



## Descrição

Implementar um sistema de recomendação baseado em filtragem colaborativa utilizando a decomposição em valores singulares (SVD) e analisar o impacto dos autovalores e autovetores na qualidade das recomendações.

Usando os dados do grouplens.org, converta os dados em uma matriz de ratings usuário-item, preenchendo os valores ausentes com zeros ou a média dos ratings do usuário/item. Normalize os dados, se necessário.

Implemente um SVD usando funções de autovalores e autovetores. Aplique a SVD na matriz de ratings para obter as matrizes  $U, \Sigma, V^T$ . Utilize as matrizes  $U, V^T$  para representar usuários e itens em um espaço de menor dimensão (redução de dimensionalidade). Reconstrua a matriz de ratings aproximada utilizando um número reduzido de valores singulares. Faça previsões de ratings (valores ausentes que foram preenchidos) com base na matriz aproximada.

- Para avaliar a performance, esconda 20% das tuplas usuário-item para comparar com os valores encontrados pelo SVD.
- Analise os autovalores da matriz Sigma e interprete seu significado em relação à variância explicada pelos componentes principais.
- Visualize os autovetores da matriz  $UeV^T$  para interpretar as relações entre usuários e itens no espaço de menor dimensão.
- Discuta o impacto da escolha do número de valores singulares na qualidade das recomendações e na complexidade do modelo.

Melhore a forma de promover o preenchimento de valores faltantes com as seguintes alternativas:

- Imputação por KNN: Utilizar o algoritmo k-Nearest Neighbors para estimar os valores faltantes com base nos valores de vizinhos mais próximos. Essa abordagem pode ser mais precisa, mas é computacionalmente mais cara.
- Imputação por Regressão: Utilizar modelos de regressão para prever os valores faltantes com base em outros atributos. Essa abordagem pode ser poderosa, mas requer cuidado para evitar overfitting.
- Implementando alguma estratégia discutida no seguinte paper de Hastie et al.

PS: para algoritmos mais complexos, pode-se usar o SVD da biblioteca do Jax/numpy. Mas a primeira implementação deve ser feita usando diretamente autovalores/autovetores.

## Tarefas

### Entregáveis

Um notebook Jupyter (ou similar) contendo:

- Implementações.
- Experimentos comparando mostrando desempenho da fatorização com e sem alternativas mais robustas de imputação de dados.
- Análise e discussão dos resultados, incluindo gráficos e tabelas relevantes.

### CrITÉrios de Avaliação

- Clareza do código e organização do notebook.
- Correção e eficiência das implementações.
- Qualidade da análise e discussão dos resultados.
- Profundidade da compreensão dos conceitos e técnicas exploradas.