

# Desafio Álgebra Linear - Notas de perfumes

Thalita Carvalho Routh

May 2023

## 1 Introdução

Este projeto consiste na análise de mil componentes de notas de fragrâncias em inglês, a fim de identificar correlações de similaridade e, assim, sugerir outros perfumes com características semelhantes.

Os dados utilizados neste trabalho estão disponíveis na plataforma Kaggle, no conjunto de dados chamado "Perfume Recommendation Dataset", criado pelo usuário Nandini Bansal. É possível fazer o download do conjunto de dados pelo link: <https://www.kaggle.com/datasets/nandini1999/perfume-recommendation-dataset?resource=download>.

## 2 Fundamentação Teórica

### 2.1 Cálculo da Similaridade do Cosseno

A fórmula utilizada para o cálculo do cosseno da similaridade é:

$$\text{similarity}(A, B) = \frac{A \cdot B}{\|A\| \cdot \|B\|}$$

### 2.2 Explicação da Similaridade do Cosseno

A similaridade do cosseno é um conceito fundamental em áreas como aprendizado de máquina, processamento de linguagem natural e recuperação de informações. Trata-se de uma medida que avalia a proximidade entre dois vetores por meio do cálculo do ângulo formado por eles em um espaço n-dimensional, utilizando a função de similaridade do cosseno. Por ser uma métrica amplamente utilizada, é importante compreender seus fundamentos teóricos e práticos para aplicá-la de forma eficaz em diferentes contextos.

## 3 Ferramentas e Metodologias Adotadas

Foram utilizadas as seguintes ferramentas e bibliotecas para análise, tratamento:

- Biblioteca Pycharm (PYTHON)
- Biblioteca Pandas (pd)
- Colab

- Biblioteca Tfidfvectorizer
- Biblioteca Sklearn
- Biblioteca Cosine\_similarity
- Biblioteca Mathplotlib as plt
- Látex

## 4 Código Aplicado em Python

Código utilizado para a verificação da similaridade:

```
import pandas as pd
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.metrics.pairwise import cosine_similarity
import math

# carregar dados do arquivo CSV
dados = pd.read_excel('excel_perfume.xlsx')

# selecionar as colunas relevantes para comparação
dados_selecionados = dados[['Name', 'Notes']]

# transformar os dados em um formato de texto unificado para vetorização
dados_texto = dados_selecionados.apply(lambda x: ' '.join(x.astype(str)), axis=1)

# criar o vetorizador TF-IDF
vetorizador = TfidfVectorizer()

# vetorizar as descrições dos produtos
vetores = vetorizador.fit_transform(dados_texto)

# exemplo das nota perfume fornecido pelo usuário
nota_perfume = input('Digite a nota de perfume que mais te agrada: ').split()

# transformar o exemplo em um vetor numérico
nota_perfume_vetor = vetorizador.transform([' '.join(map(str, nota_perfume + [''] * (len(dados_selecionados.columns) - len(nota_perfume))))])

# calcular a similaridade do cosseno entre a nota_perfume fornecido pelo usuário e outra nota_perfume
similaridades = cosine_similarity(nota_perfume_vetor, vetores)

# obter os índices da nota_perfume ordenada por ordem decrescente de similaridade
indices_similares = similaridades.argsort()[0][::-1]

# selecionar os 10 perfumes com notas mais similares,
top_similares = []
for i in indices_similares:
    if dados.loc[i, 'Name'] != nota_perfume:
        top_similares.append(i)
    if len(top_similares) >= 10:
        break

# adicionar as colunas de similaridade do cosseno e ângulo do cosseno
dados_selecionados['Cosine Similarity'] = similaridades[0]
dados_selecionados['Cosine Angle'] = [math.degrees(math.acos(similarity)) for similarity in similaridades[0]]

# imprimir os 10 perfumes mais similares com as colunas adicionais
print("A nota_perfume mais similar a", ' '.join(map(str, nota_perfume)), "são:")
display(dados_selecionados.loc[top_similares, ['Name', 'Notes', 'Cosine Similarity', 'Cosine Angle']])
```

### 4.1 Dados Qualitativos

#### 4.1.1 Retirada de Stops Words

Inicialmente realizamos a limpeza das palavras (Stops Words), deixando apenas as palavras importantes.

### 4.1.2 Texto em Vetor


Criar um vetor numérico para cada frase. Para isso, é necessário selecionar a lista contendo as frases ou a coluna correspondente do dataframe e inseri-la na função responsável pela geração do vetor.

### 4.1.3 Similaridade

É preciso calcular a similaridade do cosseno entre os dois conjuntos de vetores. A função encarregada dessa tarefa recebe a variável mencionada anteriormente e produz como resultado uma matriz de similaridade que mostra a relação entre cada par de frases, de forma semelhante à matriz de correlação.

### 4.1.4 Finalização

Depois de gerada a matriz de similaridade, é preciso realizar uma busca na matriz, onde o item desejado pode ser selecionado pelo índice ou pela coluna correspondente. Ao selecionar o item, é possível obter uma lista dos produtos e sua respectiva similaridade do cosseno em relação ao item buscado, apresentando portanto o ângulo do cosseno.

	Name	Notes	Cosine Similarity	Cosine Angle	
709	Holy Water Eau de Parfum	frankincense, sandalwood, orange, orange blos...	0.459755	62.628677	
693	No. 1 Tonic Blanc Eau de Parfum	bergamot, mandarin, orange, neroli, orange fl...	0.371928	68.165440	
806	Sanguine Eau de Parfum	Mediterranean blood orange, orange rind, citr...	0.364921	68.597276	
93	Gold II Sahara Parfum Extrait	Saffron, orange, elemi, rose, orange blossom,...	0.364142	68.645207	
731	Sumo Wrestler Eau de Parfum	Orange, Eucalyptus, Anise, Cinnamon, Heliotro...	0.336710	70.323462	
558	Kalan Eau de Parfum	Blood Orange, Black Pepper, Lavender, Solar n...	0.335371	70.404915	
343	Jardin du Poete Eau de Toilette	Green orange, sweet orange, bitter orange, gr...	0.325212	71.021602	
812	Kolonya Eau de Toilette	Bergamot, Neroli, Petitgrain, Clove, Nutmeg, ...	0.315771	71.592606	
873	Orange X Santal Eau de Parfum	Bitter orange, Egyptian basil, natural oakmos...	0.313995	71.699845	
884	Cap Neroli Eau de Toilette	Petitgrain, orange, mandarin, rosemary, mint,...	0.313010	71.759270	