Redução de Dimensionalidade, DCDistance, e CARFRE

Fabrício Olivetti de França

Universidade Federal do ABC

Tópicos

1. Redução de Dimensionalidade
2. Análise de Componentes Principais
3. DCDistance - Document-Class Distance
4. CARFRE - Categorical Average Rating Feature Reduction

UFABC

Redução de Dimensionalidade

Redução de Dimensionalidade

Quando trabalhamos com muitos atributos, alguns destes podem ser:

- **Redundantes:** dois ou mais representam a mesma informação (Ex.: tempo em segundos e tempo em minutos)
- **Ruidosos:** possuem valores aleatórios sem significado (Ex.: sinais ruidosos)
- **Irrelevantes:** não possuem relação com a tarefa desejada (Ex.: termos irrelevantes para classificação de sentimentos)

Redução de Dimensionalidade

Além disso, quando possível, é desejável compactar esses atributos para uma espaço de menor dimensão.

Curse of Dimensionality: quanto maior a dimensionalidade, mais difícil se torna perceber diferenças de similaridades; complexidade de algoritmos em função de d, e número de amostras de exemplo necessárias cresce exponencialmente.

Análise de Componentes Principais

Principal Components Analysis (PCA)

Identifica as direções de maior variação de valores.

Rotaciona o eixo para que cada eixo rotacionado represente da maior para a menor variação.

Principal Components Analysis (PCA)

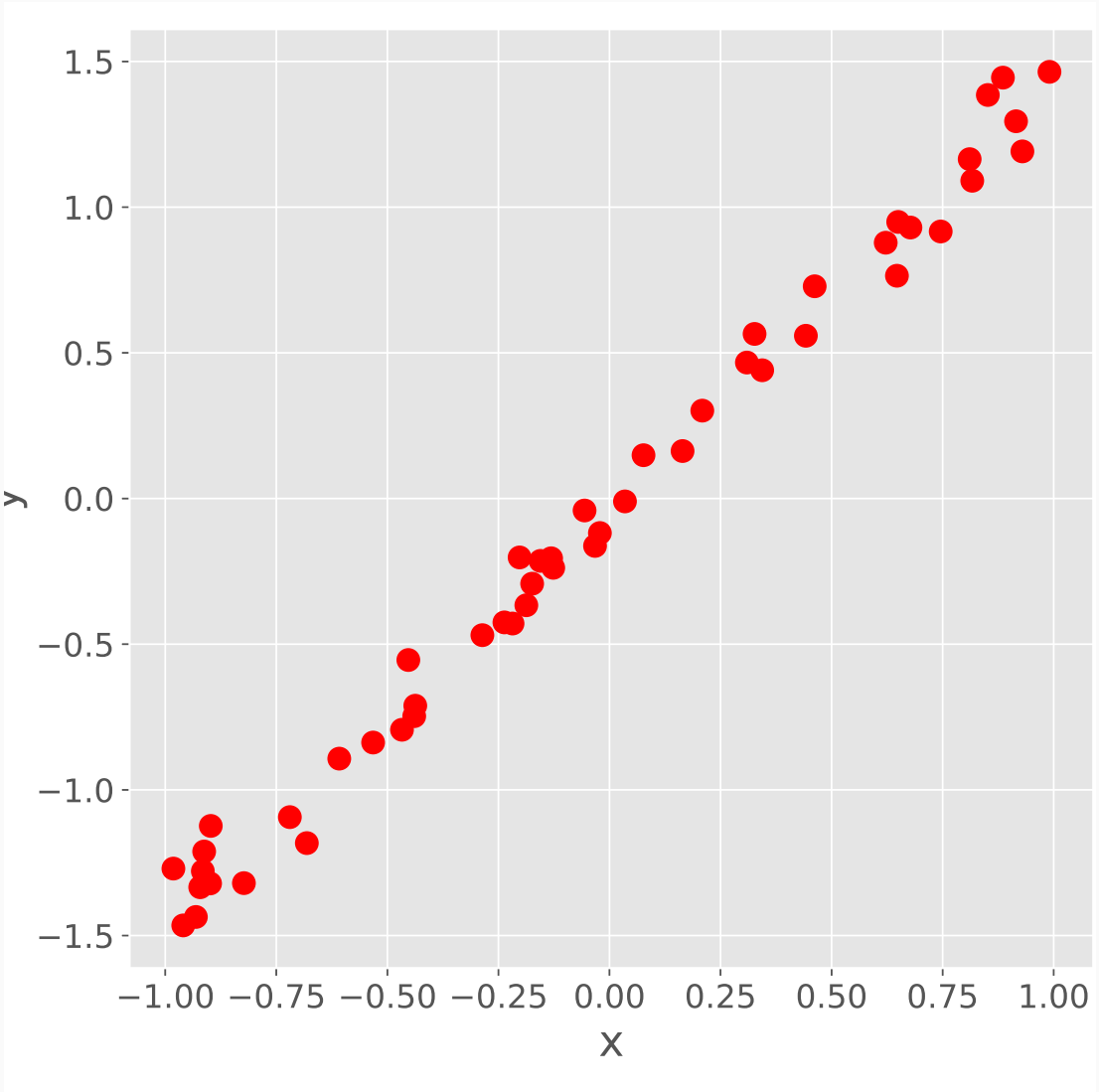

Figura 1: Eixos originais.

UFABC

Principal Components Analysis (PCA)

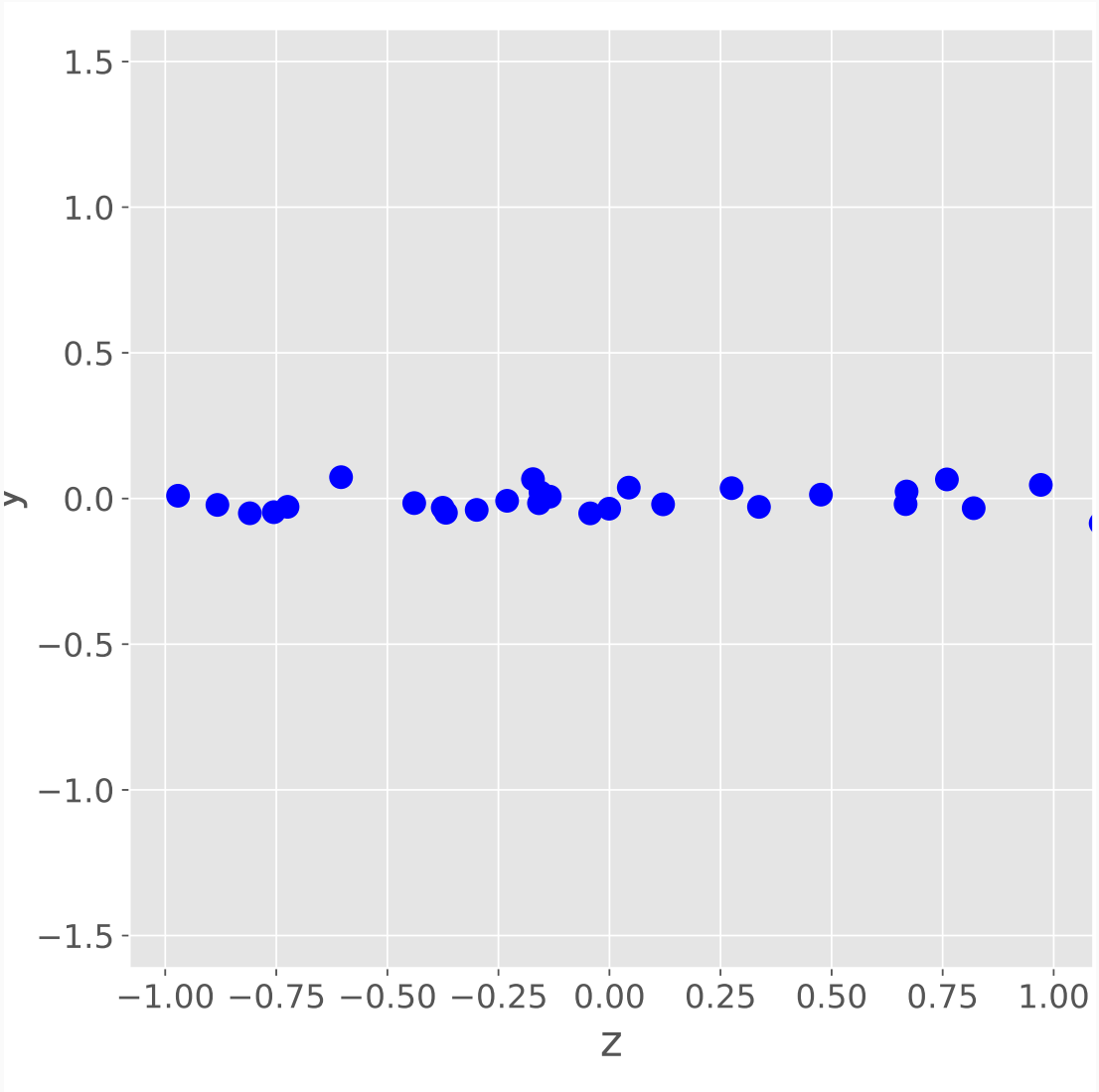

Figura 2: Eixos rotacionados.

UFABC

Principal Components Analysis (PCA)

Para isso utiliza a informação de autovalores e autovetores da matriz de covariância dos atributos.

UFABC

Covariância dos Atributos

Dada uma matriz de dados $X \in \Re^{n \times d}$, centralizamos os pontos para que fiquem com média zero:

$$x'_{i,j} = x_{i,j} - \hat{x}_j,$$

com $\hat{x}_j$ sendo a média dos valores do atributo j.

UFABC

Covariância dos Atributos

A covariância dos atributos pode ser calculada como:

$$Cov = \frac{1}{n} X'^T X',$$

que resulta em uma matriz $Cov \in \Re^{d \times d}$.

UFABC

Covariância dos Atributos

O elemento i, j dessa matriz representa a correlação entre o atributo i e o atributo j.

A diagonal indica a variância do respectivo atributo.

UFABC

Autovalores e Autovetores

Dessa matriz podemos extrair um total de $\mathbf{d}$ [1] autovalores (λ) e autovetores (e) tal que:

$$Cov \cdot e = \lambda \cdot e$$

[1] apenas se todos os atributos forem independentes

UFABC

Autovalores e Autovetores

Se ordenarmos todos os autovalores/autovetores pela ordem do maior autovalor para o menor, temos que:

- Cada autovetor i representa a i-ésima direção de maior variação
- O autovalor correspondente quantifica essa variação

UFABC

Principal Components Analysis (PCA)

Cada autovetor representa uma combinação linear dos atributos originais de tal forma a capturar a variação descrita pelo autovalor.

Basicamente a matriz de autovetores é uma base de dados rotacionada que captura a variação em ordem crescente.

Principal Components Analysis (PCA)

Se um autovalor for muito pequeno, significa que não existe variação naquele eixo e, portanto, ele pode ser descartado.

Imagine um problema de classificação utilizando apenas uma variável x_j com variância baixa. É fácil perceber que tal variável não tem poder discriminatório pois, para toda classe, ela apresenta um valor muito similar.

Rotacionando a base de dados

De posse da matriz $E \in \Re^{d \times k}$ dos k primeiros autovetores com um valor significativo de λ, é possível transformar a matriz de dados centralizada X' com:

$$X^* = X' \times E.$$

Isso transforma a matriz em uma matriz $X^* \in \Re^{n \times k}$ com $k < d$.

Principal Components Analysis (PCA)

Usos do PCA:

- Transformar a base de dados mantendo a dimensão.
- Reduzir a base de dados, eliminando eixos com pouca variância.
- Reduzir para duas dimensões para visualizar os dados.

Principal Components Analysis (PCA)

Quando utilizado para redução de dimensionalidade, verifique o custo-benefício de remover um eixo dado sua variância.

Além disso, uma vez que a base é transformada, os atributos perdem totalmente seu significado original.

Se temos os atributos m^2, garagens, andar, bairro; uma vez aplicado o PCA não sabemos que combinação linear cada novo atributo representa.

PCA - Algoritmo

```
Xcenter = X - X.mean(axis=0)
Covar = Xcenter.T @ Xcenter
eva, eve = np.linalg.eigh(Covar)
idx = np.argsort(-eva)
pcaMtx = eve[:,idx[:k]]
X_red = Xcenter @ pcaMtx
```

UFABC

PCA - Scikit-Learn

```
from sklearn.decomposition import PCA

pca = PCA(k)
X_red = pca.fit_transform(X)
```

UFABC

DCDistance - Document-Class Distance

Relembrando: Extraindo atributos textuais

Para ser possível aplicar algoritmos de Aprendizado de Máquina em dados textuais é necessário estruturar a informação.

Uma das formas é utilizando o esquema Bag-of-Words.

Relembrando: Extraindo atributos textuais

"Essa aula está bem interessante, quero ganhar pontos"

"Não usarei os pontos que ganhei nessa aula, quero fazer prova"

Se traduziria para algo como:

aula	interessante	ganhar	pontos	usar	fazer	prova
1	1	1	1	0	0	0
1	0	1	1	1	1	1

UFABC

Relembrando: Extraindo atributos textuais

É fácil perceber que conforme a quantidade e o tamanho dos textos aumentam, a dimensão da representação vetorial dos documentos aumenta consideravelmente.

Nem todos os termos são úteis para classificar.

UFABC

Relembrando: Extraindo atributos textuais

Se eu quero saber se o aluno gostou da aula ou não, muitos dos termos extraídos anteriormente são inúteis. Talvez bastaria verificar a ocorrência do termo *interessante*.

Relembrando: Extraindo atributos textuais

Se eu quero saber se vai usar os pontos ou não, talvez fosse necessário obter o atributo *usar* e *não usar*.

Para obter o atributo *não usar* temos que definir os tokens como sequência de duas palavras, aumentando exponencialmente o número total de atributos.

DCDistance: Document-Class Distance

Uma forma de extrair novos atributos a partir dos atributos estruturados, é através do algoritmo DCDistance.

Esse algoritmo verifica a distância entre cada documento e um vetor representativos de cada classe. Se temos k classes, teremos como resultado vetores de dimensão k.

UFABC

DCDistance

Dada a base de dados de exemplo:

Tabela 1: Base de Dados

doc	data	label
D1	doll ball car	C1
D2	doll ball toy bird bear	C1
D3	doll ball car toy bear	C1
D4	dog toy cat doll	C2
D5	dog toy cat deer doll bird	C2

DCDistance

Extraímos os atributos para uma forma estruturada (podemos utilizar tf ou tf-idf):

Tabela 2: Bag-of-Words

doc	features	label
D1	1 1 1 0 0 0 0 0 0	C1
D2	1 1 0 1 1 1 0 0 0	C1
D3	1 1 1 1 0 1 0 0 0	C1
D4	1 0 0 1 0 0 1 1 0	C2
D5	1 0 0 1 1 0 1 1 1	C2

UFABC

DCDistance

Utilizando apenas a base de treino, selecionamos todos os documentos da classe C1 e calculamos um vetor representativo através de uma operação de agregação. Nesse exemplo utilizamos a soma:

Tabela 3: Agregando informações da classe C1 (soma)

doc	features	label
D1	1 1 1 0 0 0 0 0 0	C1
D2	1 1 0 1 1 1 0 0 0	C1
D3	1 1 1 1 0 1 0 0 0	C1
Soma	3 3 2 2 1 2 0 0 0	

UFABC

DCDistance

Repetimos para todas as classes:

Tabela 4: Bag-of-Words

doc	features	label
D4	1 0 0 1 0 0 1 1 0	C2
D5	1 0 0 1 1 0 1 1 1	C2
Soma	2 0 0 2 1 0 2 2 1	

UFABC

DCDistance

Com isso temos os vetores representantes de cada classe. Note que a intensidade de cada elemento do vetor representa a importância daquele termo na classe observada.

Tabela 5: Vetores representativos da classe C1 e C2

3 3 2 2 1 2 0 0 0	C1
2 0 0 2 1 0 2 2 1	C2

UFABC

DCDistance

Calculamos a distância entre cada documento para cada uma das classes (nesse exemplo utilizamos a distância euclidiana). A primeira coluna da tabela abaixo representa a distância entre cada documento e a classe C1, a segunda coluna representa a distância para a classe C2:

Tabela 6: Novos atributos

doc	features	label
D1	4.24 4.12	C1
D2	3.74 3.60	C1
D3	3.46 3.87	C1
D4	5.00 2.44	C2
D5	5.00 2.00	C2

DCDistance

Com isso temos uma base de dados com dimensão bem reduzida em relação a original.

Em testes práticos, as reduções obtidas foram de mais de 99% da dimensão original enquanto o resultado obtido por uma simples Regressão Logística foram iguais ou melhores do que algoritmos avançados de classificação aplicados na representação original.

DCDistance - Algoritmo

Dado *X_train*, *X_test* as bases de treino e teste, respectivamente, já em representação vetorial:

```
from sklearn.metrics.pairwise import pairwise_distances

representantes = np.zeros((klasses.shape[0], X_train.shape[1]))

for i, k in enumerate(klasses):
    representantes[i,:] = X_train[yTrain==k,:].sum(axis=0)
    DCD_train = pairwise_distances(X=X_train,
                        Y=representantes, metric='cosine')
    DCD_test  = pairwise_distances(X=X_test,
                        Y=representantes, metric='cosine')
```

UFABC

CARFRE - Categorical Average Rating Feature Reduction

Sistemas de Recomendação

Uma aplicação popular atualmente na área de Aprendizado Supervisionado.

Tenho (com algumas variações):

- Uma matriz binária representando dados categóricos sobre meus produtos.
- Uma matriz binária representando dados categóricos sobre meus clientes.
- Uma matriz numérica e esparsa associando notas dos meus clientes aos produtos que eles consumiram.

Meu objetivo é aprender um $f(u, i) = \hat{r}_{u,i}$ que retorne uma predição da nota que o usuário u atribuirá ao item i.

Também pode ser uma função de probabilidade do usuário aceitar comprar aquele item.

Temos isso:

	Filmes									
	Ação						*Comédia*			
Clientes	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
A	5				4	4			5	4
B	5		5	5	5		2	1		2
C		5	5	5		5	1	2	2	
D		5	5	5		5	1	3	2	
E	1		2	2		1	5	5	5	
F	2		1	1	2		5	5		5
G	2	1			2	1			5	5

E queremos isso:

	Filmes									
	Ação						*Comédia*			
Clientes	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
A	5	5	4	5	4	4	4	3	5	4
B	5	5	5	5	5	5	2	1	1	2
C	5	5	5	5	5	5	1	2	2	3
D	5	5	5	5	5	5	1	3	2	2
E	1	1	2	2	1	1	5	5	5	5
F	2	1	1	1	2	2	5	5	4	5
G	2	1	2	1	2	1	5	4	5	5

Queremos uma função que retorne um valor numérico entre 1 e 5. Que técnica devemos utilizar?

Algoritmo de Regressão!!

Mas como adaptar um Algoritmo de Regressão para nosso problema?

Regressão por Item

Podemos definir que todos os valores ausentes tem o valor 0.

Criamos um modelo de regressão para cada item, fazendo com que cada objeto seja representado pelas notas dada pelo usuário a todos os outros itens.

Como um valor ausente multiplicado pelo peso correspondente é 0, ele não terá influência no resultado final, mas os ajustes dos pesos será influenciado por esses valores.

Regressão por Item

Se existe um certo item que é bem determinante para a nota do item de interesse, o algoritmo funciona muito bem.

Exemplo: quem atribuiu nota alta para *Star Wars* atribuirá nota alta para os outros filmes da série.

Todos os outros atributos podem ter um peso 0 associado.

Regressão por Usuário

Similar ao anterior, só que agora temos um modelo por usuário.

Cada objeto é representado pelas notas dadas aos itens pelos outros usuários.

Regressão por Usuário

Problemas:

- Em uma empresa de médio porte, temos milhões de itens e usuários. A dimensão na ordem de milhões demanda amostras na casa de centenas de milhões.
- As matrizes de notas geralmente são muito esparsas, a base de dados contém informação insuficiente para gerar um modelo adequado.

Valores Agregados

Uma forma de aliviar esse problema é representando cada objeto da base como valores agregados, então dados um usuário u e um item i, desejamos obter:

$$\hat{r}_{u,i} = w_u \cdot \mu_u +_i \cdot \mu_i + w \cdot \mu,$$

com μ_u, μ_i, μ sendo as médias das notas dadas pelo usuário u, a média das notas dadas para o item i e as médias de todas as notas, e w_u, w_i, w sendo os pesos do modelo de regressão.

Valores Agregados

Esse modelo tem a vantagem de possuir apenas 3 variáveis, reduzindo a demanda por muitas amostras para gerar um modelo de regressão.

Além disso, os valores nulos não são levados em consideração durante a média, não influenciando no ajuste dos pesos.

Valores Agregados

Apesar disso, esse modelo linear é uma visão muito simplista do processo de decisão utilizado pelos usuários na escolha de um item.

Os usuários tendem a tomar decisões baseadas não apenas nas notas mas também as características dos produtos e pessoais.

CARFRE - Categorical Average Rating Feature Reduction

O CARFRE foi criado para extrair valores agregados utilizando o contexto dos dados categóricos da descrição dos itens e dos usuários.

Cada objeto da base de dados contém atributos referentes ao usuário e ao item que deseja obter a predição. Esses atributos são valores agregados correspondentes a cada um dos atributos categóricos considerados.

CARFRE - Categorical Average Rating Feature Reduction

Basicamente a predição das notas se dará através das informações de média que usuários da mesma idade deu a esse item, média que filmes desse mesmo gênero recebeu desse usuário, etc.

CARFRE - Categorical Average Rating Feature Reduction

Vamos tomar como exemplo a seguinte base de dados, cada linha representa a descrição de um usuário combinado com a descrição de um filme:

	uid	**gender**	**age**	**iid**	**year**	**genre**	**r**
x_1	1	F	25	10	1998	action	5
x_2	1	F	25	35	1982	drama	2
x_3	2	M	40	100	1965	comedy	3
..	..	..	..	..	..	..	
x_n	1000	M	18	35	1982	drama	5

UFABC

CARFRE - Categorical Average Rating Feature Reduction

Após a transformação gerada pelo CARFRE, teremos uma base de dados no seguinte formato:

	μ_**uid**	μ_**gender**	μ_**age**	μ_**iid**	μ_**year**	μ_**genre**	**r**
x_1	4.5	4.3	4.2	3.9	4.1	3.8	5
x_2	4.5	4.3	4.2	4.3	3.6	4.65	4
x_3	3.7	3.9	3.85	3.75	4,4	3.5	3
..	..	..	..	..	..	..	
x_n	4.2	3.9	3.7	4.3	3.6	4.65	5

UFABC

CARFRE - Categorical Average Rating Feature Reduction

Cada tupla (usuário, filme) é representada pelas informações das médias de:

- Notas dada por aquele usuário.
- Notas dada pelos usuários do mesmo gênero.
- Notas dada por usuários da mesma idade.
- Notas dada aos filmes do mesmo ano.
- Notas dada aos filmes do mesmo gênero.

CARFRE - Categorical Average Rating Feature Reduction

Esse tipo de transformação permite uma informação mais rica sobre o comportamento dos usuários e dos itens no modelo de tomada de decisão de notas.

Além disso, a dimensionalidade da base de dados se mantém baixa em relação ao uso das variáveis originais.

Os resultados desse algoritmo se mostraram competitivos com os algoritmos estado-da-arte da literatura. Embora não obtenha os melhores resultados, esse algoritmo consegue chegar perto do melhor resultado em menos tempo e com menos uso de memória.

CARFRE - Categorical Average Rating Feature Reduction

Embora obtenha bom resultado com um modelo único e global de regressão, gerar um modelo por usuário obtém resultados melhores e com possibilidade de extração de informação extra.

Interpretabilidade

Considere o seguinte modelo gerado para um dos usuários:

$$r_{u,i} = 0.833 \cdot \mu\_uid - 0.685 \cdot \mu\_gender + 0.810 \cdot \mu\_age$$
$$-0.062 \cdot \mu\_job + 0.874 \cdot \mu\_mid - 0.015 \cdot \mu\_year$$
$$+0.055 \cdot \mu\_main\_genre$$

UFABC

Interpretabilidade

Os fatores que mais contribuem para a definição das notas é a média das notas do usuário (critério pessoal), as notas da mesma faixa etária e a média das notas do filme em questão.

$$r_{u,i} = \mathbf{0.833} \cdot \mu\_\mathbf{uid} - 0.685 \cdot \mu\_gender + \mathbf{0.810} \cdot \mu\_\mathbf{age}$$
$$-0.062 \cdot \mu\_job + \mathbf{0.874} \cdot \mu\_\mathbf{mid} - 0.015 \cdot \mu\_year$$
$$+0.055 \cdot \mu\_main\_genre$$

Interpretabilidade

Isso significa que cada usuário possui um critério próprio de avaliação, podendo ser mais ou menos exigente.

$$r_{u,i} = \mathbf{0.833} \cdot \mu\_\mathbf{uid} - 0.685 \cdot \mu\_gender + \mathbf{0.810} \cdot \mu\_\mathbf{age}$$
$$-0.062 \cdot \mu\_job + \mathbf{0.874} \cdot \mu\_\mathbf{mid} - 0.015 \cdot \mu\_year$$
$$+0.055 \cdot \mu\_main\_genre$$

UFABC

Interpretabilidade

Além disso, usuários da mesma faixa de idade tendem a ter os mesmo gosto para filmes. Crianças obviamente vão dar notas positivas para filmes infantis, enquanto adolescentes podem preferir comédia ou filmes de ação.

$$\begin{aligned} r_{u,i} = \mathbf{0.833} \cdot \mu\_\mathbf{uid} - 0.685 \cdot \mu\_gender + \mathbf{0.810} \cdot \mu\_\mathbf{age} \\ -0.062 \cdot \mu\_job + \mathbf{0.874} \cdot \mu\_\mathbf{mid} - 0.015 \cdot \mu\_year \\ +0.055 \cdot \mu\_main\_genre \end{aligned}$$

UFABC

Interpretabilidade

A nota média recebida pelo filme é um aspecto importante da decisão, uma média alta implica em um filme que atraiu a atenção do público geral.

$$r_{u,i} = \mathbf{0.833} \cdot \mu\_\mathbf{uid} - 0.685 \cdot \mu\_gender + \mathbf{0.810} \cdot \mu\_\mathbf{age} \\ -0.062 \cdot \mu\_job + \mathbf{0.874} \cdot \mu\_\mathbf{mid} - 0.015 \cdot \mu\_year \\ +0.055 \cdot \mu\_main\_genre$$

UFABC

Interpretabilidade

Um outro aspecto interessante é o peso negativo que o gênero do usuário tem na nota, isso significa que o gosto desse usuário em particular tem uma tendência oposta dos da média dos usuários do mesmo gênero.

$$r_{u,i} = 0.833 \cdot \mu\_uid - \mathbf{0.685} \cdot \mu\_\mathbf{gender} + 0.810 \cdot \mu\_age$$
$$- 0.062 \cdot \mu\_job + 0.874 \cdot \mu\_mid - 0.015 \cdot \mu\_year$$
$$+ 0.055 \cdot \mu\_main\_genre$$

UFABC

Interpretabilidade

Isso pode ter diversas explicações: conta compartilhada, usuário ser de uma faixa etária menos predominante, etc.

$$r_{u,i} = 0.833 \cdot \mu\_uid - \mathbf{0.685} \cdot \mu\_\mathbf{gender} + 0.810 \cdot \mu\_age$$
$$-0.062 \cdot \mu\_job + 0.874 \cdot \mu\_mid - 0.015 \cdot \mu\_year$$
$$+0.055 \cdot \mu\_main\_genre$$

UFABC

Próxima Aula

Na próxima aula aprenderemos sobre redução de dimensionalidade e extração de atributos, mais especificamente os algoritmos:

- PCA
- DCDistance
- CARFRE

UFABC

Atividade 08

Complete os Laboratórios:

Dimensionality_Reduction_Exercises.ipynb

UFABC