Sistemas de recomendação utilizando o software R

Leonardo Filgueira

14 de dezembro de 2018

Introdução

- Técnicas de *machine learning* que filtram um grande conjunto de dados, tendo como base informações dos usuários, prevendo avaliações dos usuários aos itens para recomendar o(s) item(ns) que obtiveram uma nota prevista maior.
- Variedade de informações ofertadas e coletadas no ambiente digital, principalmente.
- Desafio de lidar com os dados coletados.
- Utilização das avaliações dos usuários para os itens.
- Técnicas de filtro dos dados para sugerir itens para usuários.
- Proposta na década de 90 e já utilizada pelo Yahoo no período.
- Sugestão itens que acredita-se ser do desejo do usuário.
- Facilitação da busca do usuário pelo item.
- Fidelização do usuário.

Objetivos

- Comparar a acurácia das recomendações utilizando filtragem colaborativa para todo o conjunto de dados com as recomendações utilizando filtragem colaborativa para cada cluster de usuários.
- Descrever informações de filmes e usuários.
- Analisar filmes avaliados e recomendados para determinado usuário.
- Comparar o tempo de execução da recomendação para as configurações escolhidas.

Estado da arte

As avaliações são disponíveis da seguinte maneira:

	Item 1	Item 2	$\cdots$	Item m
Usuário 1	$r_{(1,1)}$		$\cdots$	
Usuário 2		$r_{(2,2)}$	$\cdots$	$r_{(2,m)}$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
Usuário n			$\cdots$	$r_{(n,m)}$

Onde $r_{u,i}$ é a avaliação do usuário u para o item i.

- Matriz esparsa.
- Vários dados faltantes.

Estado da arte

- Formatos de avaliação:
  - Avaliações numéricas;
  - Avaliações qualitativas;
  - Avaliações binárias;
  - Avaliação unária.
- Utilizar avaliações observadas para prever as faltantes.

Estado da arte

Tipos de sistemas de recomendação:

- Filtragem baseada em conteúdo:
  - Uso do histórico do usuário;
  - Associação entre itens;
  - Buscar itens mais associados aos do histórico;
  - Técnicas de *text mining* para associação;
  - Não considera comportamento de outros usuários;

Estado da arte

Tipos de sistemas de recomendação:

- Filtragem colaborativa
  - Uso das informações de outros usuários.
  - Agrupamento de itens/usuários.
- Tipos:
  - Baseada no item,
  - Baseada no usuário.

Estado da arte

- Filtragem colaborativa baseada no item:
  - Cálculo da similaridade entre itens i e j:
  - Coeficiente de correlação de Pearson:

$$w_{i,j} = \frac{\sum_{u \in U} (r_{u,i} - \overline{r}_i)(r_{u,j} - \overline{r}_j)}{\sqrt{\sum_{u \in U} (r_{u,i} - \overline{r}_i)^2 \sum_{u \in U} (r_{u,j} - \overline{r}_j)^2}} \quad (1)$$

- U: conjunto de usuários que avaliaram os dois itens.
- $r_{u,i}$ o rating dado pelo usuário u ao item i.
- $\overline{r}_i$ o rating médio recebido pelo item i dado por todos os usuários que o avaliaram.

Estado da arte

- Filtragem colaborativa baseada no item:
- Cosseno entre vetores de avaliações:

$$w_{i,j} = \cos(\vec{r}_i, \vec{r}_j) = \frac{\vec{r}_i \cdot \vec{r}_j}{\|\vec{r}_i\| \times \|\vec{r}_i\|} = \frac{\sum_{u=1}^{n} r_{u,i} r_{u,j}}{\sqrt{\sum_{u=1}^{n} r_{u,i}^2 \sum_{u=1}^{n} r_{u,j}^2}} \quad (2)$$

- *rating* previsto:

$$p_{a,i} = \frac{\sum_{j \in k} r_{a,i} - w_{i,j}}{\sum_{j \in k} |w_{i,j}|} \quad (3)$$

- Onde k é a vizinhança do item i.

Estado da arte

- Filtragem colaborativa baseada no usuário:
- Cálculo da similaridade entre usuários a e u.
- Coeficiente de correlação de Pearson:

$$w_{a,u} = \frac{\sum_{i \in I} (r_{a,i} - \overline{r}_a)(r_{u,i} - \overline{r}_u)}{\sqrt{\sum_{i \in I} (r_{a,i} - \overline{r}_a)^2 \sum_{i \in I} (r_{u,i} - \overline{r}_u)^2}} \tag{4}$$

- Cosseno entre os vetores de avaliações:

$$w_{a,u} = \cos(\vec{r}_a, \vec{r}_u) = \frac{\vec{r}_a \cdot \vec{r}_u}{\|\vec{r}_a\| \times \|\vec{r}_u\|} = \frac{\sum_{i=1}^{m} r_{a,i} r_{u,i}}{\sqrt{\sum_{i=1}^{m} r_{a,i}^2 \sum_{i=1}^{m} r_{u,i}^2}} \tag{5}$$

Estado da arte

- Filtragem colaborativa baseada no usuário:
- Cálculo do valor previsto da avaliação:

$$p_{a,i} = \overline{r}_a + \frac{\sum_{u \in k} (r_{u,i} - \overline{r}_u) w_{a,u}}{\sum_{u \in k} |w_{a,u}|} \tag{6}$$

- Onde k é a vizinhança do usuário a.

Clusterização

Dois algoritmos utilizados no agrupamento de usuários:

- k-means.
- CLARA.

Para a clusterização, utilizou-se o rating médio por gênero e proporção de filmes assistidos por gênero.

Clusterização

k-means

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.

Clusterização

CLARA (Clustering Large Applications)

- Execução do algoritmo PAM (Partitioning Around Medoids) utilizando amostragem.
- Aplicável para um grande conjunto de dados.

Clusterização

PAM (Partitioning Around Medoids)

1. Definir aleatoriamente k medoides.
2. Associar cada um dos elementos restantes a um cluster, sendo pertencente ao grupo 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.

Clusterização

CLARA

Selecionada uma amostra da base e aplica-se o algoritmo PAM.

Função de custo:

$$C(m, D) = \frac{\sum_{i=1}^{n} d(x_i, cl(m, x_i))}{n} \tag{7}$$

Onde:

- m são os medoides encontrados.
- $cl(m, x_i)$ é o medoide mais próximo de um ponto x_i.
- $d(x_i, cl(m, x_i))$ é uma medida de similaridade entre x_i e seu medoide mais próximo.
- n é o número de observações na base de dados D.

O agrupamento da iteração com menor função de custo é retornado.

Análise dos resultados

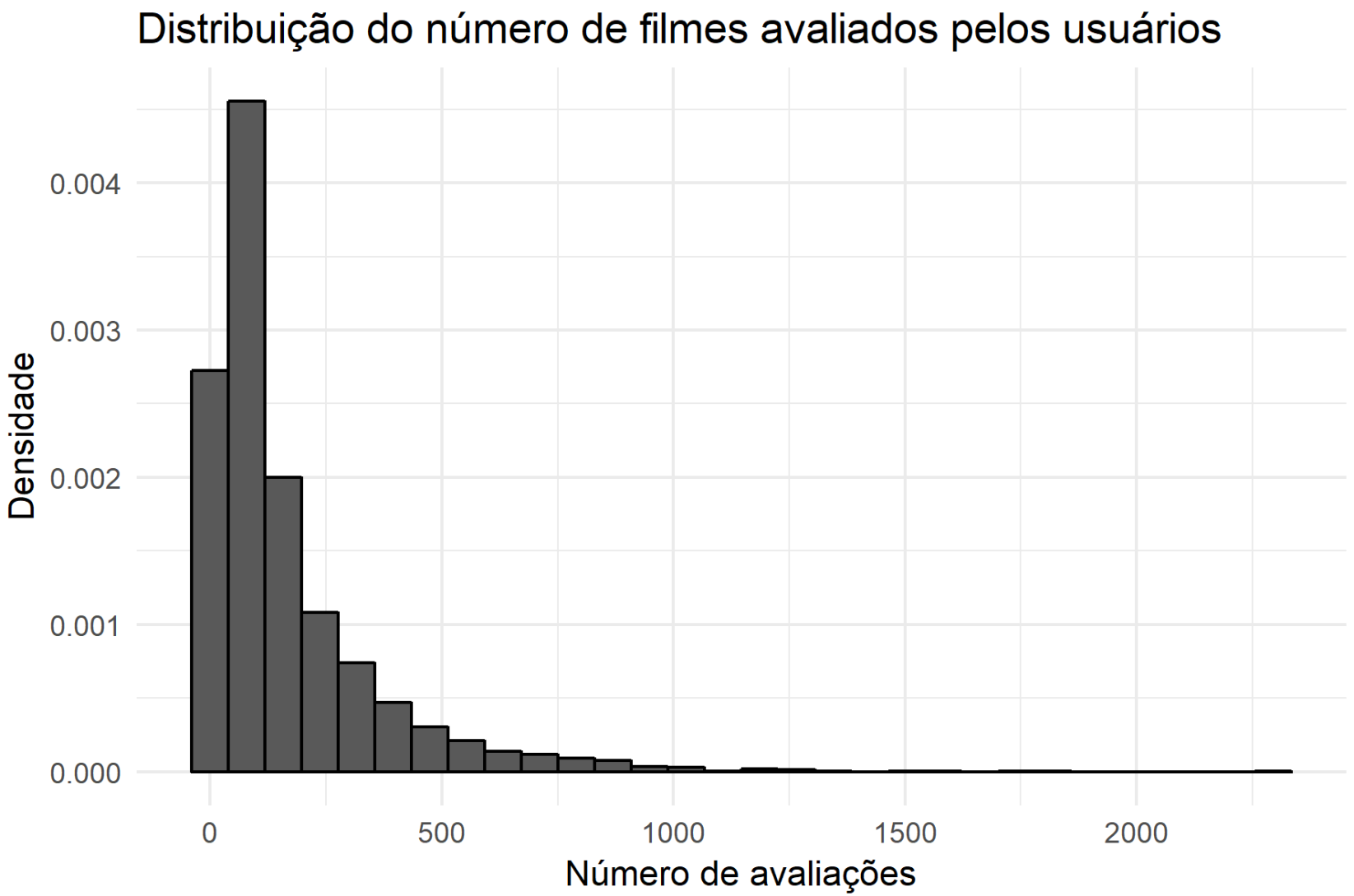

Figure 1: Número de filmes avaliados

Análise dos resultados

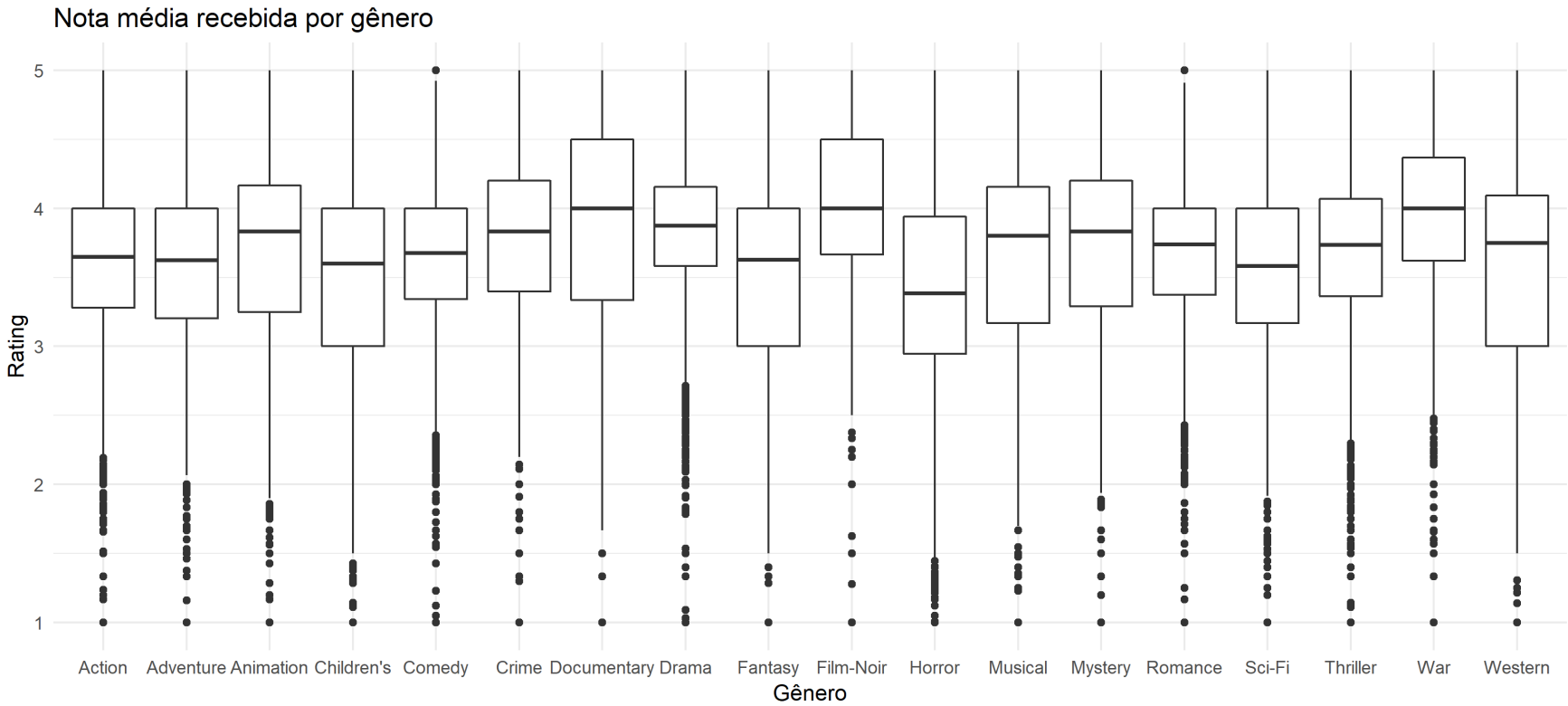

Figure 2: Distribuição da avaliação médio por gênero

Análise dos resultados

Table 1: 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

Análise dos resultados

Table 2: 10 maiores notas previstas 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

Análise dos resultados

Table 3: 10 menores erros de recomendação

Algoritmo	**Informação**	**Clusters**	**Raiz do EQM**	**EQM**	**EMA**
CLARA	Rating	3	1.0055	1.0199	0.7983
k-means	Proporção	12	1.0213	1.0444	0.8093
k-means	Proporção	8	1.024	1.0495	0.8162
k-means	Rating	12	1.0251	1.052	0.817
CLARA	Rating	4	1.0254	1.0634	0.8188
k-means	Proporção	15	1.0266	1.055	0.8182
k-means	Rating	11	1.0278	1.0568	0.8156
CLARA	Proporção	5	1.0288	1.0599	0.816
k-means	Rating	15	1.03	1.0624	0.8219
CLARA	Proporção	7	1.0315	1.0668	0.825

- Sem cluserização:
  $\sqrt{EQM} = 1.0341, EQM = 1.0693, EMA = 0.8221$

Análise dos resultados

Table 4: 10 maiores erros de recomendação

Algoritmo	**Informação**	**Clusters**	**Raiz do EQM**	**EQM**	**EMA**
CLARA	Rating	13	1.0948	1.2513	0.8915
CLARA	Rating	10	1.1087	1.2502	0.9061
CLARA	Rating	9	1.1091	1.2444	0.8942
CLARA	Rating	14	1.1165	1.3091	0.9202
CLARA	Rating	11	1.1167	1.2909	0.9197
CLARA	Rating	5	1.1194	1.2743	0.9096
CLARA	Rating	7	1.1282	1.299	0.928
CLARA	Rating	8	1.1377	1.3082	0.9302
CLARA	Rating	12	1.1867	1.5236	0.9671
CLARA	Rating	15	1.1926	1.4865	0.9851

Análise dos resultados

Table 5: Tempos de execução da recomendação (em segundos)

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

Conclusões

- Pequeno ganho na acurácia em algumas configurações, mas erros maiores gerados na maior parte das clusterizações.
- Recomendações como a apresentada fazem sentido em relação aos itens previamente avaliados.
- Decréscimo no tempo de execução da recomendação conforme o número de clusters aumenta.

Referências

- ISINKAYE, F.; FOLAJIMI, Y.; OJOKOH, B. Recommendation systems: Principles, methods and evaluation. Egyptian Informatics Journal, Elsevier, v. 16, n. 3, p. 261–273, 2015.
- 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.
- TAKAHASHI, M. M.; JR, R. H. Estudo comparativo de algoritmos de recomendação. USP. São Paulo, 2015.
- SHAPIRA, B. et al. Recommender systems handbook. [S.l.]: Springer New York, 2011.
- MELVILLE, P.; SINDHWANI, V. Recommender systems. In: Encyclopedia of machine learning. [S.l.]: Springer, 2011. p. 829–838.

Referências

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

Referências

- HAHSLER, M. recommenderlab: A framework for developing and testing recommendation algorithms. [S.I.], 2015.
- SARWAR, B. et al. Item-based collaborative filtering recommendation algorithms. In: ACM. Proceedings of the 10th international conference on World Wide Web. [S.I.], 2001. p. 285–295.
- VALE, M. N. do. Agrupamentos de dados: Avaliação de Métodos e Desenvolvimento de Aplicativo para Análise de Grupos. Tese (Doutorado) — PUC-Rio, 2005. [14] 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.
- 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.

Referências

- MINING, W. I. D. Data mining: Concepts and techniques. Morgan Kaufinann, 2006.
- R Core Team. R: A Language and Environment for Statistical Computing. Vienna, Austria, 2018. Disponível em: https://www.R-project.org/.
- RStudio Team. RStudio: Integrated Development Environment for R. Boston, MA, 2016. Disponível em: http://www.rstudio.com/.
- HAHSLER, M. recommenderlab: Lab for Developing and Testing Recommender Algorithms. [S.l.], 2017. R package version 0.2-2. Disponível em: https://CRAN.Rproject.org/package=recommenderlab.

Obrigado!