import pandas as pd

dados= pd.read\_csv('dataset/tracking.csv')

dados.head(5)

comprou=dados.comprou

dados.drop(columns='comprou',inplace=True)

comprou.head()

from sklearn.model\_selection import train\_test\_split

treino, teste, classe\_treino, classe\_teste = train\_test\_split(dados, comprou,random\_state=42, test\_size=0.3)

from sklearn.svm import LinearSVC

modelo= LinearSVC()

Modelagem 3D

Caio

Lafaiete

João

https://dontpad.com/semanamltarde

https://lamfo-unb.github.io/2017/07/27/tres-tipos-am/

https://youtu.be/Qw4dV8PCigw?si=bMU5RoQSpFpPRyIh

padrão: 75% 25%

https://linktr.ee/semanadatascience

https://dontpad.com/semanamltarde

https://youtube.com/playlist?list=PLGwqoftZstLZUQGt3GeLpI-QAZaT8ccVG&si=pAYz3W0nrSq1F-PB

{#Aula de Hoje#}

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

[[[[[ Parte 1 - Programação Orientada a Objetos Avançada ]]]]]]]]

(notebook) - executar

no final desafio

ml\_pln\_01.ipynb

https://colab.research.google.com/drive/1dh5IwS8fC56gqHPbESLADXMpQVf99Xqw?usp=sharing

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Parte 2 - Machine Learning com PLN e Sacola de Palavras

(notebook) - executar

no final desafio

ml\_pln\_02.ipynb

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

df\_resenhas['clasificação'] = classificacao

df\_resenhas.head()

https://colab.research.google.com/drive/1GGVFaVqNxeng1e2XEb4\_I2TyqF3s6X8\_?usp=sharing

link da imagem

https://github.com/romulosilvestre/semanamachinelearning-datasets/blob/main/imagem.PNG

from IPython.display import Image, display

# URL da imagem

url = "https://raw.githubusercontent.com/romulosilvestre/semanamachinelearning-datasets/main/imagem.PNG"

# Exibir a imagem

display(Image(url=url))

toarray() - conversão considerando uma matriz densa (inclui tudo)

sparse.from\_matrix()- cria DataFrame sparso diretamente da matriz sparsa.

Vantagem:

1. Eficiencia

2. Memória

3. Especializa em muitos zeros

4. Performance

link do notebook

https://colab.research.google.com/drive/1dOTQ\_GhA\_AeeaE0\_9RQhYFH6SDfvy-5O?usp=sharing

{# link do henrique #}

https://github.com/henriqueserafin/projeto-diario

html diario bordo

###########################

<!DOCTYPE html>

<html lang="pt-BR">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Formulário de Dados</title>

<style>

body {

display: flex;

justify-content: center;

align-items: center;

height: 100vh;

margin: 0;

font-family: Arial, sans-serif;

background-color: #f0f0f0;

}

.container {

background-color: white;

padding: 20px;

border-radius: 8px;

box-shadow: 0 4px 10px rgba(0, 0, 0, 0.1);

width: 300px;

text-align: center;

}

h1 {

color: #333;

}

label {

display: block;

margin: 10px 0 5px;

text-align: left;

}

input[type="datetime-local"],

textarea {

width: 100%;

padding: 8px;

margin-bottom: 15px;

border: 1px solid #ccc;

border-radius: 4px;

}

button {

background-color: #007bff;

color: white;

border: none;

border-radius: 4px;

padding: 10px;

cursor: pointer;

transition: background-color 0.3s;

}

button:hover {

background-color: #0056b3;

}

</style>

</head>

<body>

<div class="container">

<h1>Envio de Dados</h1>

<form action="/" method="post">

<label for="datahora">Data e Hora:</label>

<input type="datetime-local" id="datahora" name="datahora" required>

<label for="texto">Texto:</label>

<textarea id="texto" name="texto" required></textarea>

<button type="submit">Enviar</button>

</form>

</div>

</body>

</html>

-------------------------------------------------------

@app.route('/diariobordo')

def diariobordo():

return render\_template('diariodebordo.html')

-------------------------------------------------------------------------------------------------------------------------

diario de bordo PAINEL PROFESSOR html (victor)

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Painel do Professor</title>

<style>

/\* Resetando o estilo padrão \*/

\* {

margin: 0;

padding: 0;

box-sizing: border-box;

}

body {

font-family: 'Roboto', sans-serif;

background: linear-gradient(135deg, #f7f7f7, #d6d6d6);

min-height: 100vh;

margin: 0;

display: flex;

justify-content: center;

align-items: center;

}

.container {

background-color: #ffffff;

padding: 30px;

border-radius: 12px;

box-shadow: 0 6px 18px rgba(0, 0, 0, 0.15);

width: 400px;

text-align: center;

transition: transform 0.3s ease;

}

.container:hover {

transform: translateY(-8px);

}

h1 {

font-size: 26px;

color: #2c3e50;

margin-bottom: 20px;

text-transform: uppercase;

letter-spacing: 1px;

font-weight: bold;

}

ul {

list-style: none;

padding: 0;

}

li {

margin-bottom: 15px;

}

a {

text-decoration: none;

font-size: 18px;

color: white;

background-color: #34495e;

padding: 12px 25px;

border-radius: 5px;

transition: background-color 0.3s ease, transform 0.3s ease;

box-shadow: 0 4px 12px rgba(0, 0, 0, 0.1);

display: inline-block;

margin-bottom: 10px; /\* Espaçamento ajustado entre os botões \*/

}

a:hover {

background-color: #2c3e50;

transform: scale(1.03);

}

a:active {

background-color: #1c2833;

}

/\* Adicionando responsividade \*/

@media (max-width: 400px) {

.container {

width: 100%;

padding: 20px;

}

h1 {

font-size: 22px;

}

a {

font-size: 16px;

padding: 10px 20px;

}

}

</style>

</head>

<body>

<div class="container">

<h1>Painel do Professor</h1>

<ul>

<li><a href="/cadastro">Cadastrar Aluno</a></li>

<li><a href="/alunos">Listar Todos os Alunos</a></li>

<li><a href="/configuracoes">Configurações</a></li>

</ul>

</div>

</body>

-----------------------------------------------------

https://github.com/andrerodriguesfeitosa/schooltracker