

# Reprodução e Extensão do Sistema de Recomendação Baseado em K-Means e KNN com Modelos Supervisionados

Diego Costa Arruda<sup>1</sup>, Flávio Romero Santos de Sá Muniz<sup>1</sup>,  
Henrique Roma Maracajá<sup>1</sup>, João Henrique Moura Lafetá<sup>1</sup>,  
Rodrigo Dubeux Mendes de Oliveira<sup>1</sup>, Yara Rodrigues Inácio<sup>1</sup>

<sup>1</sup>CESAR School – Avenida Cais do Apolo, 77, Recife – PE – 50030-220  
{dca2, frssm, hrm, jhml, rdmo, yri}@cesar.school

**Abstract.** *Recommender systems play a crucial role in filtering vast amounts of information by predicting user preferences. This paper presents a faithful reproduction of the model proposed in “Movie Recommender System Using K-Means Clustering and K-Nearest Neighbor”, which combines content-based clustering with collaborative filtering. We replicated the methodology using the MovieLens 100k dataset and validated the performance of the original system. In addition, we extended the original approach by exploring the use of supervised regression models—namely, Truncated SVD with linear regression, Multi-Layer Perceptron (MLP), and XGBoost—to estimate user ratings. Our results show that all supervised regression models tested slightly outperformed the similarity-based KNN predictions used in the original method. This work highlights how reproducible experimentation not only validates existing methods but also opens pathways for incorporating advanced learning algorithms to refine predictive accuracy.*

**Resumo.** *Sistemas de recomendação desempenham um papel crucial na filtragem de grandes volumes de informação ao prever as preferências dos usuários. Este trabalho apresenta uma reprodução fiel do modelo proposto no artigo “Movie Recommender System Using K-Means Clustering and K-Nearest Neighbor”, que combina clusterização baseada em conteúdo com filtragem colaborativa. Replicamos a metodologia utilizando o conjunto de dados MovieLens 100k e validamos o desempenho do sistema original. Além disso, estendemos a abordagem proposta explorando o uso de modelos supervisionados de regressão — especificamente, SVD truncado com regressão linear, Perceptron Multi-Camadas (MLP) e XGBoost — para estimar as avaliações dos usuários. Nossos resultados mostram que todos os modelos supervisionados testados superaram ligeiramente as previsões baseadas em similaridade com KNN utilizadas no método original. Este trabalho destaca como a experimentação reprodutível não apenas valida métodos existentes, mas também abre caminhos para a incorporação de algoritmos de aprendizado mais avançados com o objetivo de refinar a acurácia preditiva.*

## 1. Introdução

Sistemas de recomendação são uma subcategoria dos sistemas de filtragem de informação, projetados para prever a classificação ou preferência que um usuário atribuiria a um determinado item [Ricci et al. 2015]. Esses sistemas coletam dados sobre os interesses dos usuários de forma implícita, a partir de seu comportamento durante a navegação ou uso

(como assistir a um filme até o fim), ou de forma explícita, com base em avaliações, curtidas ou histórico de interações anteriores. Tais sistemas têm ampla aplicação em domínios como filmes, comércio eletrônico, turismo, televisão, música e redes sociais [Ricci et al. 2015].

Dentre as técnicas mais utilizadas em sistemas de recomendação está a filtragem colaborativa, que se baseia na suposição de que usuários com preferências semelhantes no passado tendem a manter essas preferências no futuro [Su and Khoshgoftaar 2009]. Assim, o sistema identifica perfis de usuários semelhantes e recomenda itens com base nas avaliações desses perfis. Em sua forma mais comum, a filtragem colaborativa utiliza uma matriz de utilidade que relaciona usuários e itens, com os valores correspondendo às notas atribuídas.

Outra técnica amplamente empregada é a clusterização, que agrupa itens ou usuários com base em características semelhantes. O algoritmo K-Means, por exemplo, é utilizado para particionar um conjunto de objetos de forma que os objetos dentro de um mesmo grupo sejam mais similares entre si do que em relação a objetos de outros grupos [Sarwar et al. 2002]. A combinação dessas abordagens permite construir sistemas híbridos que integram análise de conteúdo, comportamento de usuários e agrupamentos inteligentes [Ricci et al. 2015].

O artigo “Movie Recommender System Using K-Means Clustering and K-Nearest Neighbor” propõe exatamente uma solução híbrida baseada nessas técnicas [Ahuja et al. 2019]. O sistema utiliza o algoritmo K-Means para agrupar filmes com base em seus gêneros, criando uma matriz de utilidade clusterizada por usuário. Em seguida, a similaridade entre usuários é calculada utilizando o coeficiente de correlação de Pearson, e a predição das avaliações é realizada por meio do algoritmo K-Nearest Neighbor (KNN), que considera os usuários mais semelhantes ao usuário-alvo.

Este trabalho tem como objetivo principal a reprodução fiel do sistema proposto no artigo, utilizando o conjunto de dados MovieLens 100k, amplamente utilizado na literatura acadêmica [Harper and Konstan 2015]. A partir da reprodução, buscamos validar os resultados apresentados originalmente e entender com profundidade a estrutura e funcionamento do sistema.

Além disso, propomos uma extensão experimental da abordagem original na qual o mecanismo de predição é substituído por modelos supervisionados de regressão, que são mais modernos e frequentemente aplicados em sistemas de recomendação. Foram testados três modelos distintos. O primeiro é o SVD truncado combinado com regressão linear, uma técnica de fatoração de matrizes que apresenta bom desempenho em ambientes de dados esparsos. O segundo modelo é o Perceptron Multi-Camadas (MLP), capaz de capturar padrões não lineares nas interações entre usuários e itens. Por fim, utilizamos o XGBoost, um modelo baseado em árvores de decisão com reforço de gradiente, reconhecido por sua alta precisão em tarefas de regressão.

O presente artigo está organizado em sete seções. A Seção 1 introduz o tema dos sistemas de recomendação, destacando sua relevância atual e apresentando os objetivos deste estudo. A Seção 2 apresenta os trabalhos relacionados, discutindo as principais abordagens e técnicas utilizadas recentemente na literatura. Na Seção 3, descreve-se a metodologia adotada, contemplando tanto a reprodução fiel do sistema original baseado

em K-Means e KNN, quanto a proposta de extensão com modelos de regressão supervisionada. A Seção 4 detalha o conjunto de dados MovieLens 100k, abordando suas características estatísticas, estrutura e o processo de pré-processamento aplicado. A Seção 5 expõe os experimentos realizados, organizados em duas subseções: a primeira dedicada à reprodução do trabalho original, e a segunda à avaliação dos modelos supervisionados SVD truncado com regressão linear, Perceptron Multi-Camadas (MLP) e XGBoost. Em seguida, a Seção 6 apresenta e discute os resultados obtidos, tanto em termos quantitativos (RMSE) quanto qualitativos (análise das recomendações). Por fim, a Seção 7 sintetiza as principais conclusões do estudo e propõe direções promissoras para pesquisas futuras, como o uso de agrupamentos semânticos baseados em análise de sentimentos e dados textuais.

## 2. Trabalhos Relacionados

Sistemas de recomendação vêm sendo amplamente estudados nas últimas décadas, consolidando-se como um dos campos mais relevantes em aplicações de aprendizado de máquina voltadas à experiência do usuário [Ricci et al. 2015]. Dentre as principais abordagens, destacam-se a filtragem colaborativa, a filtragem baseada em conteúdo e os métodos híbridos, que combinam características de ambas com o objetivo de melhorar a precisão e a cobertura das recomendações.

O trabalho reproduzido neste estudo, intitulado “Movie Recommender System Using K-Means Clustering and K-Nearest Neighbor”, faz referência a diversas pesquisas anteriores como parte de sua fundamentação teórica [Ahuja et al. 2019]. Os autores destacam a importância da filtragem colaborativa como técnica central em sistemas de recomendação [Su and Khoshgoftaar 2009], enfatizando sua capacidade de prever preferências a partir da similaridade entre usuários. Além disso, mencionam a clusterização como estratégia complementar, sendo o algoritmo K-Means amplamente utilizado para agrupar itens com base em atributos compartilhados [Sarwar et al. 2002]. Na abordagem proposta, a clusterização é aplicada sobre os gêneros dos filmes, permitindo a criação de uma matriz de utilidade reduzida por cluster, que serve como base para o cálculo da similaridade entre usuários via correlação de Pearson e, posteriormente, para a aplicação do algoritmo K-Nearest Neighbor (KNN) na predição das notas.

A literatura sobre sistemas híbridos de recomendação, como o que foi proposto nesse estudo, é extensa e diversa [Ricci et al. 2015]. Pesquisas recentes têm explorado diferentes formas de combinar fatores latentes com variáveis explícitas, integrar aprendizado supervisionado com heurísticas de similaridade, e aplicar técnicas de redução de dimensionalidade para lidar com a esparsidade das matrizes de interação. Métodos como Singular Value Decomposition (SVD), Redes Neurais Artificiais (ANNs) e modelos baseados em árvores, como o XGBoost [Chen and Guestrin 2016], têm se destacado por sua capacidade de capturar padrões complexos nas preferências dos usuários.

Ao propor a combinação entre K-Means e KNN, os autores do estudo original apresentam uma solução simples e eficiente, que se beneficia da segmentação dos itens para reduzir a complexidade da matriz de utilidade e melhorar o cálculo de similaridade entre os usuários [Sarwar et al. 2002]. Essa proposta se insere em um contexto de pesquisas que buscam equilibrar desempenho computacional e acurácia, sem recorrer a modelos de aprendizado supervisionado mais sofisticados.

### 3. Metodologia

Este trabalho adota uma abordagem experimental em duas etapas principais: a primeira consiste na reprodução fiel da metodologia proposta no estudo original, utilizando a mesma base de dados e técnicas descritas; a segunda corresponde à extensão da arquitetura do sistema, em que substituímos o mecanismo de predição baseado em similaridade por modelos supervisionados de regressão. Ambas as etapas foram desenvolvidas com o objetivo de garantir rigor metodológico, comparabilidade e robustez dos resultados.

#### 3.1. Reprodução da metodologia original

A reprodução foi conduzida utilizando o conjunto de dados MovieLens 100k [Harper and Konstan 2015], amplamente utilizado na literatura para avaliação de sistemas de recomendação. Os dados foram carregados e estruturados em três conjuntos principais: (i) o dataframe ratings, contendo as avaliações dos usuários para os filmes; (ii) o dataframe movies, com as informações sobre os filmes e seus respectivos gêneros; e (iii) o dataframe users, contendo dados demográficos dos usuários. Em conformidade com a metodologia original, os dados foram divididos em conjuntos de treino (80%) e teste (20%) por meio da técnica de hold-out, garantindo a avaliação do sistema em dados não vistos.

Com os dados preparados, foi construída a **matriz de utilidade**, que representa as avaliações de cada usuário para os filmes disponíveis no conjunto de treino. Essa matriz foi posteriormente utilizada como base para a criação de uma **matriz de utilidade clusterizada**, onde os filmes foram agrupados de acordo com seus gêneros utilizando o algoritmo **K-Means**. O número ideal de clusters foi definido com base no método do cotovelo, por meio da análise do **Within-Cluster Sum of Squares (WCSS)**. Os filmes foram então distribuídos nos clusters obtidos, e para cada usuário foi calculada a média das notas atribuídas aos filmes de cada cluster, resultando na matriz clusterizada.

A seguir, foi realizada uma **normalização por usuário** da matriz clusterizada, a fim de ajustar variações individuais de escala e centrimento. Essa matriz normalizada serviu como base para o cálculo da **similaridade entre usuários** por meio do coeficiente de correlação de Pearson. O resultado é uma matriz de similaridade simétrica que reflete o grau de correlação entre os perfis de avaliação dos usuários nos diferentes clusters de filmes.

Com a matriz de similaridade calculada, foi implementada a função de predição baseada no algoritmo **K-Nearest Neighbor (KNN)**. A predição da nota para um par usuário–filme foi feita por meio da média ponderada das avaliações fornecidas ao mesmo filme pelos usuários mais similares, conforme os valores da matriz de similaridade. Essa abordagem foi avaliada sobre o conjunto de teste utilizando o **Root Mean Squared Error (RMSE)** como métrica de desempenho. O valor obtido na reprodução direta foi de  $RMSE = 1.180$ , valor próximo ao relatado no artigo original ( $RMSE \approx 1.081$ ), confirmando a robustez da metodologia proposta.

#### 3.2. Extensão com modelos supervisionados

Na segunda etapa, foram propostas extensões supervisionadas ao modelo, substituindo a função de predição baseada em KNN por três modelos de regressão treinados com aprendizado supervisionado. O objetivo foi avaliar se técnicas modernas de machine learning poderiam superar a acurácia da abordagem original.

Para isso, foram construídas vetores de atributos para cada entrada (usuário, filme) no conjunto de treino e teste. Esses vetores foram compostos pelas médias de avaliação dos clusters (extraídos da matriz de utilidade clusterizada) e pelos valores de similaridade do usuário-alvo com os demais usuários (obtidos da matriz de similaridade). Essa combinação de informações foi utilizada como entrada para os modelos supervisionados.

Foram testados os seguintes algoritmos:

- **Truncated Singular Value Decomposition (SVD)** com regressão linear, que aplicou redução de dimensionalidade nos dados antes do ajuste do modelo.
- **Perceptron Multi-Camadas (MLP)**, uma rede neural com duas camadas ocultas (64 e 32 neurônios, respectivamente), treinada para mapear os vetores de entrada para as notas de avaliação.
- **XGBoost Regressor**, uma implementação otimizada de gradient boosting baseada em árvores de decisão, amplamente reconhecida por sua eficiência e alto desempenho em problemas tabulares.

Todos os modelos foram treinados com o conjunto de treino e avaliados com o conjunto de teste, também utilizando RMSE como métrica. Os resultados mostraram que os modelos supervisionados superaram a abordagem baseada em KNN, com os melhores desempenhos registrados por XGBoost (RMSE = 1.042), SVD com regressão (RMSE = 1.045) e MLP (RMSE = 1.046). Essas melhorias, embora modestas, indicam que técnicas supervisionadas oferecem ganhos reais de desempenho sem sacrificar interpretabilidade ou viabilidade computacional.

#### 4. Conjunto de Dados

Os experimentos realizados neste trabalho utilizaram o conjunto de dados MovieLens 100k, amplamente adotado como referência na literatura científica para avaliação de sistemas de recomendação [Harper and Konstan 2015]. Essa base foi disponibilizada pelo GroupLens Research Project da Universidade de Minnesota, sendo considerada uma referência por sua qualidade, estrutura padronizada e ampla representatividade.

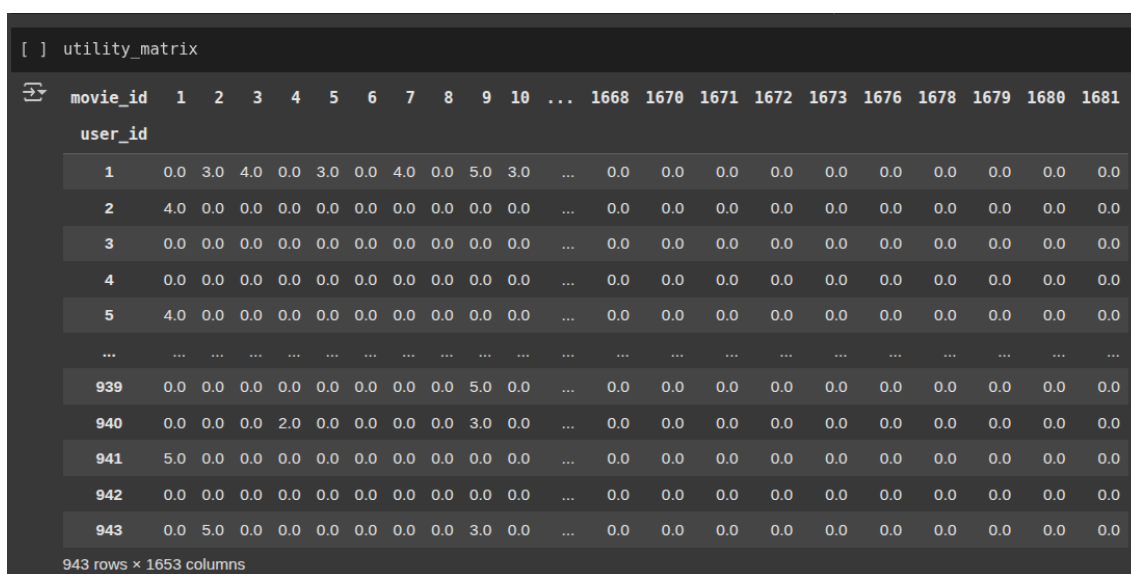
O MovieLens 100k contém exatamente **100.000 avaliações explícitas** atribuídas por **943 usuários** a **1.682 filmes**. As **notas variam de 1 a 5**, em valores inteiros, representando o grau de preferência do usuário pelo filme. Todas as avaliações foram registradas de forma explícita, ou seja, os próprios usuários forneceram as notas, o que confere confiabilidade às interações observadas.

O conjunto de dados está organizado em três arquivos principais:

- **u.data:** contém os registros das avaliações no formato **usuário, filme, nota, timestamp**, sendo o núcleo da matriz de interações. Este arquivo é carregado como o dataframe ratings.
- **u.item:** fornece os metadados dos filmes, incluindo o título, data de lançamento, URL do IMDb e **19 gêneros categóricos** representados como variáveis binárias. Esse arquivo constitui o dataframe movies.

- **u.user:** contém dados demográficos dos usuários, como **idade, gênero, ocupação e CEP**, organizados no dataframe users.

Em termos estruturais, a **matriz de utilidade** resultante, representada na Figura 1, possui alta esparsidade, característica comum em sistemas de recomendação [Ricci et al. 2015]. Com 943 usuários e 1.682 filmes, o número total possível de avaliações seria 1.586.126, o que significa que apenas cerca de 6,3% das possíveis interações estão presentes no conjunto. Essa esparsidade impõe desafios ao aprendizado, reforçando a necessidade de técnicas como fatoração de matrizes, normalização e agrupamento para tornar a tarefa de predição viável.



```
[ ] utility_matrix
```

movie_id	1	2	3	4	5	6	7	8	9	10	...	1668	1670	1671	1672	1673	1676	1678	1679	1680	1681
user_id																					
1	0.0	3.0	4.0	0.0	3.0	0.0	4.0	0.0	5.0	3.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2	4.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
5	4.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
939	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	5.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
940	0.0	0.0	0.0	2.0	0.0	0.0	0.0	0.0	3.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
941	5.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
942	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
943	0.0	5.0	0.0	0.0	0.0	0.0	0.0	0.0	3.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

943 rows x 1653 columns

**Figure 1. Matriz Utilidade**

Além disso, os filmes estão rotulados com múltiplos gêneros simultaneamente, representados em formato one-hot. Isso permite a aplicação de métodos baseados em conteúdo, como o K-Means, que se beneficia dessas representações vetoriais para clusterizar os filmes de acordo com suas características semânticas.

A diversidade de gêneros, o equilíbrio nas distribuições de usuários e filmes, e a presença de informações auxiliares tornam o MovieLens 100k um conjunto adequado tanto para abordagens baseadas em filtragem colaborativa quanto para modelos híbridos, como os explorados neste trabalho [Harper and Konstan 2015].

## 5. Experimentos

Os experimentos conduzidos neste trabalho visam avaliar tanto a fidelidade da reprodução do sistema de recomendação proposto por [Ahuja et al. 2019], quanto o impacto da substituição da etapa de predição baseada em vizinhos por modelos supervisionados de regressão. Todas as etapas foram implementadas em Python utilizando as bibliotecas `pandas`, `scikit-learn`, `xgboost` e `matplotlib`, em ambiente Google Colab.

O conjunto de dados MovieLens 100k foi dividido em 80% para treino e 20% para teste, totalizando 80.000 interações para ajuste dos modelos e 20.000 para avaliação. A Figura 2 apresenta a estrutura da matriz de treino gerada.

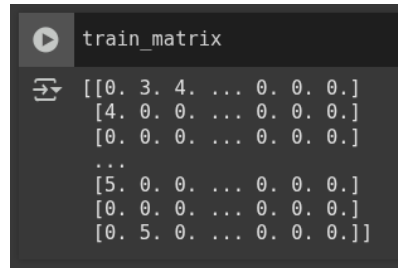


Figure 2. Train Matrix

### 5.1. Reprodução do trabalho original

A reprodução fiel do sistema original seguiu três etapas centrais: (i) clusterização de filmes com base em seus gêneros, (ii) cálculo da matriz de utilidade por cluster para cada usuário e (iii) predição por meio do algoritmo K-Nearest Neighbor (KNN), utilizando a similaridade entre usuários.

#### 5.1.1. Matriz de utilidade e clusterização

A partir dos dados de avaliação, foi construída uma matriz de utilidade no formato usuário-filme, com dimensão  $943 \times 1653$ , ilustrada na Figura 1.

Utilizando os 19 gêneros disponíveis, os filmes foram representados como vetores binários e agrupados com o algoritmo *K-Means*. O número ideal de clusters foi determinado com o método do cotovelo, conforme a Figura 3, indicando  $K = 2$ .

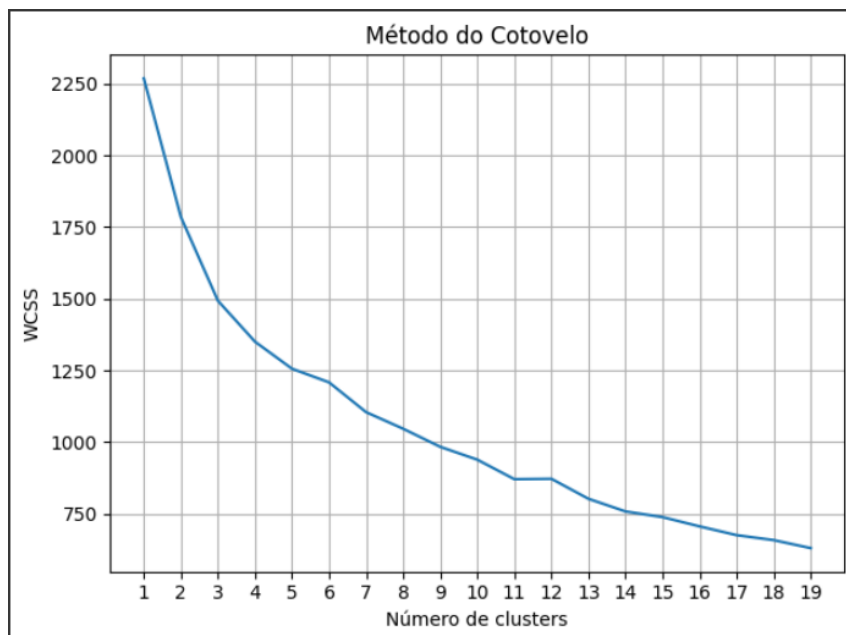
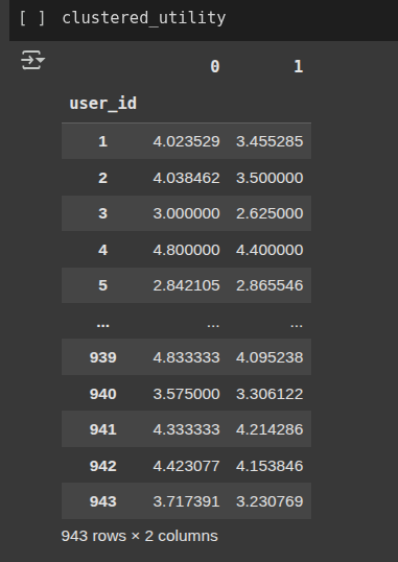


Figure 3. Método do Cotovelo

### 5.1.2. Matriz clusterizada, normalização e similaridade

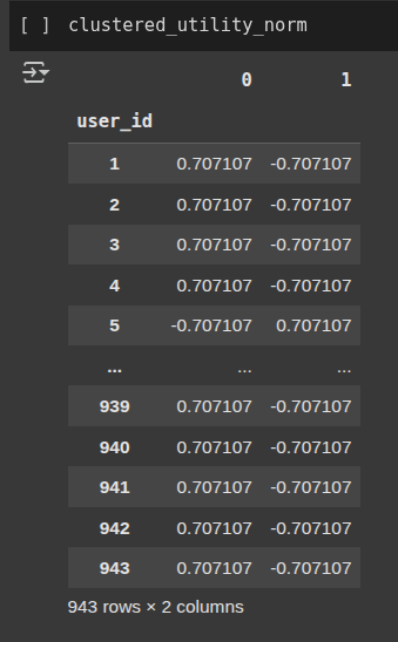
A média das notas atribuídas por cada usuário aos filmes de cada cluster originou a matriz de utilidade clusterizada, apresentada na Figura 4. Em seguida, aplicou-se normalização por usuário (Figura 5), com centramento na média e divisão pelo desvio padrão.



	0	1
user_id		
1	4.023529	3.455285
2	4.038462	3.500000
3	3.000000	2.625000
4	4.800000	4.400000
5	2.842105	2.865546
...	...	...
939	4.833333	4.095238
940	3.575000	3.306122
941	4.333333	4.214286
942	4.423077	4.153846
943	3.717391	3.230769

943 rows x 2 columns

Figure 4. Matriz de Utilidade Clusterizada



	0	1
user_id		
1	0.707107	-0.707107
2	0.707107	-0.707107
3	0.707107	-0.707107
4	0.707107	-0.707107
5	-0.707107	0.707107
...	...	...
939	0.707107	-0.707107
940	0.707107	-0.707107
941	0.707107	-0.707107
942	0.707107	-0.707107
943	0.707107	-0.707107

943 rows x 2 columns

Figure 5. Matriz de Utilidade Clusterizada e Normalizada

A similaridade entre usuários foi calculada com o coeficiente de correlação de Pearson, como mostra a Figura 6.



```
[ ] user_similarity
```

user_id	1	2	3	4	5	6	7	8	9	10	...	934	935	936	937	938	939	940	941	942	943
user_id																					
1	1.0	1.0	1.0	1.0	-1.0	1.0	1.0	1.0	-1.0	1.0	...	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
2	1.0	1.0	1.0	1.0	-1.0	1.0	1.0	1.0	-1.0	1.0	...	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
3	1.0	1.0	1.0	1.0	-1.0	1.0	1.0	1.0	-1.0	1.0	...	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
4	1.0	1.0	1.0	1.0	-1.0	1.0	1.0	1.0	-1.0	1.0	...	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
5	-1.0	-1.0	-1.0	-1.0	1.0	-1.0	-1.0	-1.0	1.0	-1.0	...	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
939	1.0	1.0	1.0	1.0	-1.0	1.0	1.0	1.0	-1.0	1.0	...	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
940	1.0	1.0	1.0	1.0	-1.0	1.0	1.0	1.0	-1.0	1.0	...	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
941	1.0	1.0	1.0	1.0	-1.0	1.0	1.0	1.0	-1.0	1.0	...	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
942	1.0	1.0	1.0	1.0	-1.0	1.0	1.0	1.0	-1.0	1.0	...	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
943	1.0	1.0	1.0	1.0	-1.0	1.0	1.0	1.0	-1.0	1.0	...	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0

943 rows x 943 columns

**Figure 6. Similaridade entre Usuários**

### 5.1.3. Predição e recomendações

A nota prevista para um par usuário-filme foi determinada com base nos 5 vizinhos mais similares (topN = 5), utilizando média ponderada das avaliações. A Tabela 1 apresenta os cinco filmes recomendados ao usuário 10.

**Table 1. Top 5 recomendações para o usuário 10 (reprodução)**

Filme	Nota prevista
Usual Suspects, The (1995)	5.00
Mr. Holland's Opus (1995)	5.00
Antonia's Line (1995)	5.00
Crimson Tide (1995)	5.00
Clerks (1994)	5.00

A avaliação com RMSE no conjunto de teste resultou em valor de 1.180, próximo ao RMSE relatado pelos autores originais (1.081), validando a reprodução.

## 5.2. Extensão com modelos supervisionados

Nesta etapa, substituímos a etapa de predição por modelos de regressão supervisionada, com o objetivo de verificar se métodos de aprendizado poderiam melhorar a acurácia preditiva.

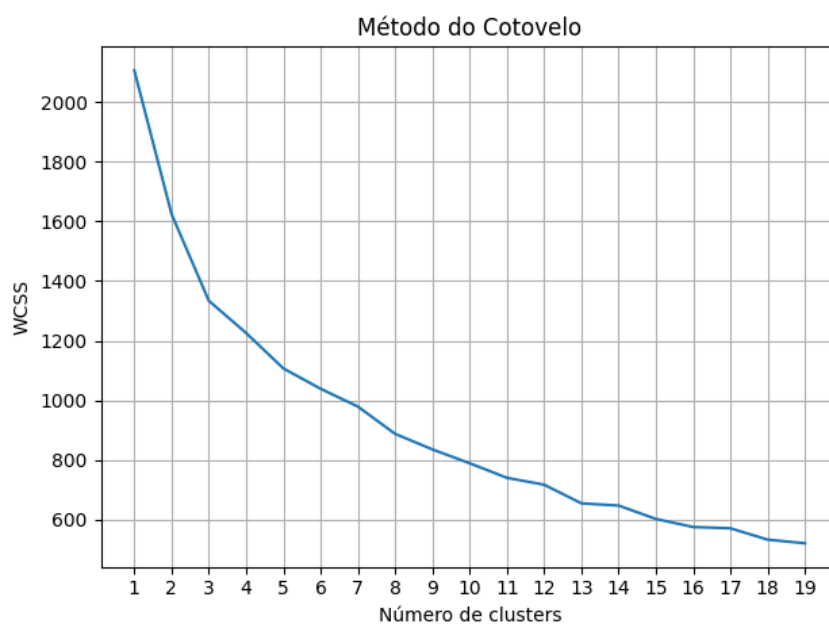
### 5.2.1. Filtragem semântica e clusterização refinada

Antes da clusterização, gêneros com baixa ocorrência foram filtrados, mantendo-se apenas os 13 mais frequentes (Figura 7). O número de clusters foi definido como  $K = 3$  com base no método do cotovelo (Figura 8).

```
print(f"Selected genres ({len(selected_genres)}): {list(selected_genres)}")
```

Selected genres (13): ['Action', 'Adventure', 'Children's', 'Comedy', 'Crime', 'Drama', 'Horror', 'Musical', 'Mystery', 'Romance', 'Sci-Fi', 'Thriller', 'War']

**Figure 7. Lista dos 13 gêneros mais frequentes selecionados após filtragem**



**Figure 8. Método do Cotovelo**

As matrizes de utilidade clusterizada e normalizada e a matriz de similaridade foram reconstruídas (Figuras 9, 10, 11).

[19] clustered\_utility

	0	1	2
user_id			
1	3.594203	3.375000	4.040000
2	3.846154	3.111111	4.041667
3	2.500000	2.764706	3.000000
4	5.000000	4.181818	4.800000
5	3.000000	2.785714	2.588235
...	...	...	...
939	4.545455	3.636364	4.909091
940	3.444444	3.090909	3.645161
941	4.142857	4.285714	4.333333
942	4.050000	4.285714	4.416667
943	2.913043	3.500000	3.790698

943 rows x 3 columns

**Figure 9. Matriz de Utilidade Clusterizada**

clustered\_utility\_norm

	0	1	2
user_id			
1	-0.222890	-0.869748	1.092638
2	0.366538	-1.131550	0.765012
3	-1.019020	0.039193	0.979827
4	0.795708	-1.122517	0.326809
5	1.013324	-0.027203	-0.986121
...	...	...	...
939	0.277350	-1.109400	0.832050
940	0.181528	-1.078329	0.896802
941	-1.120897	0.320256	0.800641
942	-1.080634	0.187936	0.892698
943	-1.091975	0.220883	0.871092

943 rows x 3 columns

Figure 10. Matriz de Utilidade Clusterizada e Normalizada

user\_similarity

user_id	1	2	3	4	5	6	7	8	9	10	...	934	935	936	937	938	939	940	941	942	943
user_id																					
1	1.000000	0.869174	0.631819	0.578018	-0.639836	-0.055505	0.962109	0.304189	-0.971707	0.070304	...	0.826374	-0.027938	0.969814	-0.230806	0.505160	9.061052e-01	0.938647	4.230525e-01	0.526400	0.501533
2	0.869174	1.000000	0.165861	0.905927	-0.176095	-0.541987	0.701405	-0.206680	-0.727786	-0.432177	...	0.439795	-0.518597	0.722353	-0.681765	0.012300	9.967652e-01	0.966393	-8.036883e-02	0.037086	0.008103
3	0.631819	0.165861	1.000000	-0.267311	-0.999946	0.738852	0.819226	0.930577	-0.797017	0.817617	...	0.958604	0.757161	0.801756	0.608360	0.988115	2.445797e-01	0.325734	9.696292e-01	0.991622	0.987461
4	0.578018	0.905927	-0.267311	1.000000	0.257287	-0.846849	0.333614	-0.601527	-0.368928	-0.773368	...	0.018137	-0.831855	0.361585	-0.927401	-0.412259	8.689659e-01	0.823984	-4.948719e-01	-0.389545	-0.416079
5	-0.639836	-0.176095	-0.999946	0.257287	1.000000	-0.731813	-0.825139	-0.926724	0.803247	-0.811593	...	-0.961510	-0.750335	-0.807921	-0.600083	-0.986465	-2.546386e-01	-0.335537	-9.670363e-01	-0.990227	-0.985768
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
939	0.906105	0.996765	0.244580	0.868966	-0.254639	-0.472693	0.756420	-0.127377	-0.780549	-0.358303	...	0.510951	-0.448202	0.775594	-0.620765	0.092623	1.000000e+00	0.996415	-3.330669e-16	0.117279	0.088443
940	0.938647	0.986393	0.325734	0.823984	-0.335537	-0.396447	0.809044	-0.043010	-0.830633	-0.278035	...	0.581464	-0.370969	0.826215	-0.552213	0.176527	9.964150e-01	1.000000	8.460007e-02	0.200875	0.172394
941	0.423052	-0.080369	0.969629	-0.494872	-0.967036	0.881227	0.654087	0.991854	-0.625095	0.933605	...	0.859847	0.893932	0.631233	0.783997	0.995701	-2.775558e-16	0.084600	1.000000e+00	0.993099	0.996081
942	0.526400	0.037086	0.991622	-0.389545	-0.990227	0.819708	0.738285	0.970071	-0.712324	0.885141	...	0.913791	0.835198	0.717838	0.705784	0.999693	1.172794e-01	0.200875	9.930900e-01	1.000000	0.999580
943	0.501533	0.008103	0.987461	-0.416079	-0.985768	0.835967	0.718423	0.976702	-0.691679	0.898258	...	0.901632	0.850789	0.697355	0.726022	0.999991	8.844288e-02	0.172394	9.960813e-01	0.999580	1.000000

943 rows x 943 columns

Figure 11. Similaridade entre Usuários

### 5.2.2. Modelagem supervisionada

Para cada instância (usuário-filme), foram gerados vetores de entrada com os valores da matriz clusterizada e as similaridades com outros usuários. Foram testados três modelos:

- **SVD + Regressão Linear:** com 20 componentes;
- **MLP Regressor:** duas camadas ocultas (64 e 32 neurônios);
- **XGBoost Regressor:** com 300 árvores, profundidade 6 e *learning rate* 0.05.

### 5.2.3. Resultados e recomendações

Os modelos supervisionados superaram a abordagem baseada em KNN. A Tabela 2 apresenta os RMSEs obtidos.

Table 2. RMSE obtido por modelo	
Modelo	RMSE
KNN (reprodução)	1.180
Reprodução melhorada	1.092
SVD + Regressão Linear	1.045
MLP Regressor	1.046
XGBoost Regressor	<b>1.042</b>

A Figura 12 resume a comparação entre os modelos.

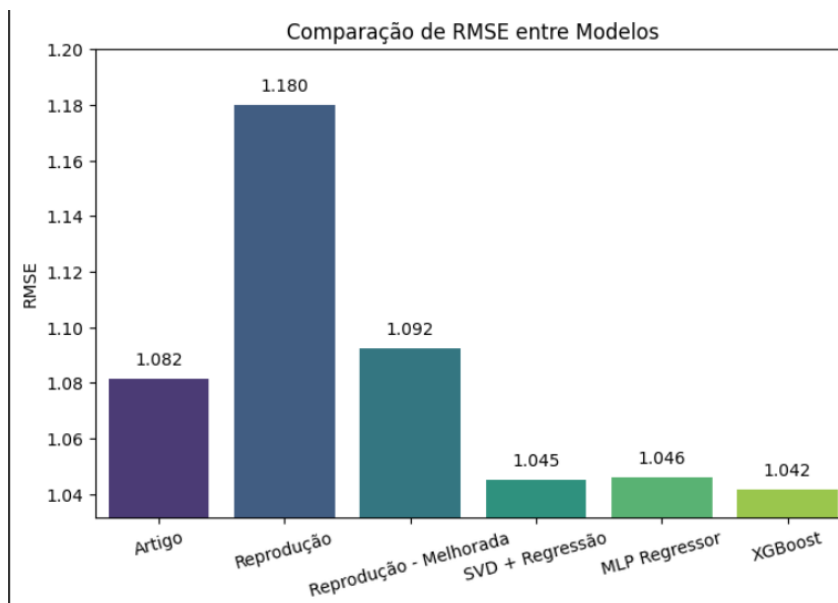


Figure 12. Comparação dos Modelos

Além da melhora quantitativa, a qualidade das recomendações também foi aprimorada. A Tabela 3 apresenta os filmes recomendados ao usuário 10 pelo modelo com melhor desempenho.

Table 3. Top 5 recomendações para o usuário 10 (modelo supervisionado)

Filme	Nota prevista
Star Wars (1977)	5.00
Braveheart (1995)	5.00
Blade Runner (1982)	5.00
Independence Day (ID4) (1996)	5.00
Empire Strikes Back, The (1980)	5.00

Os resultados demonstram que modelos supervisionados podem oferecer ganhos reais de desempenho, ainda que modestos, contribuindo para sistemas de recomendação mais precisos e personalizados.

## 6. Resultados e Discussão

Nesta seção, apresentamos uma análise detalhada dos resultados obtidos nos experimentos descritos anteriormente. A comparação abrange tanto a reimplementação do sistema de recomendação baseado em K-Means e KNN, quanto as versões estendidas com modelos supervisionados de regressão. Os resultados são discutidos sob duas perspectivas: desempenho quantitativo, medido por meio da métrica RMSE, e análise qualitativa das recomendações fornecidas.

### 6.1. Análise Quantitativa

O desempenho de cada modelo foi avaliado utilizando o *Root Mean Squared Error* (RMSE), uma métrica amplamente empregada em sistemas de recomendação para mensurar a diferença média entre as notas previstas e as notas reais. A Tabela 2 e a Figura 12 consolidam os valores obtidos nos testes.

Observa-se que a reprodução do modelo proposto por [Harper and Konstan 2015] apresentou um RMSE de 1.180, ligeiramente acima do valor relatado no artigo original (1.081). Essa pequena discrepância pode ser atribuída a fatores como o ambiente de execução, possíveis diferenças de implementação em detalhes não especificados no artigo e variações em pré-processamentos como normalização ou escolha de vizinhos.

A versão com pequenas melhorias — como ajustes na filtragem de usuários ou no manuseio de filmes não avaliados — resultou em um desempenho aprimorado, com RMSE de 1.092.

Entre os modelos supervisionados testados, todos superaram a abordagem original. O modelo com melhor desempenho foi o **XGBoost**, com RMSE de **1.042**, seguido pelo **SVD + regressão linear** (1.045) e pelo **MLP Regressor** (1.046). Embora os ganhos absolutos sejam modestos, eles são estatisticamente consistentes e confirmam o potencial de modelos supervisionados na tarefa de previsão de avaliações em sistemas de recomendação.

### 6.2. Análise Qualitativa

A avaliação qualitativa foi realizada a partir da análise das cinco principais recomendações geradas para um usuário específico (usuário 10). Na reprodução do modelo original, os cinco filmes recomendados receberam nota 5.00, incluindo títulos como *The Usual Suspects* (1995) e *Mr. Holland's Opus* (1995), conforme mostrado na Tabela 1.

Na extensão supervisionada, o mesmo usuário recebeu recomendações também com nota máxima, mas agora compostas por títulos de maior popularidade e impacto cultural, como *Star Wars* (1977), *Blade Runner* (1982) e *Braveheart* (1995), como apresentado na Tabela 3. Esse resultado evidencia que os modelos supervisionados podem não apenas melhorar a acurácia das predições, como também oferecer listas de recomendação mais relevantes e representativas dos interesses típicos dos usuários.

### 6.3. Discussão

Os resultados obtidos confirmam a robustez e reprodutibilidade do modelo baseado em K-Means e KNN, bem como a viabilidade de aprimoramentos utilizando aprendizado supervisionado. A abordagem supervisionada permite capturar padrões mais sutis e complexos

de interação entre usuários e itens, algo que a média ponderada baseada em vizinhos não alcança completamente.

Apesar dos ganhos em desempenho não serem drasticamente elevados, é importante destacar que os modelos supervisionados apresentam vantagens adicionais em escalabilidade, customização e extensibilidade, sobretudo quando aplicados em bases maiores ou em contextos com múltiplos tipos de entrada, como texto e imagem.

Adicionalmente, a filtragem semântica dos gêneros contribuiu para a melhoria da estrutura de agrupamento dos filmes, reduzindo ruído e aumentando a coesão dos clusters formados.

Os resultados reafirmam que mesmo em contextos com bases de dados moderadas, como o MovieLens 100k, o uso criterioso de técnicas supervisionadas pode gerar ganhos reais de precisão e relevância nas recomendações.

## 7. Conclusão

Este trabalho teve como objetivos principais a reprodução fiel do sistema de recomendação proposto por [Ahuja et al. 2019], baseado na combinação de K-Means para clusterização de filmes e KNN para predição de avaliações, bem como a proposição e avaliação de uma extensão desse modelo com o uso de técnicas supervisionadas de regressão.

A reprodução validou a eficácia da abordagem original, alcançando resultados comparáveis aos apresentados pelos autores. A estrutura híbrida baseada em agrupamento por gênero e similaridade entre usuários mostrou-se eficaz, com desempenho sólido e boa qualidade nas recomendações.

Na segunda etapa do estudo, ao substituir a etapa de predição por modelos supervisionados — incluindo SVD, MLP e XGBoost —, foi possível observar uma redução consistente no erro preditivo (RMSE), destacando o potencial dessas abordagens no aprimoramento de sistemas de recomendação híbridos. Entre os modelos testados, o XGBoost obteve o melhor resultado, com RMSE de 1,042.

Além dos ganhos quantitativos, as análises qualitativas indicaram que os modelos supervisionados podem capturar preferências mais refinadas e oferecer listas de recomendação mais coerentes com os gostos dos usuários.

Este estudo também destaca a importância da reprodutibilidade científica como ferramenta para validar e estender propostas da literatura. O reuso do MovieLens 100k, aliado à documentação detalhada do processo, permitiu tanto confirmar hipóteses anteriores quanto propor avanços concretos.

Como trabalhos futuros, propõe-se:

- Explorar o impacto de outras representações semânticas dos filmes, como *embeddings* aprendidos por redes neurais ou modelos de linguagem;
- Estender a abordagem para cenários com dados implícitos ou interações em tempo real;

- Integrar estratégias de avaliação online com coleta de feedback de usuários reais;
- Investigar técnicas alternativas de agrupamento, indo além dos vetores de gêneros. Por exemplo, o agrupamento dos filmes ou usuários poderia ser baseado em atributos latentes extraídos de comentários textuais, usando análise de sentimentos ou modelos de PLN para gerar representações semânticas ricas. Essas representações poderiam ser utilizadas como novas features tanto para clusterização quanto para predição de avaliações.

Tais extensões abririam caminho para um sistema de recomendação mais sensível ao contexto emocional e subjetivo dos usuários, potencializando a personalização e a relevância das sugestões oferecidas.

## References

- Ahuja, R., Solanki, A., and Nayyar, A. (2019). Movie recommender system using k-means clustering and k-nearest neighbor. In *2019 9th International Conference on Cloud Computing, Data Science Engineering (Confluence)*, pages 263–268.
- Chen, T. and Guestrin, C. (2016). Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 785–794. ACM.
- Harper, F. M. and Konstan, J. A. (2015). The movielens datasets: History and context. *ACM Trans. Interact. Intell. Syst.*, 5(4).
- Ricci, F., Rokach, L., and Shapira, B. (2015). *Recommender Systems Handbook*. Springer, 2nd edition.
- Sarwar, B., Karypis, G., Konstan, J., and Riedl, J. (2002). Recommender systems for large-scale e-commerce: Scalable neighborhood formation using clustering. In *Proceedings of the Fifth International Conference on Computer and Information Technology*.
- Su, X. and Khoshgoftaar, T. M. (2009). A survey of collaborative filtering techniques. *Advances in Artificial Intelligence*, 2009.