Helio Cavalcante Silva Neto

Data StorytellingDatasetMoviesElo7

São Paulo

Novembro 2018

Introdução

Com o objetivo de responder as perguntas do Teste, foram utilizadas as seguintes ferramentas e bibliotecas:

- Orange[1], ferramenta para testes rápidos de programação visual;
- Bibliotecas de programação e Data Science em Python: Pandas[2], Numpy[3], Matplotlib[4], Seaborn[5] e Dask[6];
- Jupyter[7], IDE para programação em Python;
- Banco de dados SQLite. A ferramenta para manipulação do banco foi SQLite Studio[8];
- GitHub[9], repositório de código e conteúdo utilizado para responder o exercício.

Link para o GitHub: https://github.com/helio69/Elo7

Ao iniciar a análise no arquivo da base *MovieLens,* foi percebido que se tratava de um dataset de tamanho considerável, em especial no dataset "ratings.csv" (20.000.263 instâncias) e "genome-scores.cvs" (11.709.768 instâncias). Por motivo de limitações de hardware para o processamento e análise da base, houve a necessidade de importar todos os datasets para o banco SQLite e realizar todas as manipulações via SQL e gerar os arquivos "cvs" e realizar as análises e conclusões de todo o dataset.

Tratamento do DatasetMovies

[1]https://orange.biolab.si/
[2]https://pandas.pydata.org/
[3]http://www.numpy.org/
[4]https://matplotlib.org/
[5]https://seaborn.pydata.org/
[6]https://dask.org/
[7]https://jupyter.org/
[8]https://sqlitestudio.pl/index.rvt
[9]https://github.com/

Como mencionado na seção anterior, a base (“movies.csv”) foi importada para oSQLite e foi tratada também na ferramenta OpenSUSE[10]LibreOffice[11].A necessidade de tratamento no LibreOffice será explicada na seção da análise exploratória logo abaixo.

O tratamento dedados dos Filmes (“movies.csv”)foi:

1. O atributo “genres” continha mais de um gênero por filme, assim, estes gêneros contidos em cada instância do atributo estavam separados pela barra (“|”). Ao separar estes gêneros, foram encontrados cinco níveis de gênero por filme. Para cada tipo de gênero, foi atribuído um ID de identificação e os níveis que não tinha gênero continuou vazio.
2. Realizei um merge data entre os dataset “movies.csv” e “link.csv”. O atributo comparativo para realizar o merge data foi o “movieId”. Este merge data foi salvo no arquivo “movies_link.csv”.

Análise Exploratória do DatasetMovies

Ao iniciar a análise do dataset “movies.csv”, verificou-se a existência de 27.278 tipos de filmes e que cada filme tem sua classificação de gênero.

Para verificar quantos gêneros de filmes existem no dataset, foi utilizada a ferramenta SQLite Studio query “Selectgenres,count(*)frommoviesgroupbygenres”. Isto possibilitou entender que os 5 gêneros mais presentes são Drama, Comedy, Documentary, Comedy com Drama e Drama com Romance.

Tabela 1: Top Gêneros e Qtd.

genres	count(*)
Drama	4520
Comedy	2294
Documentary	1942
Comedy|Drama	1264
Drama|Romance	1075

Conforme tabela abaixo, identificamos 607 filmes com gênero único.

[10]https://www.opensuse.org/
[11]https://pt-br.libreoffice.org/

Tabela 2: Bot Gêneros e Qtd.

genres	count(*)
Action|Adventure|Animation|Children|Comedy|Romance	1
Action|Adventure|Animation|Children|Comedy|Sci-Fi|IMAX	1
Action|Adventure|Animation|Children|Comedy|Western	1
Action|Adventure|Animation|Comedy	1
Action|Adventure|Animation|Comedy|Crime|Mystery	1
Action|Adventure|Animation|Comedy|Drama|Fantasy|Romance	1
Action|Adventure|Animation|Comedy|Fantasy	1
Action|Adventure|Animation|Comedy|Fantasy|Mystery|Sci-Fi	1
Action|Adventure|Animation|Comedy|Fantasy|Sci-Fi	1
Action|Adventure|Animation|Comedy|Thriller	1
Action|Adventure|Animation|Crime|Sci-Fi	1
Action|Adventure|Animation|Drama	1
Action|Adventure|Animation|Drama|Sci-Fi	1
Action|Adventure|Animation|Fantasy|Horror	1
Action|Adventure|Animation|Fantasy|IMAX	1
Action|Adventure|Animation|Horror	1
Action|Adventure|Animation|Horror|Sci-Fi	1
Action|Adventure|Animation|Mystery|Sci-Fi	1
Action|Adventure|Animation|Sci-Fi|Thriller	1
Action|Adventure|Children|Comedy|Crime	1

Verificamos, no entanto, que na tabela acima, a investigação quantitativa dos gêneros não obteve uma precisão desejável, tendo em vista que os 607 filmes, embora estejam classificados em um único gênero, apresentam vários subgêneros.

Os subgêneros podem ser Action, Adventure, Animation, Children's, Comedy, Crime, Documentary, Drama, Fantasy, Film-Noir, Horror, Musical, Mystery, Romance, Sci-Fi, Thriller, War, Western e (no genreslisted).

Por este fator, houve a necessidade de separar os filmes em níveis e, assim, proporcionar uma investigação quantitativa mais satisfatória.

Ao separar os subgêneros em níveis, foi possível verificar que havia 5 níveis de classificação dos tipos de subgêneros dos filmes.

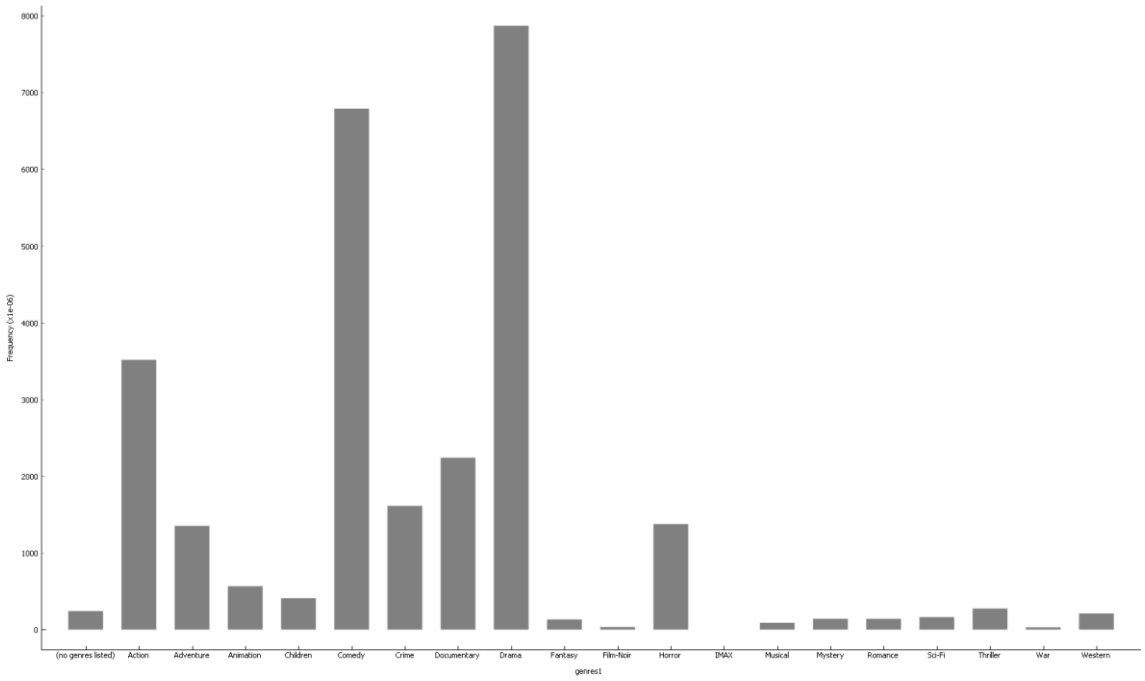

Gráfico 1: Gênero nível 1

Analisando o gráfico de dispersão dos filmes por classificação do gênero no nível 1,os 5 gêneros mais presentes foram Drama, Comédia, Ação, Documentário e Crime. Já, os 3menos presentes foram IMAX, War e Musical.

Logo abaixo, é possível visualizar o gráfico do nível 2, onde se verifica que os gêneros mais presentes foram Drama, Romance e Thriller. Os menos frequentes no nível 2 foram IMAX, Film-Noir e Western.

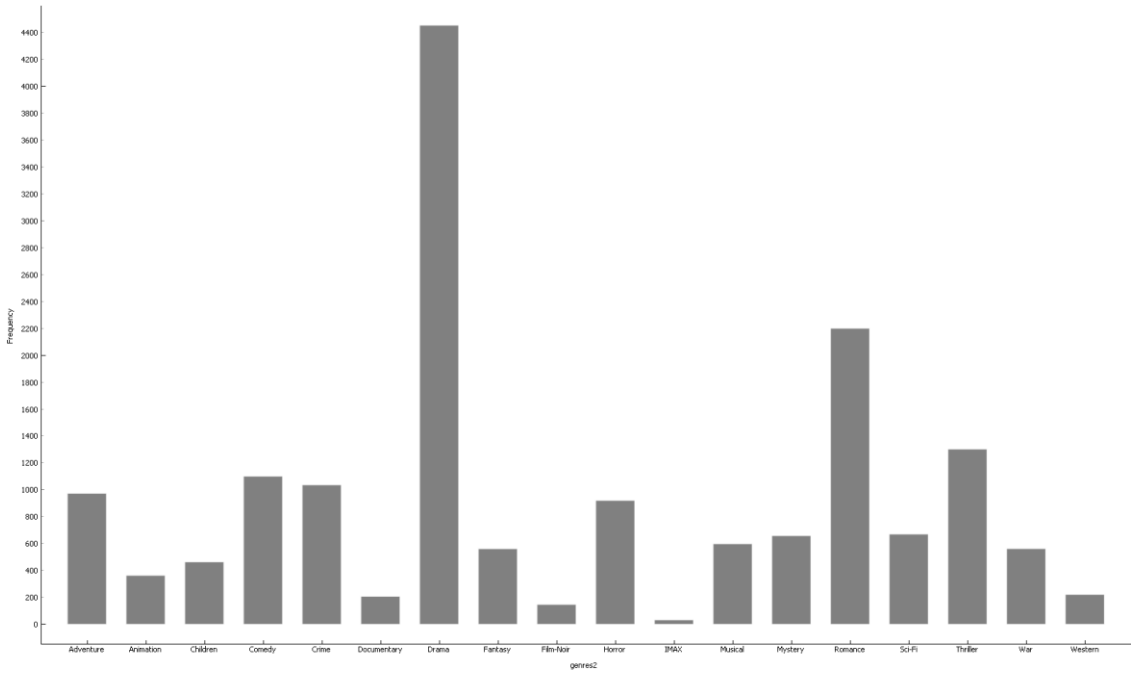

Gráfico 2: Gênero nível 2

No nível 3, vemos como mais presentes o gêneroThriller, Romance e Drama. Os menos freqüentes foram IMAX, Documentário e Animação.

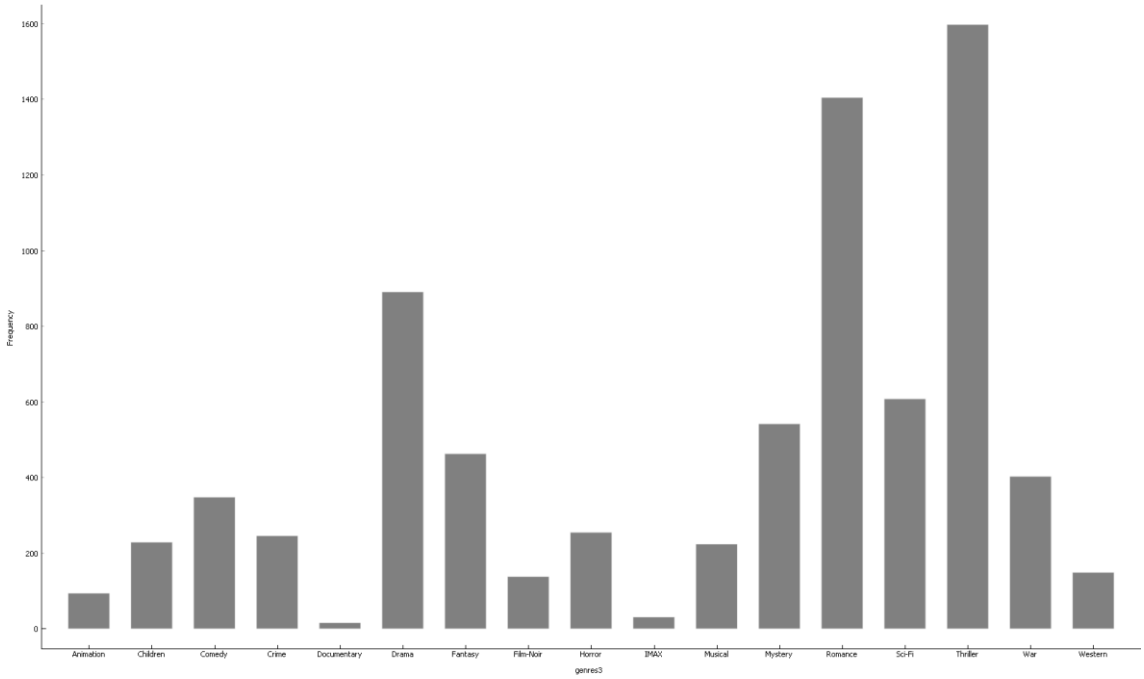

Gráfico 3: Gênero 3

No nível 4, os mais frequentes foramThriller, Romance e Sci-Fi.Já, os menos presentesforam Documentário, Film-Noir e Children.

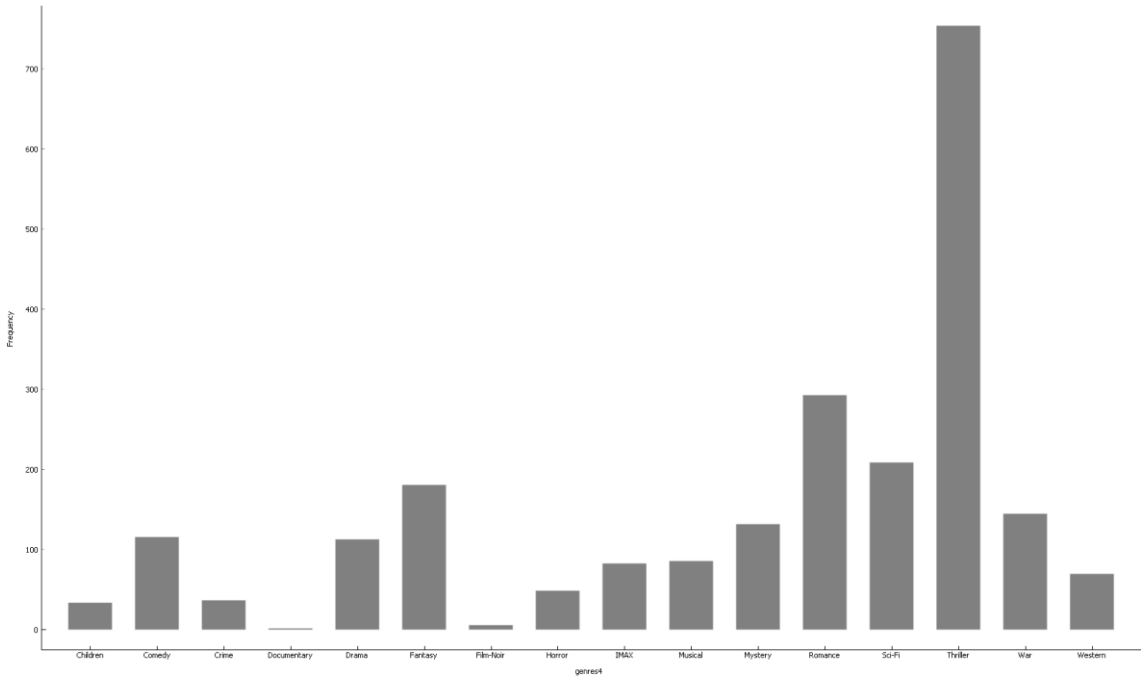

Gráfico 4: Gênero 4

Por fim, no nível 5, verificamos que os gêneros mais presentes foram Thriller, Sci-Fi e Fantasy.Os menos freqüentes foramFilm-Noir, Crime e Horror.

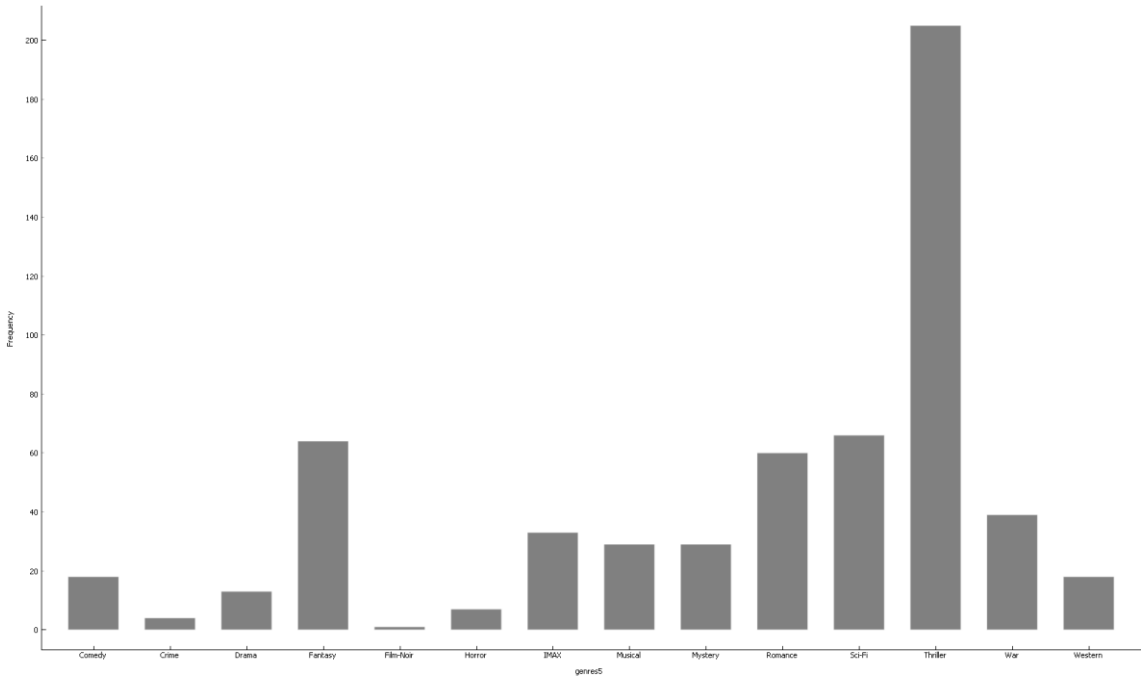

Gráfico 5: Gênero 5

Análise Exploratória do DatasetRating

Como primeiro passo de análise exploratória do dataset das notas dos filmes (“ratings.csv”) fornecidas pelos telespectadores, foi criado um gráfico de distribuição das notas. Assim, foi possível perceber que a maior concentração de notas é quatro, seguida de três e cinco. É importante destacar que não existe nota zero para os filmes e a menor nota fornecida pelos telespectadores é meio ponto.

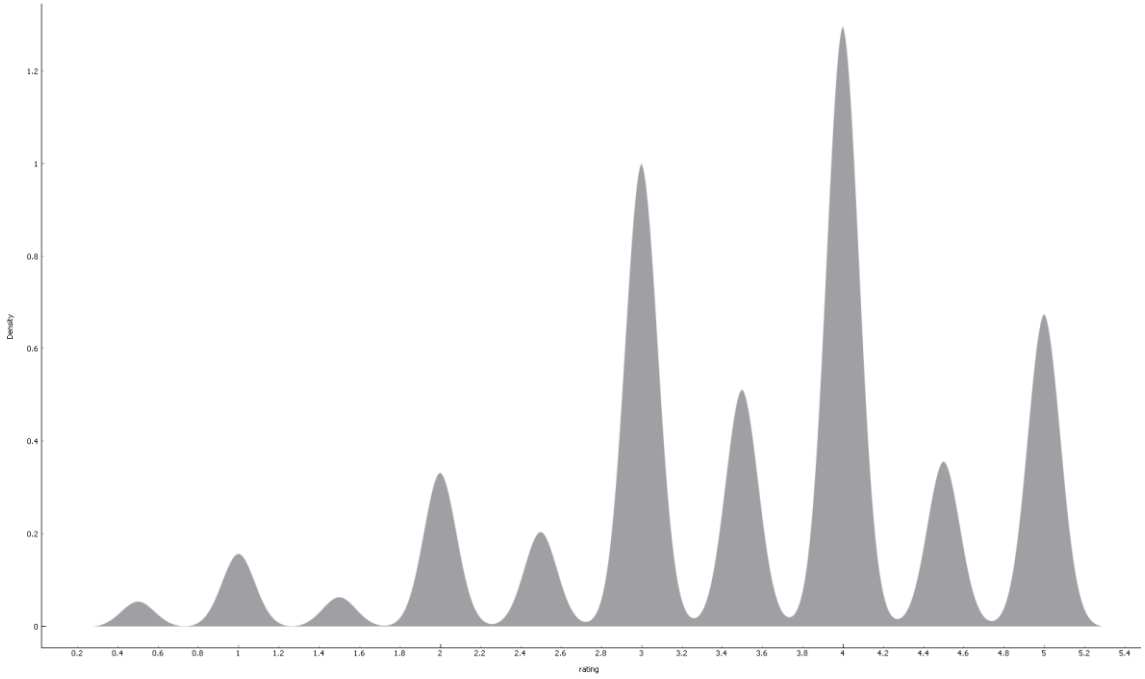

Gráfico 6: Gráfico de distribuição das notas dos filmes

Em continuidade a análise das notas dos filmes, foi gerado um gráfico boxplot das notas. Assim, temos como nota máxima cinco e nota mínima meio. A média das notas é 3,53 com desvio padrão de 1,05.

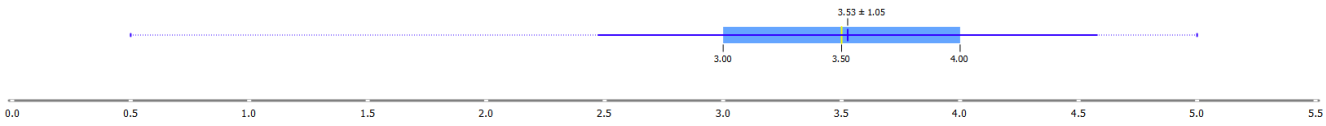

Gráfico 7:Gráfico Boxplot notas dos filmes

Para analisar nota média de cada filme foi criado uma query com “SELECT a.userId, b.genres, count(*), sum(a.rating), sum(a.rating)/count(*) from ratings a innerjoinmovies b onb.movieId = a.movieIdgroupbya.userId, b.genres” e exportado um novo csv chamado “movies_ratings_media.csv”. Por meio deste dataset criado, é possível encontrar os piores filmes e os melhores filmes por nota média fornecida

pelos telespectadores. Abaixo, temos o ScatterPlot da média de notas por Id dos filmes. O comportamento da amostra dos filmes é igual o já mencionado no Boxplot.

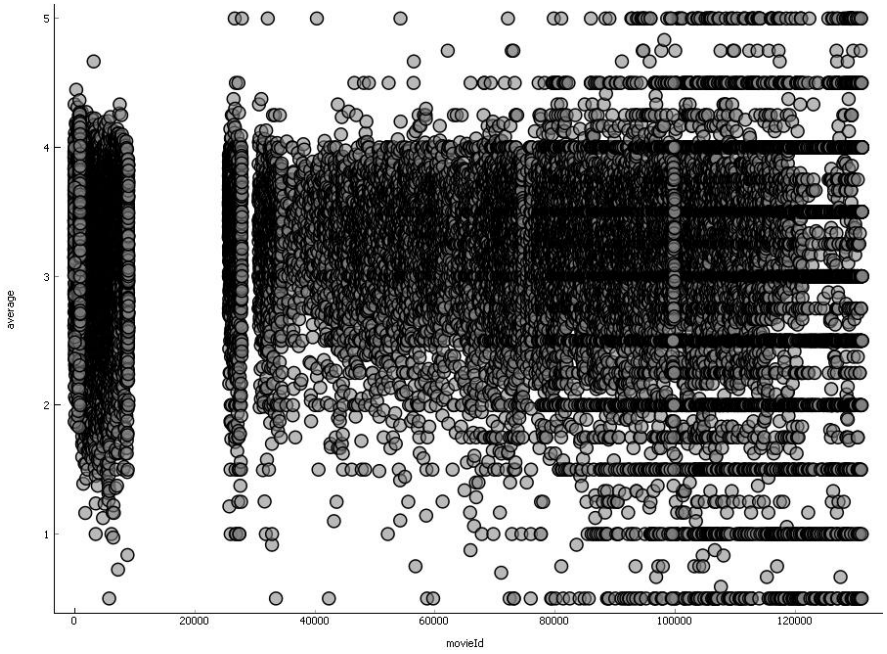

Gráfico 8: Scatterplot média notas filmes

Outra perspectiva para análise exploratória dos filmes é o datasetGenome com Genome_tag. Assim, foi gerada uma query “selecta.movieId, b.tagId, b.tag, c.averagefromgenomescores a innerjoingenometags b onb.tagId = a.tagIdinnerjoinmovie_ratings c onc.movieId = a.movieIdwhererelevance> '0.85'” (“modeid_tag_Genome_maior85.cvs”), ou seja, consideramos as genometags apenas as que o algoritmo reconheceu com a precisão de 85% para cima. Com este csv é possível analisar as características de cada filme segundo a perspectiva do algoritmo de reconhecimento de imagem que foi gerado. Por exemplo, o filmes de Id 1 têm as características demonstradas na tabela abaixo.

Tabela 3: Taggenomes filme id 1

movield	tag
1	adventure
1	animated
1	animation
1	cartoon
1	childhood
1	children
1	computer anim...
1	disney
1	disney animate...
1	friendship
1	fun
1	great movie
1	imdb top 250
1	kids
1	kids and family
1	light
1	original
1	pixar
1	pixar animation
1	story
1	toys

Analisando-se na perspectiva dos telespectadores, houve a necessidade de criar um novo dataset com a quantidade de filmes assistido por cada usuário (“userid_cont.csv”). A query que criou a base foi “SELECT userId,count(*) from ratings groupbyuserId”. Verificando a quantidade de registro de userId foi constatado que existem o histórico de 138.493 telespectadores.

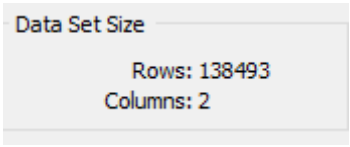

Imagem 1: Total userId

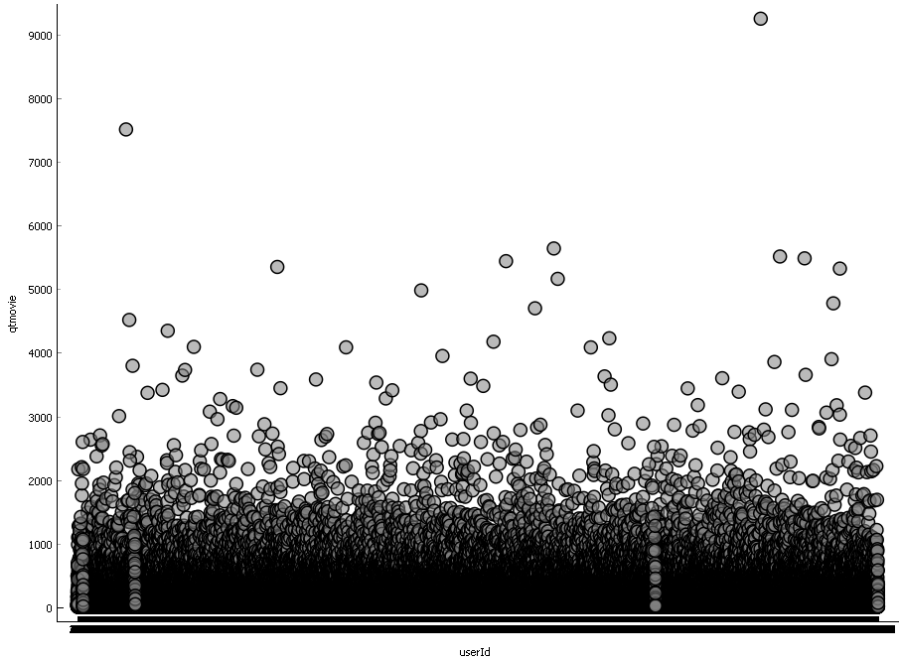

Gráfico 9: Gráfico scatterplot usuários e qt. filme

No gráfico acima é possível verificar que os telespectadores se concentram entre 20 amais ou menos mil filmes assistidos (20<=qt. Filmes assistido por userId<= +ou-1000).Em continuidade com a análise, abaixo temos os tops telespectadores que representam osoutliers da amostra e ao lado estão os valores base da amostra como já mencionado na descrição do dataset[12].

Tabela 4: Máximo e Mínimo filmes assistido

userId	count(*)
118205	9254
8405	7515
82418	5646
121535	5520
125794	5491
74142	5447
34576	5356
131904	5330
83090	5169
59477	4988

userId	count(*)
100012	20
10004	20
100057	20
100104	20
100131	20
100152	20
100153	20
100197	20
100225	20
100257	20
100268	20
10029	20

Através do atributo “timestamp” do dataset“ratings.csv”, é possível verificar o histórico de filmes assistido pelos telespectadores durante o intervalo de tempo

[12]http://files.grouplens.org/datasets/movielens/ml-20m-README.html

registrado. Para extrair a informação, foi gerado um dataset chamado “userid_cont_tempo.csv” com a query “SELECT userId, strftime('%Y%m ', datetime(timestamp, 'unixepoch')), count(*) from ratings groupbyuserId, strftime('%Y%m ', datetime(timestamp, 'unixepoch'))”. Importante destacar que o atributo “timestamp” foi tratado com mês e ano, ou seja, foi possível verificar a quantidade de filmes visto pelos telespectadores por mês e ano. Assim, temos por exemplo uma lista de usuários com a quantidade de filmes pela data[13].

Tabela 5: quantidade filme por usuário e data

userId	data	qt
1	200409	46
1	200504	129
2	200011	61
3	199912	187
4	199608	28
5	199612	66
6	199703	24
7	200201	276
8	199606	70
9	200107	35
10	199911	38
11	200901	419
11	200908	66
11	201101	19
12	199703	36
13	199611	62
14	200810	243
15	199608	49
16	200105	60
17	199912	1
17	200101	20
17	200105	3
17	200108	2
18	200711	63
18	200802	1
18	200803	1
18	200805	2
18	200806	1
18	200810	1
18	200902	4
18	200903	28
18	200904	1
18	200905	2
18	201001	7
18	201002	4
18	201009	6
19	199702	50
20	200509	28

Para analisar a média de nota dada pelos usuários por gênero, temos a query “SELECT a.userId, b.genres, count(*), sum(a.rating), sum(a.rating)/count(*) from ratings a innerjoinmovies b onb.movieId = a.movieIdgroupbya.userId, b.genres” (“userId_genres_media_nota.csv”). Com este csv, é possível entender por média de

[13]Lê-se ano e mês: 200409 -> ano 2004, mês 09.

nota de quais gêneros cada usuário costuma assistir, por exemplo, o userId 1. Este usuário é mais presente para filmes do gênero Action|Adventure|Fantasy com nota média considerável alta pela quantidade, já na perspectiva da nota para este usuário com uma quantia de filmes, temos o gênero Action|Adventure|Sci-Fi com média de nota 4.

Tabela 6: Usuário 1 filmes e médias

userId	genres	count(*)	n(a.rating) / coun
1	Action|Adventure|Fantasy	9	3.611111111111...
1	Adventure|Fantasy	7	4.214285714285...
1	Action|Adventure|Sci-Fi	5	4.000000000000...
1	Adventure|Comedy|Fantasy	4	3.875000000000...
1	Drama	4	3.625000000000...
1	Comedy	4	3.625000000000...
1	Comedy|Fantasy	3	3.833333333333...
1	Action|Crime|Drama|Thriller	3	3.833333333333...
1	Horror	3	3.666666666666...
1	Drama|War	3	3.666666666666...
1	Adventure|Children|Fantasy	3	3.666666666666...
1	Action|Adventure	3	3.666666666666...
1	Horror|Mystery|Thriller	3	3.500000000000...
1	Comedy|Horror|Thriller	2	4.000000000000...
1	Comedy|Crime|Thriller	2	4.000000000000...
1	Adventure|Fantasy|Romance	2	4.000000000000...
1	Action|Sci-Fi|Thriller	2	4.000000000000...
1	Action|Adventure|Sci-Fi|Thriller	2	4.000000000000...
1	Horror|Thriller	2	3.750000000000...
1	Drama|Horror|Thriller	2	3.750000000000...
1	Drama|Horror|Mystery|Thriller	2	3.750000000000...
1	Comedy|Fantasy|Romance	2	3.750000000000...
1	Adventure|Comedy|Fantasy|Sci-Fi	2	3.750000000000...
1	Action|Drama|War	2	3.750000000000...
1	Mystery|Thriller	2	3.500000000000...
1	Horror|Sci-Fi	2	3.500000000000...
1	Horror|Mystery	2	3.500000000000...
1	Crime|Mystery|Thriller	2	3.500000000000...
1	Comedy|Horror	2	3.250000000000...
1	Action|Crime	2	3.250000000000...
1	Action|Adventure|Thriller	2	3.250000000000...
1	Action|Adventure|Children|Fantasy	2	3.250000000000...
1	Crime|Drama|Horror	1	5.000000000000...
1	Action|Adventure|Drama|Fantasy	1	5.000000000000...
1	Action|Adventure|Sci-Fi|IMAX	1	4.500000000000...
1	Thriller	1	4.000000000000...
1		1	4.000000000000

Sistema de Recomendação

Como sistema de recomendação através do dataset analisado, há a possibilidade de criar três tipos de sistemas:

a) Sistema de Filtragem Baseada em Conteúdo: faz-se a sugestão de itens que sejam semelhantes aos que o usuário demonstrou interesse no passado e/ou sobre as configurações de preferências do usuário;

b) Similaridade de Item/Usuário: consiste em descobrir itens similares aos que o usuário já adquiriu ou descobrir similaridade entre os usuários (vizinhos mais próximo).

c) Método Híbrido: baseia-se na combinação das duas técnicas acima descritas.

Possíveis estratégias de modelos:

1. Busca Booleana tipo Similaridade de Item/Usuário: Com o objetivo de criar um modelo de sistema de recomendação para os telespectadores que assistiram algum tipo de filme e para os que assistirão no futuro. Permite-se criar um sistema relacionado ao gênero do filme assistido baseado na nota que o telespectador deu ao filme. Assim, para os telespectadores que assistirem algum filme de gênero X e pontuarem como nota quatro ou mais (4<=nota<=5), o sistema recomendaria um filme com gênero semelhante baseado com a notas fornecidas pelo telespectador.
2. Outra perspectiva de Similaridade de Item/Usuário: Montar uma matriz em que cada elemento ij representa a avaliação média do usuário i no gênero j. Dado um novo usuário com as suas avaliações médias em cada gênero, correlaciona-se às avaliações médias desse novo usuário com cada usuário na matriz. Supondo que o novo usuário possui alta correlação com um usuário i, recomendar ao novo usuário os filmes que o usuário i já assistiu.
3. Na perspectiva de Método Híbrido para recomendação: Cria-se um método que calcula a distância euclidiana[14] entre os filmes ou os usuários do dataset. O cálculo da distância é feito em duas perspectivas: nas notas que os filmes

[14]https://pt.wikipedia.org/wiki/Dist%C3%A2ncia_euclidiana

receberam ou/e nas notas que os usuários deram para cada filme (perfil do usuário). Com isto, torna-se possível calcular a similaridade dos filmes e a similaridade dos usuários no dataset. Por fim, cria-se um sistema tanto para recomendar produtos no estoque (filmes no catálogo), quanto para usuários que não viram os filmes, ou seja, um sistema de recomendação desimilaridade de itens ou usuários.

Em continuidade, na proposta de solução para o sistema de recomendação, foi escolhida a estratégia de sistema Híbrido (escolha número três), assim, para testar a solução desenvolvida foi utilizado o dataset resumido fornecido pelo *MovieLens*.

A solução implementada e testada encontra-se no notebook "SistemaDeRecomendacao.ipynb". Como já mencionado anteriormente, o sistema atende a perspectiva dos filmes do catálogo e dos perfis dos usuários baseando-se nas notas.

Na solução, temos os métodos de leitura e tratamento do dataset para o sistema (carregaMovieLensUsuario e carregaMovieLensFilme). Existe também o método que calcula a distância euclidiana (euclidiana. O retorno varia de zero a cem por cento, ou seja, se o retorno for mais próximo de zero, os objetos são distantes (diferentes) e, caso for mais próximo de cem, os objetos são parecidos (semelhantes).

O método "getSimilares" retorna uma lista de similaridade de filmes no dataset filme para com o filme desejado ou similaridade de usuário para o usuário desejado e o retorno também é variado de zero a cem seguindo o conceito apresentado acima. Já o método "calculaObjetosSimilares" retorna todas as similaridades de filmes ou usuários do dataset variando de zero a cem por cento.

Por fim, o método "getRecomendacoesObjetos" retorna à recomendação de filmes que são semelhantes ao filme que foi visto e a possível nota que irá receber, já na perspectiva do usuário é recomendado o filme para ele e a possível nota que irá receber caso o usuário venha a assistir o filme (predição da nota varia de zero a cinco).

Avaliação do Sistema de Recomendação

Em análise ao sistema de recomendação implementado e testado, alguns pontos são importantes de serem abordados:

- Ao utilizar os métodos“getSimilares” e “calculaObjetosSimilares” em um dataset de tamanho considerável, poderá ocasionar um problema de processamento (memória) e tempo de resposta, assim, os métodos da maneira que foram implementados irão correr todo o dataset e calcular a similaridade. Para solucionar o problema de memória e tempo de resposta, o ideal é dividir o dataset em subgrupos ou clusters tanto para filmes quanto para usuário. Assim, não será necessário correr todo o dataset.
- Como estratégia de criação de subgrupos de filmes, poderia ser por similaridade de notas, por gênero ou por“genomes”. Já os usuários poderiam ser divididos em similaridade de notas, quantidade de filmes vistos, grupos de idade, região ou país.
- No contexto da recomendação de filmes para os usuários (getRecomendacoesObjeto), é possível aplicar uma heurística de recomendação através da média das notas que o usuário costuma dar, por exemplo, o usuário 1 assistiu o filme ToyStory e deu nota 2.0, no filme Copycat deu a nota 3.0, no filme Taxi Driver nota 4.0 e no filme Apollo 13 deu a nota 3.0. Assim, o cálculo heurístico é h=(2+3+4+3)/4 o resultado é 3. Então, o sistema recomendaria os filmes em que a predição da nota fosse maior ou igual a três (nota>=3). Com a heurística aplicada, o resultado seria filtrado e preciso na hora da recomendação.
- Heurística para organizar os filmes do catalogo: No caso em análise, o catálogo dos filmes ou produtos em estoque de uma loja são mais estáticos, ou seja, não mudam constantemente em comparação com a base de usuários. Isto possibilita a oportunidade de criar uma tabela de similaridade dos itens do estoque ou catálogos dos filmes em tempo desejado pelo negócio. Com esta heurística, evitará sobrecarga de processamento de memória e tempo de resposta da recomendação. Com isto, antes mesmo que o usuário termine de

assistir o filme, já existe recomendações para ele. Esta estratégia pode ser aplicada para casos de usuários Cold-Start (não se sabe o perfil do usuário), o sistema recomendaria alguns itens ou filmes que tenha similaridade baseando-se no item que o usuário clicou ou chegou a ver alguma informação.