DATA SCIENCE CAPSTONE PROJECT EXPLANATION

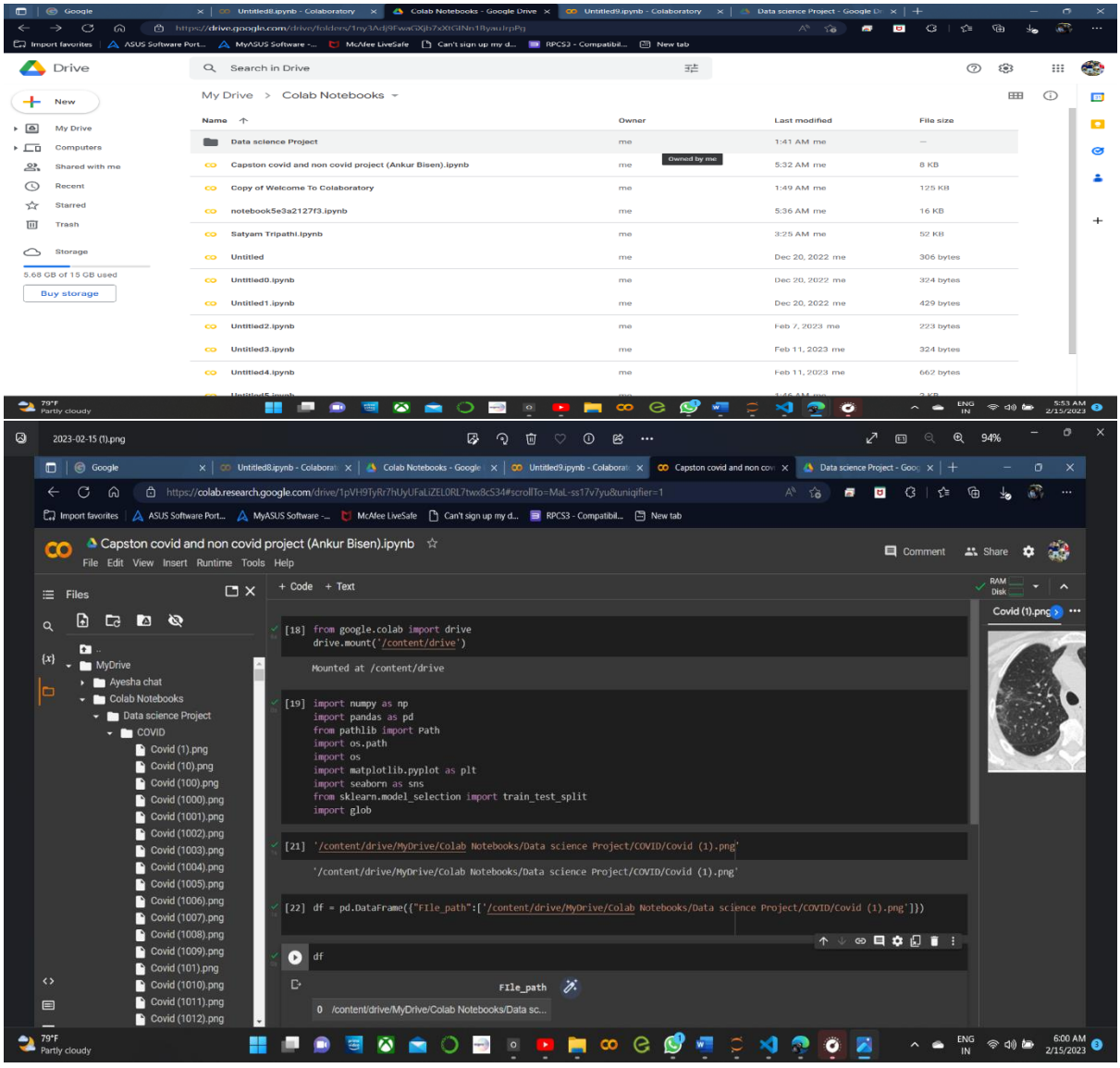
SARS-COV-2 Ct-Scan Dataset

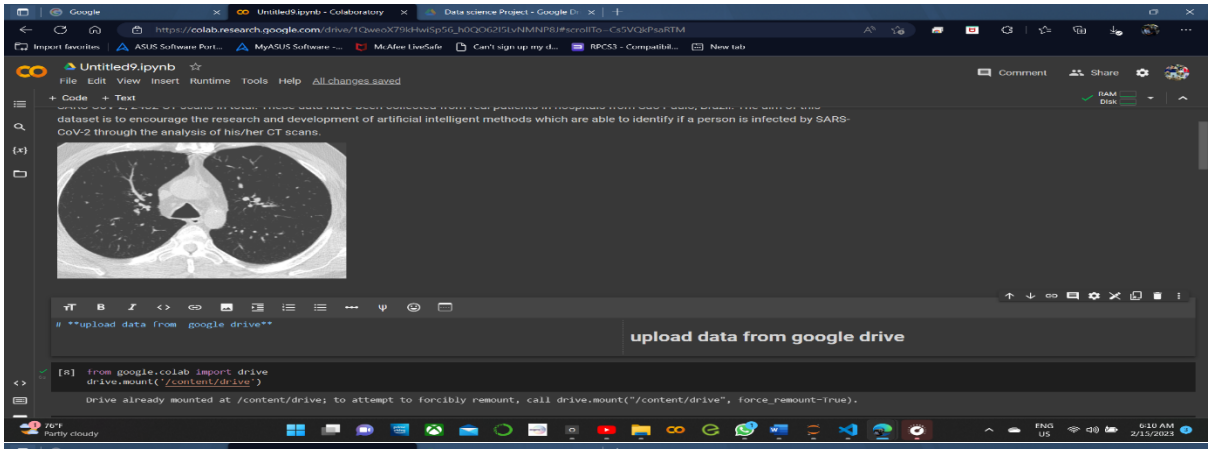
A large dataset of CT scans for SARS-CoV-2 (COVID-19) identification

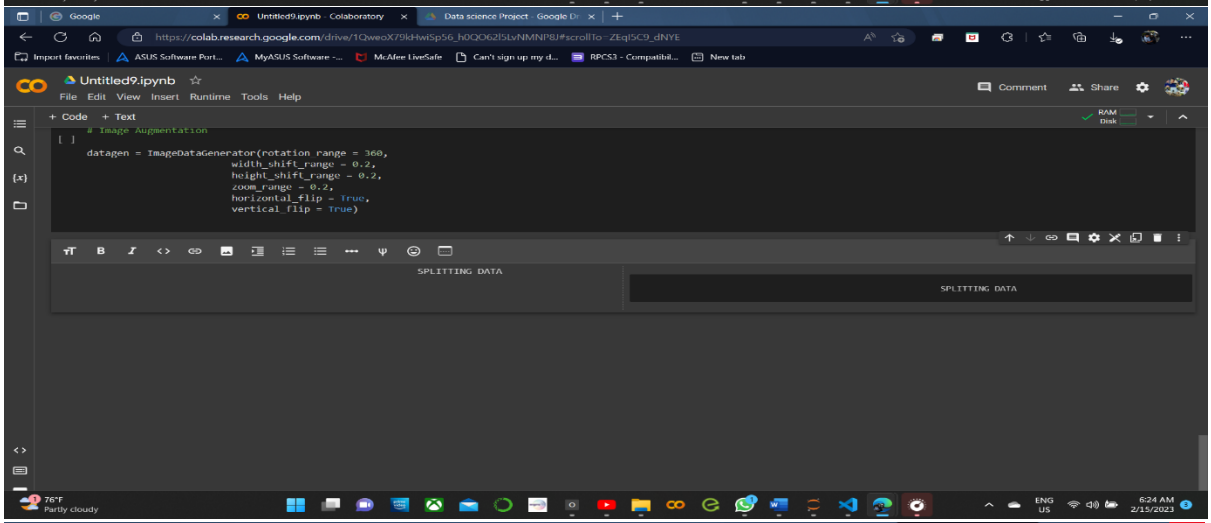
Data Overview

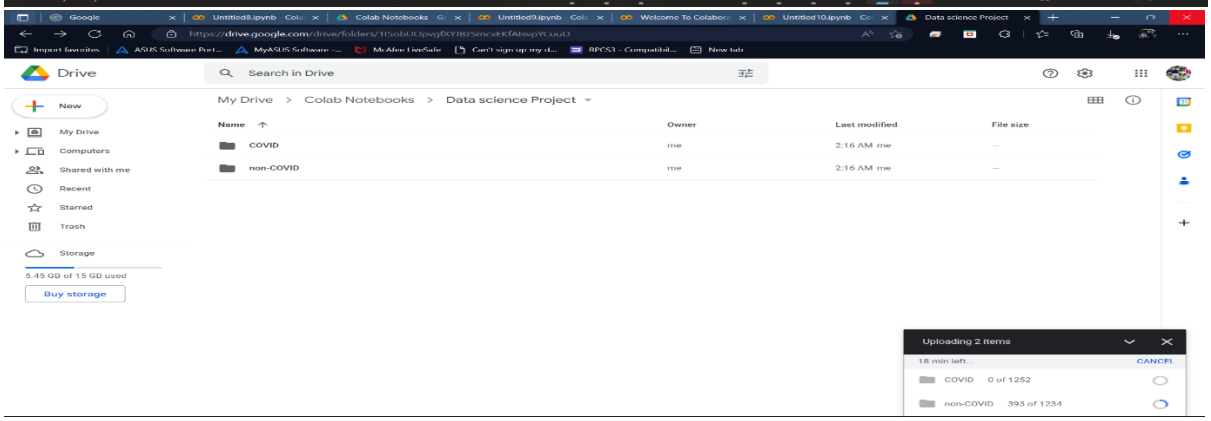
This dataset contains 1252 CT scans that are positive for SARS-CoV-2 infection (COVID-19) and 1230 CT scans for patients non-infected by SARS-CoV-2, 2482 CT scans in total. These data have been collected from real patients in hospitals from Sao Paulo, Brazil. The aim of this dataset is to encourage the research and development of artificial intelligent methods which are able to identify if a person is infected by SARS-CoV-2 through the analysis of his/her CT scans.

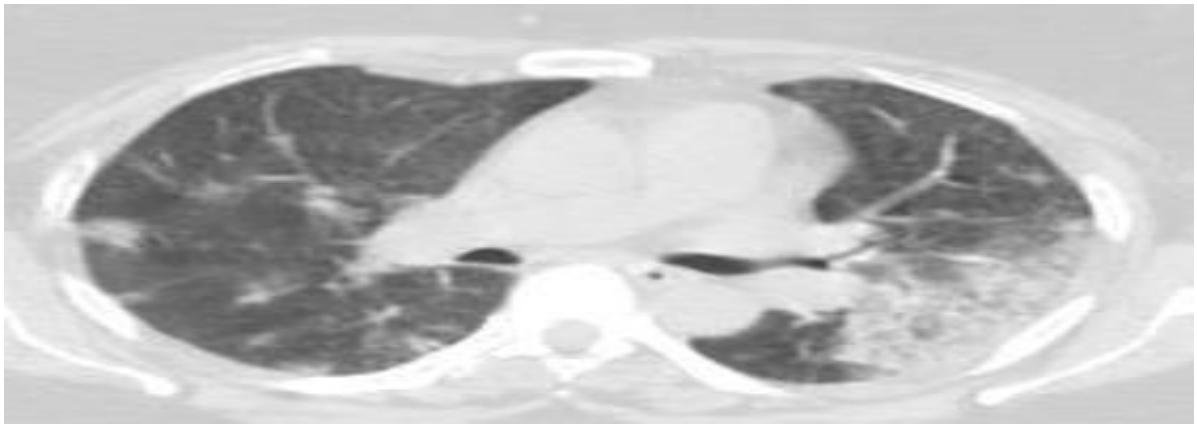
WORKING ON PROJECT




Untitled9.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
+ Code + Text
dataset is to encourage the research and development of artificial intelligent methods which are able to identify if a person is infected by SARS-CoV-2 through the analysis of his/her CT scans.
upload data from google drive
upload data from google drive
[8] from google.colab import drive
drive.mount('/content/drive')
Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=True).


Untitled9.ipynb
File Edit View Insert Runtime Tools Help
Comment
Share
+ Code + Text
# Image Augmentation
datagen = ImageDataGenerator(rotation_range = 360,
width_shift_range = 0.2,
height_shift_range = 0.2,
zoom_range = 0.2,
horizontal_flip = True,
vertical_flip = True)
SPLITTING DATA
SPLITTING DATA


Drive
Search in Drive
New
My Drive
Computers
Shared with me
Recent
Starred
Trash
Storage
5.45 GB of 15 GB used
Buy storage
My Drive > Colab Notebooks > Data science Project
Name
Owner
Last modified
File size
COVID
me
2:16 AM me
non-COVID
me
2:16 AM me
Uploading 2 items
18 min left...
CANCEL
COVID 0 of 1252
non-COVID 393 of 1234



1 – The images are in different sizes so you have to take a fixed size on which you have to work-

We can use the Python Imaging Library (PIL) to resize images where we have to work.

First, import the PIL library:

```
import PIL
```

Then, open the image you want to resize.

```
Img = PIL.Image.open("image_name.jpg")
```

Now you can set the desired size for the image:

```
new_width = 200 new_height = 300
```

Finally, you can use the resize function of the PIL library to resize the image:

```
img.resize((new_width, new_height))
```

The image is now resized to the desired size.

SPLITTING DATA

The data can be split into train and test sets using the train_test_split() method from the scikit-learn library. This method takes two parameters: X, which is the feature set, and y, which is the target variable. The test size parameter is used to specify the proportion of the data that should be split into the test set. The random_state parameter is used to ensure that the same split is used each time the code is run.

Bulding and Visualizing the model using ResNet-50

ResNet-Building50 is a convolutional neural network that is trained on more than a million images from the ImageNet database. It is a 50-layer deep convolutional network and is a variant of the ResNet architecture. ResNet-50 can be used for image classification, object detection, and other computer vision tasks.

In this example, ResNet-50 will be used to build and visualize a model for image classification. The code below imports the necessary libraries, loads the ResNet-50 model, and prepares the data for training.

END