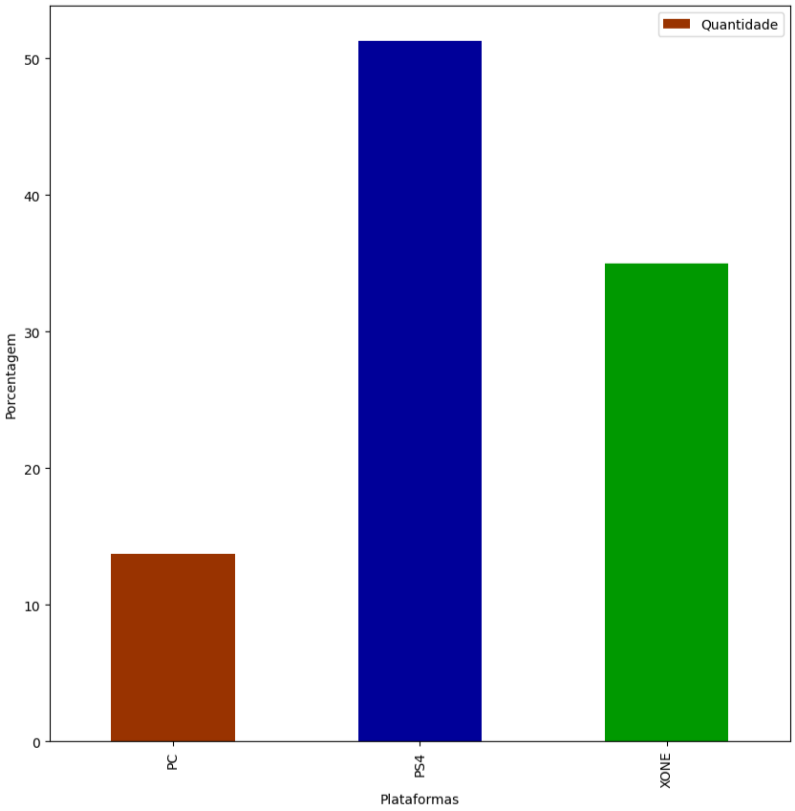

Figura 4. Porcentagem de Jogadores em cada Plataforma

Pelo gráfico, fica aparente uma enorme diferença entre a popularidade de cada plataforma. Os consoles (”X One”e ”PS4”) são muitos mais populares que PC, sendo apenas o PS4 representante de mais da metade da população de jogadores (51.3

Há diversos possíveis fatores para esse resultado, entre eles: consoles são mais baratos que comprar/montar um PC, ou o preço do jogo difere dependendo da plataforma, sendo mais barato (e atraente) no PS4.

5.5. Duração da Rodada

Nossa análise consiste em observar em média quanto dura um ”round”dependendo do rank. A análise dessa informação nos daria uma informação de como o aumento das habilidades dos jogadores correlacionam com a duração de uma rodada.

Para tal, criamosum gráfico representando a questão acima:

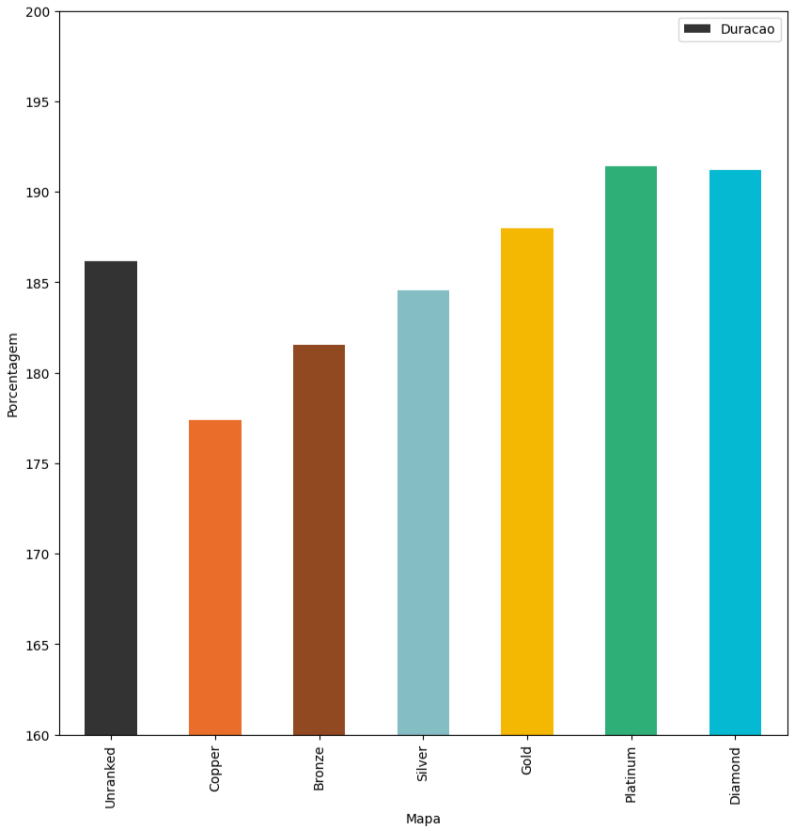

Figura 5. Duração Média de uma Rodada por Rank

Pelo gráfico, notamos que com o crescimento quanto maior o rank, maior a duração do round. Além disso, as partidas dos jogadores não rankeados ”unranked”aparece exatamente como uma média geral da duração.

Dentre possíveis fatores que explicam esse resulado, citamos o que achamos os mais prováveis: Jogadores nos ranks altos tendem a ser mais cautelosos, ou eles são mais pacientes e tendem permanecer em uma batalha por mais tempo antes de continuar a partida.

Essa informação foi tão interessante, que decidimos remotar ela em outra análise que fizemos. Dessa vez, aplicando métodos de ”Regressão”.

6. Testes de hipótese

6.1. Mapas

6.1.1. Explixação

Os mapas têm um papel muito importante na partida, ditando as áreas de embate entre os jogadores e contendo objetivos que obriga os participantes a tomarem ação para chegar a vitória. Naturalmente cada mapa é especial, gerando diferenças na jogatina e criando variedade nas estratégias.

No entanto, é de extrema importância que os mapas sejam bem elaborados. Mapas que favoreçam a equipe atacte ou a defensora (seja pela proximidade de um objetivo,

ou áreas que limitam a mobilidade de uma equipe) podem gerar frustração, pois estar em desvantagem desde o ínicio da partida leva a momentos onde o jogador se encontra trabalhando mais duro que seus oponentes para chegar a vitória.

Por isso, elaboramos um estudo, visando encontrar quais mapas favorecem um time acima do outro. Essa pesquisa é essencial, pois pode acarretar em um estudo procurando padrões nos mapas que geram esse desequilíbrio. Tais padrões podem ser retirados dos mapas, além de auxiliar na construção de futuros mapas, a fim de evitar novamente esses casos. No fim, fizemos a seguinte pergunta:

Pergunta 1. *Há mapas que favorecem uma equipe acima da outra? Se sim, quais?*

6.1.2. Metodologia

Para o experimento, a cada mapa, elaboramos as seguintes hipóteses:

- **Hipótese Nula (H_0):** O mapa não beneficia nenhum time;
- **Hipótese Alternativa 1(H_1):** O mapa beneficia o time dos atacantes;
- **Hipótese Alternativa 2(H_2):** O mapa beneficia o time dos defensores.

Para o experimento, foi usado o teste de permutação, que consiste em alterar a ordem dos elementos da base de dados e retirar uma amostra para fazer a análise. Retiramos 1000 amostras no total, cada uma contendo 50 elementos no total. Então, olhamos a porcentagem de vitórias do lado do atacante (arbitrariamente escolhida) de dentro da amostra. Criada essa lista de porcentagens, tiramos a média final, junto do intervalo de confiança de que a média real está dentro. Retiramos também o histograma de cada mapa, mostrando a distribuição dessas porcentagens:

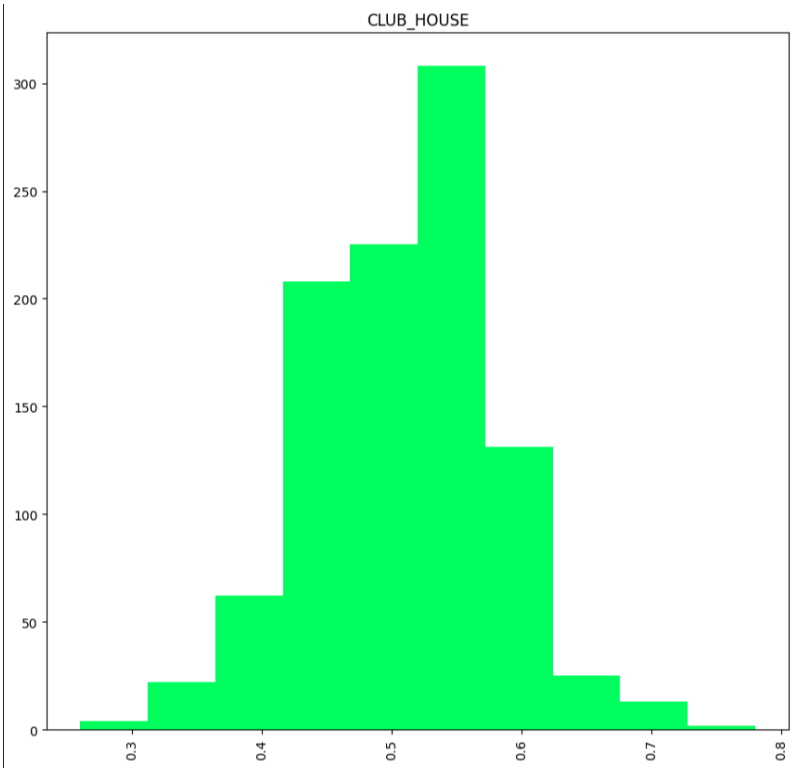

Figura 6. Histograma da porcentagem de vitórias dos atacantes em Club House

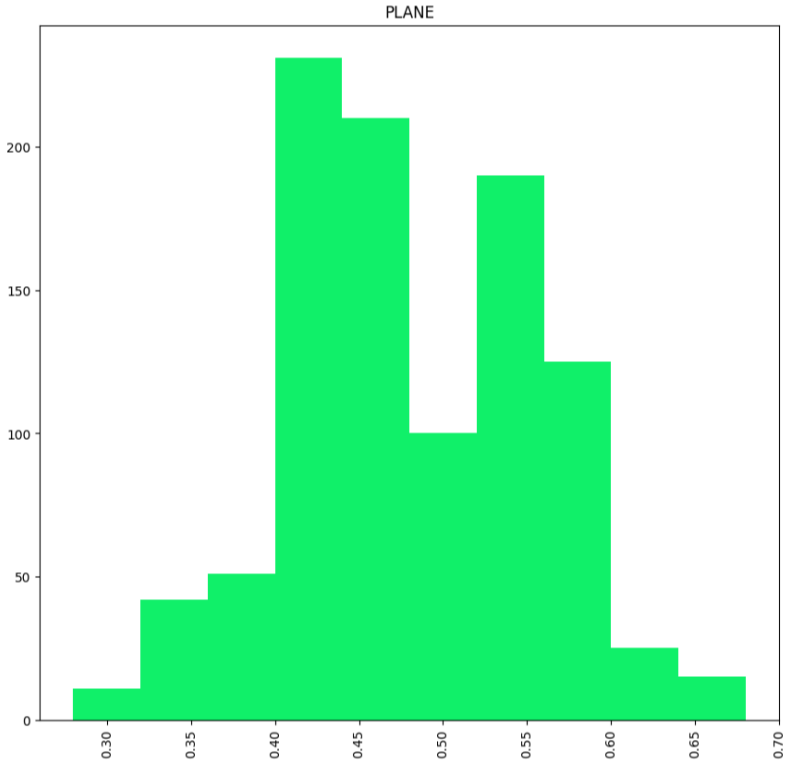

Figura 7. Histograma da porcentagem de vitórias dos atacantes em Plane

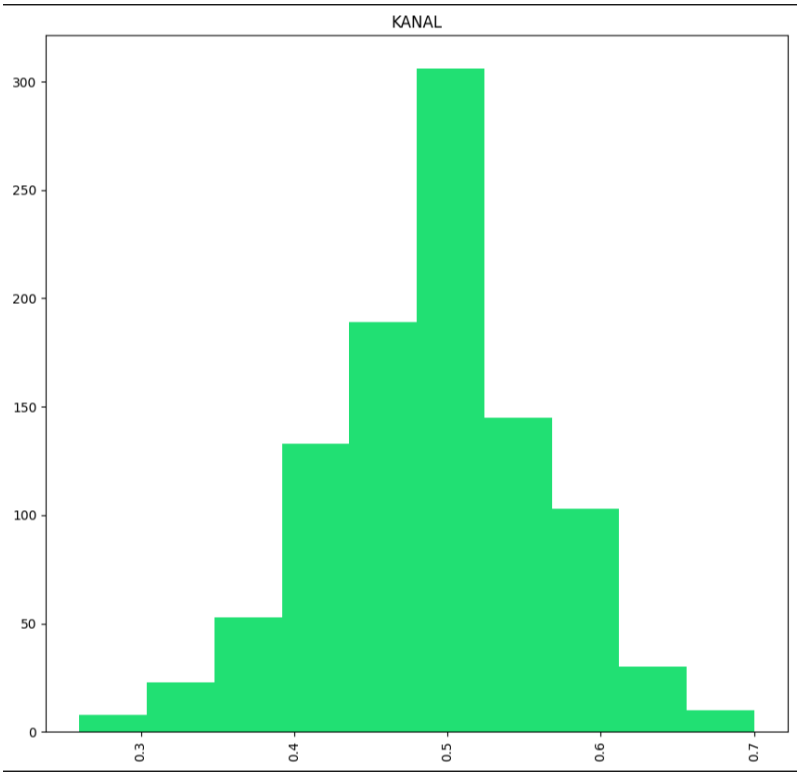

Figura 8. Histograma da porcentagem de vitórias dos atacantes em Kanal

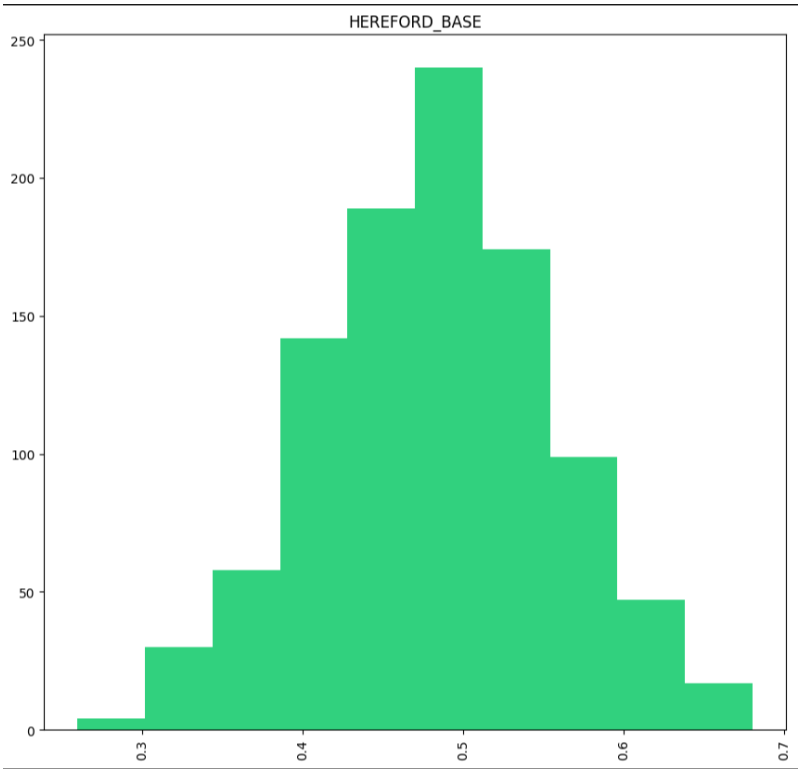

Figura 9. Histograma da porcentagem de vitórias dos atacantes em Hereford Base

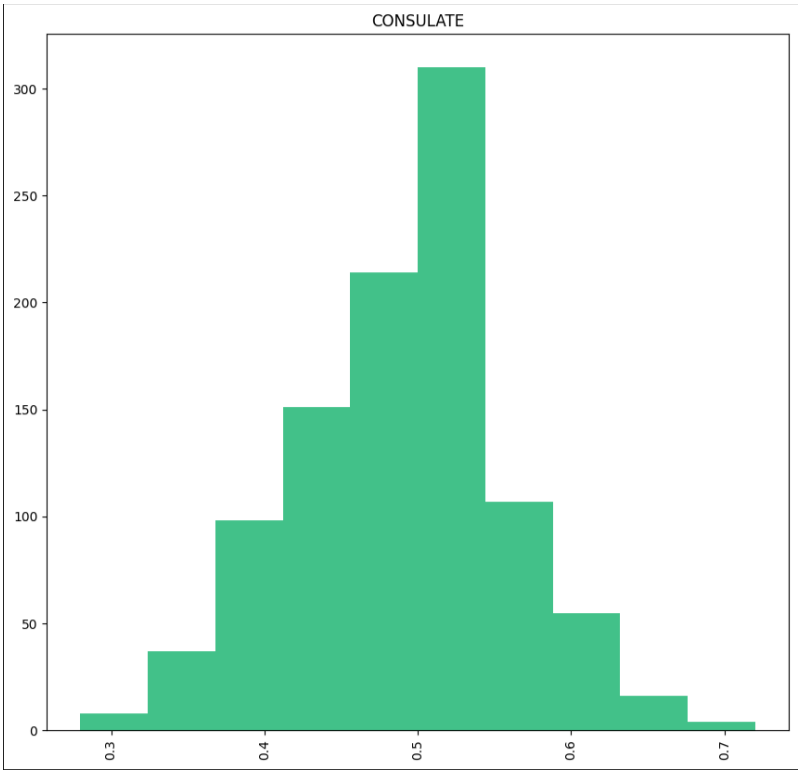

Figura 10. Histograma da porcentagem de vitórias dos atacantes em Consulate

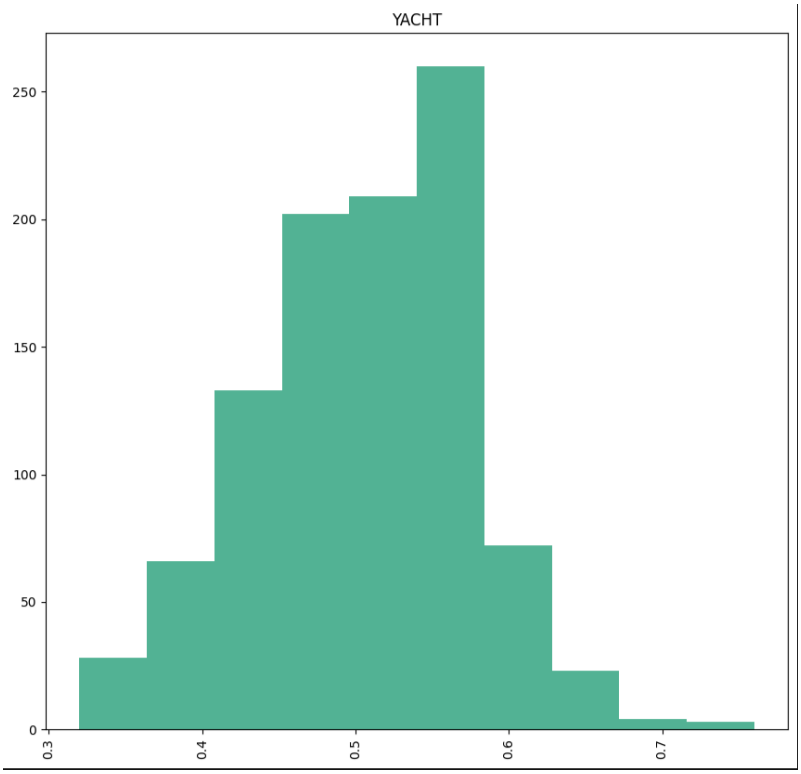

Figura 11. Histograma da porcentagem de vitórias dos atacantes em Yacht

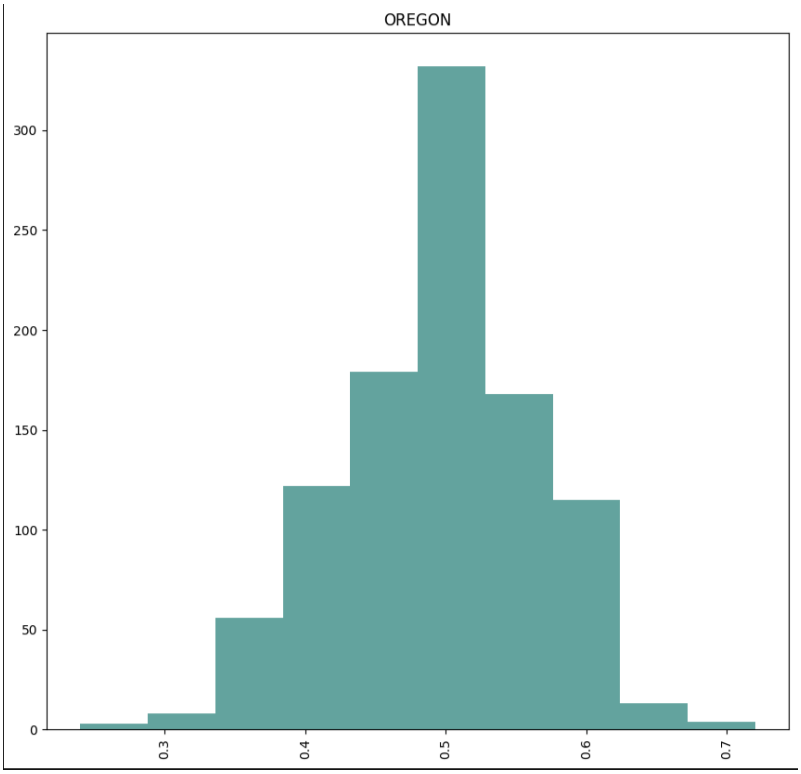

Figura 12. Histograma da porcentagem de vitórias dos atacantes em Oregon

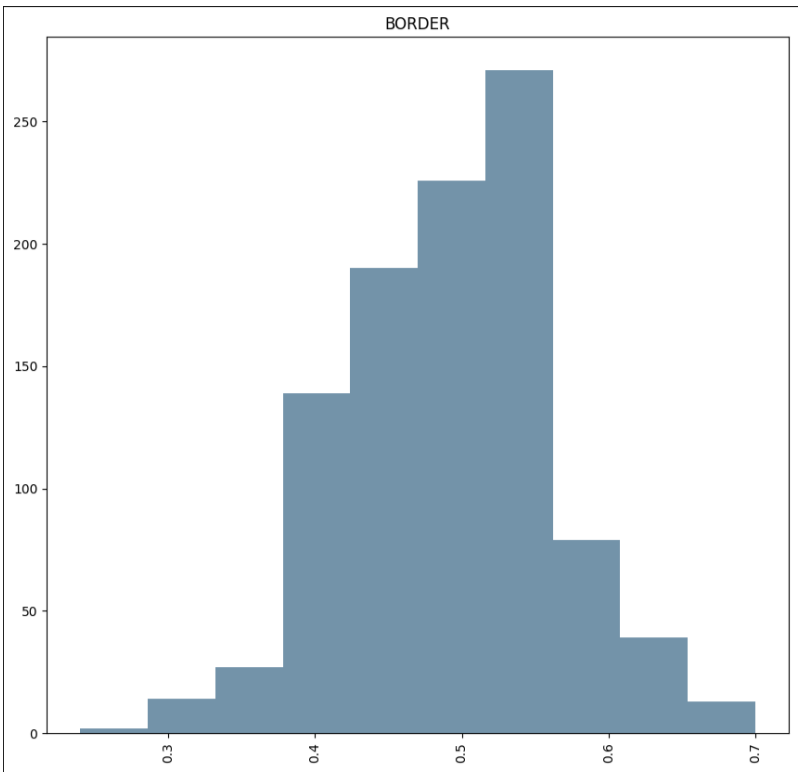

Figura 13. Histograma da porcentagem de vitórias dos atacantes em Border

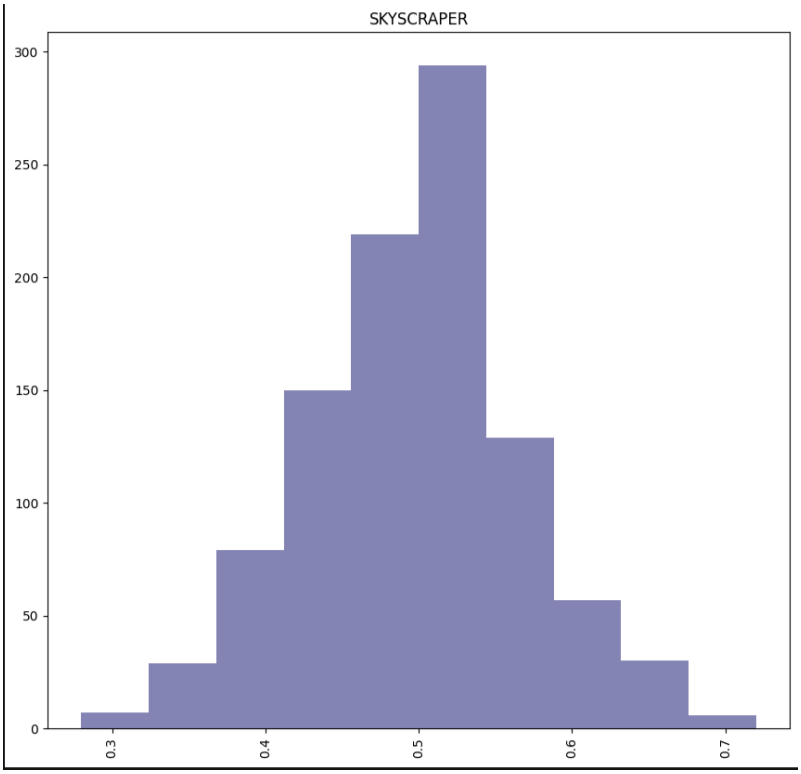

Figura 14. Histograma da porcentagem de vitórias dos atacantes em Skyscraper

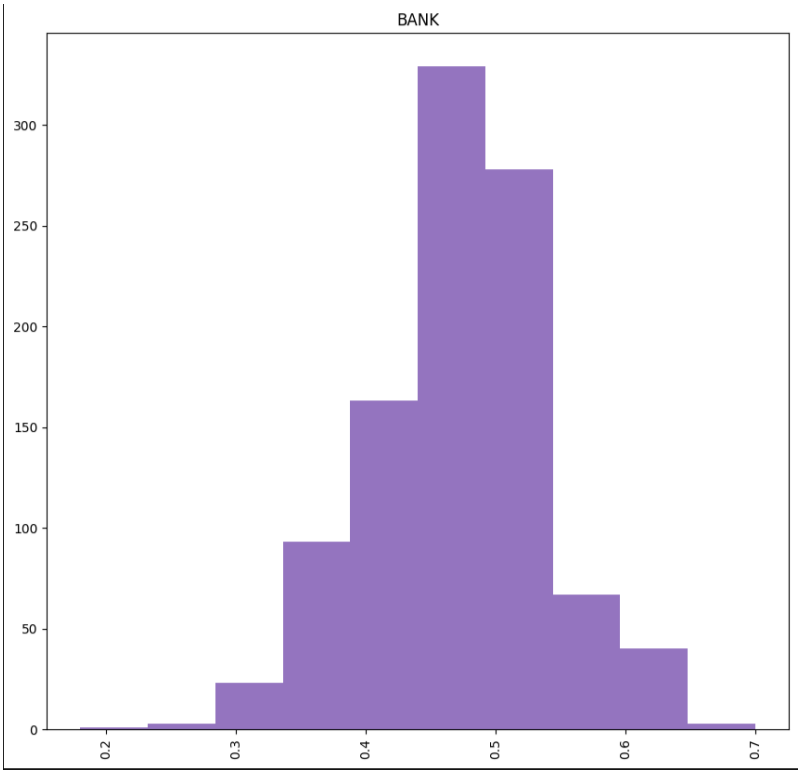

Figura 15. Histograma da porcentagem de vitórias dos atacantes em Bank

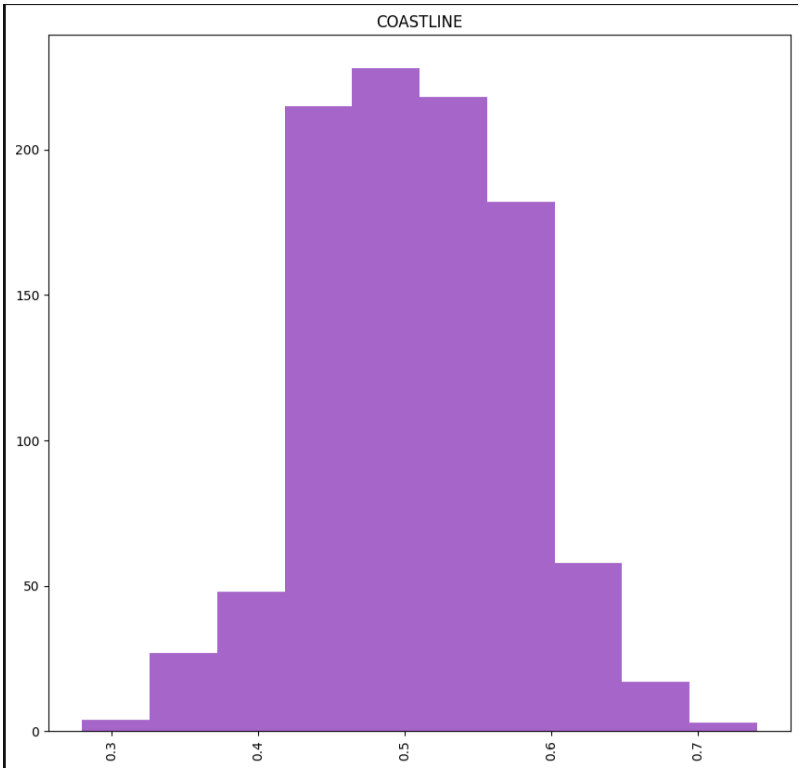

Figura 16. Histograma da porcentagem de vitórias dos atacantes em Coastline

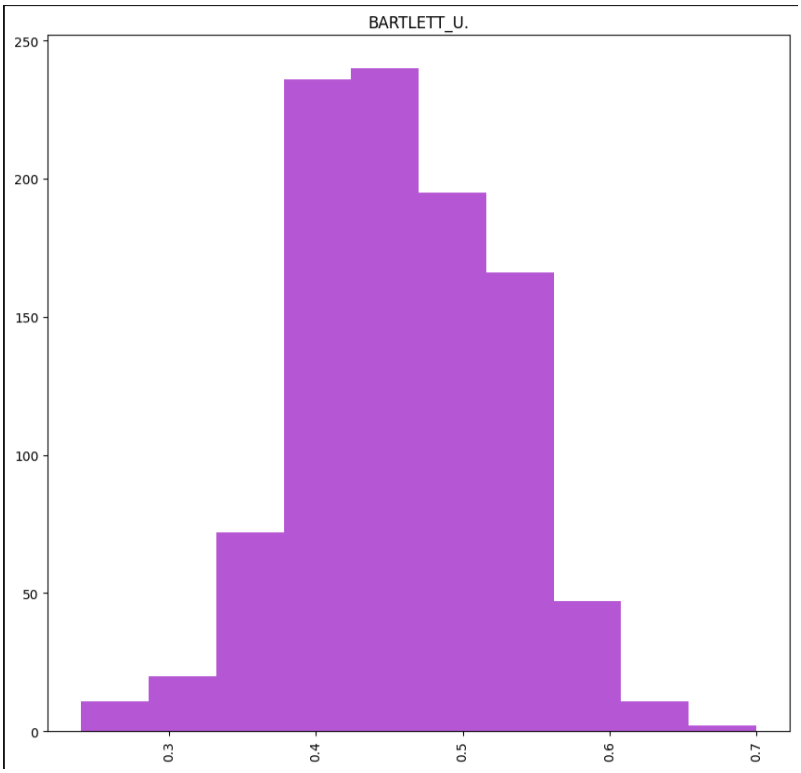

Figura 17. Histograma da porcentagem de vitórias dos atacantes em Barlett U.

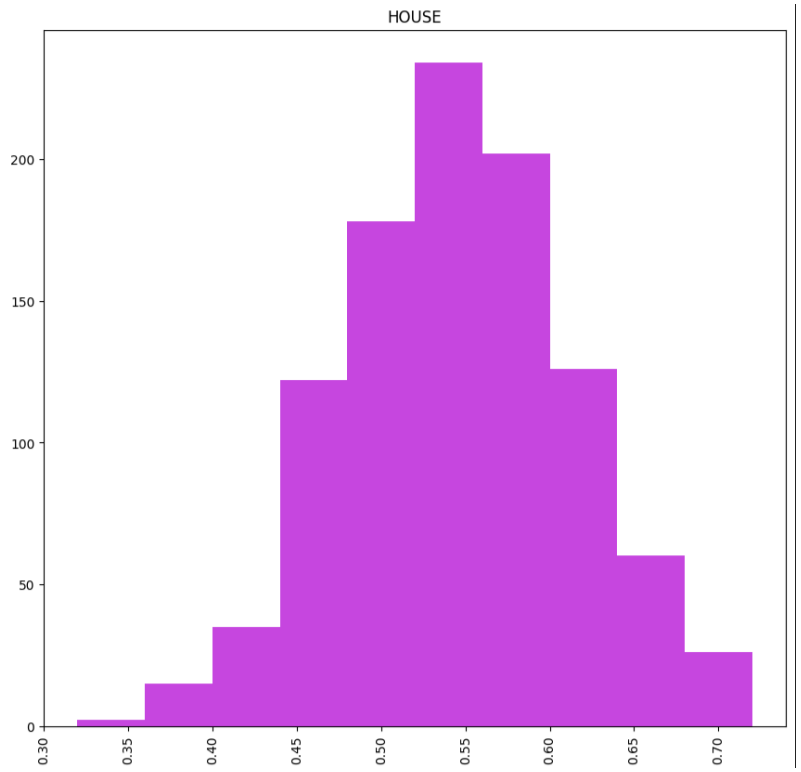

Figura 18. Histograma da porcentagem de vitórias dos atacantes em House

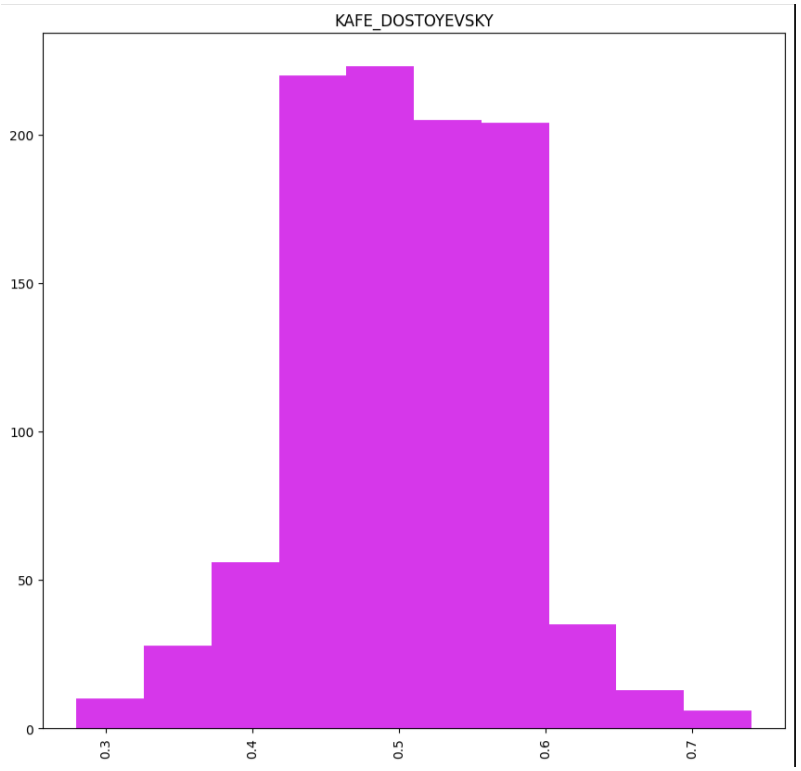

Figura 19. Histograma da porcentagem de vitórias dos atacantes em Kafe Dostoyevski

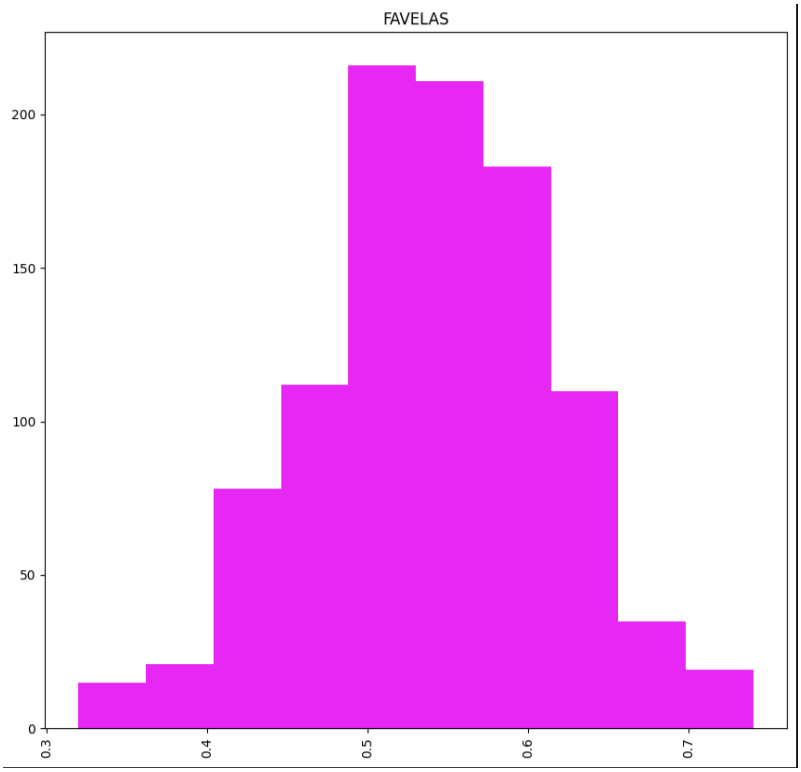

Figura 20. Histograma da porcentagem de vitórias dos atacantes em Favelas

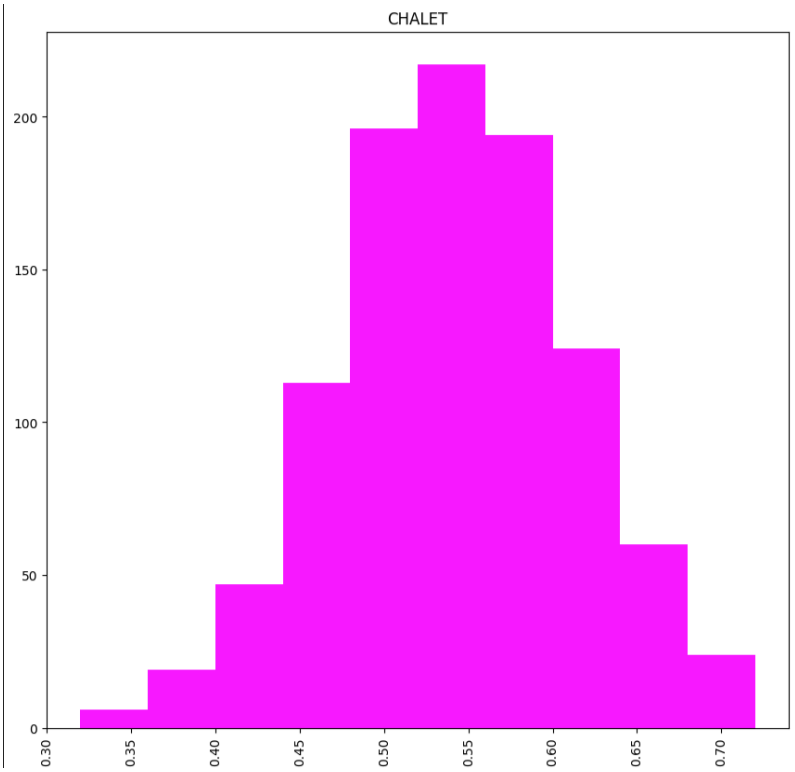

Figura 21. Histograma da porcentagem de vitórias dos atacantes em Chalet

Retiramos tais intervalos de confiança de cada mapa:

Tabela 1. IC (95%) para a taxa de vitória dos atacantes por mapa

Mapa	Intervalo de confiança
Club House	$[50{,}14\%,\ 51{,}02\%]$
Plane	$[47{,}68\%,\ 48{,}57\%]$
Kanal	$[48{,}34\%,\ 49{,}24\%]$
Hereford Base	$[47{,}78\%,\ 48{,}65\%]$
Consulate	$[48{,}33\%,\ 49{,}21\%]$
Yacht	$[49{,}88\%,\ 50{,}74\%]$
Oregon	$[48{,}81\%,\ 49{,}68\%]$
Border	$[48{,}85\%,\ 49{,}74\%]$
Skyscraper	$[49{,}04\%,\ 49{,}92\%]$
Bank	$[46{,}50\%,\ 47{,}37\%]$
Coastline	$[50{,}30\%,\ 51{,}16\%]$
Barlett U.	$[45{,}36\%,\ 46{,}23\%]$
House	$[53{,}14\%,\ 53{,}98\%]$
Kafe Dostoyevski	$[49{,}82\%,\ 50{,}70\%]$
Favelas	$[53{,}53\%,\ 54{,}43\%]$
Chalet	$[52{,}78\%,\ 53{,}66\%]$

6.1.3. Conclusão

Com os resultados condensados, podemos avaliar qual hipótese é válida. Como cada mapa tem seu próprio valor, agrupamos eles junto de qual hipóteses é verdadeira para seu caso.

- **Hipótese Nula (H_0):** Club House, Yacht, Border, Skyscraper, Barlett U., Kafe Dostoyevski;
- **Hipótese Alternativa 1(H_1):** Coastline, House, Favelas, Chalet;
- **Hipótese Alternativa 2(H_2):** Plane, Kanal, Hereford Base, Consulate, Oregon, Bank.

É notável que poucos mapas favorecem o time dos atacantes. Pelo contrário, há mais mapas que favorecem defensores que atacantes (6 mapas contra 4 mapas). No entanto, há mais mapas bem balanceados que as outras categorias, mas por pouco. Se juntarmos os grupos das duas hipóteses em uma categoria, onde pelo menos um time é desfavorecido, então a maioria dos mapas são desbalanceados.

Portanto, concluimos que há uma boa gama de mapas que precisariam de uma revisão, pois eles não permitem um jogo justo entre as equipes. Mas, ainda há uma boa seleção que cumpre com a proposta de ser uma arena entre os jogadores, onde o mais habilidoso deveria sempre vencer.

6.2. Uso do Operador Jager

6.2.1. Observação

Como observado na análise exploratória de dados, Jager é o operador popular entre comunidade de R6. Entretanto, observamos que nos ranks mais altos (Platina e Diamante), a proporção do uso do operador em questão é ainda maior.

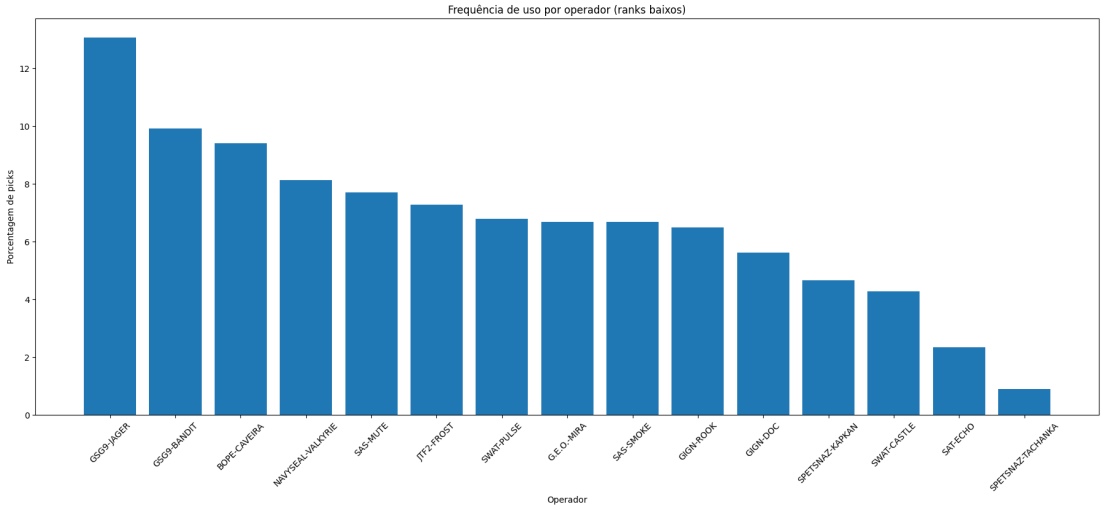

Figura 22. Uso de operadores de defesa em ranks baixos

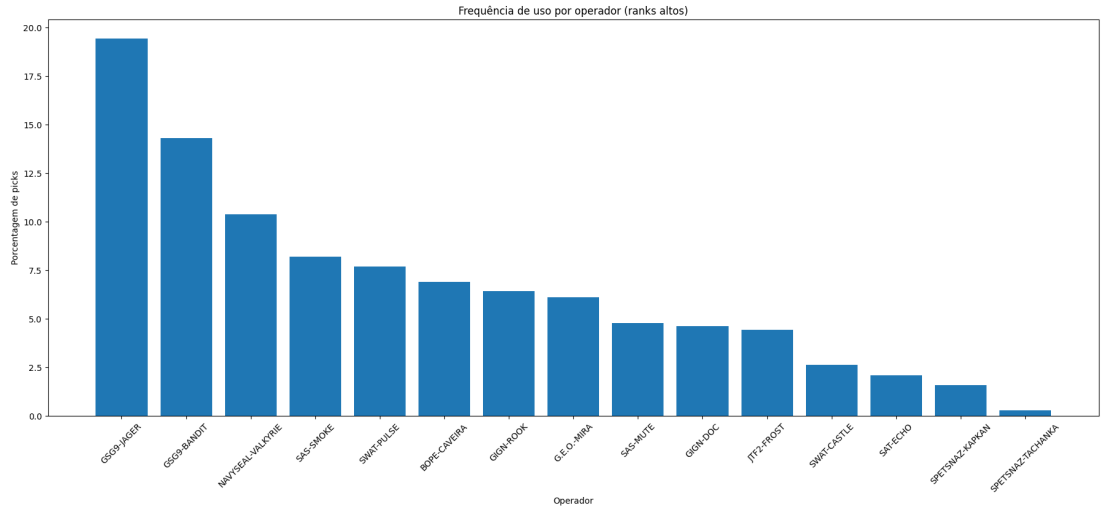

Figura 23. Uso de operadores de defesa em ranks altos

Analisando os gráficos acima, é evidente que a taxa de uso do Jager em ranks baixos (cerca de 13%) é menor do que taxa de uso em ranks altos (cerca de 19%). Portanto, foi observado que jogadores nos ranks Platina e Diamante tem uma preferência maior pelo operador Jager (quando estão jogando como defensores) do que os players de de ranks baixos. Isso deu origem a uma pergunta !

Pergunta 2. *Jogadores de ranks altos usam o operador Jager em uma proporção maior do que os jogadores de ranks baixos ?*

6.2.2. Metodologia

Para responder essa pergunta foi feito o seguinte teste de hipótese:

- **Hipótese Nula (H_0):** O operador *Jäger* tem a mesma taxa de escolha em ranks baixos e altos.
- **Hipótese Alternativa (H_1):** O operador *Jäger* tem taxas de escolha diferentes entre ranks baixos e altos.

Com o intuito de realizar o teste de hipótese, foi construido o intervalo de confiança para a porcentagem de uso do operador Jager em ranks baixos e altos.

Os intervalos de confiança foram construido via bootstrap. Foram constuidas 10.000 permutações com reposição do conjunto de dados. A partir delas, as estatísticas da proporção do uso do operador Jager tanto em ranks baixos e altos foram coletadas. Por fim chegamos nos seguintes IC's (95%):

Tabela 2. IC (95%) para a proporção de uso do operador *Jager*

Grupo de jogadores	Intervalo de confiança
Ranks baixos	$[0{,}1289,\ 0{,}1311]$
Ranks altos	$[0{,}1906,\ 0{,}1973]$

É importante mencionar que pelo fato do operador Jager ser defensor, nas análises realizadas só foram considerados de operadores de defesa.

Com as estatísticas das permutações computadas utilizando o algoritmo bootstrap, fizemos um boxplot com a intenção de deixar os resultados obtidos mais fáceis de interpretar.

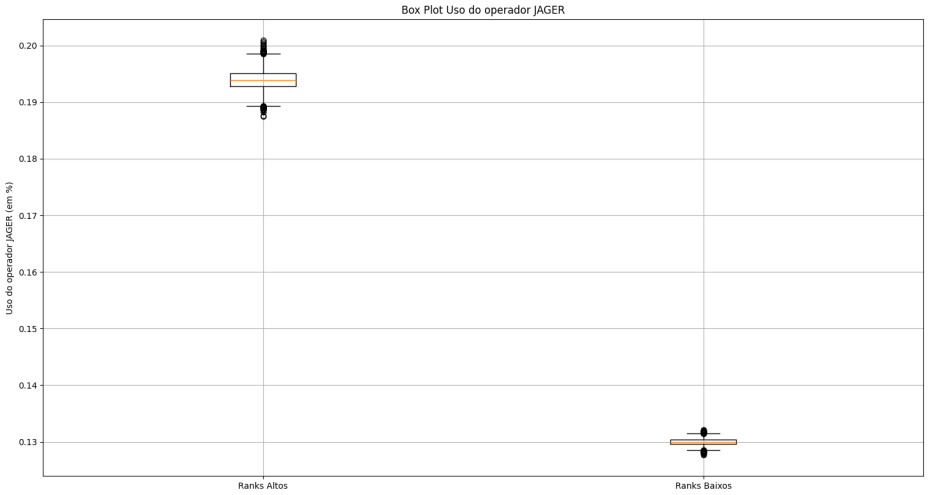

Figura 24. Boxplot da proporção de uso do operador Jager

6.2.3. Conclusão

Percebe-se observando o boxplot e a tabela com os intervalos de confiança que os IC's da proporção de uso do operador Jager em ranks baixos e altos são conjuntos totalmente disjuntos. Além disso, a diferença entre os IC's também é significativa. Dessa maneira é possível rejeitar a hipótese nula e concluir que o operador Jager é selecionado com uma maior frequência nos ranks mais altos.

É possível teorizar os motivos pelos quais o uso do Jager é maior nos ranks mais altos. No entanto, antes de tudo, é necessário caracterizar o operador em questão. O Jager é um operador cujo o gadget (Active Defense System) é capaz de neutralizar granadas e projéteis ofensivos dos atacantes. Isso faz com que o mesmo seja crucial para controlar áreas do mapa e também para diminuir o impacto das ações ofensivas dos atacantes. Além de tudo isso, o Jager é bastante veloz e é equipado com a carabina 416-C (uma das aramas com o maior potencial de eliminação de todo o R6).

Considerando que em ranks baixos os jogos tendem a ser menos táticos e orientados a combates diretos, o gadget do Jager passa a não ter uma utilidade significativa. Com isso, é possível especular que o Jager continua sendo um operador popular nos ranks baixos por ser ágil e possuir um armamento excelente. Todavia, teorizamos que o Jager é ainda mais utilizado nos ranks mais altos pelo fato de o seu gadget possuir um grande impacto em rounds coordenados e pelos motivos mencionados anteriormente.

7. Classificação e Regressão

É importante dizer que tanto no modelo de classificação quanto no de regressão, foi feito um undersample randomizado do dataset. Isso foi feito pelo fato de que não era computacionalmente viável construir os modelos a partir de todo o dataset.

7.1. Modelo de Classificação

7.1.1. Objetivo

Gostaríamos de verificar se é possível classificar um jogador por *rank* de acordo com as suas estatísticas. Logo, vimos uma ótima oportunidade para implementar uma modelação de classificação capaz de determinar o nível de habilidade do jogador.

7.1.2. Tratamento de Dados e algoritmo usado

Primeiramente, as colunas `dateid`, `mapname`, `matchid`, `roundnumber`, `objectivelocation`, `winrole`, `role`, `team`, `endroundreason` foram removidas do conjunto de dados que foi utilizado para o desenvolvimento do modelo. Essas colunas foram removidas pois o objetivo dessa classificação é utilizar as colunas `platform`, `gamemode`, `roundduration`, `clearancelevel`, `skillrank`, `haswon`, `operator`, `nbkills`, `isdead` e `speciality` para realizá-la. Isso porque o trabalho muito se baseia no uso dessas informações e suas relações com o rank do jogador.

Em sequência, foi feito o *one hot encoding* das colunas `operator`, `gamemode`, `platform`, `SPECIALTY`. Performamos essa alteração no *dataset* pois elas estão em formato textual, e só é possível treinar modelos de classificação com variáveis numéricas. Portanto, foi necessário transformar as variáveis categóricas em variáveis numéricas.

É importante dizer que o *dataset* foi dividido em 80% para treino e teste e 20% para validação e o algoritmo de classificação usado foi o random forest, um algoritmo de *machine learning* que usa múltiplas árvores de decisão para realizar predições.

7.1.3. Modelo Original

A primeira tentativa de modelo para classificar os jogadores de acordo com seu *rank* foi utilizando todo o *dataset* somente com as alterações realizadas na subseção Tratamento de Dados e algoritmo usado, obtendo os seguintes resultados:

Tabela 3. Acurácia do modelo na classificação dos ranks dos jogadores de R6

Classe	Acurácia	Corretas	Total
Bronze	0,0186	6.111	329.390
Copper	0,3558	116.761	328.154
Diamond	0,0108	203	18.829
Gold	0,8202	726.756	886.042
Platinum	0,1941	57.947	298.559
Silver	0,0711	35.570	500.489
Unranked	0,2725	48.005	176.187
Acurácia Geral	**0,3907**	–	–

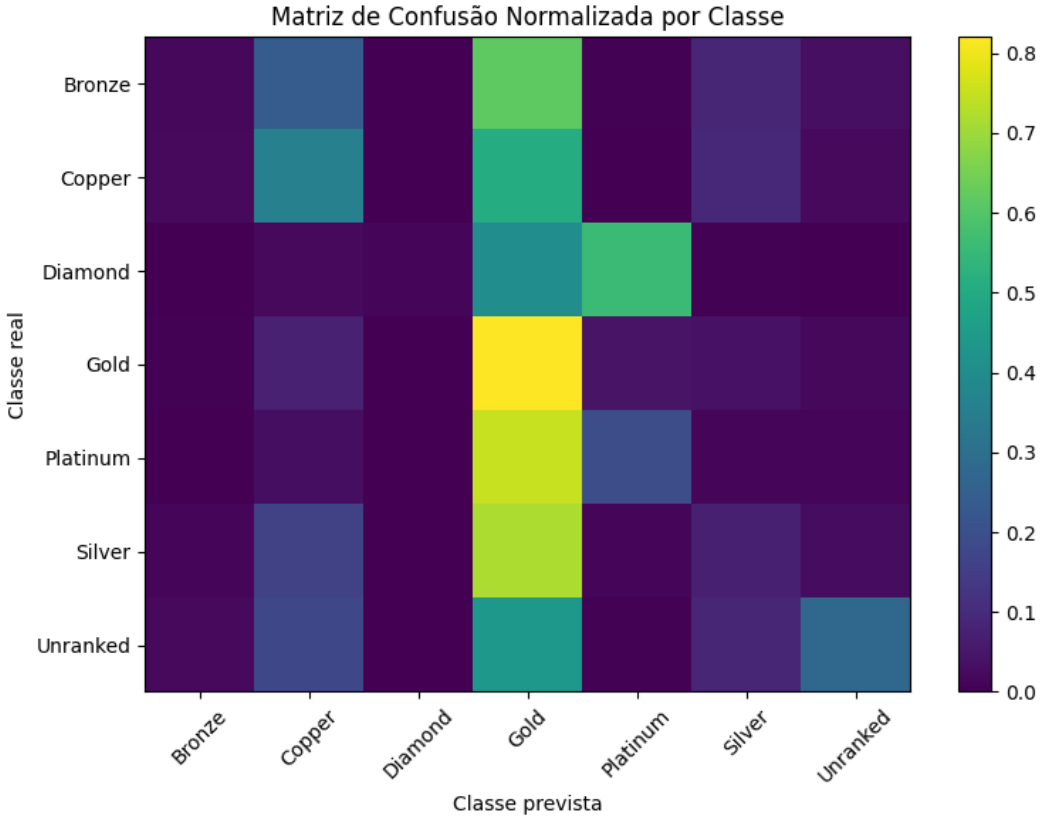

Figura 25. Matriz de Confusão do modelo de classificação original

Analisando os resultados obtidos acima, ficou evidente que os resultados ficaram agrupados na classe Gold, uma vez que este era o *rank* com o maior número de instâncias. Isso mostra um possível viés do modelo.

7.1.4. Modelo com Agrupamento

A segunda abordagem de modelo de classificação foi agrupar as classificações a fim de diminuir a diferença de instânicas em cada classe, diminuir o viés do modelo e melhorar sua acurácia. Foram feitos os seguintes agrupamentos: `Copper--Unranked` e `Diamond--Platinum`.

A partir do modelo treinado com a base de dados agrupada, foram obtidos os seguintes resultados:

Tabela 4. Acurácia do modelo na classificação dos ranks dos jogadores de R6

Classe Agrupada	Acurácia	Corretas	Total
Bronze	0,0063	2.065	329.390
Copper–Unranked	0,6083	306.766	504.341
Diamond–Platinum	0,2339	74.226	317.388
Gold	0,7590	672.478	886.042
Silver	0,0254	12.733	500.489
Acurácia Geral	**0,4210**	–	–

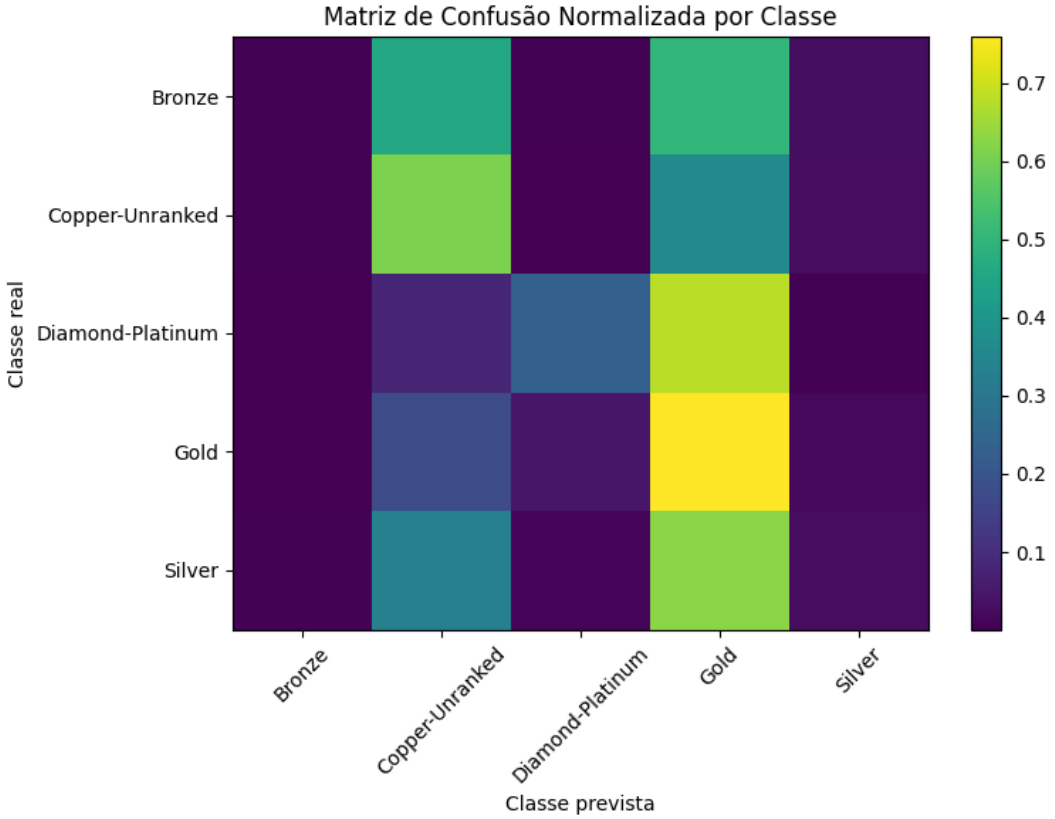

Figura 26. Matriz de Confusão do modelo de classificação com agrupamento

Um aspecto desse modelo é que o erro se agrupa em duas classes dominantes (`Copper-Unranked`, `Gold`, uma vez que são as duas classes com maior quantidade de instâncias).

7.1.5. Modelo Binário

Por fim, criamos dois agrupamentos: casual (`Unranked`, `Copper`, `Bronze`, `Silver`) e competitivo (`Gold`, `Platinum`, `Diamond`). O modelo de classificação foi treinado considerando a base de dados agrupada.

A partir do modelo binário, obtivemos os seguintes resultados:

Tabela 5. Acurácia do modelo na classificação entre modos de jogo

Modo de Jogo	Acurácia	Corretas	Total
Casual	0,7754	1.034.516	1.334.220
Competitivo	0,6287	756.586	1.203.430
Acurácia Geral	**0,7058**	–	–

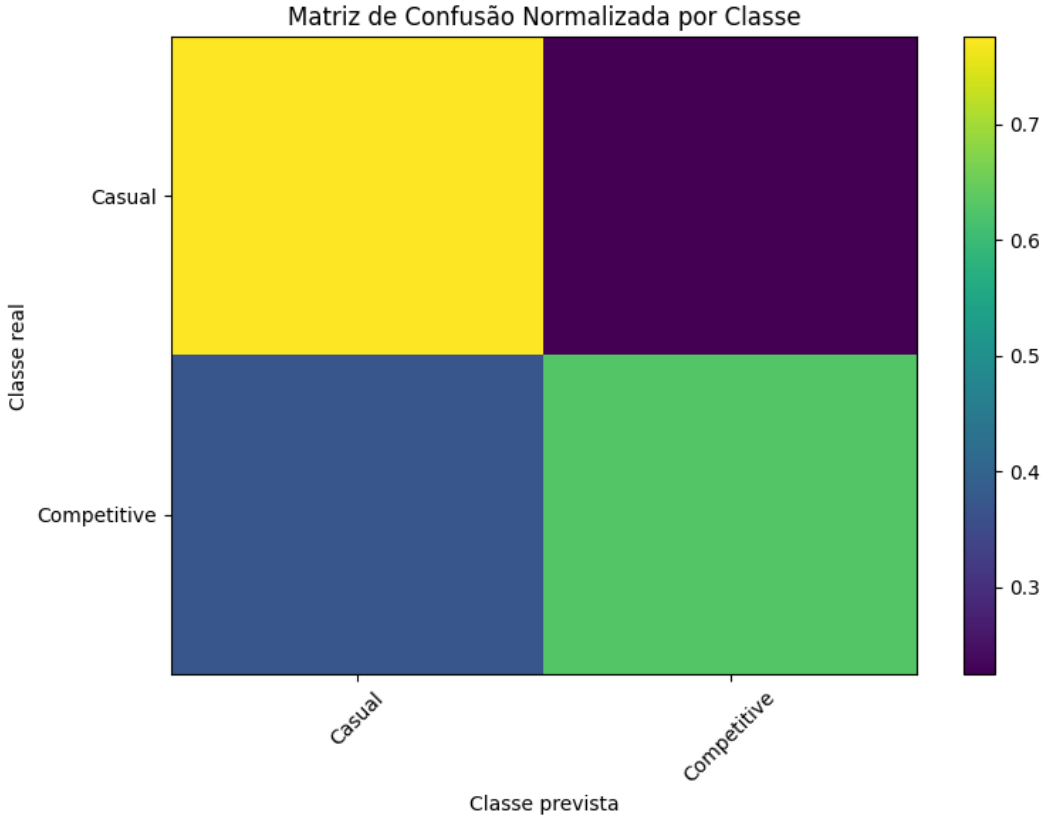

Figura 27. Matriz de Confusão do modelo de classificação com replacement e agrupamento

Analisando a matriz de confusão, é interessante notar que o modelo mais erra ao falar que uma instância competitiva é casual, do que uma casual é competitiva.

7.1.6. Modelo com Replacement

A terceira abordagem de modelo de classificação foi construir um modelo treinado com uma base de dados com *replacement*. Com isso, o número de instâncias não iria variar significativamente de *rank* para *rank*.

A partir do modelo treinado com a base de dados em que foi efetuado o *replacement*, obtivemos os seguintes resultados:

Tabela 6. Acurácia do modelo na classificação dos ranks dos jogadores de R6

Classe Agrupada	Acurácia	Corretas	Total
Bronze	0,2056	65.174	317.046
Copper–Unranked	0,5824	184.023	315.966
Diamond–Platinum	0,6257	198.303	316.919
Gold	0,2467	78.304	317.444
Silver	0,1718	54.521	317.362
Acurácia Geral	**0,3662**	–	–

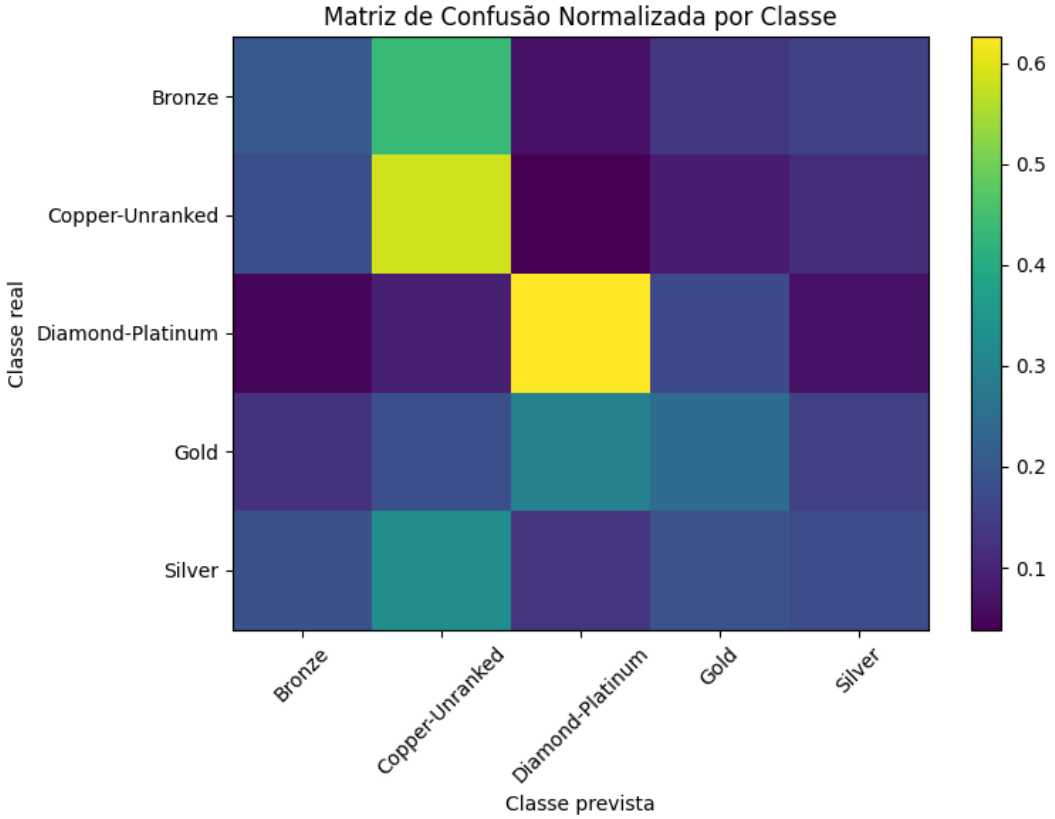

Figura 28. Matriz de Confusão do modelo de classificação com replacement e agrupamento

7.1.7. Modelo Binário

Por fim, criamos dois agrupamentos: casual (`Unranked, Copper, Bronze, Silver`) e competitivo (`Gold, Platinum, Diamond`). O modelo de classificação foi treinado considerando a base de dados agrupada.

A partir do modelo binário, obtivemos os seguintes resultados:

Tabela 7. Acurácia do modelo na classificação entre modos de jogo

Modo de Jogo	Acurácia	Corretas	Total
Casual	0,7754	1.034.516	1.334.220
Competitivo	0,6287	756.586	1.203.430
Acurácia Geral	**0,7058**	–	–

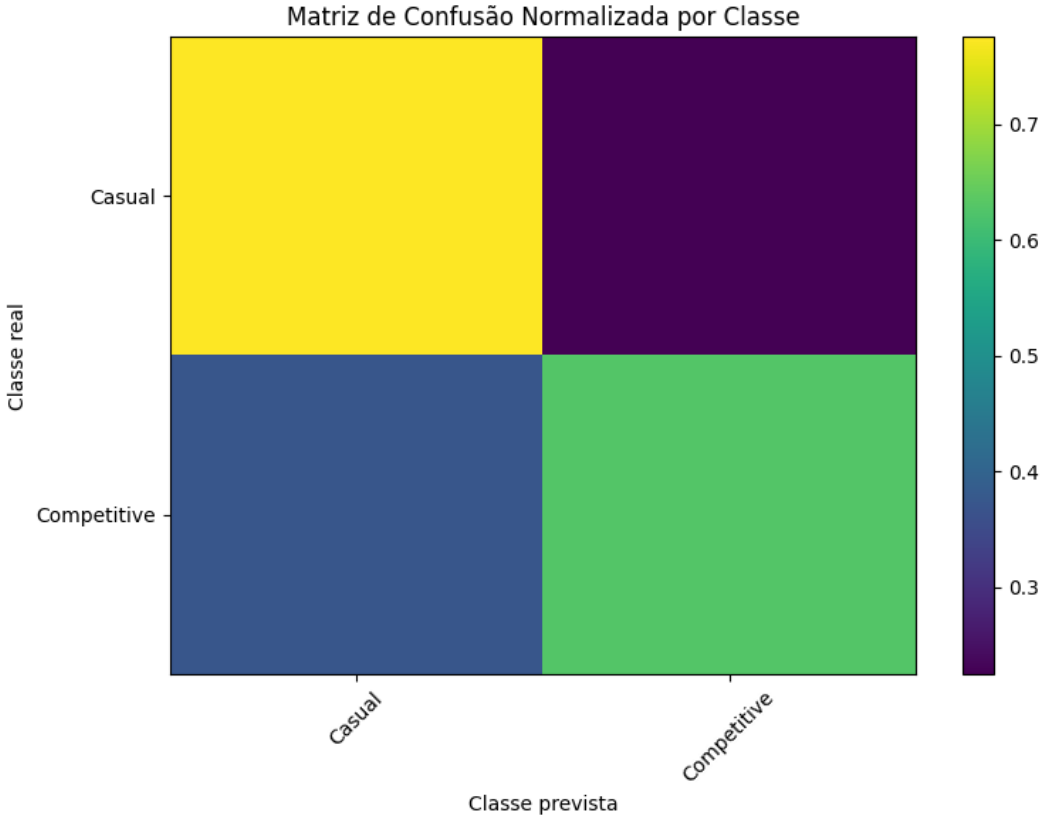

Figura 29. Matriz de Confusão do modelo de classificação com replacement e agrupamento

7.2. Modelo de Regressão

7.2.1. Objetivo

Durante as análises de dados, percebeu-se que o tempo médio de duração de round variava de acordo com o rank dos jogadores. Ademais, notamos que existe um comportamento linear entre o aumento do tempo médio de um round e o rank dos jogadores. Diante disso, surgiu a seguinte pergunta:

Pergunta 3. *É possível prever a duração de um round ?*

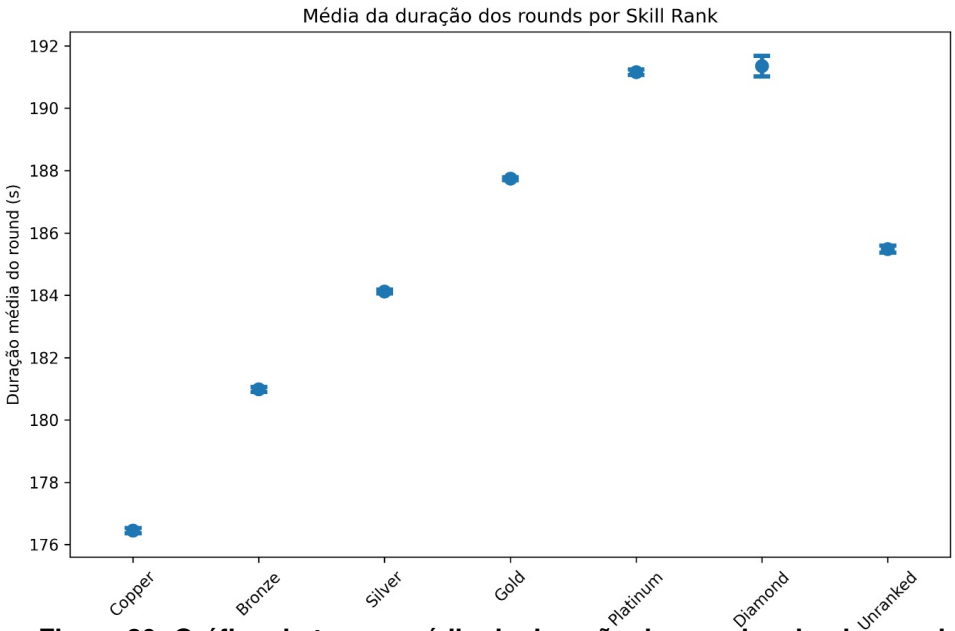

Figura 30. Gráfico do tempo médio de duração de round variando o rank

7.2.2. Tratamento de Dados

A primeira medida de tratamento de dados foi remover todos os rounds que duravam menos de 30 segundos ou mais de 300 segundos, uma vez que eles representam ruído.

Pelo fato das colunas `skillrank`, `mapname`, `gamemode`, `winrole`, `endroundreason`, `role`, `platform` armazenarem valores categóricos performamos um original enconding (enumeração de 1 a n dos valores da coluna, sendo n o número de valores distintos que a coluna possui) nessas mesmas colunas. O original enconding foi escolhido ao invés do one hot enconding pelo fato de que o dataset produzido fica menor, assim viabilizando trabalhar com colunas que podem assumir muitos valores (como `mapname`). Com tudo isso, treinar os modelos de regressão se torna uma tarefa com um custo computacional um pouco menor.

As colunas selecionadas para treinar o modelo de regressão foram `skillrank`, `mapname`, `gamemode`, `winrole`, `endroundreason`, `role`, `platform`

Em seguida, fizemos com que o dataset de treino tivesse apenas as seguintes colunas `skillrank`, `mapname`, `gamemode`, `winrole`, `endroundreason`, `role`, `platform`. As colunas `skillrank`, `mapname`, `gamemode`, `endroundreason`, `winrole`, `role` foram selecionadas pelo fato de que elas estão diratamente relacionadas com as estratégias e com o estilo de jogo dos jogadores, que são aspectos que contribuem significativamente para a duração do round. A coluna `platform`, apesar de não ser um fator do round, pode afetar sim a o tamanho de um round.

7.3. Regressão Linear

Foi feita uma regressão linear sobre o conjunto de dados tratado. Os seguintes resultados foram obtidos:

Tabela 8. Métricas da Regressão Linear

Métrica	Valor
Erro Quadrático Médio (MSE)	1247,40
Coeficiente de Determinação (R^2)	0,209

A partir dos resultados acima, percebemos que a cada round o modelo errou em média 35 segundos. Entretanto, 35 segundos é uma quantidade significativa considerando a duração de rounds de R6. Desse modo, é afirma-se que o nosso modelo de regressão possui uma acurácia baixa.

Diante desse resultado, resolvemos transformar o modelo de classificação que vai classificar os rounds em rouds curtos, médios e longos, com o intuito de melhorar o mesmo.

7.4. Classificação

Definimos que: rounds com menos de 90 segundos são curtos, rounds com mais de 90 segundos e 180 segundos são médios e rounds com mais de 180 segundos são longos.

A partir do mesmo dataset utilizado para treinar o modelo de regressão, foi treinado um modelo de classificação cujo objetivo é tentar definir se um round será curto, médio ou longo. Com esse modelo de classificação, foram obtidos os seguintes resultados:

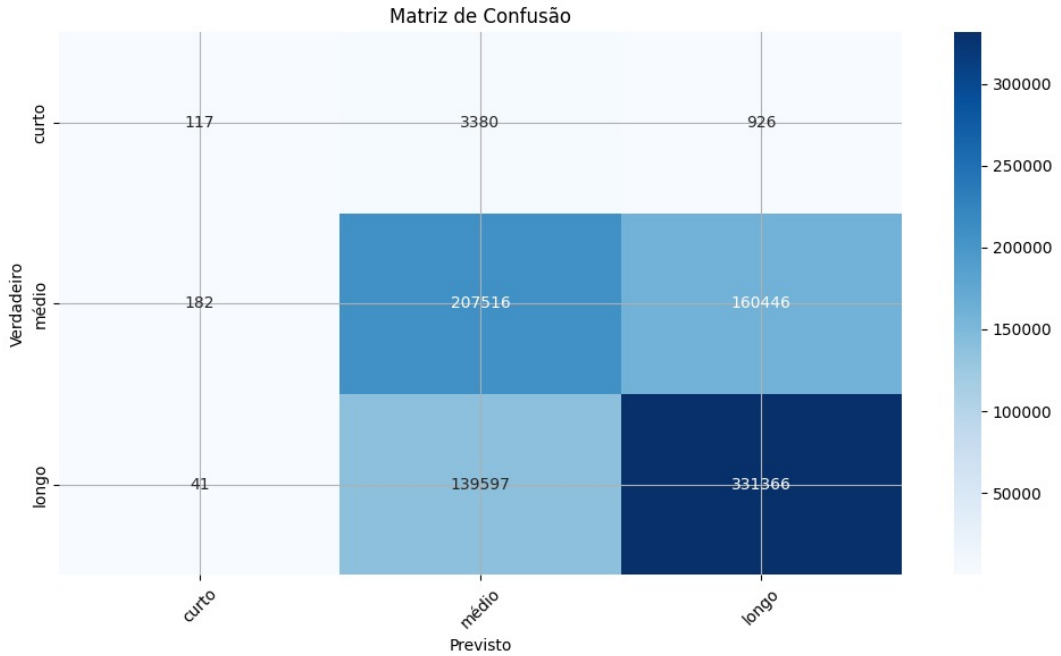

Figura 31. Matriz de confusão do modelo de classificação por tempo

Tabela 9. Relatório de Classificação do Modelo

Classe	Precisão	Revocação	F1-score	Suporte
Curto	0,34	0,03	0,05	4423
Longo	0,67	0,70	0,69	471004
Médio	0,59	0,56	0,58	368144
Média macro	0,54	0,43	0,44	843571
Média ponderada	0,64	0,64	0,64	843571

Acurácia geral: 0,64

Interpretando os resultados acima, nota-se que o a acurácia do modelo para rounds curtos é baixa, uma vez que o modelo não arrisca classificr rounds como curtos. Ademais, percebe-se que a acurácia do modelo para classificar rounds médios e longos é aceitável.

Portanto, fica evidente, que os resultados desse modelo foram melhores do que os do modelo anterior.

8. Conclusão

Por fim, como apresentado ao longo desse relatório, Rainbow Six Siege em que algumas estatísticas mudam significativamente ao fazer uma análise por rank. Além disso, foi visto que o jogo possui algumas pequenas falhas de game design, como mapas que favorecem a equipe atacante ou defensora.

Esse trabalho foi enriquecedor uma vez que ele exigiu que nós implementassemos técnicas clássicas de análise de dados. Também tivemos a oportunidade de botar os nossos conhecimentos de python em prática, o que foi bastante interessante.

Um dos principais desafios desse TP foi lidar com um dataset grande. Isso nos obrigou a explorar alternativas que tornassem nossas análises viáveis e que fossem executadas em uma quantidade de tempo efetiva. Outro desafio inerente ao trabalho foi ter o senso crítico de como explorar os dados de forma a econtrar padrões e relações.

No final, nos sentimos satisfeitos com o trabalho, e acreditamos que crescemos como estudantes na área de computação.