IV SEMINÁRIO INTERNACIONAL DE ESTATÍSTICA COM R
R & PYTHON E AS TENDÊNCIAS DE COLABORAÇÃO
NITERÓI, 21 A 23 DE MAIO DE 2019

SISTEMAS DE RECOMENDAÇÃO USANDO O SOFTWARE R

Leonardo Filgueira[1]

Luciane Ferreira Alcoforado[2]

Rodrigo Otávio de Araújo Ribeiro[3]

Resumo

O artigo apresenta algumas técnicas utilizadas na realização de sistemas de recomendação e realiza um estudo de caso, ao utilizar uma base de dados de avaliações de filmes por usuários. É aplicada sobre essa base uma técnica de filtragem colaborativa a fim obter recomendações de filmes aos usuários. Informações como gêneros dos filmes, avaliação dos usuários e número de filmes avaliados serão utilizadas de forma a agrupar os usuários em clusters, utilizando as técnicas de particionamento CLARA e K-means, de forma a aplicar a filtragem colaborativa para cada um dos clusters, em separado. Realiza-se uma comparação de acurácia dos modelos, fazendo a divisão em base de treino (para construir o modelo) e base de teste (para verificar a acurácia), além de uma verificação do tempo de execução, com o objetivo de verificar se, nas condições desse estudo, a clusterização torna a recomendação mais acurada.
Palavras-chave: sistemas de recomendação, filtragem colaborativa, clusterização.

Abstract

The article presents some techniques used in the recommendation systems realization and performs a case study, using a user movie rating database. A collaborative filtering technique is applied on the database in order to acquire movie recommendations to users. Informations like movie genre, users rating and number of rated movies will be used to group the users in clusters, using the partitioning techniques CLARA and K-means, in order to apply the collaborative filtering to each cluster, separated. An accuracy comparison is performed in the models, doing the separation in training base (to build the model) and test base (to verify the accuracy), besides a runtime verification, for the purpose of verifying if, in this study conditions, clustering makes the recommendation more accurate.
Keywords: recommender systems, collaborative filtering, clustering.

Introdução

A partir do aumento de informação disponível com a popularização da Internet e com a possibilidade de armazená-las, surge o desafio de lidar com este grande conjunto de dados (ISINKAYE et al, 2015). Este aumento de informações desafia o site (loja, rede social,

[1] Universidade Federal Fluminense (UFF), leonardo_filgueira@id.uff.br

[2] Universidade Federal Fluminense (UFF), lucianea@id.uff.br

[3] Universidade do Estado do Rio de Janeiro (UERJ), rodrigo.ribeiro@ibopedtm.com

streaming de vídeo/músicas) que recebe todos os dados dos usuários que visitam o endereço, mas também pode se tornar um problema para o usuário que, diante da grande quantidade de produtos disponíveis para compra, pode levar muito tempo para achar o produto desejado (MILD et al, 2002).

Sistemas de recomendação podem ser definidos como técnicas de aprendizado de máquina que filtram um grande conjunto de dados, tendo como base informações dos usuários e itens (TAKAHASHI et al, 2015). A partir dessas técnicas são indicados item(ns) aos usuários. De maneira mais simples, sistemas de recomendação são técnicas que fornecem sugestões de itens, de forma que os usuários possam tomar melhores decisões (GORAKALA et al, 2015), a depender do contexto onde a recomendação se aplica. Os sistemas de recomendação têm como objetivo recomendar itens que interessariam aos usuários (MELVILLE et al, 2011), beneficiando o usuário e a loja, pois eles aumentam o desempenho da loja, fazendo-a vender uma quantidade maior de produtos, e também facilitam a procura do usuário fazendo-o achar o(s) produto(s) desejado(s) em um menor tempo (ISINKAYE et al, 2015).

É facilmente perceptível no cotidiano o uso de sistemas de recomendação em ambientes on-line. Ao usar a *Netflix*, sugestões para o usuário são oferecidas, baseadas nas atrações já assistidas e/ou avaliadas. Sites de compras como a *Amazon* também oferecem sugestões de produtos ao usuário baseado em visitas à página dos produtos ou no comportamento de outros usuários que compraram um mesmo produto. Também em redes sociais, como no *YouTube*, são sugeridos vídeos baseados no histórico do internauta e nas suas avaliações, ou então no *Facebook*, que recomenda lista de pessoas que o usuário pode conhecer (GORAKALA et al, 2015).

Uma forma de comunicar ao usuário a recomendação processada é por meio do e-mail marketing, que é um canal de comunicação direto com o cliente (TAKAHASHI et al, 2015). Após a pessoa aceitar receber tais mensagens, pode-se utilizar dessa comunicação direta para enviar ao usuário os itens para ele recomendados. Assim, podem-se aplicar recomendações também no meio físico, bastando cadastrar os clientes e utilizar o e-mail marketing.

Existem 3 tipos de sistemas de recomendação: filtragem baseada em conteúdo, filtragem colaborativa e sistemas de recomendação híbridos (MELVILLE et al, 2011), podendo o último tipo ser a aplicação das duas primeiras filtragens separadamente ou um único modelo que une as duas abordagens (TAKAHASHI et al, 2015), aproveitando suas vantagens e buscando eliminar suas desvantagens (SHAPIRA et al, 2011).

Na filtragem baseada em conteúdo recomenda-se itens similares aos que o usuário gostou no passado (GORAKALA et al, 2015). Para isso é necessário utilizar informações das características de um produto (SHAPIRA et al, 2011) e comparar com o perfil do usuário, de acordo com itens já conhecidos pelo indivíduo. Por outro lado, a filtragem baseada em conteúdo não leva em conta a similaridade de preferência entre os usuários, mas apenas o histórico do usuário e as características dos itens (GORAKALA et al, 2015).

Na filtragem colaborativa são recomendados itens de acordo com as avaliações de todos os usuários (MELVILLE et al, 2011). Existem duas maneiras principais de realizar essa filtragem: baseado em memória ou em modelo (DAKHEL et al, 2011). Nos algoritmos baseados em memória, verifica-se a similaridade entre usuários ou entre itens (vizinhança), de acordo com suas avaliações passadas. A filtragem colaborativa é o tipo mais utilizado para realizar recomendações (SHAPIRA et al, 2011). Os algoritmos que executam essa tarefa utilizam uma matriz, que contém as avaliações dos usuários aos itens, como mostra a Tabela 1.

Tabela 1 – Matriz R de avaliações.

Usuário	**Item**			
	i_1	i_2	$\cdots$	i_m
u_1	$r_{(1,1)}$		$\cdots$	
u_2		$r_{(2,2)}$	$\cdots$	$r_{(2,m)}$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
u_n			$\cdots$	$r_{(n,m)}$

Fonte: FILGUEIRA, 2018

Onde $r_{(i,j)}$ é a avaliação dada pelo usuário i ao item j. Em geral, os usuários não tiveram contato com todos os itens, logo os itens não recebem avaliações de todos os usuários, produzindo então uma matriz esparsa (com grande quantidade de valores faltantes). Os algoritmos buscam, então, preencher a matriz de avaliações com estimativas para os valores faltantes.

À medida, porém, que os números de usuários e itens aumentam, podem surgir problemas ao realizar a filtragem, como o aumento do tempo necessário para execução, além de recursos computacionais, para executar o algoritmo, chamado de problema de escalabilidade (DAKHEL et al, 2011). Buscando reduzir o tempo de processamento e melhores medidas de acurácia podem ser utilizados métodos de agrupamento (O'CONNOR et al, 1999).

Objetivo

O objetivo geral do artigo é realizar um estudo de caso a partir de uma base de avaliação de filmes, de forma a avaliar a acurácia das recomendações utilizando filtragem colaborativa baseada em memória para todo o conjunto de dados e as recomendações utilizando filtragem colaborativa baseada em memória para cada cluster de usuários. Além disso, objetivos específicos são analisar as recomendações para um usuário específico e comparar o tempo de execução dos algoritmos.

Material e Método

Foi utilizado um conjunto de dados obtido no site *GroupLens* (disponível em *https://grouplens.org/datasets/movielens/1m/*). O conjunto de dados possui 1.000.209 avaliações de 3.900 filmes dados por 6.040 usuários (HARPER et al, 2016), que se cadastraram no site *MovieLens* no ano de 2000. De acordo com o próprio site, pessoas podem se inscrever para avaliar filmes e receber recomendações de filmes para assistir. Os usuários são representados pelo seu ID, que varia entre 1 e 6040 e os filmes possuem ID entre 1 e 3952. As avaliações têm formato numérico, de até 5 estrelas, com estrelas completas, tendo cada usuário avaliado ao menos 20 filmes.

A base de dados foi dividida em duas, treino e teste, na proporção de 70% dos dados para treinar o modelo (base de treino) e 30% da base que foi usada para testar o modelo (base de teste). Assim, as notas dadas pelos usuários pertencentes à base de teste foram utilizadas para comparar com as notas previstas pelo modelo. Foi utilizada uma segunda base, que apresenta informações sobre os filmes, como o código, nome e gêneros do filme, com o objetivo de executar a tarefa do agrupamento dos usuários. Um mesmo filme pode ter sido associado a mais de um gênero, mas nenhum filme não foi associado a algum dos 18 gêneros existentes.

Existem um conjunto de usuários $U = \{u_1, u_2, \dots, u_n\}$ e um conjunto de itens $I = \{i_1, i_2, \dots, i_m\}$, assim como as notas dos usuários aos itens, que são armazenadas na matriz $R_{n \times m}$ de avaliações (HAHSLER, 2015). O algoritmo utilizado buscará preencher os valores faltantes desta matriz, com valores na mesma escala das avaliações presentes na matriz (TAKAHASHI et al, 2015).

O algoritmo de filtragem colaborativa utilizado baseia-se no usuário e assume que usuários com preferência similar no passado terão preferências similares no futuro. Então as avaliações não observadas serão previstas a partir das avaliações de uma vizinhança e

usuários com gostos similares (HAHSLER, 2015). O coeficiente de correlação de Pearson pode ser utilizado como medida de similaridade entre dois usuários a e u, definido da seguinte maneira (MELVILLE et al, 2011):

$$w_{a,u} = \frac{\sum_{i \in I}(r_{a,i} - \bar{r}_a)(r_{u,i} - \bar{r}_u)}{\sqrt{\sum_{i \in I}(r_{a,i} - \bar{r}_a)^2 \sum_{i \in I}(r_{u,i} - \bar{r}_u)^2}}$$

Sendo I o conjunto de itens avaliados pelos dois usuários, $r_{u,i}$ a avaliação dada pelo usuário u ao item i e $\bar{r}_u$ a avaliação média do usuário u a todos os itens por ele avaliados. Em filtragem colaborativa, esse coeficiente é muito usado (SU et al, 2009) e atinge melhores resultados (BREESE et al, 1998). Por esses motivos, a correlação de Pearson será utilizada como medida de similaridade.

Já a predição da nota dada ao item i pelo usuário a é dada por:

$$p_{a,i} = \bar{r}_a + \frac{\sum_{u \in V}(r_{u,i} - \bar{r}_u) w_{a,u}}{\sum_{u \in V} |w_{au}|}$$

Sendo V a vizinhança do usuário a.

De modo a agrupar os usuários para, então aplicar a filtragem colaborativa, foram utilizadas duas técnicas de particionamento: CLARA e K-means.

A técnica CLARA (*Clustering Large Applications*) foi proposta em 1990, de forma a aplicar a técnica PAM (Partitioning Around Medoids), utilizando amostragem para a aplicação da técnica (PARK et al, 2009). O método, então, seleciona aleatoriamente uma parte da base de dados e aplica o algoritmo PAM nesta amostra. O algoritmo de agrupamento PAM é baseado na definição de *medoide*, que é o ponto com menor distância, em média, de todos os outros elementos do cluster. O algoritmo, para obter k clusters, é executado da seguinte maneira (VALE, 2005):

1. Definir aleatoriamente k medoides.
2. Associar cada um dos elementos restantes ao cluster de medoide mais próximo.
3. Calcular a dissimilaridade entre um elemento x_i e todos os outros do cluster, e a dissimilaridade entre o medoide e os outros elementos do cluster.
4. Caso a distância considerando x_i como novo medoide seja menor que a distância do medoide atual, passe a considerar x_i como medoide daquele cluster.
5. Repetir os passos 2 a 4 até não haver troca de medoides.

Uma desvantagem do PAM é a ineficiência ao ser aplicado para um grande conjunto de dados (PARK et al, 2009). Essa é a razão pela qual foi escolhida a técnica CLARA. O algoritmo que executa a técnica segue calculando a função de custo, que é uma média da similaridade

entre os medoides e os outros elementos da base (BHAT, 2014). A função de custo é definida da seguinte maneira:

$$C(m, D) = \frac{\sum_{i=1}^{n} d\left(x_i, cl(m, x_i)\right)}{n}$$

Onde m são os medoides encontrados, $cl(m, x_i)$ é o medoide mais próximo de um ponto x_i, $d\left(x_i, cl(m, x_i)\right)$ é uma medida de similaridade entre x_i e seu medoide mais próximo (tendo sido utilizada neste trabalho a distância Euclidiana) e n é o número de observações na base de dados D.

Todo o processo é repetido um número determinado de vezes e o resultado que obtiver menor função de custo é definido então como o melhor e é retornado (BHAT, 2014).

K-means é uma técnica que particiona elementos em k clusters utilizando-se de centroides, que são elementos representativos de cada cluster. Este método busca minimizar a soma das distâncias dos elementos de um mesmo cluster. Dados então, uma matriz D, de dimensão $m \times n$, e um número de clusters k, o algoritmo, então, procede da seguinte maneira (HAN et al, 2011):

1. São escolhidos, aleatoriamente, k objetos de D como sendo os centroides.
2. Cada elemento $D_{i,}$ é associado ao centroide mais próximo, de acordo com a medida de distância adotada (neste caso, a distância Euclidiana).
3. Os centroides de cada um dos clusters são calculados.
4. Repetir os passos 2 e 3 até que não haja mudanças.

A fim de realizar as técnicas de agrupamento descritas acima, foram utilizadas duas informações a partir das bases de dados disponíveis: a avaliação média dos usuários para cada categoria e a proporção de filmes assistidos pelos usuários para cada categoria. No primeiro caso, utilizando o rating médio, houveram casos em que alguns gêneros não receberam nenhuma nota, gerando um dado faltante. Para preencher os valores faltantes foi usada a média de notas por categoria. Além disso, o número de clusters foi variado entre 2 e 15.

Para verificar a acurácia do modelo foi utilizada a raiz do erro quadrático médio, sendo o segundo definido da seguinte maneira:

$$EQM = \frac{1}{nm}\sum_{i=1}^{n}\sum_{j=1}^{m}\left(r_{i,j} - p_{i,j}\right)^2$$

Todas as técnicas utilizadas foram aplicadas utilizando a linguagem R, por meio do ambiente de desenvolvimento integrado RStudio. O pacote *recommenderlab* (HAHSLER, 2017), do R foi usado para particionar a base em treino e teste, criar os modelos, fazer previsão e avaliar os resultados. Também foram usados os pacotes *dplyr (*WICKHAM, FRANÇOIS, HENRY e MÜLLER, 2018*), tidyr* (WICKHAM e HENRY, 2018)*, tibble* (MÜLLER e WICKHAM, 2018)*, plyr* (WICKHAM, 2011)*, data.table* (DOWLE e SRINIVASAN, 2018) e *cluster* (MAECHLER, ROUSSEEUW, STRUYF, HUBERT e HORNIK, 2018).

Resultados e Discussão

A matriz de avaliações obtida possui 1.000.209 elementos preenchidos. Desta maneira é possível verificar que aproximadamente 4,68% da matriz possui avaliações. Como uma matriz esparsa, nota-se que a maior parte dos elementos da matriz não possui valores.

Os 6040 usuários avaliaram pelo menos 20 filmes. Na figura 1, é possível notar que a maior parte dos usuários avaliou até 500 filmes. Além disso, nota-se que essa distribuição apresenta uma assimetria a direita.

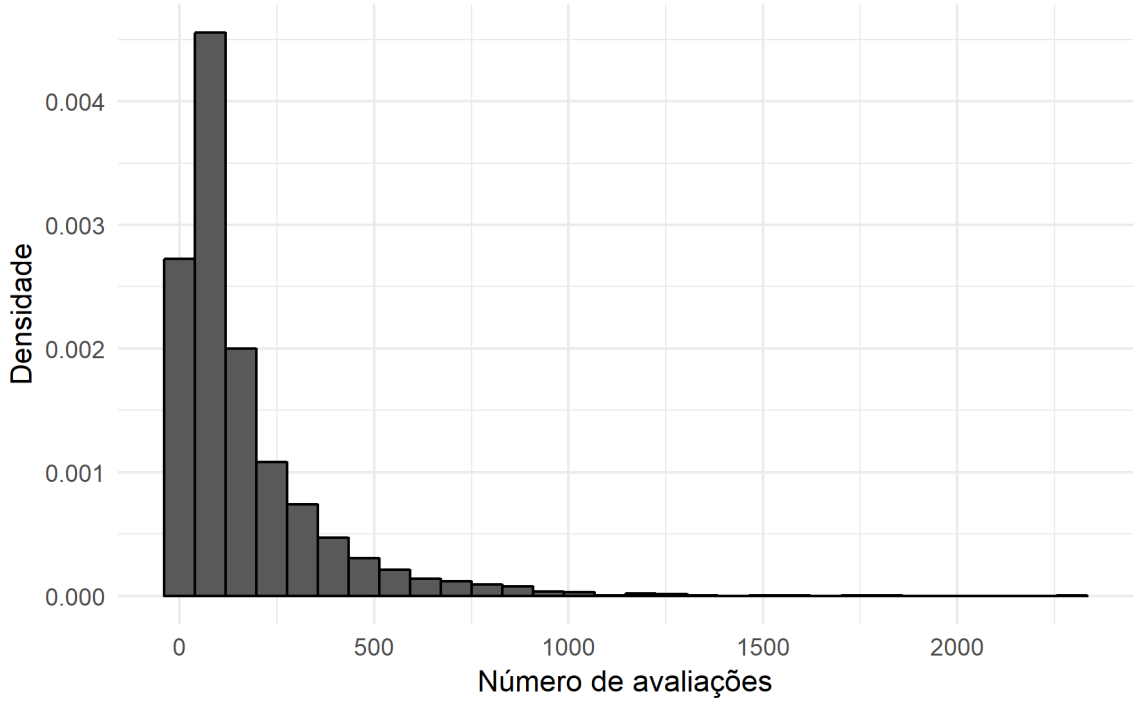

Figura 1 – Distribuição do número de filmes avaliados pelos usuários

Fonte: FILGUEIRA, 2018

A Tabela 2 apresenta algumas medidas resumo a respeito da quantidade de filmes avaliados pelos usuários. Nota-se uma grande amplitude, variância, desvio padrão e coeficiente de variação, o que indica uma grande variabilidade na quantidade de filmes avaliados. Metade dos usuários avaliou até 96 filmes. Além disso, 75% dos usuários avaliaram até 208 filmes, o que indica, como já foi indicado na Figura 1, que um número pequeno de usuários, diferentemente do comportamento da maior parte, avaliou uma grande quantidade de filmes.

Tabela 2 – Medidas resumo da quantidade de filmes avaliados por usuários.

Medida resumo	Mínimo	Mediana	Média	Máximo	Variância	Desvio padrão	Coef. de variação
	20	96	165,6	2314	37151	193	116%

Fonte: FILGUEIRA, 2018

A seguir, a Tabela 3 apresenta os 10 filmes com maior número de avaliações recebidas. Destaca-se a trilogia original de *Star Wars*, cujos filmes receberam, entre si, uma quantidade muito próxima de avaliações dos usuários.

Tabela 3 – Os 10 filmes com mais avaliações recebidas.

Filme	Número de avaliações
American Beauty (1999)	3428
Star Wars: Episode IV - A New Hope (1977)	2991
Star Wars: Episode V - The Empire Strikes Back (1980)	2990
Star Wars: Episode VI - Return of the Jedi (1983)	2883
Jurassic Park (1993)	2672
Saving Private Ryan (1998)	2653
Terminator 2: Judgment Day (1991)	2649
Matrix, The (1999)	2590
Back to the Future (1985)	2583
Silence of the Lambs, The (1991)	2578

Fonte: FILGUEIRA, 2018

A Tabela 4 apresenta a quantidade de filmes aos quais cada gênero foi atribuído e a proporção de usuários que avaliaram cada gênero. Como cada filme pode ter sido descrito

com mais de um gênero, a soma das frequências é maior que o número de filmes. Nota-se que os gêneros aos quais mais filmes foram associados são drama e comédia. O terceiro gênero com mais filmes associados, ação, representa menos de metade do número de filmes, em relação aos dois primeiros. Além desses três gêneros, aventura, romance, *thriller*, *sci-fi* e guerra possuem uma proporção acima de 95% de usuários que os avaliaram.

Tabela 4 – Gêneros existentes na base e número de filmes associados.

Gênero	Número de filmes	Proporção de usuários
Action	503	0,9954
Adventure	283	0,9758
Animation	105	0,796
Children's	251	0,8747
Comedy	1200	0,9985
Crime	211	0,9374
Documentary	127	0,3714
Drama	1603	0,9995
Fantasy	68	0,803
Film-Noir	44	0,6871
Horror	343	0,8775
Musical	114	0,7871
Mystery	106	0,8498
Romance	471	0,9869
Sci-Fi	276	0,9786
Thriller	492	0,9916
War	143	0,9551
Western	68	0,6788

Fonte: FILGUEIRA, 2018

A Figura 2 apresenta as distribuições de nota média recebida para os gêneros presentes na base. A mediana das avaliações médias encontra-se entre 3 e 4 estrelas, com apenas os gêneros *War* (guerra), *Film-Noir* (uma espécie de filme policial) e *Documentary* (documentário) atingindo uma mediana igual a 4. As medianas mais baixas encontram-se em *Horror* e *Sci-Fi* (ficção científica), com 3,38 e 3,58 como medianas, respectivamente.

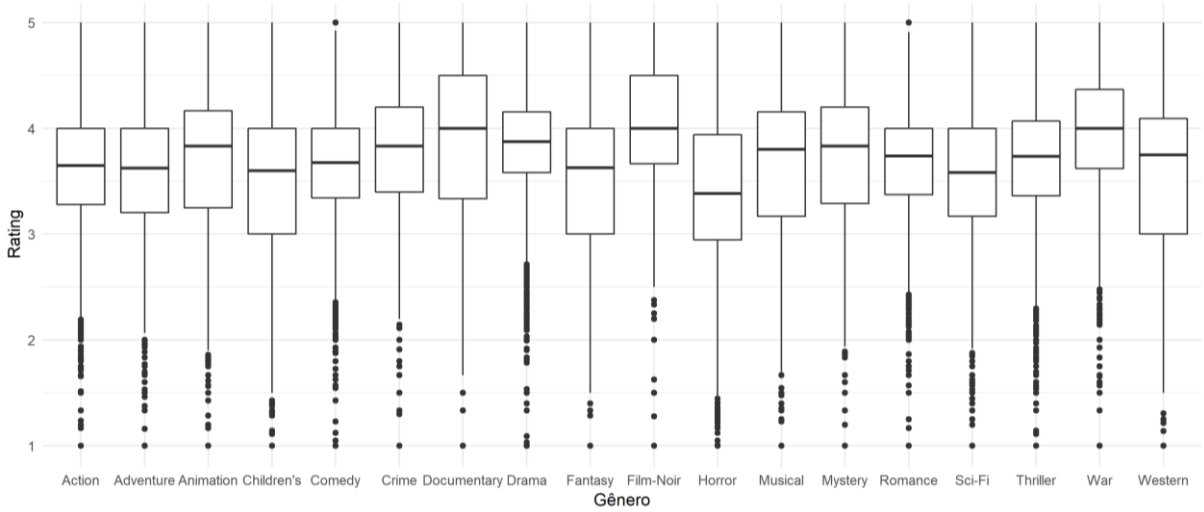

Figura 2 – Distribuição da avaliação médio por gênero

Fonte: FILGUEIRA, 2018

A seguir será verificada a recomendação para um usuário em específico, que avaliou 20 filmes. A seleção da pessoa foi feita aleatoriamente. Primeiramente devem ser apresentadas as avaliações do usuário aos filmes, para comparar com os filmes que seriam recomendados para ele. A Tabela 5 apresenta essa informação.

Tabela 5 – Avaliações do usuário 341.

Filme	Gênero	Avaliação
Nikita (La Femme Nikita) (1990)	Thriller	5
Mission: Impossible (1996)	Action, Adventure, Mystery	5
Somewhere in Time (1980)	Drama, Romance	5
East of Eden (1955)	Drama	5
Braveheart (1995)	Action, Drama, War	5
Hard-Boiled (Lashou shentan) (1992)	Action, Crime	5
Out of Sight (1998)	Action, Crime, Romance	5
American Beauty (1999)	Comedy, Drama	5
Airplane! (1980)	Comedy	5
Boat, The (Das Boot) (1981)	Action, Drama, War	5
Contact (1997)	Drama, Sci-Fi	4
Frequency (2000)	Drama, Thriller	4
Superman (1978)	Action, Adventure, Sci-Fi	4
Tank Girl (1995)	Action, Comedy, Musical, Sci-Fi	4
Alien (1979)	Action, Horror, Sci-Fi, Thriller	4

Pitch Black (2000)	Action, Sci-Fi	3
Shanghai Noon (2000)	Action	3
Run Lola Run (Lola rennt) (1998)	Action, Crime, Romance	3
Jurassic Park (1993)	Action, Adventure, Sci-Fi	3
Perfect Storm, The (2000)	Action, Adventure, Thriller	2

Fonte: FILGUEIRA, 2018

De acordo com a Tabela 5, nota-se que a maior parte dos filmes avaliados é do gênero de ação, porém alguns receberam avaliações muito boas, de 5 estrelas, enquanto alguns receberam apenas 3 ou até mesmo 2 estrelas. Os filmes que contém comédia, drama ou guerra em geral receberam boas notas, com, pelo menos 4 estrelas. Ao executar a filtragem colaborativa, selecionando as 10 maiores avaliações previstas, tem-se uma recomendação de 10 filmes para esse usuário. A Tabela 6 apresenta o que seria a recomendação dos filmes para esse mesmo usuário.

Tabela 6 – Recomendação de 10 filmes para o usuário 341.

Filme	**Gênero**	**Avaliação prevista**
Pulp Fiction (1994)	Crime, Drama	4,55
Schindler's List (1993)	Drama, War	4,55
Casablanca (1942)	Drama, Romance, War	4,52
Sixth Sense, The (1999)	Thriller	4,51
L.A. Confidential (1997)	Crime, Film-Noir, Mystery, Thriller	4,51
Gladiator (2000)	Action, Drama	4,49
Being John Malkovich (1999)	Comedy	4,48
Saving Private Ryan (1998)	Action, Drama, War	4,48
Godfather, The (1972)	Action, Crime, Drama	4,47
Shakespeare in Love (1998)	Comedy, Romance	4,47

Fonte: FILGUEIRA, 2018

A recomendação, levando em conta os gêneros, é aceitável, pois verifica-se notas altas previstas a filmes de drama, guerra, comédia, e até ação. Os três primeiros gêneros receberam apenas avaliações boas pelo usuário, mas o último recebeu avaliações positivas e também negativas, porém muitos dos filmes avaliados eram desse gênero, o que pode indicar algum interesse do usuário por este tipo de filme. Além disso, destaca-se o filme *L.A. Confidential*, que é associado ao gênero *Film-Noir*, sendo próximo de filmes de ação ou crime.

A seguir serão apresentados a raiz do erro quadrático médio e o erro quadrático médio entre avaliação prevista e observada. O número de clusters foi variado entre 2 e 15, utilizando as técnicas CLARA e k-means. Como a Tabela 6 apresenta, o método CLARA com 3 clusters encontrados baseando-se na avaliação média por categoria obteve uma acurácia maior. Além disso pode-se notar que o método CLARA apresentou melhores resultados com um número menor de clusters, de até 5 grupos, enquanto que o K-means apresentou bons resultados com um número maior de grupos, de 8 até 15, número máximo de clusters.

Tabela 7 – As 10 melhores acurácias.

Método	**Variável utilizada**	**Clusters**	**Raiz do EQM**	EQM
CLARA	Avaliação média	3	1,0055	1,0199
K-means	Proporção de filmes	12	1,0213	1,0444
K-means	Proporção de filmes	8	1,024	1,0495
K-means	Avaliação média	12	1,0251	1,052
CLARA	Avaliação média	4	1,0254	1,0634
K-means	Proporção de filmes	15	1,0266	1,055
K-means	Avaliação média	11	1,0278	1,0568
CLARA	Proporção de filmes	5	1.0288	1,0599
K-means	Avaliação média	15	1.03	1,0624
CLARA	Proporção de filmes	7	1.0315	1,0668

Fonte: FILGUEIRA, 2018

Ao executar a recomendação sem qualquer clusterização, ou seja, considerando toda a base de treino, os erros ao comparar as notas previstas com as notas observadas na base de teste foram os seguintes: Raiz do EQM = 1,0341 e EQM = 1,0693. O valor de referência é o erro obtido ao executar a recomendação sem o particionamento da base. A menor raiz do EQM, utilizando a técnica CLARA a partir do rating médio dos usuários aos gêneros, com 3 clusters, de 1.0055 é aproximadamente 3% menor que a mesma medida sem o particionamento dos usuários. Ainda vale ressaltar que a maior parte das recomendações utilizando algum dos métodos de clusterização obteve um erro maior do que o valor de referência.

Além disso, ao agrupar os usuários o tempo de processamento diminuiu. Como indica a Tabela 8, percebe-se uma tendência ao decréscimo do tempo necessário, com uma diferença de aproximadamente 100 segundos entre a recomendação com a base completa e com 15 clusters, com a técnica CLARA, tendo sido utilizada a avaliação média. O tempo foi

calculado considerando preparação da base e clusterização (quando o agrupamento foi feito), divisão da base de treino e teste, execução da filtragem colaborativa e cálculo das medidas de erro. Esses tempos são aproximados e podem variar a cada vez que o código é executado.

. **Tabela 8** – Tempos de execução da recomendação (em segundos).

Método	**Variável utilizada**	**Clusters**	**Tempo** (s)
	Sem clusterização		130
CLARA	Avaliação média	2	84
CLARA	Avaliação média	3	62
CLARA	Avaliação média	4	46
CLARA	Avaliação média	5	46
CLARA	Avaliação média	6	40
CLARA	Avaliação média	7	37
CLARA	Avaliação média	8	42
CLARA	Avaliação média	9	32
CLARA	Avaliação média	10	32
CLARA	Avaliação média	11	32
CLARA	Avaliação média	12	34
CLARA	Avaliação média	13	31
CLARA	Avaliação média	14	33
CLARA	Avaliação média	15	30

Fonte: FILGUEIRA, 2018

Conclusão

O trabalho buscou verificar se existia alguma diferença entre a acurácia da filtragem colaborativa, considerando a base completa de avaliações e a divisão dos usuários em clusters, para executar a filtragem dentro de cada grupo. A diferença entre as medidas de erro do valor de referência e da configuração que obteve o menor erro indica que a clusterização pode trazer uma melhora para a filtragem colaborativa. Por outro lado, a maior parte das execuções ao clusterizar a base teve um resultado pior, de acordo com as métricas de erro utilizadas. Esse resultado se apresenta a partir da matriz utilizada, que apresenta cerca de 4,68% de elementos preenchidos.

Considerando os resultados obtidos, trabalhos futuros poderiam estudar a relação entre critérios para escolha do número de clusters e da técnica utilizada no agrupamento com a acurácia da filtragem colaborativa. Os estudos buscariam verificar se o número de clusters

apontado como mais adequado por algumas das técnicas existentes resultaria em melhores resultados em termos de acurácia ou alguma outra medida disponível para os estudos. Também outras técnicas de clusterização poderiam ser empregadas na análise.

Ao verificar a recomendação de um usuário específico pôde ser constatado que os filmes recomendados não parecem ser completamente ao acaso, aleatórios, mas sim, são de alguma forma similares aos avaliados pelo usuário. Além disso os filmes recomendados foram classificados por gêneros em geral bem avaliados pela pessoa.

Com relação ao tempo de execução, o agrupamento dos usuários foi uma tarefa fácil, não havendo nem um momento de espera pela execução da função pelo software R. Já no momento de executar a recomendação e calcular o erro gerado, um maior tempo foi necessário, além de utilizar uma quantidade razoavelmente grande de memória do computador, considerando um dispositivo de 8Gb de memória. O tempo gasto no processamento das recomendações foi consideravelmente menor quando utilizou-se o agrupamento de usuários em relação ao tempo gasto sem particionamento da base. Com isso, uma boa escolha de clusters pode ser muito vantajosa, por economizar tempo de processamento e ter maior acurácia.

Referências

BHAT, A. **K-medoids clustering using partitioning around medoids for performing face recognition**. International Journal of Soft Computing, Mathematics and Control, Citeseer, v. 3, n. 3, p. 1–12, 2014.

BREESE, J. S.; HECKERMAN, D.; KADIE, C. **Empirical analysis of predictive algorithms for collaborative filtering**. In: MORGAN KAUFMANN PUBLISHERS INC. Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence. [S.l.], 1998. p. 43–52.

DAKHEL, G. M.; MAHDAVI, M. **A new collaborative filtering algorithm using kmeans clustering and neighbors' voting**. In: IEEE. Hybrid Intelligent Systems (HIS), 2011 11th International Conference on. [S.l.], 2011. p. 179–184.

DOWLE Matt; SRINIVASAN, Arun. **data.table**: Extension of `data.frame`. R package version 1.11.8. https://CRAN.R-project.org/package=data.table 2018.

GORAKALA, S. K.; USUELLI, M. **Building a recommendation system with R**. [S.l.]: Packt Publishing Ltd, 2015.

HAHSLER, Michael. **recommenderlab**: Lab for Developing and Testing Recommender Algorithms. R package version 0.2-3. https://CRAN.R-project.org/package=recommenderlab 2018.

HAHSLER, M. **recommenderlab**: A framework for developing and testing recommendation a/lgorithms. [S.l.], 2015.

HAN, J.; PEI, J.; KAMBER, M. **Data mining**: concepts and techniques. [S.l.]: Elsevier, 2011.

HARPER, F. M.; KONSTAN, J. A. **The movielens datasets**: History and context. Acm transactions on interactive intelligent systems (tiis), ACM, v. 5, n. 4, p. 19, 2016.

HU, Y.; KOREN, Y.; VOLINSKY, C. **Collaborative filtering for implicit feedback datasets**. In: IEEE. Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on. [S.l.], 2008. p. 263–272.

ISINKAYE, F.; FOLAJIMI, Y.; OJOKOH, B. **Recommendation systems**: Principles, methods and evaluation. Egyptian Informatics Journal, Elsevier, v. 16, n. 3, p. 261–273, 2015.

MAECHLER, M., ROUSSEEUW, P., STRUYF, A., HUBERT, M., HORNIK, K. **cluster**: Cluster Analysis Basics and Extensions. R package version 2.0.7-1 2018.

MELVILLE, P.; SINDHWANI, V. **Recommender systems**. In: Encyclopedia of machine learning. [S.l.]: Springer, 2011. p. 829–838.

MILD, A.; NATTER, M. **Collaborative filtering or regression models for internet recommendation systems?** Journal of Targeting, Measurement and Analysis for marketing, Springer, v. 10, n. 4, p. 304–313, 2002.

MÜLLER, Kirill; WICKHAM, Hadley. **tibble**: Simple Data Frames. R package version 1.4.2. https://CRAN.R-project.org/package=tibble 2018.

OARD, D. W.; KIM, J. et al. **Implicit feedback for recommender systems**. In: WOUONGONG. Proceedings of the AAAI workshop on recommender systems. [S.l.], 1998. v. 83.

O'CONNOR, M.; HERLOCKER, J. **Clustering items for collaborative filtering**. In: UC BERKELEY. Proceedings of the ACM SIGIR workshop on recommender systems. [S.l.], 1999. v. 128.

PARK, H.-S.; JUN, C.-H. **A simple and fast algorithm for k-medoids clustering**. Expert systems with applications, Elsevier, v. 36, n. 2, p. 3336–3341, 2009.

R Core Team. **R**: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/ 2018.

REATEGUI, E. B.; CAZELLA, S. C. **Sistemas de recomendação**. In: XXV Congresso da Sociedade Brasileira de Computação. [S.l.: s.n.], 2005. p. 306–348.

SHAPIRA, B. et al. **Recommender systems handbook**. [S.l.]: Springer New York, 2011.

SU, X.; KHOSHGOFTAAR, T. M. **A survey of collaborative filtering techniques**. Advances in artificial intelligence, Hindawi, v. 2009, 2009.

TAKAHASHI, M. M.; JR, R. H. **Estudo comparativo de algoritmos de recomendação**. USP. São Paulo, 2015.

VALE, M. N. do. **Agrupamentos de dados**: Avaliação de Métodos e Desenvolvimento de Aplicativo para Análise de Grupos. Tese (Doutorado) — PUC-Rio, 2005.

WICKHAM, Hadley; FRANÇOIS, Romain; HENRY, Lionel; MÜLLER, Kirill. **dplyr**: A Grammar of Data Manipulation. R package version 0.7.8. https://CRAN.R-project.org/package=dplyr 2018.

WICKHAM, Hadley; HENRY, Lionel. **tidyr**: Easily Tidy Data with 'spread()' and 'gather()' Functions. R package version 0.8.2. https://CRAN.R-project.org/package=tidyr 2018.

WICKHAM, Hadley. **The Split-Apply-Combine Strategy for Data Analysis**. Journal of Statistical Software, 40(1), 1-29. URL http://www.jstatsoft.org/v40/i01/ 2011.

Anexo

Todos os scripts executados para a obtenção desses resultados estão disponíveis no endereço https://github.com/leo-filgueira/trab_recom/.