

Sistemas de Recomendação

Filtragem Colaborativa

Model-based



PUC Minas

Premissa CF: as preferências passadas dos usuários revelam seus interesses.

Extrai modelos dos usuários e itens para a tarefa de predição.

- **Machine Learning:**
 - Redes Neurais, Deep Learning, e outros.
- **Redução de Dimensionalidade:**
 - Fatorização de Matrizes (SVD e PCA).

Redução de Dimensionalidade

- Dado uma matriz de ratings de m usuários e n itens.
 - *User-based CF* é m -dimensional
 - *Item-based CF* é n -dimensional
- **Objetivo:** Reduzir as dimensões da matriz mantendo a representatividade das informações dos usuários e itens.

Exemplo

Star Wars

Star Trek

Die Hard

Harry Potter

Twilight

Pretty Woman

Exemplo

Star Wars

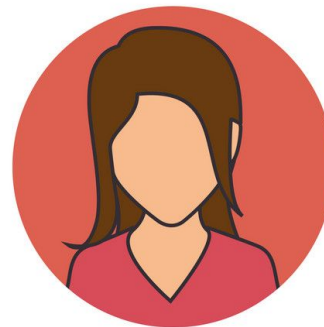
Star Trek

Die Hard

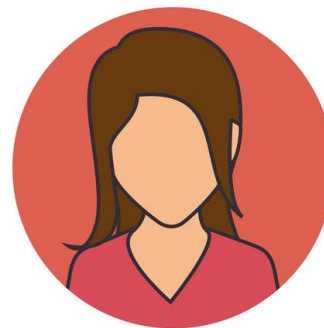
Harry Potter

Twilight

Pretty Woman



Exemplo



male
female
reality
fantasy

“fantasy”

Sue

Star Wars

Star Trek

Twilight

Harry Potter

Bob

Mary

Alice

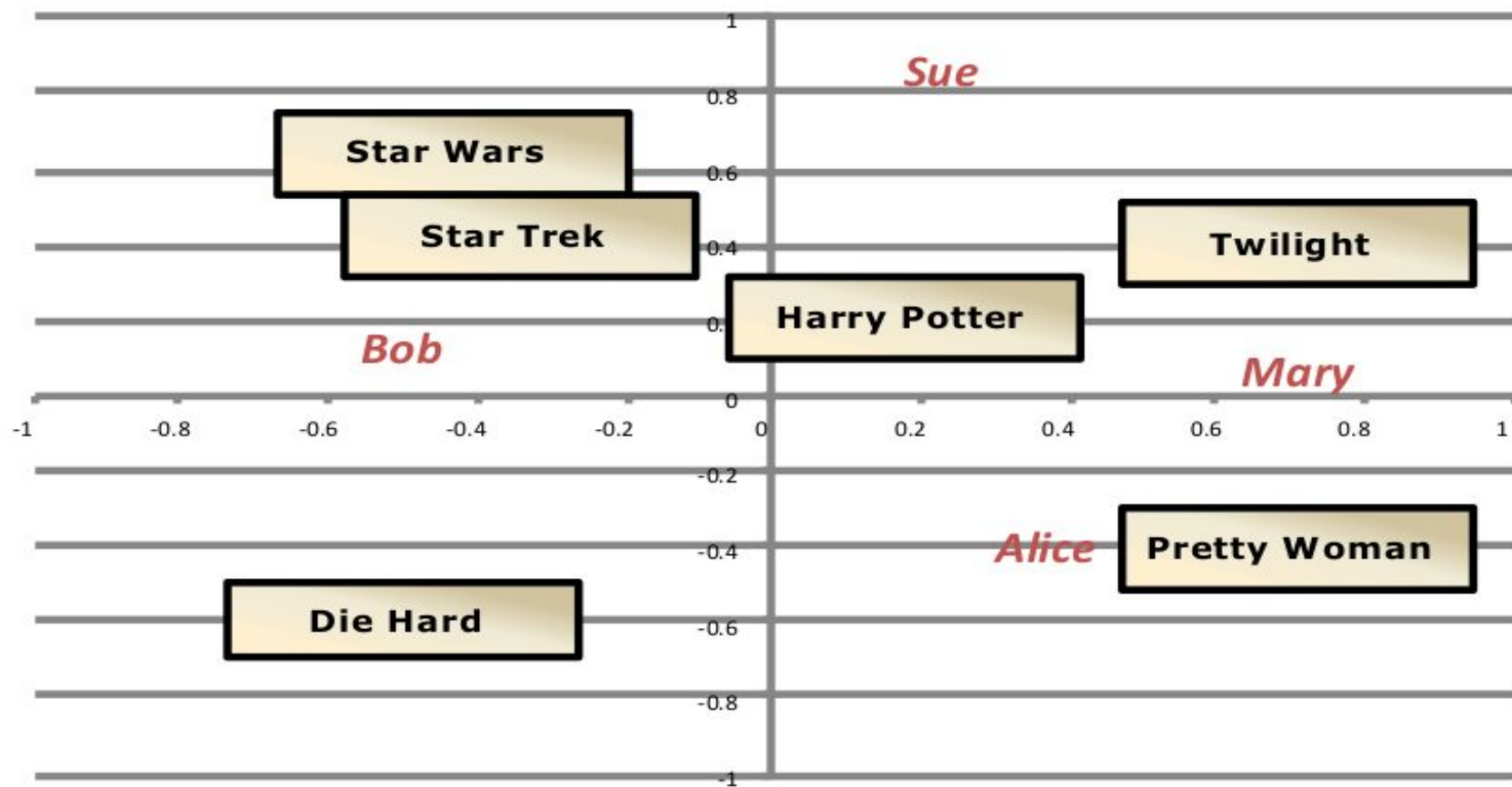
Pretty Woman

Die Hard

“reality”

“female”

“male”

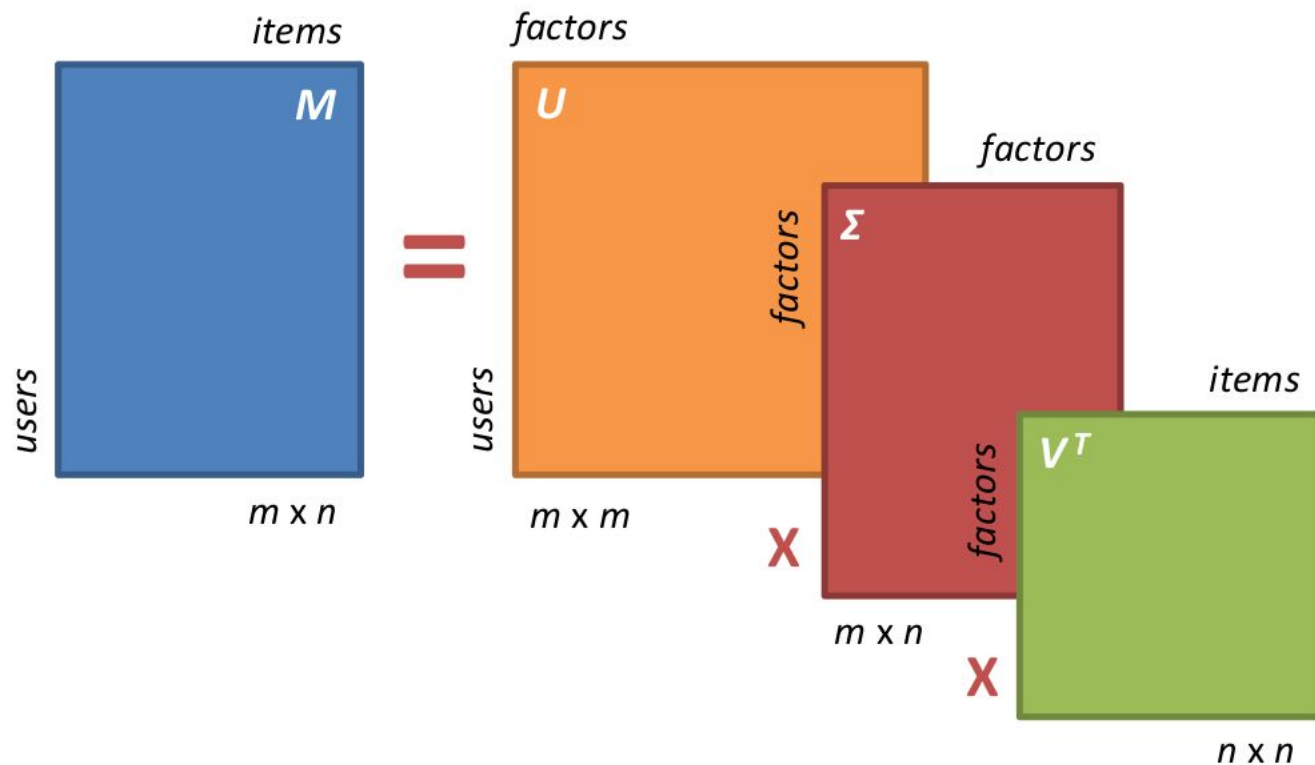


- Latent Semantic Indexing (LSI)
 - Deerwester et al., ASIS 1988
 - Representa consultas e documentos em um espaço compacto e robusto de tópicos latentes.
- Latent Dirichlet Allocation (LDA)
 - Uma estratégia de modelagem de tópicos, que leva em conta a esparsidade dos modelos.

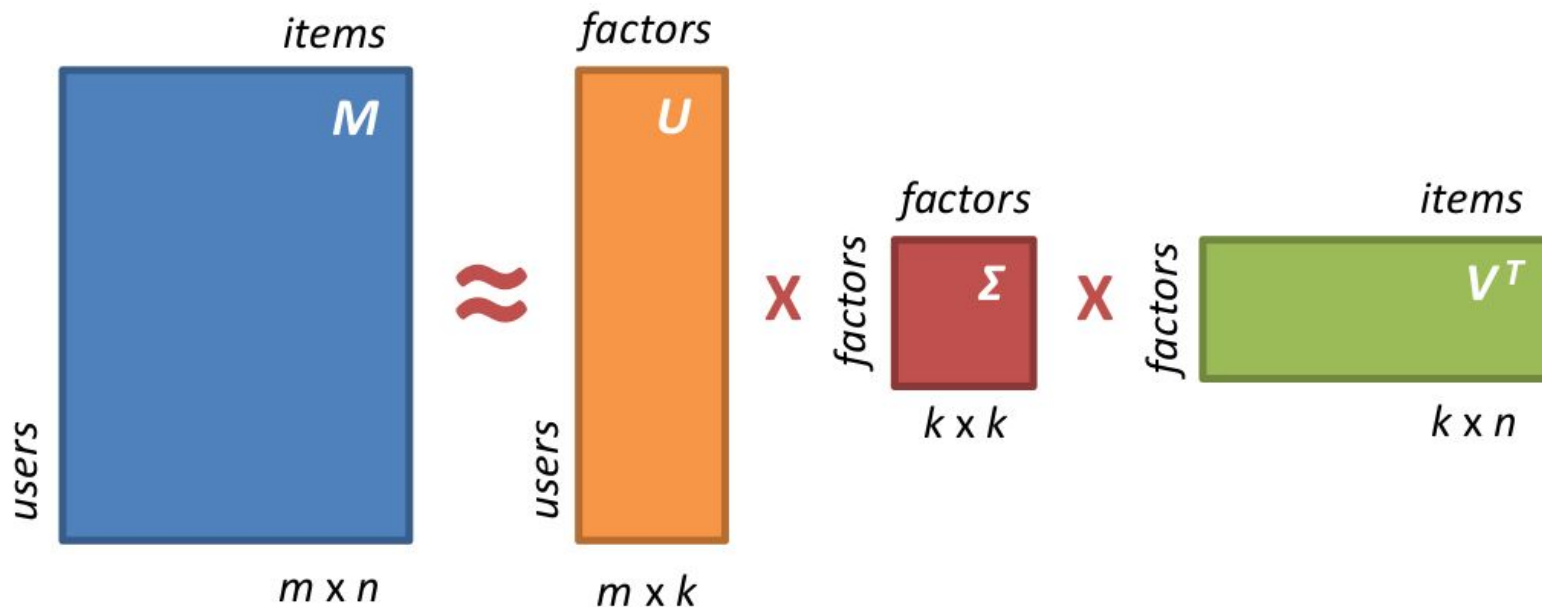
Singular Value Decomposition

- Reduz a dimensionalidade de uma matriz de valores reais.
 - A ‘dimensionalidade’ está relacionado a quantidade de dados.
 - Reduzir significa diminuir as colunas e linhas da matriz.
- Resulta em um conjunto pequeno de dados.
 - Torna o modelo mais rápido.
- Mantém as informações mesmo com menos dimensões.
 - Constrói uma rede densa de vizinhos.

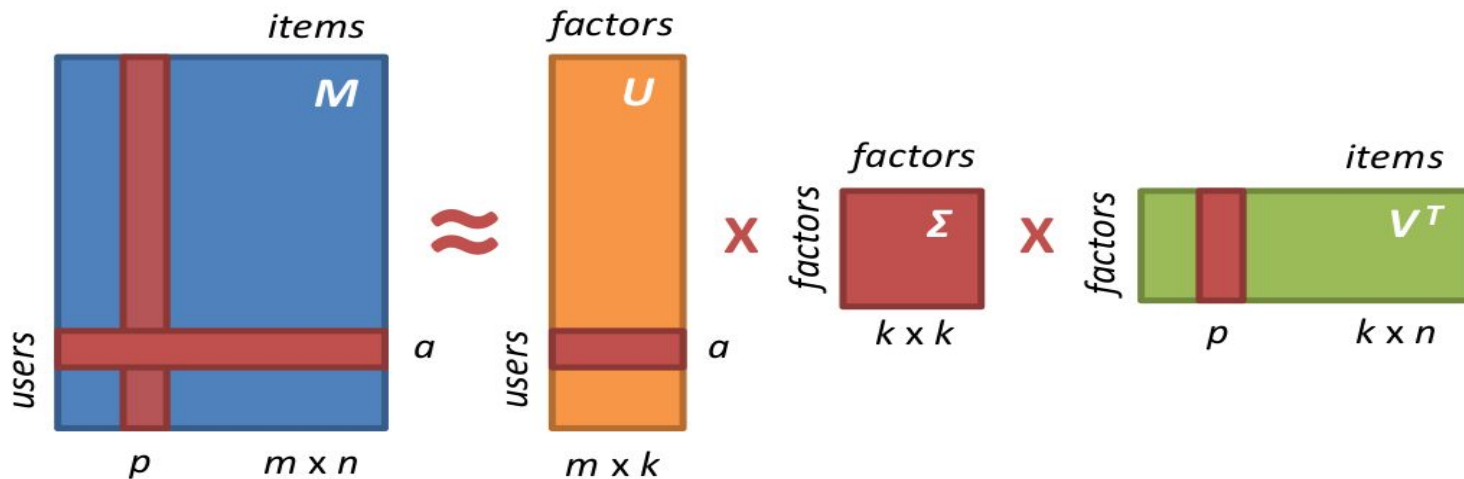
Fatorização com SVD



Fatorização com SVD



Predição com SVD



$$\hat{r}_{ap} = \bar{r}_a + \frac{\sum_{b \in N} \text{sim}(a, b) \times (r_{bp} - \bar{r}_b)}{\sum_{b \in N} \text{sim}(a, b)} \approx \hat{r}_{ap} = \bar{r}_a + U_a \times \Sigma \times V_p^T$$

- A qualidade da predição geralmente aumenta.
 - Evitamos os ruídos dos dados.
 - Conseguimos detectar correlações não triviais.
- Depende do tamanho da redução
 - Normalmente valores pequenos são melhores.
 - Valores em torno de 20 a 100 (Koren, KDD 2009).
 - Depende do domínio de aplicação.

- Falta transparência
 - Os valores encontrados não possuem um significado semântico.
- Sofre com a esparsidade dos dados
 - SVD não se comporta muito bem para matrizes muito esparsas
 - A matriz de ratings é muito esparsa
- Possui uma alta complexidade
 - Computar o SVD é $O(m^2n + n^3)$

Esparsidade dos Dados

- SVD assume uma matriz completa
 - Mas, se a matriz é completa, não precisamos de um recomendador!
- O que fazer nesse caso?
 - Assumimos um valor (e.g., média)
 - Normalizamos (e.g., assumimos ser 0)
 - Ignoramos!

- Identificar o valor ideal de autovalores é complicado.
 - Precisamos de um processo de otimização
 - $Erro = rating\ original - rating\ estimado$

- Identificar o valor ideal de autovalores é complicado.
 - Precisamos de um processo de otimização
 - $Erro = rating\ original - rating\ estimado$
- Funk SVD
 - Uma das melhores propostas no Netflix Prize (3º lugar)
 - Simon Funk propôs uma otimização via gradiente descendente

1. Pure SVD

- Abordagem simples proposta por Paolo Cremonesi no RecSys'10

2. Normalized SVD

- Recomendação via SVD sobre a matriz de ratings normalizada

- Uma análise prática dos recomendadores na tarefa de ranking.
- Proposta de um modelo simples e eficiente para o SVD.

Performance of Recommender Algorithms on Top-N Recommendation Tasks

Paolo Cremonesi, Yehuda Koren, Roberto Turrin

- [Recommender Systems \(2010\)](#)

- Dado uma matriz de ratings R , aplicamos o SVD:

$$\hat{\mathbf{R}} = \mathbf{U} \cdot \mathbf{\Sigma} \cdot \mathbf{Q}^T$$

- \mathbf{U} é a matriz de m usuários por f fatores
 - $\mathbf{\Sigma}$ é a matriz diagonal quadrada dos f fatores
 - \mathbf{Q} é matriz de n itens por f fatores
- **Predição:** usa-se o histórico de \mathbf{u} e os fatores latentes de \mathbf{i}

$$\hat{r}_{ui} = \mathbf{r}_u \cdot \mathbf{Q} \cdot \mathbf{q}_i^T$$

- Artigo com diversos exemplos práticos de algoritmos CF.
- Uma ótima leitura **extra** para entender detalhes da implementação.

A careful assessment of recommendation algorithms related to dimension reduction techniques

Chun-Xia Yin, Qin-Ke Peng

- Knowledge-Based Systems (2012)

Normalized SVD

Existem cinco passos fundamentais:

1. Converter a matriz esparsa de ratings em uma matriz densa.
2. Normalizar a matriz de dados via Z-score.
3. Aplicar o SVD na matriz normalizada.
4. Adquirir as matrizes de menores dimensões.
5. Computar os ratings a serem preditos.

1. Matriz Densa

$$A_{5 \times 8} = \begin{pmatrix} -2 & ? & 5 & 1 & ? & 3 & ? & ? \\ -1 & 3 & 2 & ? & 4 & ? & 5 & ? \\ ? & ? & -4 & 2 & 4 & -2 & ? & 5 \\ -2 & 2 & 4 & 2 & 5 & ? & ? & ? \\ -3 & ? & -5 & 1 & 3 & 1 & ? & 4 \end{pmatrix} \xRightarrow{\text{filling}} B_{5 \times 8} = \begin{pmatrix} -2 & 0 & 5 & 1 & 0 & 3 & 0 & 0 \\ -1 & 3 & 2 & 0 & 4 & 0 & 5 & 0 \\ 0 & 0 & -4 & 2 & 4 & -2 & 0 & 5 \\ -2 & 2 & 4 & 2 & 5 & 0 & 0 & 0 \\ -3 & 0 & -5 & 1 & 3 & 1 & 0 & 4 \end{pmatrix}$$

- Dado a matriz A, com ratings positivos e negativos
- Completamos a matriz esparsa com 0s.

2. Normalização Z-score

$$B_{5 \times 8} = \begin{pmatrix} -2 & 0 & 5 & 1 & 0 & 3 & 0 & 0 \\ -1 & 3 & 2 & 0 & 4 & 0 & 5 & 0 \\ 0 & 0 & -4 & 2 & 4 & -2 & 0 & 5 \\ -2 & 2 & 4 & 2 & 5 & 0 & 0 & 0 \\ -3 & 0 & -5 & 1 & 3 & 1 & 0 & 4 \end{pmatrix} \xrightarrow{\text{Z-scores}} \begin{pmatrix} -0.351 & -0.707 & 0.997 & -0.239 & -1.664 & 1.431 & -0.447 & -0.723 \\ 0.526 & 1.414 & 0.347 & -1.434 & 0.416 & -0.220 & 1.789 & -0.723 \\ 1.403 & -0.707 & -0.953 & 0.956 & 0.416 & -1.321 & -0.447 & 1.285 \\ -0.351 & 0.707 & 0.780 & 0.956 & 0.936 & -0.220 & -0.447 & -0.723 \\ -1.228 & -0.707 & -1.170 & -0.239 & -0.104 & 0.330 & -0.447 & 0.884 \end{pmatrix}$$

- Subtrai o valor pela média dos ratings
- Divide pelo desvio padrão dos ratings

$$Z = \frac{X - E[X]}{\sigma(X)}$$

3. Aplicação do SVD

$$\begin{pmatrix} 0.371 & -0.657 & -0.075 & -0.475 & -0.447 \\ 0.531 & 0.609 & 0.378 & -0.067 & -0.447 \\ -0.713 & 0.250 & -0.016 & -0.479 & -0.447 \\ 0.072 & 0.138 & -0.780 & 0.409 & -0.447 \\ -0.260 & -0.340 & 0.492 & 0.612 & -0.447 \end{pmatrix} \times \begin{pmatrix} 3.616 & 0.000 & 0.000 & 0.000 & 0.000 \\ 0.000 & 3.285 & 0.000 & 0.000 & 0.000 \\ 0.000 & 0.000 & 2.211 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.000 & 1.802 & 0.000 \\ 0.000 & 0.000 & 0.000 & 0.000 & 0.000 \end{pmatrix} \times \begin{pmatrix} -0.154 & 0.387 & -0.058 & -0.796 & -0.090 \\ 0.339 & 0.453 & -0.136 & 0.242 & -0.501 \\ 0.441 & -0.054 & -0.503 & -0.242 & 0.606 \\ -0.387 & -0.080 & -0.634 & -0.002 & -0.288 \\ -0.165 & 0.492 & -0.229 & 0.489 & 0.416 \\ 0.347 & -0.471 & 0.075 & 0.044 & -0.063 \\ 0.328 & 0.415 & 0.382 & -0.083 & 0.099 \\ -0.511 & -0.014 & 0.344 & 0.012 & 0.322 \end{pmatrix}$$

U
 Σ
 V

- Seleciona-se a matriz de Z-score e aplica o SVD:

- U é a matriz de usuários por fatores.
- Σ é a matriz diagonal de autovalores.
- V é a matriz de itens por fatores.

$$Z = U \Sigma V^T$$

4. Seleção das dimensões

$$\begin{aligned}
 Z_d^{d=2} &= \begin{pmatrix} 0.371 & -0.657 \\ 0.531 & 0.609 \\ -0.713 & 0.250 \\ 0.072 & 0.138 \\ -0.260 & -0.340 \end{pmatrix} \begin{pmatrix} 3.616 & 0.000 \\ 0.000 & 3.285 \end{pmatrix} \begin{pmatrix} -0.154 & 0.387 \\ 0.339 & 0.453 \\ 0.441 & -0.054 \\ -0.387 & -0.080 \\ -0.165 & 0.492 \\ 0.347 & -0.471 \\ 0.328 & 0.415 \\ -0.511 & -0.014 \end{pmatrix}^T \\
 &\quad \quad \quad U_d \quad \quad \quad \Sigma_d \quad \quad \quad V_d \\
 &= \begin{pmatrix} -1.041 & -0.522 & 0.706 & -0.346 & -1.283 & 1.481 & -0.455 & -0.656 \\ 0.479 & 1.557 & 0.738 & -0.904 & 0.666 & -0.277 & 1.459 & -1.009 \\ 0.715 & -0.503 & -1.180 & 0.932 & 0.830 & -1.280 & -0.505 & 1.307 \\ 0.136 & 0.294 & 0.090 & -0.137 & 0.180 & -0.124 & 0.273 & -0.139 \\ -0.288 & -0.825 & -0.355 & 0.454 & -0.394 & 0.200 & -0.772 & 0.496 \end{pmatrix}
 \end{aligned}$$

Apenas algumas dimensões (as mais importantes) são selecionadas.

5. Predição de Ratings

A predição é feita por:

- A média das notas dos itens.
- O desvio padrão dessas mesmas notas
- O valor adquirido pelas matrizes do SVD.

$$p_{ij} = \overline{B^j} + \sigma_j^*(Z_d)_{ij}$$

- Abordagens Model-based tendem a ser as mais eficazes em SsR
 - Vale a pena estudá-las
 - Vale a pena implementá-las no seu modelo.