

Análise de avaliações de jogos: O que afeta a opinião dos fãs?

Modelagem Estatística

George Dutra

Ciência de Dados e I.A. - 5º Período



Escola de Matemática Aplicada
Fundação Getulio Vargas
Brasil
2024

Sumário

1	Resumo	2
2	Introdução	2
2.1	Objetivo	2
2.2	Método	2
3	Conjunto de Dados	3
3.1	Tratamento	3
3.2	Análise Exploratória	4
4	Modelos	8
4.1	Previsão de Rating	8
4.1.1	Beta	8
4.2	Previsão do MetaCritic	11
4.3	Beta	11
5	Conclusão	12
A	Descrição do Dataset	14

1 Resumo

Este projeto busca utilizar dados de uma rede social de avaliação de jogos digitais para analisar os efeitos de diferentes variáveis (como tempo de jogo, gênero, número de conquistas) na opinião pública a respeito dos jogos. O objetivo deste estudo é discernir quais decisões de negócio mais afetam o desempenho de um jogo durante seu desenvolvimento. Para isto, tais variáveis serão aplicadas em GLM's (Modelos Lineares Generalizados) e analisados segundo algumas das métricas aprendidas ao longo do curso.

2 Introdução

Neste projeto, foi definido como objeto de estudo um conjunto de dados retirados do site RAWG.io[1], um site/rede social com foco em jogos digitais, em que usuários podem avaliar, comentar e discutir os jogos que eles jogaram, seja no computador ou em qualquer outra plataforma. Através da [API](#) disponibilizada pelo site, foi possível extrair dados referentes aos jogos que o site possuía até o ano de 2020.

2.1 Objetivo

O objetivo do projeto é analisar o efeito de algumas variáveis na avaliação pública do jogo. Esse tipo de análise seria útil para, por exemplo, uma desenvolvedora de jogos que deseja saber quais decisões de negócio tomar durante o desenvolvimento de um jogo, no intuito de maximizar a esperança de sua popularidade.

O site nos entrega duas informações que serão tratadas como variáveis objetivo: a avaliação média do MetaCritics, e a avaliação média dos usuários do site. Ambas as avaliações seguem o modelo de estrelas muito comum nesse tipo de plataformas, ou seja, cada avaliador escolhe um valor inteiro entre 0 e 5 para avaliar o jogo, sendo 0 o valor mais negativo (no sentido de não gostar da obra) e 5 o valor mais positivo (significando que o jogador amou a obra). Por se tratar da média, a variável de interesse estará sempre entre 0 e 5, o que será levado em consideração para a análise.

Tais variáveis foram escolhidas pois retratam bem o interesse das desenvolvedoras ao lançar um jogo. O MetaCritics é um comitê de “profissionais” que avaliam e publicam críticas de jogos, e ter uma nota alta com esse órgão auxilia na popularidade do jogo e nas indicações do estúdio para concorrer a prêmios, que podem levar a contratos de publicidade e outros benefícios. A avaliação pública, é claro, traduz diretamente na satisfação dos jogadores e reflete o sucesso das vendas do jogo.

2.2 Método

Visto a natureza dos dados de entrada e das variáveis de interesse, opta-se por utilizar de GLM's (Modelos Lineares Generalizados) para desenvolver-se o estudo deste trabalho. Os objetivos principais com o uso do GLM são a capacidade de limitar o espaço das variáveis preditas, e lidar com o formato dos dados, especialmente quanto às variáveis categóricas não auto-excludentes.

Em particular, serão utilizados modelos de GLM de família

3 Conjunto de Dados

Em resumo, o dataset se trata de uma tabela de 474.417 linhas e 27 colunas. Cada linha se refere a um jogo, e cada coluna é um tipo de dado diferente. Muitas das colunas são informação inútil para a análise, ou precisam ser tratadas para poderem ser utilizadas. A seguir, serão descritos alguns dos tratamentos realizados, e serão citados nomes e características de colunas que podem ser conferidos mais a fundo no Apêndice A. O processo de tratamento de dados pode ser conferido no [Repositório GitHub\[2\]](#) do projeto.

3.1 Tratamento

Em relação aos dados, podemos descartar dados que possuam número de reviews iguais a zero e valor nulo na avaliação do MetaCritic, pois isso indica que não há avaliação de nenhuma das fontes desejadas. Isso já filtra uma grande parte do nosso dataset, mas ainda mantém mais de 30 mil jogos, o que é mais que suficiente para o que desejamos. Além disso, não é interessante para essa análise jogos que ainda não foram lançados, portanto serão removidos todos os jogos que possuem valor verdadeiro na coluna ‘tba’, e esta será removida.

Algumas das colunas se tratam apenas de identificadores dos jogos, como nomes, siglas e apelidos, sendo inúteis para análise e portanto sendo descartadas, pois já existe a garantia de que cada linha é um jogo único. Algumas colunas que envolvem contagem absoluta de usuários (*ratings_count*, *suggestions_count*, *reviews_count*) serão removidas pois não possuem tanta utilidade para a análise, além de arriscarem um alto nível de correlação entre si. Algumas outras colunas também serão removidas somente por não refletirem características interessantes para o problema, como colunas referentes a datas, *website*, *developers* e *publishers*.

A coluna *platforms* mostra uma informação importante, pois no meio de jogos digitais, a plataforma em que os jogos são lançados praticamente segregam a comunidade de *games* em sub-comunidades que costumam ser muito fiéis às suas plataformas favoritas. Algumas dessas plataformas inclusive possuem estratégias de mercado que envolvem contratos de exclusividade de lançamento; em troca de financiamento por parte da empresa, o jogo é lançado somente na plataforma dela, ou lançado com muito atraso nas concorrentes, o que mostra o quão interessante é tal decisão para o lançamento de um jogo. Como existem muitas plataformas, e ainda por cima estão separadas por geração, além de ser uma coluna mal modelada para análise (se trata de uma string com separadores) serão feitos os seguintes tratamentos:

1. Eliminar jogos que não possuem informação de plataforma de lançamento, pois se tratam apenas de cerca de 4 mil jogos, e isso configura informação faltante, afinal todo jogo precisa ser lançado em alguma plataforma;
2. Agregar as gerações segundo as empresas que as comandam, de forma que por exemplo, as classes ‘PlayStation 2’ e ‘PlayStation 3’ sejam tratadas como uma mesma classe ‘PlayStation’, pois ambas se referem à plataformas de uma mesma empresa;
3. Filtrar apenas as plataformas que dominam o mercado atual, sendo essas: Playstation (da Sony), Xbox (da Microsoft), consoles da Nintendo, plataformas Mobile (que é um nicho de mercado bem particular, dominado pelas empresas

Google e Apple), e claro, Computadores (apesar de existirem diferentes lojas de jogos dentro do nicho de computadores, essa informação não está presente no conjunto de dados, se restringindo apenas a indicar o Sistema Operacional com o qual o jogo é compatível, que não é tão relevante). Nos casos que não se encaixam em nenhuma das classes acima, a classe de plataforma do jogo será resumida como "Other", pois não se trata de uma empresa relevante no mercado atual;

4. Devido as características da variável, a string resultante será enfim reinterpretada em 6 colunas de booleanos, cada uma representando o pertencimento à uma das classes descritas acima. A escolha dessa modelagem se dá pois a plataforma se trata de uma variável categórica não excludente, ou seja, é possível que um jogo seja lançado em uma, várias, ou até todas as plataformas.

Todas as colunas de status serão removidas, porém elas serão utilizadas para criar uma nova coluna com uma informação interessante: Dividindo a coluna de status "Jogado, mas não concluído" pela soma dela com a coluna de status "Concluído", é possível saber, dos jogadores que começaram e pararam de jogar o jogo, qual porcentagem deles desistiram no meio do caminho ao invés de concluir o jogo. Esse comportamento no meio de jogos digitais é chamado "*dropar* um jogo", e é um forte indicativo de que o jogo é chato, cansativo, ou que não atendeu às expectativas do jogador.

Por último mas não menos importante, a coluna de gênero é, talvez, a variável mais interessante para o objetivo deste projeto. Se tratando de uma variável categórica não excludente, seu tratamento se assemelhará ao tratamento da coluna de plataforma. A única diferença, é que nenhum gênero será filtrado ou resumido, pois não há um número tão grande, e cada um em particular interessa ao objetivo da análise. Com isso, serão criadas 19 novas colunas booleanas, cada uma referente ao pertencimento ou não do jogo à um determinado gênero dentre os possíveis: Action, Adventure, Puzzle, Platformer, Simulation, Indie, Strategy, RPG, Shooter, Casual, Arcade, Racing, Sports, Educational, Board Games, Fighting, Family, Card, e Massively Multiplayer.

3.2 Análise Exploratória

Para ter alguns *insights* dos dados, foram realizadas algumas análises escolhidas pelo autor para tentar extrair informações úteis dos dados.

A primeira análise feita foi um histograma dos valores das avaliações dos jogos, buscando entender a distribuição dos dados. Foram desconsiderados tanto avaliações do MetaCritics classificadas como NaN (ou seja, jogos não avaliados) quanto jogos cuja avaliação média do público é zero (geralmente causado por jogos com nenhum ou poucos avaliadores). De mais de 33 mil jogos totais, cerca de 22 mil possuem avaliação pública igual a 0, e 29 mil não possuem avaliação do MetaCritics. Ainda assim, com os dados restantes, é possível observar uma distribuição interessante nas avaliações:

Como é possível ver abaixo na figura 1, apesar de possuírem suporte diferente, ambas as avaliações possuem uma distribuição similar. Não há muitas avaliações muito próximas dos extremos, mas a maioria delas se concentra um pouco acima da mediana, demonstrando uma tendência ao otimismo tanto da parte do público quanto da parte dos críticos.

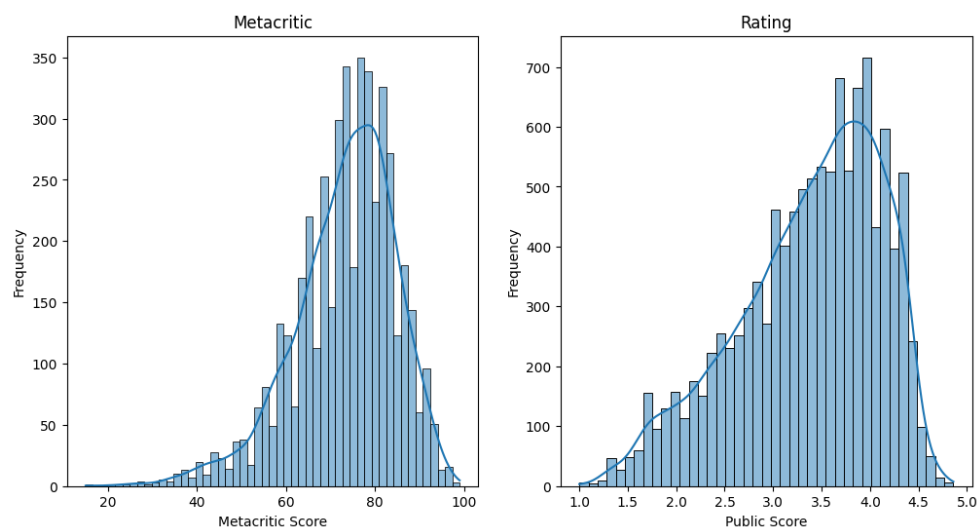


Figura 1: Histogramas de Avaliação

Na figura 2, tentou-se ainda visualizar a correlação par-a-par de algumas variáveis (rating, metacritic, playtime, achievements count, game series count e users drop percent), colocando-as num scatterplot umas contra as outras, mas desse teste só se concluiu o que já era óbvio com os histogramas, que a avaliação do público possui tendência similar à dos críticos.

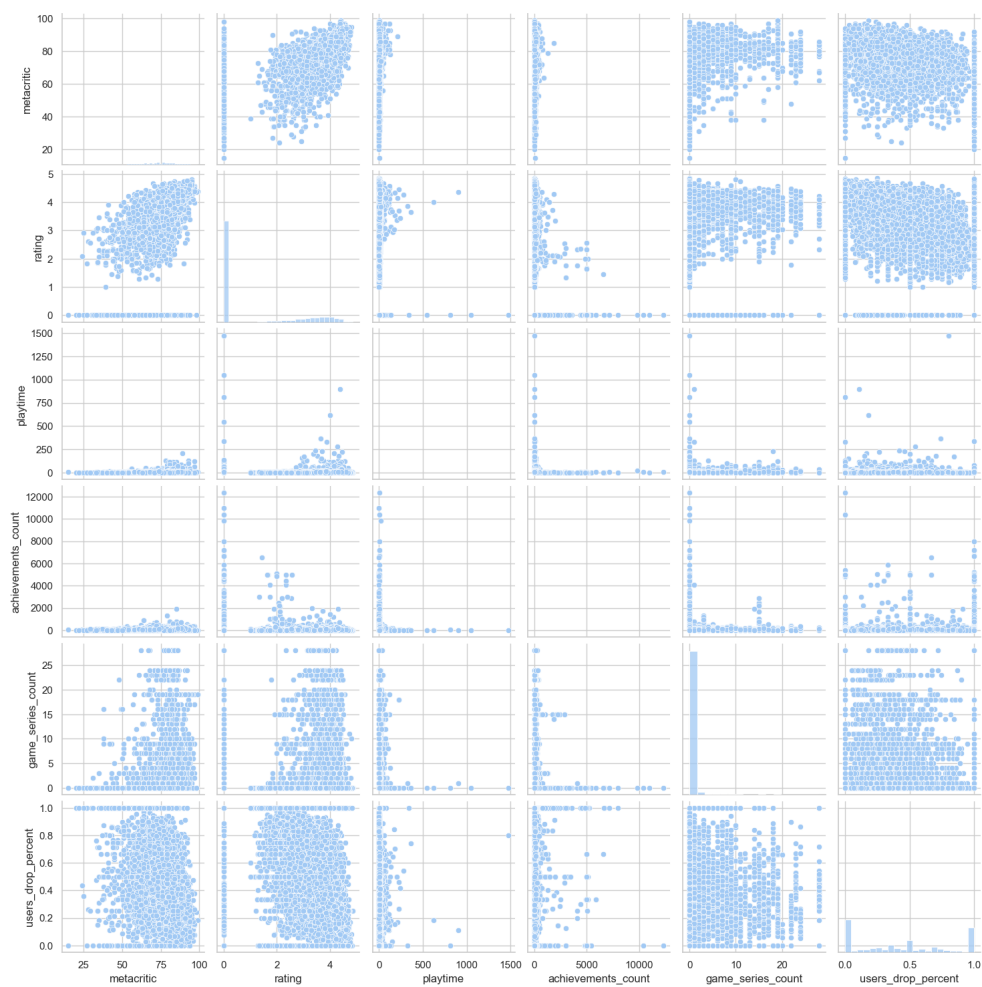


Figura 2: Gráficos de Scatter par-a-par

Na figura 3, observa-se que as avaliações médias tanto por parte do público quanto por parte dos críticos não possuem uma variância muito grande entre gêneros e nem entre plataformas. Porém, é interessante reparar que na ordem de melhor a pior avaliados, há uma certa discordância entre o público e os críticos, tanto na categoria de gêneros quanto nas plataformas. Como não há uma variância grande, no entanto, é difícil inferir conclusões baseadas numa diferença tão pequena.

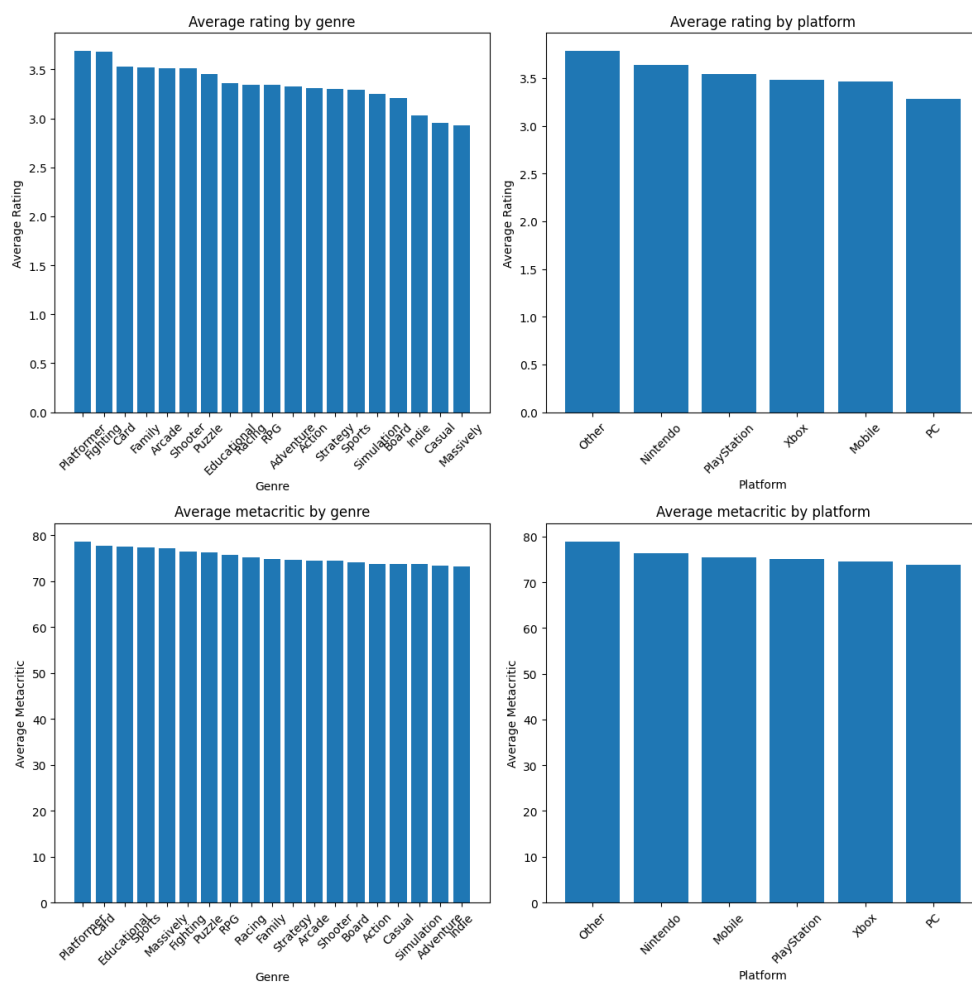


Figura 3: Avaliação Média por Categoria

Por fim, é possível observar pela matriz de correlação de variáveis que há pouca ou nenhuma dependência colinear entre as colunas. Isso é um ponto extremamente positivo para a aplicação do método desejado, pois seria o maior impedimento a um modelo de GLM.

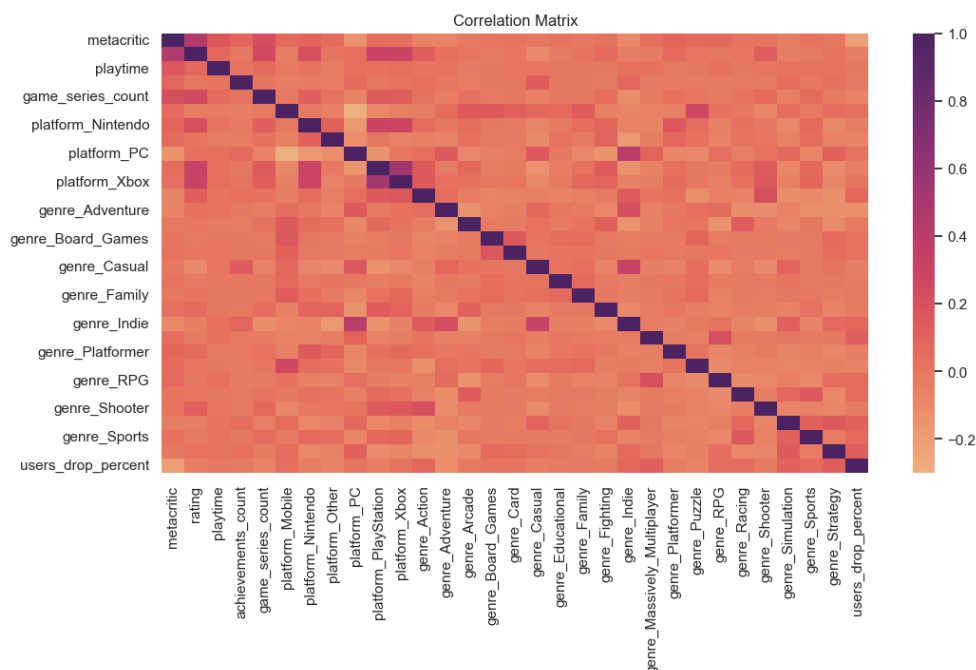


Figura 4: Matriz de correlação das colunas

4 Modelos

4.1 Previsão de Rating

Nessa etapa de exploração, buscou-se entender os efeitos das variáveis na predição da coluna de rating. Por isso, retirou-se a coluna “Metacritic”, por possuir uma correlação significativa com ela, e seguiu-se a aplicação do modelo.

4.1.1 Beta

Analisando a aparência dos dados na análise exploratória, optou-se por iniciar os testes com modelos GLM de família Beta, com função de ligação Logit. Nestes testes, a coluna criada artificialmente de porcentagem de usuários que “droparam” o jogo causava um erro de inversão de matrizes, e por isso foi deixada de lado. No primeiro teste, utilizamos **todas** as variáveis disponíveis. Os resultados de métricas de erro não foram muito ruins:

Type of estimator: ML (maximum likelihood)

Log-likelihood: $1.196e + 05$ on $36Df$

Pseudo R-squared: 0.2006

Number of iterations: $71(BFGS) + 3$ (Fisher scoring)

Como pode-se ver, o Pseudo R-Quadrado possui um valor razoável, apesar deste ser apenas um primeiro teste. Quanto aos efeitos das variáveis, é possível observar dois fatores interessantes. Primeiramente percebe-se que boa parte das variáveis possuem probabilidade baixa de extrapolar o intervalo de confiança definido, com a maioria alcançando um nível menor que $10e-3$, que é considerado muito satisfatório (marcado pelo R com três estrelas). Isso indica um baixo nível de incerteza em relação aos parâmetros estudados, e pode indicar um bom encaixe do modelo aos dados escolhidos.

Além disso, a análise utilizando todas as variáveis demonstra que algumas variáveis levantam mais incerteza que outras, podendo nos indicar o caminho que desejamos de quais variáveis afetam mais a avaliação pública. Até o momento, tais variáveis com menor nível de incerteza foram as plataformas em que o jogo foi lançado e a classificação indicativa dos jogos.

Por fim, também é possível observar a análise do parâmetro *Phi*:

```
Phi coefficients (precision model with identity link):
      Estimate Std. Error z value Pr(>|z|)
(phi)  0.89635    0.00747    120  <2e-16 ***
```

Figura 5: Análise do coeficiente *Phi*

Na sequência, observaram-se os efeitos de certas variáveis aplicadas de forma independente no modelo. Começando pelas classes de plataforma, testaram-se seus efeitos sem interação;

```
Standardized weighted residuals 2:
      Min      IQ  Median      3Q      Max
-2.3070 -0.3217 -0.3217  0.7730  1.2914

Coefficients (mean model with logit link):
              Estimate Std. Error z value Pr(>|z|)
(Intercept)   -1.94545    0.01839 -105.783  < 2e-16 ***
platform_Mobile  0.13222    0.01803   7.332 2.27e-13 ***
platform_Nintendo 0.24769    0.01902  13.026  < 2e-16 ***
platform_Other  0.07319    0.02720   2.690 0.00714 **
platform_PC     0.27546    0.01631  16.892  < 2e-16 ***
platform_PlayStation 0.43500    0.01884  23.092  < 2e-16 ***
platform_Xbox   0.55735    0.02306  24.167  < 2e-16 ***

Phi coefficients (precision model with identity link):
      Estimate Std. Error z value Pr(>|z|)
(phi)  0.838512  0.006904  121.5  <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Type of estimator: ML (maximum likelihood)
Log-likelihood: 1.188e+05 on 8 Df
Pseudo R-squared: 0.1406
Number of iterations: 35 (BFGS) + 3 (Fisher scoring)
```

Figura 6: Efeitos das classes de plataformas

E também seus efeitos com interação entre si:

```

platform_Mobile:platform_PC:platform_PlayStation      0.72211
platform_Nintendo:platform_PC:platform_PlayStation    4.42e-05 ***
platform_Other:platform_PC:platform_PlayStation        0.39433
platform_Mobile:platform_Nintendo:platform_Other:platform_PC
platform_Mobile:platform_Nintendo:platform_Other:platform_PlayStation
platform_Mobile:platform_Nintendo:platform_PC:platform_PlayStation
platform_Mobile:platform_Other:platform_PC:platform_PlayStation
platform_Nintendo:platform_Other:platform_PC:platform_PlayStation
platform_Mobile:platform_Nintendo:platform_Other:platform_PC:platform_PlayStation 0.71554

Phi coefficients (precision model with identity link):
Estimate Std. Error z value Pr(>|z|)
(phi) 0.847284 0.006981 121.4 <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Type of estimator: ML (maximum likelihood)
Log-likelihood: 1.189e+05 on 34 Df
Pseudo R-squared: 0.158
Number of iterations: 64 (BFGS) + 4 (Fisher scoring)

```

Figura 7: Efeitos das plataformas com interação

A interação mútua entre variáveis causa um aumento muito grande nas variáveis, afinal o modelo testa todas as combinações possíveis, o que leva a uma piora no modelo por instabilidade ou excesso de incerteza. Ainda assim, essa análise nos traz uma conclusão bem interessante. Além das variáveis independentes, 3 variáveis de interação mantêm um nível de incerteza consistentemente aceitável: PC, Nintendo, e PlayStation. Se o modelo for construído apenas com essas três, o resultado é um tanto interessante:

```

      Min      IQ      Median      3Q      Max
-1.6684 -0.3355 -0.3355  0.7844  1.2932

Coefficients (mean model with logit link):
(Intercept)      -1.83541    0.01969 -93.201 < 2e-16 ***
platform_Nintendo  0.47157    0.03487  13.525 < 2e-16 ***
platform_PC       0.19294    0.02009   9.602 < 2e-16 ***
platform_PlayStation 0.46971    0.03150  14.910 < 2e-16 ***
platform_Nintendo:platform_PC -0.12655    0.05086  -2.488  0.0128 *
platform_Nintendo:platform_PlayStation -0.40084    0.06354  -6.309 2.82e-10 ***
platform_PC:platform_PlayStation  0.36963    0.04032   9.168 < 2e-16 ***
platform_Nintendo:platform_PC:platform_PlayStation 0.17840    0.08183   2.180  0.0292 *

Phi coefficients (precision model with identity link):
Estimate Std. Error z value Pr(>|z|)
(phi) 0.824941 0.006772 121.8 <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Type of estimator: ML (maximum likelihood)
Log-likelihood: 1.185e+05 on 9 Df
Pseudo R-squared: 0.1271
Number of iterations: 37 (BFGS) + 2 (Fisher scoring)

```

Figura 8: Interações entre as 3 maiores plataformas

Apesar de não ser 100% claro, o modelo aparenta reagir com uma melhora nas métricas, e quase todas as interações apresentam números agradáveis de incerteza. Especulando um pouco sobre os motivos que podem levar a essa interação, das plataformas que escolhemos para análise, essas são possivelmente as mais competitivas no que diz respeito a exclusividade de jogos e comunidades. A grande maioria dos jogos de Nintendo nunca saem para hardwares que não sejam da própria empresa, e a Sony também investe muito em seus títulos exclusivos de forma a competir com seus rivais. Diferente delas, o Xbox não possui tantos exclusivos (pois sua empresa também fabrica computadores) e o mercado Mobile possui suas próprias particularidades de mercado. No computador, por outro lado, apesar de não possuir tanto destaque com exclusivos, é seguro dizer que a maioria dos jogos que estão disponíveis ao mesmo tempo no computador e em outras plataformas, possuem uma comunidade muito maior e mais ativa no ambiente de computadores do que nos demais. Essa relação de plataformas com a opinião pública dos jogos pode estar refletindo a leal-

dade das comunidades com suas respectivas plataformas favoritas, e isso certamente é algo a se levar em consideração para o desenvolvimento de jogos em qualquer uma dessas plataformas.

4.2 Previsão do MetaCritic

Seguindo a linha da análise anterior, serão analisados os modelos aplicados para previsão das avaliações dos profissionais do MetaCritics.

4.3 Beta

Assim como anteriormente, será aplicada uma GLM de família Beta com função de ligação Logit. Apesar de tecnicamente se tratar de uma variável discreta e não contínua, por ter um suporte maior, acredita-se que para esta análise, uma aproximação através do modelo logit seja suficiente para uma boa previsão dos resultados. Seguindo a linha anterior, os testes são iniciados com um modelo simples sem interação entre as variáveis:

```
Phi coefficients (precision model with identity link):
      Estimate Std. Error z value Pr(>|z|)
(phi)  16.5269      0.3344  49.42  <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Type of estimator: ML (maximum likelihood)
Log-likelihood:  4039 on 36 Df
Pseudo R-squared: 0.1485
Number of iterations: 44 (BFGS) + 2 (Fisher scoring)
```

Figura 9: Predição do MetaCritics sem interação

Já logo de início é possível observar alguns detalhes na aplicação do modelo que contrastam com o modelo anterior. Além do desempenho do modelo ter sido um pouco melhor, é interessante observar que para a análise dos críticos, as variáveis de classificação indicativa não são tão impactantes quanto na análise do público. É possível que isso reflita justamente no caráter das avaliações em si; enquanto os críticos avaliam todos os jogos igualmente de um ponto de vista técnico e objetivo, o público que avalia os jogos provavelmente está concentrado numa certa faixa de idade que possui preferências por jogos com classificação indicativa voltada para esta parcela dos usuários.

Quanto à interação das plataformas de jogos:

```

(Intercept)                                Pr(>|z|)
platform_Nintendo                          < 2e-16 ***
platform_PC                                0.08739 .
platform_PlayStation                       0.00096 ***
platform_Xbox                              0.80743
platform_Nintendo:platform_PC              0.21547
platform_Nintendo:platform_PlayStation     0.64413
platform_PC:platform_PlayStation           0.00698 **
platform_Nintendo:platform_Xbox            0.71956
platform_PC:platform_Xbox                  0.07615 .
platform_PlayStation:platform_Xbox         0.61416
platform_Nintendo:platform_PC:platform_PlayStation 0.18223
platform_Nintendo:platform_PC:platform_Xbox 0.02610 *
platform_Nintendo:platform_PlayStation:platform_Xbox 0.09192 .
platform_PC:platform_PlayStation:platform_Xbox 0.06087 .
platform_Nintendo:platform_PC:platform_PlayStation:platform_Xbox 0.31195
platform_Nintendo:platform_PC:platform_PlayStation:platform_Xbox 0.08767 .

Phi coefficients (precision model with identity link):
  Estimate Std. Error z value Pr(>|z|)
(phi)  14.7880    0.2984   49.56  <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Type of estimator: ML (maximum likelihood)
Log-likelihood: 3784 on 17 Df
Pseudo R-squared: 0.04766
Number of iterations: 24 (BFGS) + 2 (Fisher scoring)

```

Figura 10: Interação entre plataformas para o Metacritics

É possível interpretar que a teoria anterior é reforçada pelo comportamento visto aqui. Enquanto as comunidades fiéis às suas plataformas favoritas levam essa métrica a ter muito peso na previsão das suas opiniões, essa não é uma realidade entre os profissionais que trabalham com revisões de jogos. A única variável que mantém um baixo nível de incerteza nesse caso é a plataforma de PC, o que faz sentido visto que a comunidade de jogos profissionais e eventos famosos têm se centrado cada vez mais na plataforma dos computadores, por serem mais versáteis, abrangentes, e muitas vezes até mesmo mais baratos.

5 Conclusão

Com as análises realizadas aqui, foi possível concluir que apesar de seguir uma mesma tendência, as avaliações do público são afetadas de forma muito diferente das avaliações dos críticos pelas mesmas variáveis. Esse tipo de comportamento pode ser explicado de muitas formas, mas é visível seu efeito nos modelos explicativos gerados nesse projeto.

Além disso, o trabalho despendido nas análises permitiu explicitar alguns dos efeitos das variáveis analisadas nas avaliações, mostrando por exemplo como as plataformas em que um jogo é lançado e a classificação etária do jogo afetam e muito na opinião pública a respeito dele, mas não tão intensamente na opinião dos críticos, que parecem ter uma opinião mais sólida.

Ao fim deste trabalho, infelizmente sinto que gostaria ter tido mais tempo para aprofundar mais a pesquisa realizada, encontrando mais das correlações entre variáveis, e quem sabe até mesmo descobrindo qual variável afeta de forma mais intensa a opinião dos críticos.

Também era de desejo do autor explorar as variáveis de gênero, que apesar de numerosas, podem explicar muito sobre as preferências do público em relação aos lançamentos. Existem ainda muitas outras análises possíveis, como a relação de certas variáveis com o período de tempo em que os jogos foram lançados, e principalmente

a interação de outras famílias de modelos com o conjunto de dados, como modelos de família Gamma, ou até mesmo Gaussiana. Este é um trabalho que deverá ser continuado em projetos futuros.

Referências

- [1] <https://rawg.io/>
- [2] https://github.com/georgedutra/Final_Project_SM
- [3] <http://leg.ufpr.br/~lucambio/GLM/GLM.html>
- [4] https://pt.wikipedia.org/wiki/Modelo_linear_generalizado
- [5] GELMAN, A.; HILL, J. Regression and Other Stories. New York: Cambridge University Press, 2020.

A Descrição do Dataset

Aqui segue uma descrição detalhada de cada coluna do Dataset original, seguido de seu tipo:

- id: Um ID único que identifica este jogo no banco de dados RAWG. (string)
- slug: Um apelido único deste jogo no banco de dados RAWG. (string)
- name: Nome do jogo. (string)
- metacritic: Avaliação do jogo no Metacritic. ($\text{int} \in [0, 100]$)
- released: A data em que o jogo foi lançado. (date)
- tba: Se o jogo está para ser anunciado ou não. (bool)
- updated: A data em que o jogo foi atualizado pela última vez. (date)
- website: Site do jogo. (string)
- rating: Avaliação feita por usuários do RAWG. ($\text{float} \in [0, 5]$)
- rating_top: Maior valor de avaliação que este jogo já alcançou. ($\text{float} \in [0, 5]$)
- playtime: Horas necessárias para concluir o jogo. ($\text{int} > 0$)
- achievements_count: Número de conquistas no jogo. ($\text{int} > 0$)
- ratings_count: Número de usuários do RAWG que avaliaram o jogo. ($\text{int} > 0$)
- suggestions_count: Número de usuários do RAWG que sugeriram o jogo. ($\text{int} > 0$)
- game_series_count: Número de jogos na série. ($\text{int} > 0$)
- reviews_count: Número de usuários do RAWG que fizeram resenhas do jogo (a resenha é um texto que além de avaliar, detalha a experiência do usuário no jogo). ($\text{int} > 0$)
- platforms: Plataformas em que o jogo foi lançado. Separadas por ||. (string)
- developers: Desenvolvedores do jogo. Separados por ||. (string)
- genres: Gêneros do jogo. Separados por ||. (string)
- publishers: Publicadores do jogo. Separados por ||. (string)
- esrb_rating: Classificação ESRB. (categorical)
- added_status_yet: Número de usuários do RAWG que têm o jogo como "Não jogado". ($\text{int} > 0$)
- added_status_owned: Número de usuários do RAWG que têm o jogo como "Possuído". ($\text{int} > 0$)
- added_status_beaten: Número de usuários do RAWG que têm o jogo como "Concluído". ($\text{int} > 0$)

- `added_status_toplay`: Número de usuários do RAWG que têm o jogo como "A jogar". (`int > 0`)
- `added_status_dropped`: Número de usuários do RAWG que têm o jogo como "Jogado, mas não concluído". (`int > 0`)
- `added_status_playing`: Número de usuários do RAWG que têm o jogo como "Jogando". (`int > 0`)