



2ª série

Inteligência Artificial

Reconhecimento de Padrões

Rildo Oliveira



25/04/2024

OBJETO DO CONHECIMENTO: Reconhecimento de Padrões

HABILIDADE:

PCRP03 - Identificar, entender e explicar em que situações o computador pode ou não ser utilizado para solucionar um problema.

EF05HI06 - Comparar o uso de diferentes linguagens e tecnologias no processo de comunicação e avaliar os significados sociais, políticos e culturais atribuídos a elas.

OBJETIVOS:

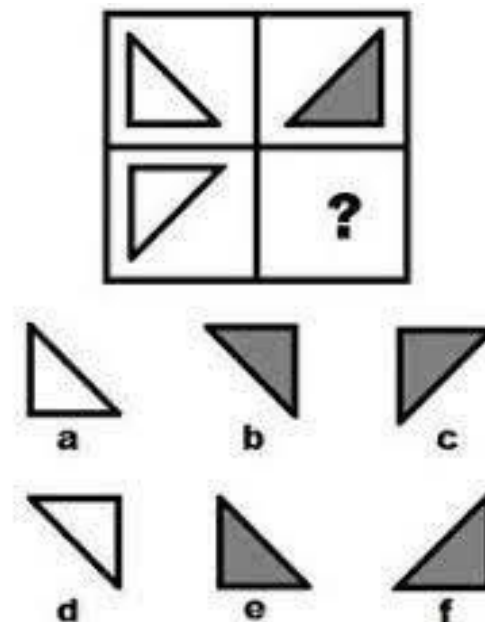
Compreender os desafios e as aplicações do reconhecimento de padrões em problemas do mundo real, como análise de imagens médicas e vigilância por vídeo.

Investigar como os algoritmos de reconhecimento de padrões são aplicados em diferentes contextos.

DA TEORIA À PRÁTICA: Uso de imagens, texto e conceitos para um melhor entendimento do tema abordado.

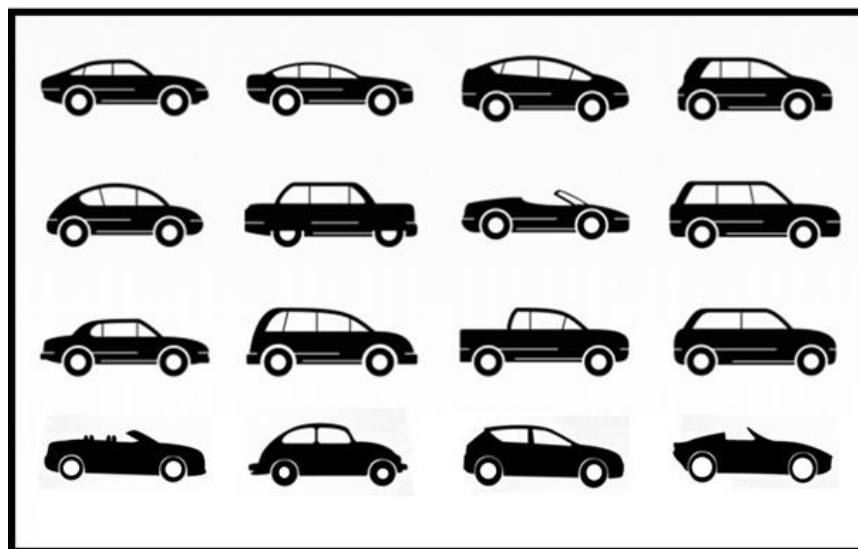
Reconhecimento de padrões

Trata-se de um sistema capaz de organizar informações com base em conjuntos específicos de dados servido para a classificação e interpretação de dados.



Definição e importância dos padrões

Um padrão pode ser definido como um conjunto de características recorrentes ou a descrição de um problema comum para o qual existe uma solução reutilizável. Essas regularidades são fundamentais para a ia, pois proporcionam estrutura e significado aos dados.

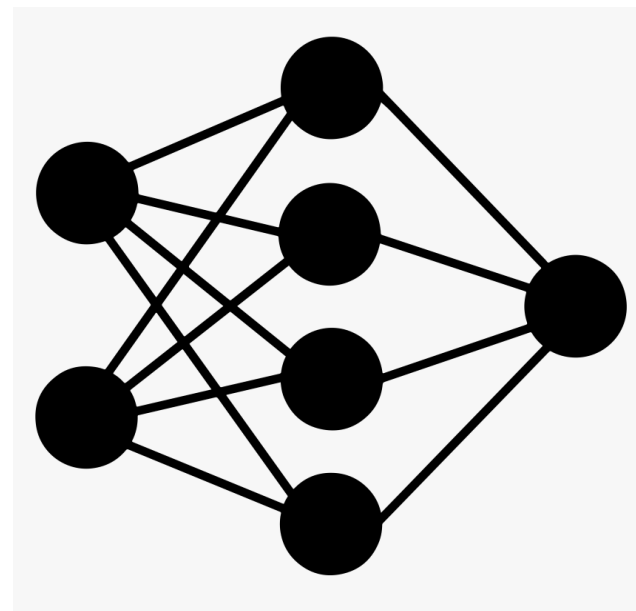


Aplicações e impacto na rotina

As aplicações do reconhecimento de padrões são vastas e impactam uma variedade de setores, desde medicina e finanças até segurança e tecnologia. Por exemplo, na área médica, sistemas de ia utilizam o reconhecimento de padrões em imagens médicas para auxiliar no diagnóstico de doenças. Em finanças, algoritmos de detecção de fraudes baseados em padrões ajudam a proteger transações bancárias.

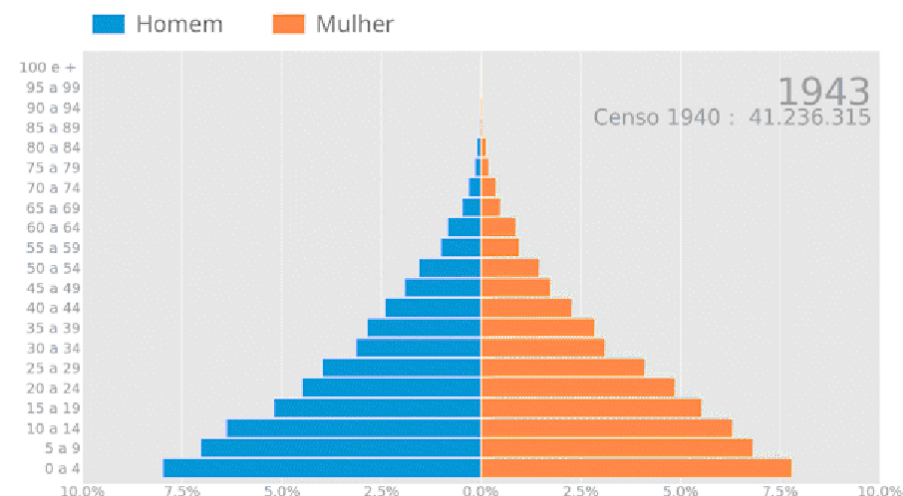
Aplicações e impacto na rotina

O reconhecimento de padrões vem capacitando sistemas a compreender e interpretar dados complexos de maneira eficiente, impulsionando a inovação e a automação em diversos campos.



Organização e classificação de dados

A inteligência artificial faz uso do reconhecimento de padrões para analisar conjuntos de dados, conhecidos como "conjuntos de treinamento", e organizá-los de acordo com padrões identificados. Esses padrões podem ser estabelecidos com base em conhecimento prévio ou em informações estatísticas derivadas de observações.



Tipos de reconhecimento

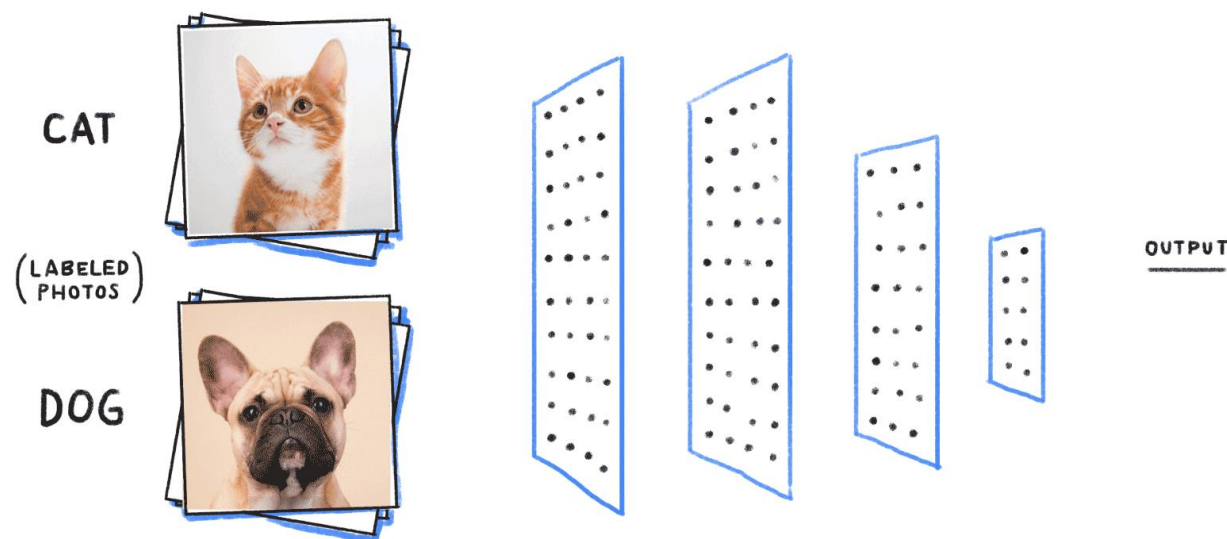
Existem dois principais tipos de reconhecimento de padrões:

Supervisionado: neste método, o algoritmo utiliza o conjunto de treinamento para classificar os dados de acordo com categorias pré-definidas.

Não supervisionado: ao contrário do método supervisionado, o reconhecimento não supervisionado não depende de categorias predefinidas. Em vez disso, o algoritmo cria novas categorias ou agrupamentos, explorando padrões intrínsecos nos dados.
















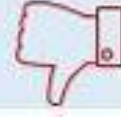
Etapa de treinamento e teste

Antes de aplicar o reconhecimento de padrões de forma efetiva, é importante realizar uma etapa de treinamento. Durante esta fase, o algoritmo é testado e ajustado para garantir que seja capaz de identificar os padrões desejados nos dados de entrada.



Filtragem da entrada

O processo começa com a coleta de informações através de sensores, que capturam as observações a serem classificadas ou descritas. Essa etapa é importante para eliminar dados desnecessários ou distorcidos, garantindo que apenas objetos relevantes sejam apresentados para o reconhecimento.

	Filme 1	Filme 2	Filme 3	Filme 4
Personagem 1				
Personagem 2				
Personagem 3				
Personagem 4				
Personagem 5				

Extração de características

Em seguida, entra em ação o mecanismo de extração de características, responsável por analisar os dados de entrada e extrair informações úteis para o processo de reconhecimento. Esta análise permite derivar características numéricas ou simbólicas das observações, que serão posteriormente utilizadas na classificação.

Classificação dos dados

O passo final do processo é a classificação dos dados, onde um esquema de classificação é aplicado para categorizar os objetos de acordo com seus padrões. Este esquema determina a que classe cada objeto pertence, com base nas características extraídas durante a etapa anterior.

Dados



Classificação



Organização sistêmica

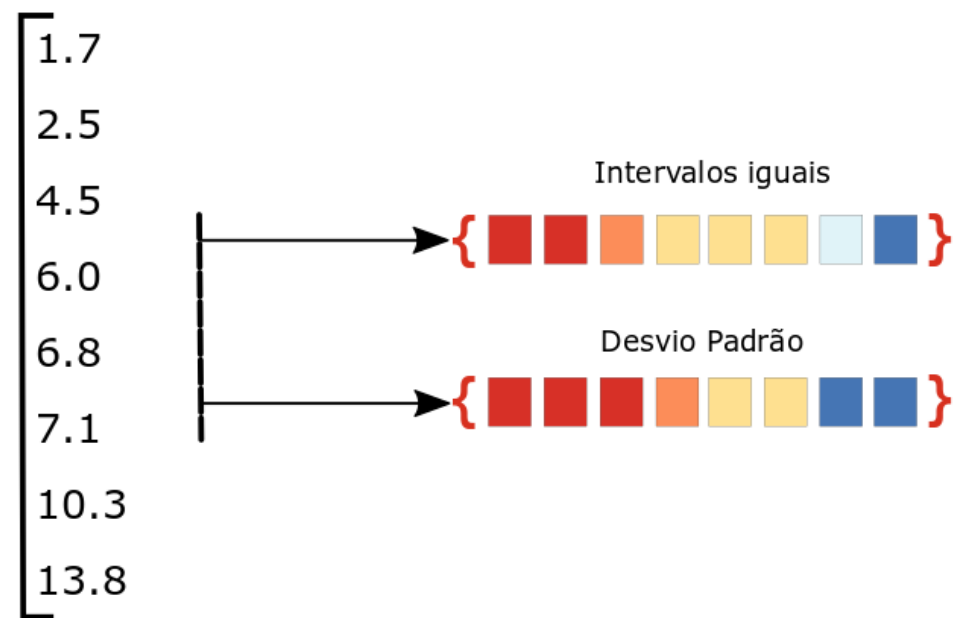


Apresentação
Gráfica



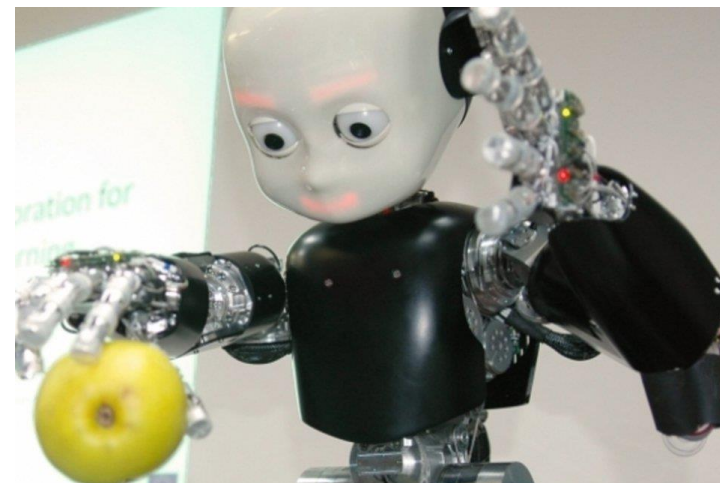
Desafios e considerações

Para que os classificadores funcionem de forma eficaz, é necessário encontrar um equilíbrio entre generalização e especificidade. Classificadores muito generalizados podem produzir categorizações vagas, enquanto classificadores muito específicos podem perder elementos relevantes devido à rigidez em relação a certos padrões.



Limitações em comparação com o reconhecimento humano

Embora o reconhecimento de padrões seja uma tarefa complexa para sistemas de ia, ainda não atingiu o mesmo nível de sofisticação encontrado no cérebro humano. A capacidade de reconhecimento humano é excepcionalmente avançada, com uma capacidade de adaptabilidade e generalização que os sistemas de ia ainda estão buscando alcançar.



Aplicações do reconhecimento de padrões

O reconhecimento de padrões possui uma ampla gama de aplicações em diversos campos, desde a identificação de rostos em fotografias até a leitura de códigos de barras em supermercados. Vamos explorar algumas dessas aplicações e como o processo é implementado em cada uma delas.

Reconhecimento facial

No reconhecimento facial, o processo envolve a identificação de características distintivas presentes em rostos humanos, como olhos, nariz e boca. O classificador busca por simetria e proporções para confirmar se a imagem corresponde a um rosto humano. Essa tecnologia é amplamente utilizada em sistemas de segurança, controle de acesso e identificação pessoal.



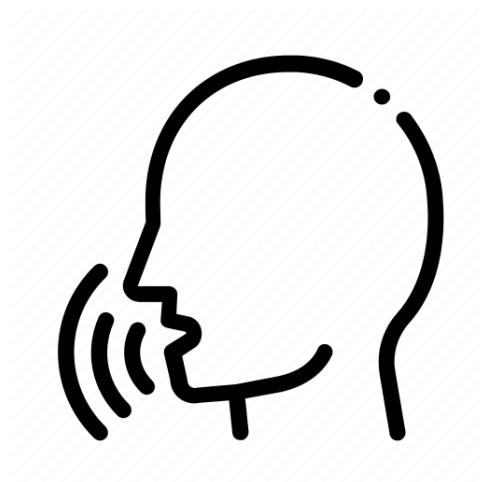
Sistemas de mensagens

Filtro antispam: um exemplo concreto de reconhecimento de padrões é o filtro antispam encontrado em serviços de e-mail. Ao analisar características como o número de destinatários e configurações pré-definidas, o sistema classifica eficientemente mensagens, separando-as em categorias relevantes ou indesejadas.



Reconhecimento de voz

No reconhecimento de voz, a fala é digitalizada e suas características espectrais são computadas. O classificador então busca por padrões fonéticos que correspondam a palavras ou sons específicos. Essa tecnologia é empregada em assistentes virtuais, sistemas de navegação por voz e reconhecimento de comandos em dispositivos eletrônicos.



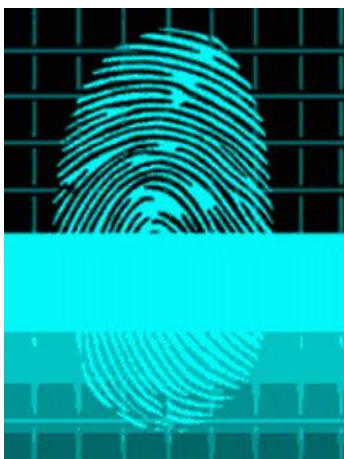
Leitura de código de barras

Na leitura de código de barras, o processo é mais direto: a máquina captura os dados do código, compara com os padrões em um banco de dados e exibe a informação associada ao código. Essa aplicação é essencial em supermercados, agilizando o processo de compra e controle de estoque.



Biometria e segurança

O reconhecimento de padrões é fundamental na biometria para identificação de indivíduos em meio à multidão, desbloqueio de dispositivos através de comandos de voz e autenticação biométrica em sistemas de segurança. Tecnologias como leitura de retina e impressão digital são exemplos de sua aplicação nesse contexto.

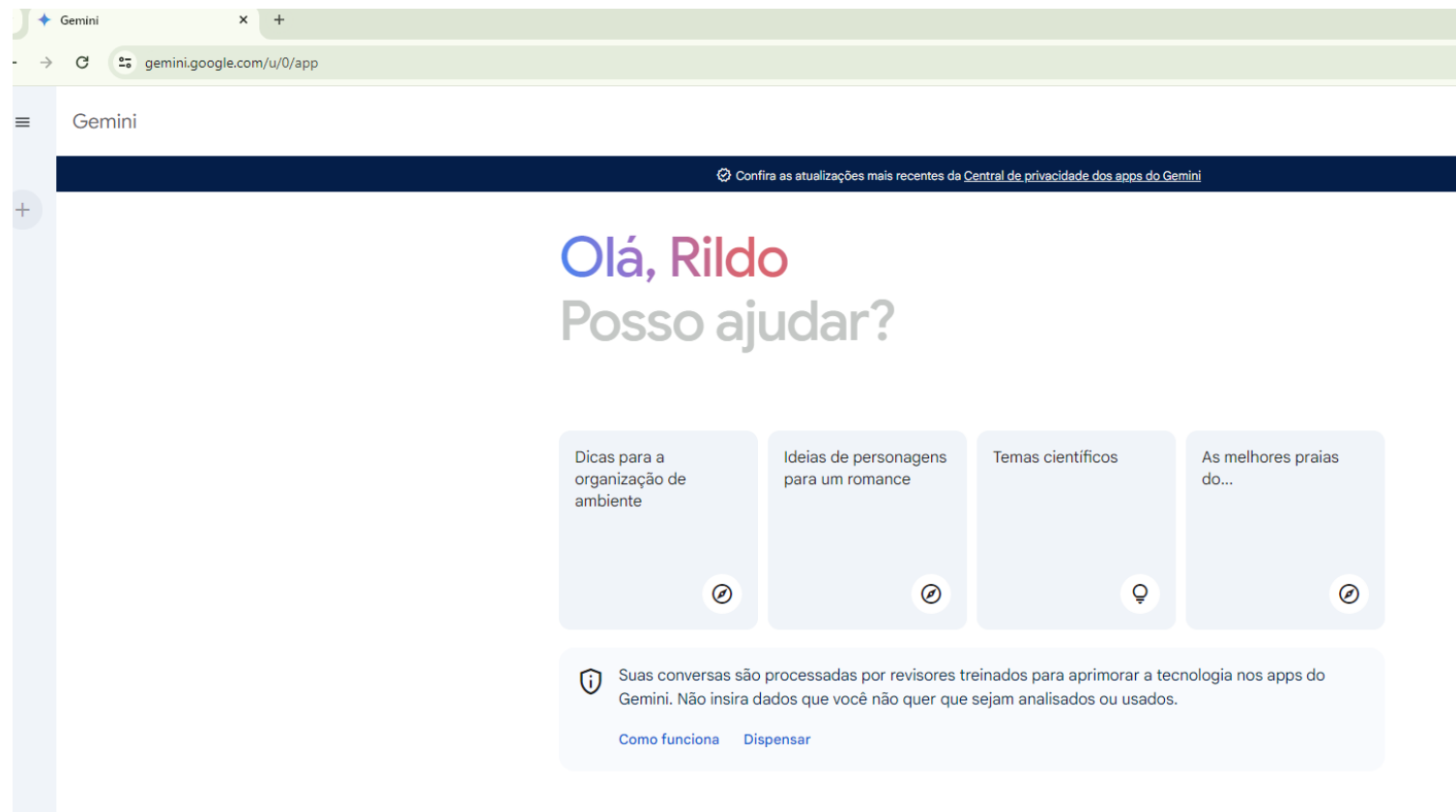


Aplicações em pesquisa científica

Além das aplicações no mundo físico, o reconhecimento de padrões é utilizado em pesquisa científica, como no estudo do genoma humano. Através de comparações de dados genéticos, os cientistas podem identificar a função de genes e entender melhor a biologia humana.

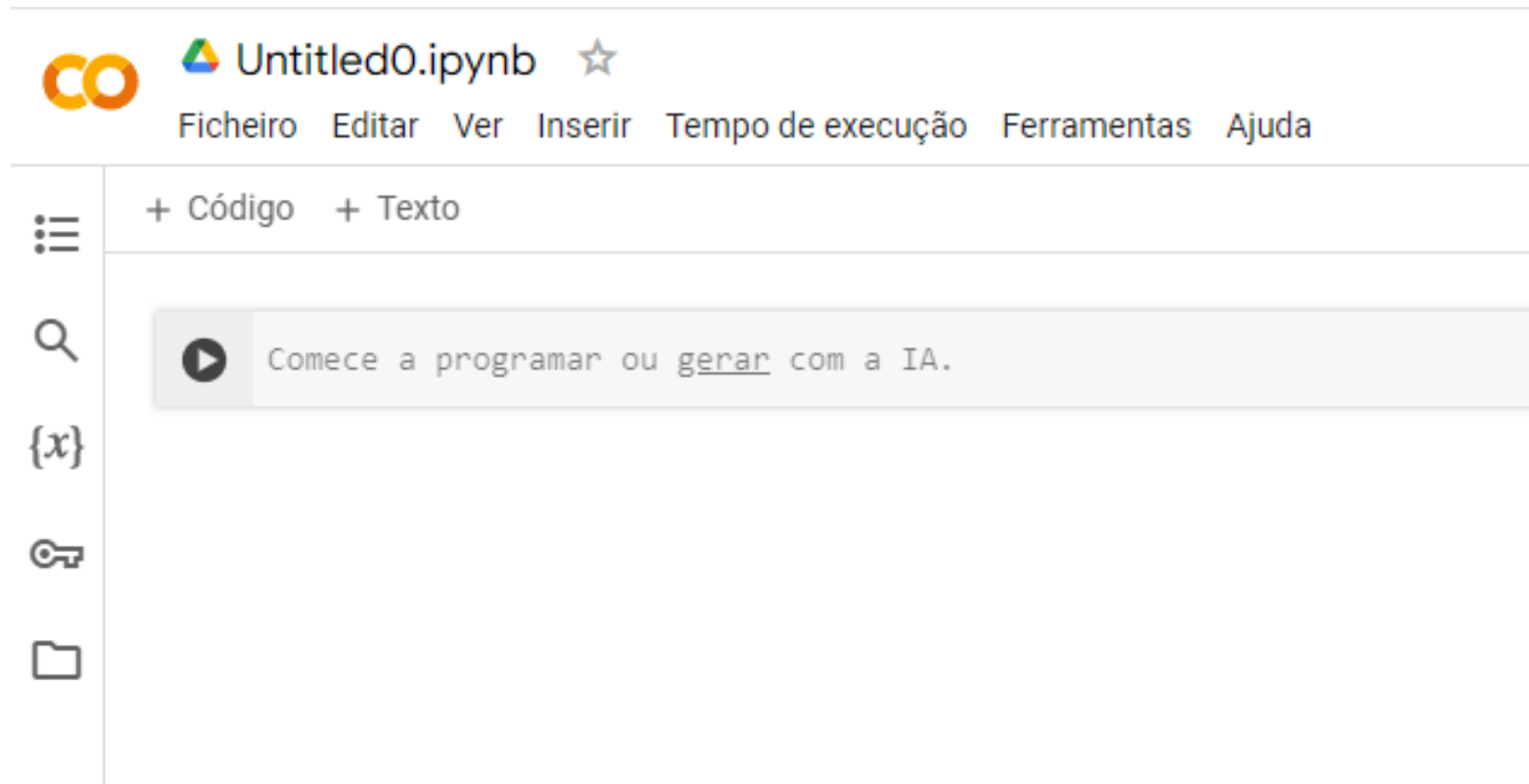


Gemini IA



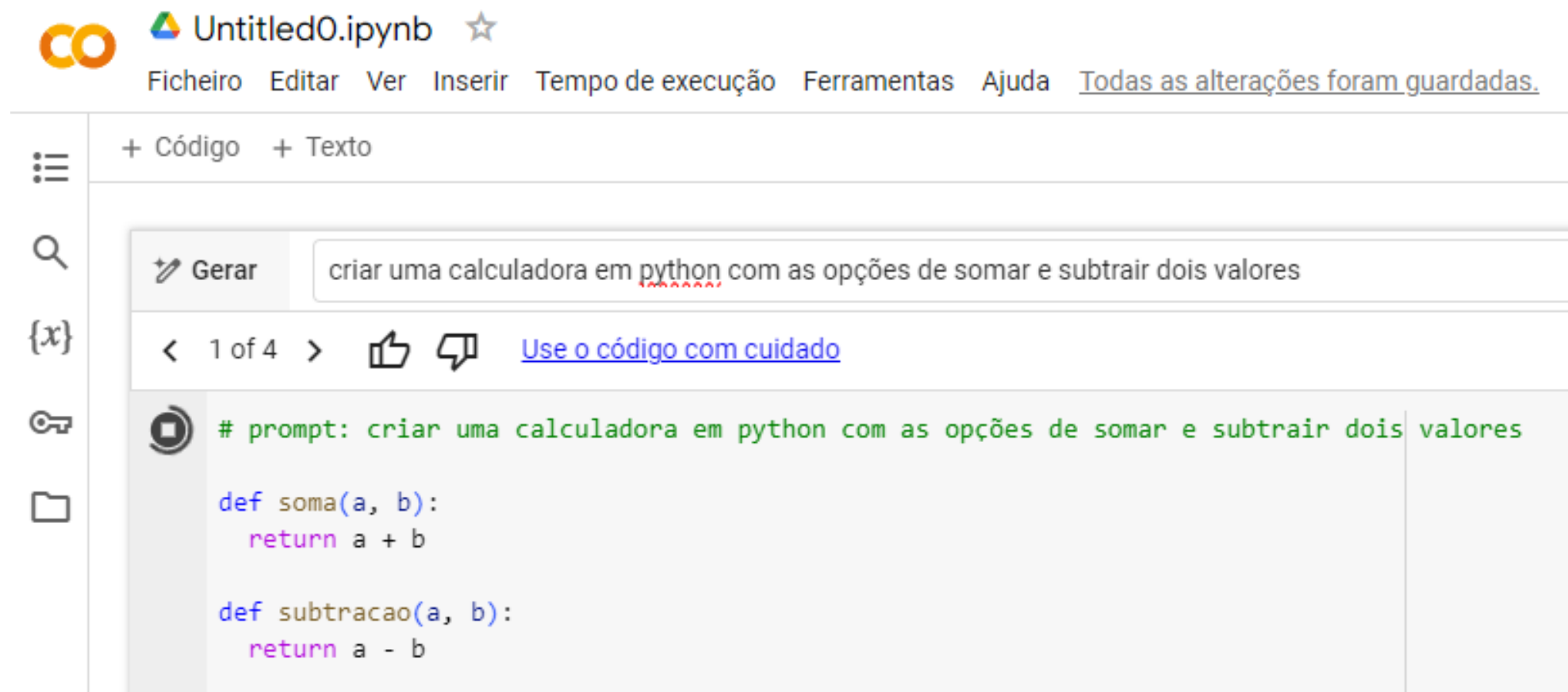
<https://gemini.google.com/u/0/app>

Colab



<https://colab.research.google.com/>

Colab




CO Untitled0.ipynb ☆

Ficheiro Editar Ver Inserir Tempo de execução Ferramentas Ajuda [Todas as alterações foram guardadas.](#)

+ Código + Texto

Gerar criar uma calculadora em python com as opções de somar e subtrair dois valores

< 1 of 4 >   [Use o código com cuidado](#)

```
# prompt: criar uma calculadora em python com as opções de somar e subtrair dois valores

def soma(a, b):
    return a + b

def subtracao(a, b):
    return a - b
```


Colab

Clique em ligar



Comentário



Partilhar



Ligar



Colab AI



Colab



co Aula de Deep MLP para caracteres escritos a mão.ipynb ☆

Ficheiro Editar Ver Inserir Tempo de execução Ferramentas Ajuda As alterações não serão guardadas.

+ Código + Texto Copiar para o Drive

```
[ ] (60000, 28, 28)
    (10000, 28, 28)

[ ] import matplotlib.pyplot as plt
    import random

def visualizaNumero(classe, X, y):
    qtImagens = 8
    print("Classe selecionada: ", classe)
    XSelecionados = X[y == classe]
    fig, axs = plt.subplots(nrows=1, ncols=qtImagens, figsize=(qtImagens,1))
    for c in range(qtImagens):
        posAleatoria = random.randint(0, len(XSelecionados) - 1)
        axs[c].imshow(XSelecionados[posAleatoria,:,:], cmap=plt.get_cmap("gray") )
        axs[c].axis("off")
```

https://colab.research.google.com/drive/1M5J_kgxQIMrNx8h5mmmcO2urqpL5orKQ?usp=sharing

Carregando os dados

✓ Carregando dados

- Essa é um base de dados embutida em keras

```
▶ import tensorflow as tf  
mnist = tf.keras.datasets.mnist
```

Executar célula (Ctrl+Enter)

A célula não foi executada nesta sessão

Treinando o modelo

✓ Dividindo para treinamento e teste

```
[ ] (XTrain, yTrain) , (XTest, yTest) = mnist.load_data()
```

Downloading data from <https://storage.googleapis.com/tensorflow/tf-keras-datasets/mnist.npz>
11493376/11490434 [=====] - 0s 0us/step

Visualizando



Visualizando os dados

- Compreensão inicial sobre os dados

✓
0 s



```
print(XTrain.shape)  
print(XTest.shape)
```


```
(60000, 28, 28)  
(10000, 28, 28)
```



```
import matplotlib.pyplot as plt  
import random
```

```
def visualizaNumero(classe, X, v):
```

Comparando a Base

✓
0 s `visualizaNumero(0, XTrain, yTrain)` Classe selecionada: 0`[] visualizaNumero(1, XTrain, yTrain)`

Comparando a Base

▼ Treinando

```
historico = model.fit(XTrainT, yTrainT, validation_split=0.15, epochs=5, batch_size=200, verbose=1, shuffle=1 )
```

... Epoch 1/5

255/255 [=====] - 6s 17ms/step - loss: 0.2980 - accuracy: 0.9145 - val_loss: 0.1336 - val_accuracy: 0.9629

Epoch 2/5

31/255 [==>.....] - ETA: 4s - loss: 0.1246 - accuracy: 0.9648

[+ Código](#)[+ Texto](#)

Resultado

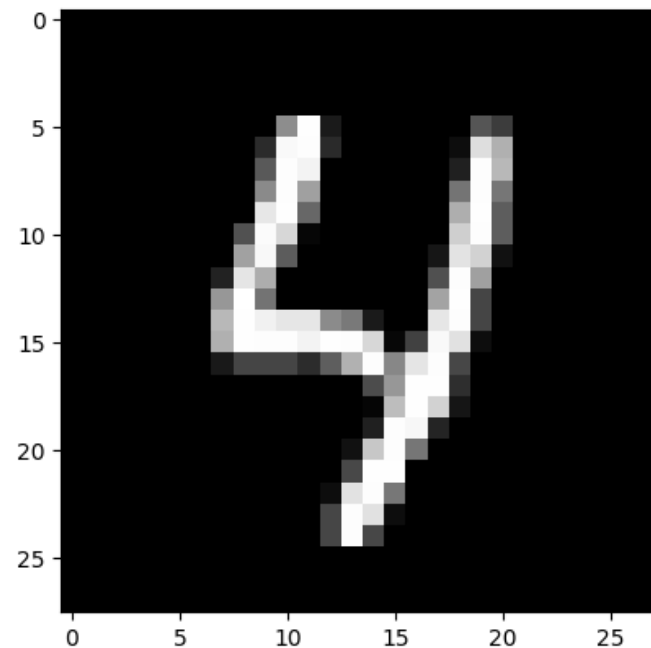


✓ Teste com uma imagem isoladamente

```
posAleatoria = random.randint(0, len(XTest) - 1)
print(posAleatoria)
imagem = XTest[posAleatoria,:,:]

plt.imshow(imagem,cmap=plt.get_cmap("gray"))
```

```
1980
<matplotlib.image.AxesImage at 0x7dbd7c920a60>
```



Referências Bibliográficas

1. Russell, S.; Norvig, P. (2016). "Artificial Intelligence: A Modern Approach". Pearson.
2. Nilsson, N. J. (2009). "The Quest for Artificial Intelligence: A History of Ideas and Achievements". Cambridge University Press.
3. McCarthy, J.; Minsky, M. L.; Rochester, N.; Shannon, C. E. (1955). "A Proposal for the Dartmouth Summer Research Project on Artificial Intelligence". AI Magazine, 27(4).
4. Kurzweil, R. (2005). "The Singularity Is Near: When Humans Transcend Biology". Viking Adult.



ATÉ A PRÓXIMA AULA!