

▼ Acesso às bases de dados da disciplina

Esse notebook tem como objetivo apenas testar o acesso aos conjuntos de dados que poderão ser utilizados no decorrer da disciplina disponibilizados no *GitHub* e *Kaggle*.

▼ Acesso pelo *GitHub*

Basta usar o comando abaixo para clonar o repositório de dados da disciplina disponível no *GitHub* pelo endereço:  
<https://github.com/malegopc/DSBD>.

+ Código

+ Texto


```
!git clone https://github.com/malegopc/DSBD
```

 Cloning into 'DSBD'...  
remote: Enumerating objects: 721, done.  
remote: Counting objects: 100% (264/264), done.  
remote: Compressing objects: 100% (150/150), done.  
remote: Total 721 (delta 140), reused 217 (delta 111), pack-reused 457 (from 1)  
Receiving objects: 100% (721/721), 14.79 MiB | 10.58 MiB/s, done.  
Resolving deltas: 100% (348/348), done.

▼ Ações da Google

Acesso ao arquivo "GOOG\_train.csv" contendo os valores das ações da Google no período de 02-01-2015 até 30-12-2019.

```
import pandas as pd
# lê arquivo de dados
df_google = pd.read_csv('/content/DSBD/Datasets/Ações_Google/GOOG_train.csv')
# mostra os dados
df_google
```



	Date	Open	High	Low	Close	Adj Close	Volume
0	2015-01-02	527.561584	529.815369	522.665039	523.373108	523.373108	1447500
1	2015-01-05	521.827332	522.894409	511.655243	512.463013	512.463013	2059800
2	2015-01-06	513.589966	514.761719	499.678131	500.585632	500.585632	2899900
3	2015-01-07	505.611847	505.855164	498.281952	499.727997	499.727997	2065000
4	2015-01-08	496.626526	502.101471	489.655640	501.303680	501.303680	3353500
...	...	...	...	...	...	...	...
1252	2019-12-23	1355.869995	1359.800049	1346.510010	1348.839966	1348.839966	883100
1253	2019-12-24	1348.500000	1350.260010	1342.780029	1343.560059	1343.560059	347500
1254	2019-12-26	1346.170044	1361.327026	1344.469971	1360.400024	1360.400024	667500
1255	2019-12-27	1362.989990	1364.530029	1349.310059	1351.890015	1351.890015	1038400
1256	2019-12-30	1350.000000	1353.000000	1334.020020	1336.140015	1336.140015	1050900

1257 rows × 7 columns

Próximas etapas:

Gerar código com df\_google


☒ Ver gráficos recomendados

New interactive sheet

▼ Tweets

Acesso ao conjunto de dados de aproximadamente 15.000 tweets sobre uma grande companhia aérea dos Estados Unidos.

```
# lê arquivo de dados
df_tweets = pd.read_csv('/content/DSBD/Datasets/Twitter/Tweets.csv')
# mostra os dados
df_tweets.head(3)
```



	tweet_id	airline_sentiment	airline_sentiment_confidence	negativereason	negativereason_confidence	airline	airline
0	570306133677760513	neutral	1.0000	NaN	NaN	Virgin America	
1	570301130888122368	positive	0.3486	NaN	0.0	Virgin America	
2	570301083672813571	neutral	0.6837	NaN	NaN	Virgin America	

Próximas etapas:

Gerar código com df\_tweets

 Ver gráficos recomendados

New interactive sheet

### Acesso pelo *Kaggle* (sem usar API tokens)

Basta executar o comando "wget" no endereço do link utilizado pelo site para fazer o download da base de dados hospedada no "Kaggle".

▼

#### Dogs vs Cats

- Dados de treino (dataset\_treino): 8.000 imagens = 4.000 imagens de cães e 4.000 imagens de gatos
- Dados de validação (dataset\_validação): 2.000 imagens = 1.000 imagens de cães e 1.000 imagens de gatos
- Dados de teste (dataset\_teste): 1.000 imagens de cães e gatos

Para acessar a base de dados execute os seguintes passos:

1. Acesse o endereço: <https://www.kaggle.com/mrcioleandrogonaves/dogs-vs-cats>
2. Clique em "download" (e pode cancelar o processo assim que iniciar)
3. Copiar o link para este processo na aba de downloads do navegador e colar no espaço entre aspas simples no comando wget abaixo (o endereço é longo!)

```
!wget 'https://storage.googleapis.com/kaggle-data-sets/1167797/1956504/bundle/archive.zip?X-Goog-Algorithm=GOOG4-RSA-SHA256&X-Goog-Cred
The destination name is too long (767), reducing to 236
--2024-08-27 13:42:46-- https://storage.googleapis.com/kaggle-data-sets/1167797/1956504/bundle/archive.zip?X-Goog-Algorithm=GOOG4-
Resolving storage.googleapis.com (storage.googleapis.com)... 74.125.20.207, 108.177.98.207, 74.125.197.207, ...
Connecting to storage.googleapis.com (storage.googleapis.com)|74.125.20.207|:443... connected.
HTTP request sent, awaiting response... 400 Bad Request
2024-08-27 13:42:46 ERROR 400: Bad Request.
```

▼

#### Renomear o arquivo (cujo nome fica longo demais)

Copie o nome do arquivo baixado e cole no espaço entre aspas simples do comando abaixo.

```
!mv 'archive.zip?X-Goog-Algorithm=GOOG4-RSA-SHA256&X-Goog-Credential=gcp-kaggle-com@kaggle-161607.iam.gserviceaccount.com%2F20211113%2F
mv: cannot stat 'archive.zip?X-Goog-Algorithm=GOOG4-RSA-SHA256&X-Goog-Credential=gcp-kaggle-com@kaggle-161607.iam.gserviceaccount.c
```

▼

#### Descompactar

```
!unzip -qu dogs_vs_cats.zip -d Dogs_vs_cats
ls -l
unzip: cannot find or open dogs_vs_cats.zip, dogs_vs_cats.zip.zip or dogs_vs_cats.zip.ZIP.
total 8
drwxr-xr-x 7 root root 4096 Aug 27 13:42 DSBD
drwxr-xr-x 1 root root 4096 Aug 23 13:20 sample_data
```

Abaixo seguem visualizações de algumas imagens do conjunto de dados de treinamento (dataset\_treino).

```
from IPython.display import Image
Image(filename = 'Dogs_vs_cats/dataset_treino/cats/cat.7.jpg')
```



```

-----
FileNotFoundError                                Traceback (most recent call last)
<ipython-input-7-8f02bf55668e> in <cell line: 2>()
      1 from IPython.display import Image
----> 2 Image(filename = 'Dogs_vs_cats/dataset_treino/cats/cat.7.jpg')

-----
3 frames
/usr/local/lib/python3.10/dist-packages/IPython/core/display.py in reload(self)
    660     """Reload the raw data from file or URL."""
    661     if self.filename is not None:
--> 662         with open(self.filename, self._read_flags) as f:
    663             self.data = f.read()
    664         elif self.url is not None:

FileNotFoundError: [Errno 2] No such file or directory: 'Dogs_vs_cats/dataset_treino/cats/cat.7.jpg'

```

Próximas etapas: [Explicar o erro](#)

Image(filename='Dogs\_vs\_cats/dataset\_treino/dogs/dog.13.jpg')

Image(filename='Dogs\_vs\_cats/dataset\_treino/cats/cat.3901.jpg')

Image(filename='Dogs\_vs\_cats/dataset\_treino/dogs/dog.3998.jpg')

## ✓ COVID-19

Conjuntos de dados de imagens de tomografias computadorizadas (CT-scans: Computerized Tomography - scans) em duas classes:

- Dados de treino: 1800 imagens = 900 imagens CT de pulmões com COVID e 900 imagens CT de pulmões sem COVID
- Dados de validação: 600 imagens = 300 imagens CT de pulmões com COVID e 300 imagens CT de pulmões sem COVID
- Dados de teste: 60 imagens = 30 imagens de pulmões com COVID e 30 imagens de pulmões sem COVID

Para acessar a base de dados execute os seguintes passos:

1. Acesse o endereço: <https://www.kaggle.com/mrcioleandrogonalves/covid19>
2. Clique em "download" (e pode cancelar o processo assim que iniciar)
3. Copiar o link para este processo na aba de downloads do navegador e colar no espaço entre aspas simples no comando wget abaixo (o endereço é longo!)

```
!wget 'https://storage.googleapis.com/kaggle-data-sets/1168490/1957615/bundle/archive.zip?X-Goog-Algorithm=GOOG4-RSA-SHA256&X-Goog-Cred
```

## ✓ Renomear o arquivo (cujo nome fica longo demais)

Copie o nome do arquivo baixado e cole no espaço entre aspas simples do comando abaixo.

```
!mv '/content/archive.zip?X-Goog-Algorithm=GOOG4-RSA-SHA256&X-Goog-Credential=gcp-kaggle-com@kaggle-161607.iam.gserviceaccount.com%2F20
```

## ✓ Descompactar

```
!unzip -qu covid-19.zip -d COVID-19
!ls -l
```

Imagens COVID

```
import matplotlib.pyplot as plt
import os

arqs_img = os.listdir('COVID-19/dataset_treino/COVID')
max_arqs = len(arqs_img)
rows = 5
cols = 5
if (rows*cols < max_arqs):
    fig, ax = plt.subplots(rows, cols, figsize=(12, 12))
    n = 0
    for i in range(rows):
        for j in range(cols):
            file_name = 'COVID-19/dataset_treino/COVID/'+arqs_img[n]
            image = plt.imread(file_name)
            ax[i, j].set_title(arqs_img[n])
            ax[i, j].set_xticks([])
            ax[i, j].set_yticks([])
            ax[i, j].imshow(image)
            n += 1
```