

SCC0652 Visualização Computacional

Documentação - Projeto 3

Grupo:

- Alexandre Norcia Medeiros - nUSP: 10295583
- Gabriel Alfonso Nascimento Salgueiro - nUSP: 10284368

Sumário

SCC0652 Visualização Computacional	1
Documentação - Projeto 3	1
Sumário	2
1. Pré-Processamento de dados	3
1.1. Sumarização	3
1.2. Dados ausentes	4
1.3. Escalonamento	6
1.4. Outliers	7
2. Visualização dos Dados	9
2.1. Visualizações realizadas	9
2.1.1. Visualização realizada para os categóricos	9
2.1.2. Visualização realizada para as vendas	10
2.1.3. Visualização realizada para as notas de avaliações	11
2.2 Dashboard e Interação da Aplicação em Conjunto	12
3. Conclusão	15

1. Pré-Processamento de dados

1.1. Sumarização

Para esse projeto escolhemos trabalhar com o dataset ["Video Games Sales with Ratings"](#), disponibilizado na plataforma Kaggle. Este conjunto de dados diz respeito aos videogames mais vendidos de todos os tempos. A figura 1 apresenta as primeiras linhas do conjunto.

O conjunto de dados tem os seguintes atributos (colunas):

- *Name*: Nome do jogo representado na linha.
- *Platform*: Plataforma onde o jogo é executado, "videogame".
- *Year_of_Release*: Ano de lançamento do jogo.
- *Genre*: Gênero do jogo.
- *Publisher*: Publicadora do jogo.
- *NA_Sales*: Vendas na América do Norte.
- *EU_Sales*: Vendas na Europa.
- *JP_Sales*: Vendas no Japão.
- *Other_Sales*: Vendas nos demais lugares do mundo.
- *Global_Sales*: Vendas globais.
- *Critic_Score*: Nota dada pela crítica ao jogo.
- *Critic_Count*: Quantidade de críticos que avaliaram.
- *User_Score*: Nota dada pelo público ao jogo.
- *User_Count*: Quantidade de usuários que avaliaram.
- *Developer*: Empresa desenvolvedora do jogo
- *Rating*: Classificação indicativa de idade para os jogadores.

Uma observação, é de que as instâncias do conjunto estão separadas por plataforma. Assim, uma linha representa a venda de determinado jogo em determinada plataforma. Ou seja, certos jogos aparecem em instâncias diferentes e para contabilizar a venda total eles devem ser agrupados (perdendo as informações categóricas).

1	Name	Platform	Year_of_Release	Genre	Publisher	NA_Sales	EU_Sales	JP_Sales	Other_Sales	Global_Sales	Critic_Score	Critic_Count	User_Score	User_Count	Developer	Rating	
2	Wii Sports	Wii	2006	Sports	Nintendo	41.36	28.96	3.77	8.45	82.53	76	51	8	322	Nintendo	E	
3	Super Mario Bros NES	NES	1985	Platform	Nintendo	29.08	3.58	6.81	0.77	40.24							
4	Mario Kart Wii	Wii	2008	Racing	Nintendo	15.68	12.76	3.79	3.29	35.52	82	73	8.3	709	Nintendo	E	
5	Wii Sports Resort	Wii	2009	Sports	Nintendo	15.61	10.93	3.28	2.95	32.77	80	73	8	192	Nintendo	E	
6	Pokemon Red/P GB	GB	1996	Role-Playing	Nintendo	11.27	8.89	10.22		1 31.37							
7	Tetris	GB	1989	Puzzle	Nintendo		23.2	2.26	4.22	0.58							
8	New Super Mario DS	DS	2006	Platform	Nintendo	11.28	9.14		6.5	2.88	29.8	89	65	8.5	431	Nintendo	E
9	Wii Play	Wii	2006	Misc	Nintendo	13.96	9.18	2.93	2.84	28.92	58	41	8.6	129	Nintendo	E	
10	New Super Mario Wii	Wii	2009	Platform	Nintendo	14.44	6.94		4.7	2.24	28.32	87	80	8.4	594	Nintendo	E
11	Duck Hunt	NES	1984	Shooter	Nintendo	26.93	0.63	0.28	0.47	28.31							
12	Nintendogs	DS	2005	Simulation	Nintendo		09.05	10.95	1.93	2.74	24.67						
13	Mario Kart DS	DS	2005	Racing	Nintendo	9.71	7.47	4.13		1.9 23.21	91	64	8.6	464	Nintendo	E	
14	Pokemon Gold/P GB	GB	1999	Role-Playing	Nintendo		9 6.18		7.2 0.71		23.1						
15	Wii Fit	Wii	2007	Sports	Nintendo	8.92		08.03	3.6 2.15		22.7	80	63	7.7	146	Nintendo	E
16	Kinect Adventure X360	X360	2010	Misc	Microsoft Game		15 4.89	0.24	1.69	21.81	61	45	6.3	106	Good Science S	E	
17	Wii Fit Plus	Wii	2009	Sports	Nintendo		09.01 8.49	2.53	1.77	21.79	80	33	7.4	52	Nintendo	E	
18	Grand Theft Auto PS3	PS3	2013	Action	Take-Two Inter		07.02	09.09 0.96	3.96	21.04	97	50	8.2	3994	Rockstar North	M	
19	Grand Theft Auto PS2	PS2	2004	Action	Take-Two Inter	9.43	0.4	0.41	10.57	20.81	95	80	9	1588	Rockstar North	M	
20	Super Mario Bros NES	NES	1990	Platform	Nintendo	12.78	3.75	3.54	0.55	20.61							
21	Brain Age Train DS	DS	2005	Misc	Nintendo	4.74		9.2 4.16		02.04 20.15	77	58	7.9	50	Nintendo	E	
22	Pokemon Diam	DS	2006	Role-Playing	Nintendo	6.38	4.46		06.04 1.36	18.25							
23	Super Mario Bros GB	GB	1989	Platform	Nintendo	10.83	2.71	4.18	0.42	18.14							
24	Super Mario Bros NES	NES	1988	Platform	Nintendo	9.54	3.44	3.84	0.46	17.28							
25	Grand Theft Auto X360	X360	2013	Action	Take-Two Inter	9.66	5.14	0.06	1.41	16.27	97	58	8.1	3711	Rockstar North	M	
26	Grand Theft Auto PS2	PS2	2002	Action	Take-Two Inter	8.41	5.49	0.47	1.78	16.15	95	62	8.7	730	Rockstar North	M	
27	Pokemon Ruby/P GBA	GBA	2002	Role-Playing	Nintendo		06.06	3.9 5.38	0.5	15.85							

Figura 1: Amostra do Conjunto de dados antes do pré-processamento.

A classe data frame do pandas contém um método chamado describe, cujo resultado pode ser observado na figura 2. Nós utilizamos ele para exibir medidas

estatísticas descritivas em cima da base. Assim, é calculado em cima de todos atributos numéricos a média, a quantidade, o desvio padrão, o mínimo, o máximo etc.

Ao observar a presença dos atributos *Critic_Score*, *Critic_Count*, e *User_Count* e a ausência do atributo similar *User_Score* da tabela de descrição, é possível concluir que existem dados inconsistentes nesse atributo ausente (*User_Score*).

	Year_of_Release	NA_Sales	EU_Sales	JP_Sales	Other_Sales	Global_Sales	Critic_Score	Critic_Count	User_Count
count	16450.000000	16719.000000	16719.000000	16719.000000	16719.000000	16719.000000	8137.000000	8137.000000	7590.000000
mean	2006.487356	0.263330	0.145025	0.077602	0.047332	0.533543	88.967679	26.360821	162.229908
std	5.878995	0.813514	0.503283	0.308818	0.186710	1.547935	13.938165	18.980495	561.282328
min	1980.000000	0.000000	0.000000	0.000000	0.000000	0.010000	13.000000	3.000000	4.000000
25%	2003.000000	0.000000	0.000000	0.000000	0.000000	0.060000	60.000000	12.000000	10.000000
50%	2007.000000	0.080000	0.020000	0.000000	0.010000	0.170000	71.000000	21.000000	24.000000
75%	2010.000000	0.240000	0.110000	0.040000	0.030000	0.470000	79.000000	36.000000	81.000000
max	2020.000000	41.380000	28.980000	10.220000	10.570000	82.530000	98.000000	113.000000	10665.000000

Figura 2: Análise descritiva dos dados, antes do pré-processamento, com atributos inconsistentes.

1.2. Dados ausentes

Para o tratamento dos dados ausentes no nosso conjunto de dados, iremos remover ou substituir os valores conforme necessário.

Nesta seção iremos trabalhar também em cima dos dados inconsistentes, como o *User_Score*.

Primeiro, se verifica quais colunas apresentam dados ausentes e a quantidade de dados ausentes. Além disso, é exibido os valores únicos de *User_Score* para tentar identificar qual inconsistência está ocorrendo no atributo (figura 3).

```
array(['8', nan, '8.3', '8.5', '6.6', '8.4', '8.6', '7.7', '6.3', '7.4',
      '8.2', '9', '7.9', '8.1', '8.7', '7.1', '3.4', '5.3', '4.8', '3.2',
      '8.9', '6.4', '7.8', '7.5', '2.6', '7.2', '9.2', '7', '7.3', '4.3',
      '7.6', '5.7', '5', '9.1', '6.5', 'tbd', '8.8', '6.9', '9.4', '6.8',
      '6.1', '6.7', '5.4', '4', '4.9', '4.5', '9.3', '6.2', '4.2', '6',
      '3.7', '4.1', '5.8', '5.6', '5.5', '4.4', '4.6', '5.9', '3.9',
      '3.1', '2.9', '5.2', '3.3', '4.7', '5.1', '3.5', '2.5', '1.9', '3',
      '2.7', '2.2', '2', '9.5', '2.1', '3.6', '2.8', '1.8', '3.8', '0',
      '1.6', '9.6', '2.4', '1.7', '1.1', '0.3', '1.5', '0.7', '1.2',
      '2.3', '0.5', '1.3', '0.2', '0.6', '1.4', '0.9', '1', '9.7'],
      dtype=object)
```

Figura 3: Valores únicos de *User_Score*.

Assim, pudemos observar a presença de um valor "tbd" dentro do atributo "User_Score". Tendo em vista que esse atributo pretende representar uma nota de avaliação do jogo entre 0 a 10, devemos remover qualquer elemento que não seja numérico. Desta forma, é necessário inicialmente remover qualquer elemento que não possa ser convertido para float, deixando o campo ausente. Além disso, precisa-se converter o campo para o mesmo se tornar de fato float.

Voltando para os dados nulos, temos, inicialmente, a tabela a seguir:

Quantidade de dados nulos por coluna:

Name	2
Platform	0
Year_of_Release	269
Genre	2
Publisher	54
NA_Sales	0
EU_Sales	0
JP_Sales	0
Other_Sales	0
Global_Sales	0
Critic_Score	8582
Critic_Count	8582
User_Score	6704
User_Count	9129
Developer	6623
Rating	6769

Figura 4: Quantidade de instâncias com dados nulos em cada atributo, antes do tratamento de campos nulos.

Assim, é possível identificar quais atributos precisam ser tratados. Assim, foi feito os seguintes tratamentos para cada um dos campos:

- Para o caso dos Nomes, decidimos remover as instâncias inválidas. Isto por conta de que este campo é o mais importante para identificarmos o jogo, assim não faz sentido instâncias com esse elemento ausente.
- Para o ano de lançamento, decidimos colocar um valor sentinela. Escolhemos a média (2006.4873556231003), pois assim ocorre uma alteração mínima em cima das estatísticas desse atributo. Além disso, é um valor real, portanto ele é facilmente identificável em comparação aos outros anos, que são valores inteiros.
- Para os campos *Publisher*, *Developer*, e *Rating*, decidimos usar o valor sentinela "Unknown". Desta forma, marcamos esses campos de texto com um valor coerente que representa a falta de informação.
- Para os campos *Critic_Score* e *User_Score* iremos usar a média (*Critic_Score*: 68.967, *User_Score*: 7.125) para preencher os valores ausentes. Dessa forma, não alteramos de forma brusca as estatísticas globais calculadas em cima desses campos. Além disso, tendo em vista a relação desses dois atributos com os campos *Critic_Count* e *User_Count*, iremos preencher os atributos que contam a quantidade de avaliação com o número 0. Assim, ainda é possível identificar se determinado jogo foi ou não avaliado.

Tendo em vista que nossos dados são descritivos, praticamente todos tratamentos necessários já foram feitos. Assim, foi salvo os dados como estão após a resolução dos

campos nulos. As figuras 5 e 6 representam como ficaram os dados após os processamentos realizados.

Quantidade de dados nulos por coluna:

```
Name          0
Platform       0
Year_of_Release 0
Genre          0
Publisher      0
NA_Sales       0
EU_Sales       0
JP_Sales       0
Other_Sales    0
Global_Sales   0
Critic_Score   0
Critic_Count   0
User_Score     0
User_Count     0
Developer      0
Rating         0
```

Figura 5: Quantidade de instâncias com dados nulos em cada atributo, após o tratamento de campos nulos.

	Year_of_Release	NA_Sales	EU_Sales	JP_Sales	Other_Sales	Global_Sales	Critic_Score	Critic_Count	User_Score	User_Count
count	16717.000000	16717.000000	16717.000000	16717.000000	16717.000000	16717.000000	16717.000000	16717.000000	16717.000000	16717.000000
mean	2006.488969	0.263255	0.145010	0.077610	0.047333	0.533462	68.967679	12.831130	7.125046	73.657056
std	5.829987	0.813475	0.503303	0.308836	0.186721	1.547956	9.723998	18.680383	1.010693	386.717446
min	1980.000000	0.000000	0.000000	0.000000	0.000000	0.010000	13.000000	0.000000	0.000000	0.000000
25%	2003.000000	0.000000	0.000000	0.000000	0.000000	0.060000	68.967679	0.000000	7.125046	0.000000
50%	2007.000000	0.080000	0.020000	0.000000	0.010000	0.170000	68.967679	0.000000	7.125046	0.000000
75%	2010.000000	0.240000	0.110000	0.040000	0.030000	0.470000	70.000000	21.000000	7.300000	20.000000
max	2020.000000	41.360000	28.960000	10.220000	10.570000	82.530000	98.000000	113.000000	9.700000	10665.000000

Figura 6: Análise descritiva dos dados, após o pré-processamento.

1.3. Escalonamento

O processo de escalonamento não é muito interessante pelo fato de ele mudar os valores dos dados. Assim, perderíamos as informações que os dados descrevem. Isto, por conta de que a maioria deles diz respeito a um número inerente à instância, como a quantidade de vendas de dado jogo ou o ano de lançamento. Os únicos casos interessantes para o escalonamento são quando se deseja comparar diretamente os jogos em relação ao jogo com valor máximo ou mínimo em determinado atributo. Um exemplo seria um gráfico que demonstra diretamente a porcentagem de venda dos jogos em relação ao jogo mais vendido da história. Isso poderia ser realizado em cima do conjunto alterado, como demonstrado nas figuras 7 e 8.

0	82.53
1	40.24
2	35.52
3	32.77
4	31.37
	...
16714	0.01
16715	0.01
16716	0.01
16717	0.01
16718	0.01

Figura 7: Vendas em milhões de unidades.

0	1.000000
1	0.487518
2	0.430320
3	0.396995
4	0.380029
	...
16714	0.000000
16715	0.000000
16716	0.000000
16717	0.000000
16718	0.000000

Figura 8: Porcentagem de vendas em relação ao mais vendido.

1.4. Outliers

Nossos data frame contém dados com muita variabilidade, assim também não faz sentido remover os dados considerados Outliers. Pois, a maior parte dos jogos listados tem baixa quantidade de venda, com poucos jogos tendo uma alta venda global. Além disso, muitos têm suas vendas concentradas em determinadas regiões, deixando os outros campos regionais próximos de zero. Também vale notar que, a maioria dos jogos que estão no dataset são jogos não populares, onde as vendas são extremamente baixas comparada aos jogos populares. Desta forma, a média dos dados fica muito concentrada em jogos não populares. Considerando tudo isso, todos jogos expressivos que tem um número de venda alta acabam sendo considerados outliers (Figura 10).

Este evento pode ser observado ao estimar a distribuição dos atributos. Por exemplo, ao estimar a distribuição da coluna *Global_Sales*, usando apenas 4000 maiores observações das 16720 existentes, temos uma curva parecida com a distribuição exponencial (Figura 9). Porém, é possível observar quão pronunciado é o declínio da curva, com praticamente apenas os 30 primeiros elementos exibindo valores significativos.

Assim, se contássemos os outliers teríamos um efeito reverso, onde apenas os jogos com números expressivos de venda seriam removidos. Desta forma, é válido considerar cortar a maior parte do conjunto fora. Com isso, trabalhamos apenas com uma pequena quantidade determinada dos jogos com as maiores vendas globais.

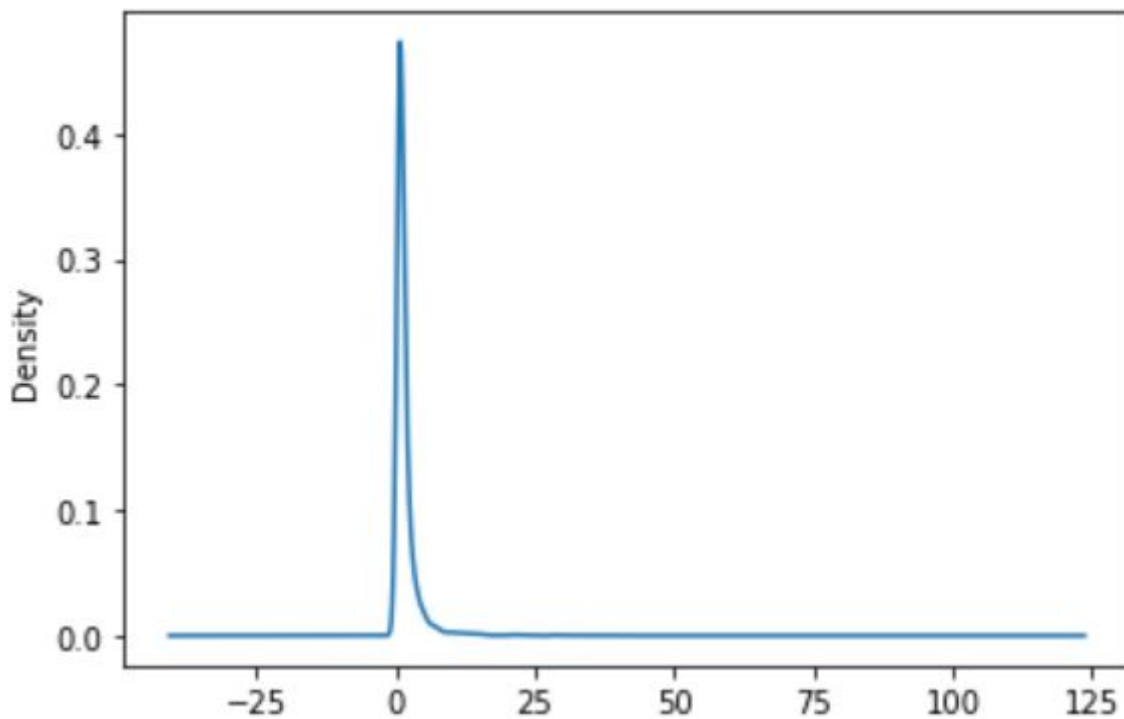


Figura 9: Aproximação de uma função de densidade das vendas dos jogos no *dataset*.

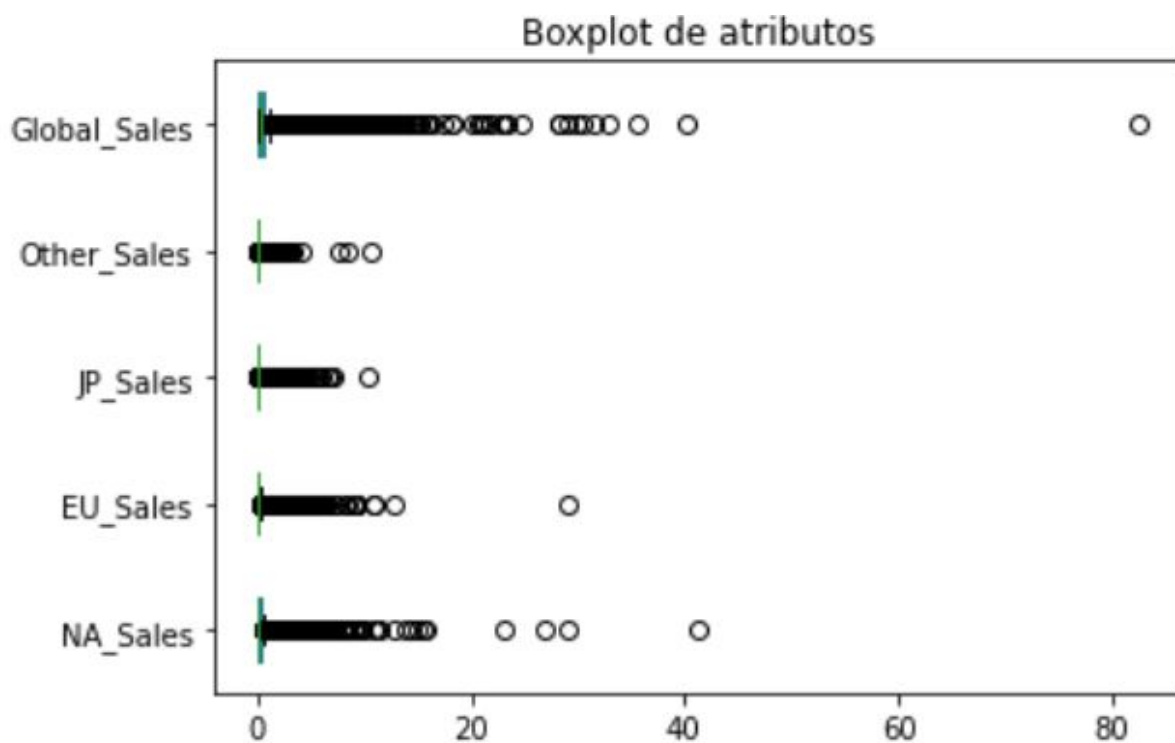


Figura 10: *Boxplot* das vendas dos jogos.

Desta forma, decidimos trabalhar só com as primeiras instâncias do conjunto, sendo elas as mais relevantes. Para isso, implementamos uma variável *threshold* que limita a quantidade de jogos que iremos visualizar, i.e, se o “*threshold* = 300”, então teremos os 300 primeiros jogos na visualização.

2. Visualização dos Dados

2.1. Visualizações realizadas

Para visualizar nosso conjunto de dados, decidimos separar as visualizações entre dados categóricos, dados sobre vendas e dados sobre avaliação. Assim, podemos construir gráficos mais concisos e informativos dentro do seu próprio contexto.

Dessa forma, optamos por construir um dashboard que mostre as visualizações em conjunto. Nele, a visualização sobre as vendas está em destaque, pois a principal informação do conjunto é o ranqueamento dos jogos em vendas. Além disso, embaixo existem duas seções de apoio, uma que exibe os jogos através de algum atributo categórico selecionado e outra que exibe os jogos através de suas avaliações recebidas.

2.1.1. Visualização realizada para os categóricos

Inicialmente pensamos em fazer o gráfico de barras ou então *pieplots/doughnuts*. Porém, a maioria dos atributos possuem valores grandes, como *Role-Playing*, além de serem em sua maioria palavras. Também, vale notar que os *pieplots* seriam muito confusos, além de já serem contra indicados, principalmente em comparações. Por fim, pensamos em gráficos no estilo *lollipop*, mas, para que a interatividade ocorra da melhor forma, optamos pelo gráfico de barras. Desta forma, fizemos gráficos de barra com o eixo invertido, exibindo as categorias no eixo y, facilitando a leitura dos valores.

Assim, os gráficos de barra construídos foram feitos em cima dos seguintes categóricos :

- *Genre*
- *Platform*
- *Rating*
- *Year_of_Release*

Eles foram feitos relacionando os valores de cada categoria com a quantidade de venda que realizaram. Outra coisa que fizemos, foi atribuir uma cor para cada valor. Assim, em gráficos onde muitos valores são exibidos, fica fácil distinguir qual barra representa qual valor. Além disso, a interatividade do gráfico permitiu separar dentro da barra os jogos. Assim, no gráfico do *Genre*, é possível observar quanto o jogo *Wii Sports* contribuiu para a categoria *Sports* em quesito de vendas. Vale notar, que o usuário pode escolher no dashboard qual gráfico dos atributos categóricos ele quer que seja exibido. Este exemplo pode ser observado na figura 11.



Figura 11: Gráfico de barras do atributo *Genre*. Nele, está em destaque o jogo *Wii Sports*, que vendeu 82.53 milhões de cópias mundialmente e é do gênero *Sports*.

2.1.2. Visualização realizada para as vendas

Para os atributos relacionados a vendas, decidimos que um gráfico de linhas seria interessante. Isto, pois mostramos a venda em cada região para determinado jogo, além de conseguirmos representar o próprio rank e a disparidade entre cada jogo ao preencher o gráfico de maneira ordenada. A interatividade do gráfico ajuda bastante no seu entendimento. O elemento de *hovering* deixa o gráfico bem mais intuitivo, mostrando informações precisas apenas em cima do elemento selecionado. Além disso, o *zoom* permite visualizar partes específicas de forma mais clara, evitando a confusão que pode ocorrer ao exibir diversas linhas no mesmo gráfico. A figura 12 representa o gráfico e como é possível interagir com ele.

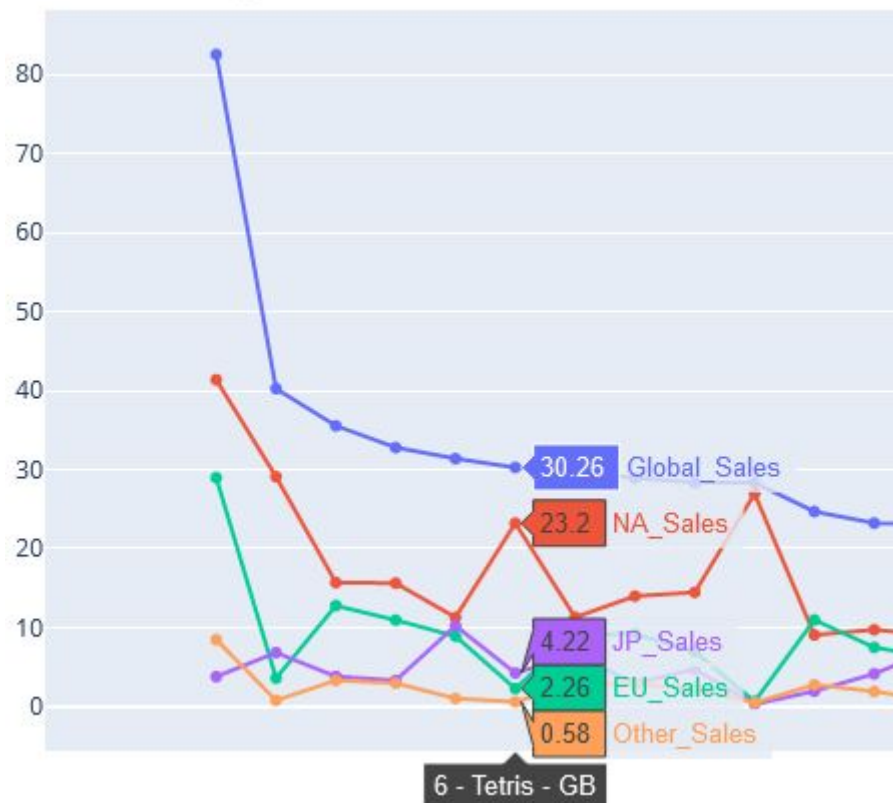


Figura 12: Gráfico de linhas representando o ranking de vendas dos jogos. Nele, o jogo *Tetris* está em destaque, onde também é exibido o número preciso de sua venda de cópias em milhões por região.

2.1.3. Visualização realizada para as notas de avaliações

Acreditamos que seria interessante poder visualizar os jogos em função da avaliação que eles receberam, tanto da crítica quanto dos usuários. Assim, criamos três *scatter plots*. O primeiro exibe os jogos em função do *Global_Sales* por *Critic_Score*. O segundo relaciona os jogos entre *Global_Sales* e *User_Score*, também exibindo o quanto a venda influencia na nota e vice-versa. Já o terceiro relaciona de forma direta as avaliações dos usuários pela crítica (*User_Score* por *Critic_Score*). Vales notar que o usuário poderá escolher qual gráfico ele quer visualizar por vez, para que se evite uma sobrecarga de informações.

Também achamos interessantes fazer desses *scatter plots* gráficos auxiliares, assim eles exibem todas informações que podem não estar sendo representadas de cada jogo através do *hovering*. Além de poderem ser colorizados por cada categoria, conforme o usuário escolher. Desta forma, ele consegue ressaltar uma informação que concorde com os outros gráficos de maneira simples.

Uma coisa importante de ser observada, é de que os *scatter plots* não apresentam todos os jogos do ranking, pois alguns dados do conjunto tinham valores inválidos de avaliações. Assim, esses gráficos mostram apenas os jogos que estão devidamente instanciados no conjunto.

A figura 13 exibe como os *scatter plots* são apresentados no *dashboard*.

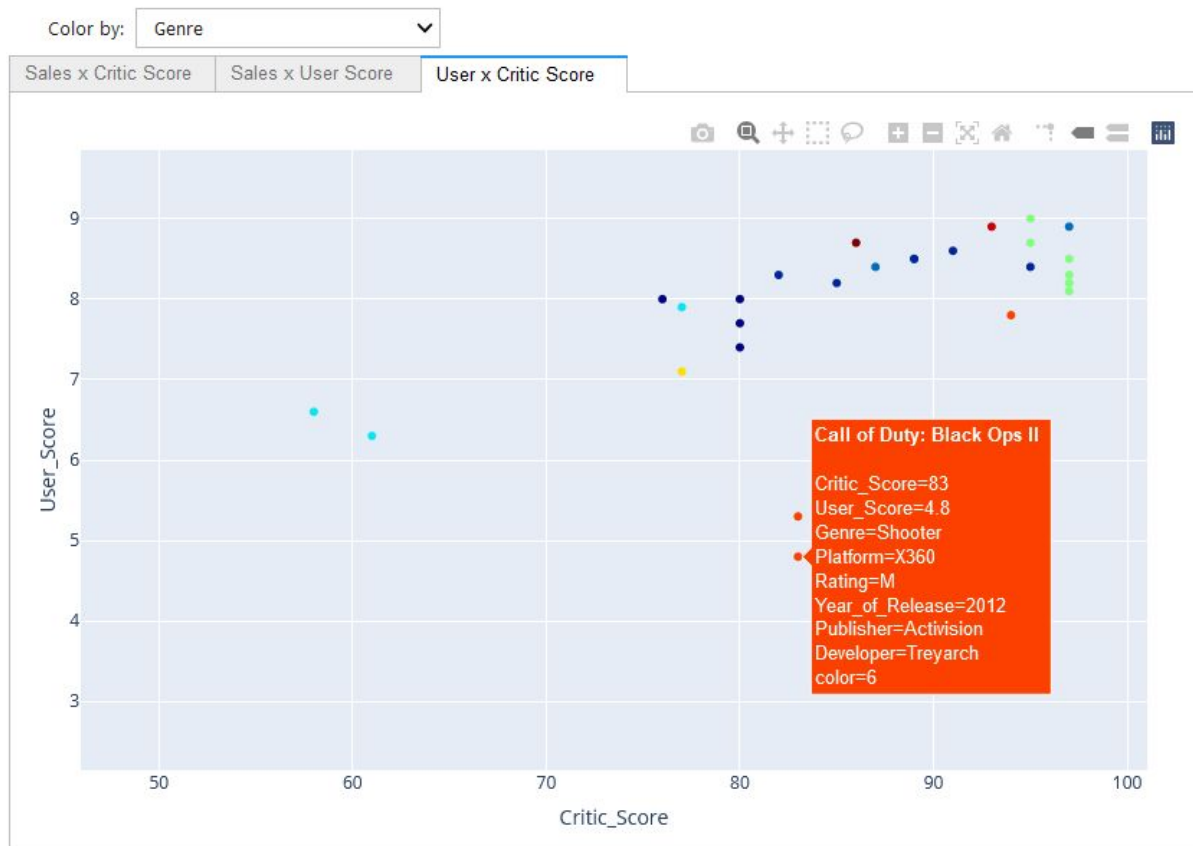


Figura 13: Gráficos de pontos, relacionando as avaliações dos críticos com as dos usuários. Nele, está em destaque o jogo *Call of Duty: Black Ops II*, que teve uma baixa nota de usuários e uma alta nota da crítica.

2.2 Dashboard e Interação da Aplicação em Conjunto



Figura 14: Configuração Padrão do *Dashboard* ao iniciar a aplicação.



Figura 15: Exemplo do *Dashboard* operando com o valor máximo do Threshold.



Figura 16: Seleção conjunta através dos gráficos presentes no *Dashboard*.

Como dito anteriormente, optamos por criar um *dashboard* conciso, que apresenta apenas três seções, exibindo somente três gráficos ao mesmo tempo. Desta forma, o usuário não recebe uma sobrecarga de informações. Mas, também colocamos abas de seleção, onde é possível selecionar o gráfico exibido em cada seção. As figuras 14, 15 e 16 exibem como é o funcionamento do *dashboard*.

Além disso, no topo da aplicação existe um seletor para o valor do *Threshold*. Assim, é possível controlar até quantos jogos serão exibidos nos gráficos em um mesmo instante. Ele é importante pois consegue limitar a visualização inicial, não deixando os gráficos com pontos demais, mas também fornece uma exploração mais aprofundada ao se aumentar o

valor. Além disso, as ferramentas de *zoom* nos gráficos ajudam a visualização ao se aumentar o *threshold*, limitando e aumentando a área de interesse.

Outra ferramenta interessante para a exploração conjunta dos gráficos é o 'box select'. Com ela, é possível selecionar elementos para serem destacados, fazendo os não selecionados ficarem apagados ao fundo. Esta seleção de elementos que ocorre em um gráfico foi propagada para os outros. Assim, qualquer seleção em qualquer gráfico fará com que os outros gráficos destaquem os mesmos jogos selecionados. O uso dessa ferramenta é visto na figura 16. Desta forma, por exemplo, fica fácil de visualizar a posição no ranking de vendas de todos os jogos do gênero *Shooter*. Basta selecionar a barra correspondente no gráfico de *Genre* e esses jogos serão destacados em todos outros gráficos, inclusive no do linhas que representa o ranking. Essa ferramenta é especialmente interessante ao relacionar os gráficos de barra com os gráficos de ponto, pois os mesmos gráficos de ponto podem ser coloridos pelas categorias representadas nas barras. Assim, são geradas diversas possibilidades de exploração, com o destaque de determinado conjunto facilitado através das seleções.

3. Conclusão

Esse trabalho fez com que fizéssemos todo *pipeline* de pré-processamento de dados em cima de um conjunto. Além disso, realizamos uma análise exploratória e conseguimos conhecer o conjunto de dados a fundo, além de, o mais importante, pudemos conhecer o processo de exploração. Durante sua realização, tivemos que entender os atributos do nosso conjunto e como seria a maneira mais adequada de representar e visualizar os dados. Isso foi feito de maneira progressiva através de etapas, onde aprendemos com nossas tentativas e corrigimos e adaptamos as visualizações, de forma que ficassem o mais intuitivo possível.

Ao final disso tudo, conseguimos criar uma aplicação moderna e intuitiva que representasse bem nosso conjunto. Além de aprendermos diversas técnicas de visualização e interação. Isso se mostra valioso ao realizar que podemos transferir esses conhecimentos obtidos durante as etapas do trabalho para outros conjuntos de dados.