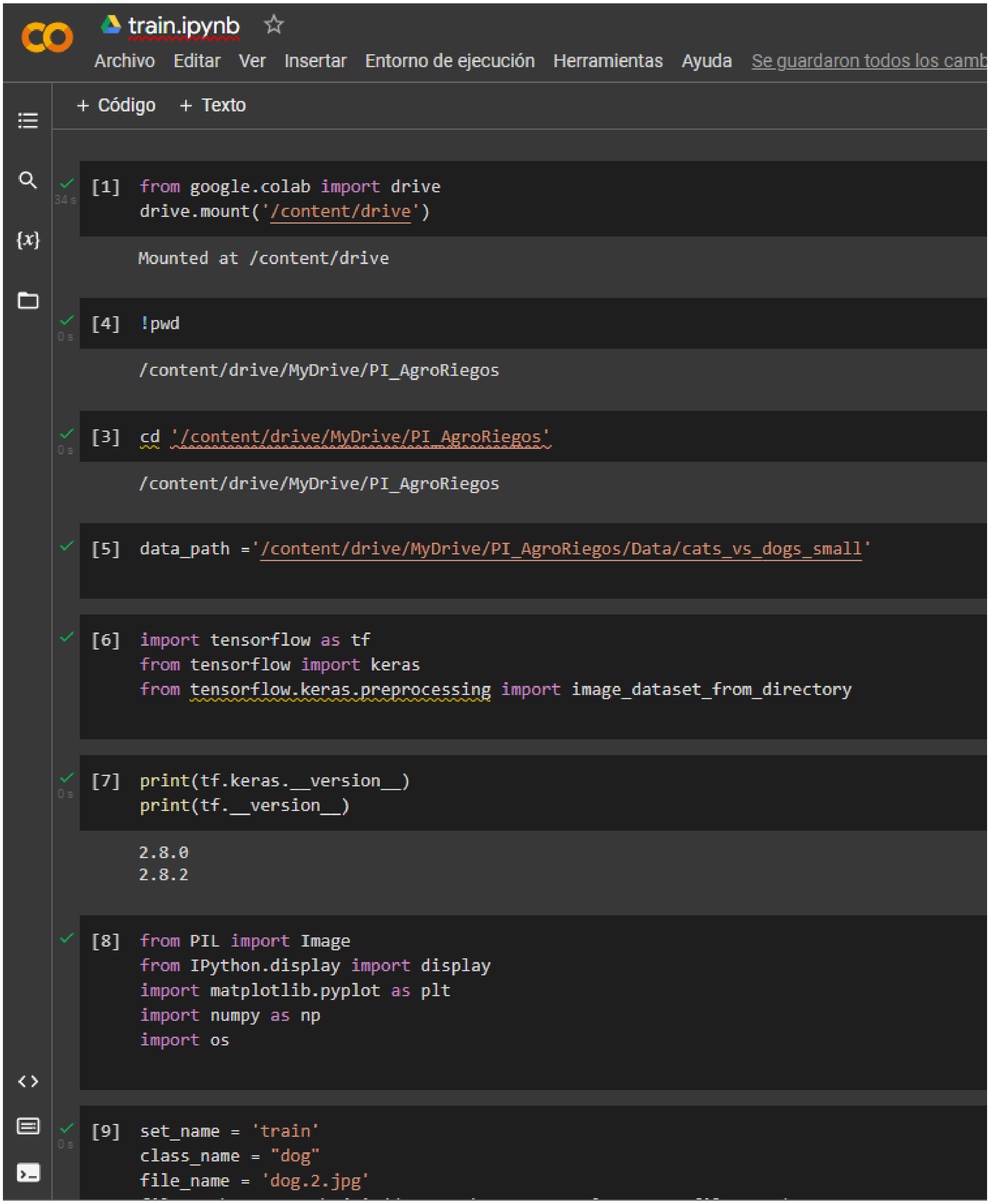


[1]
```
from google.colab import drive
drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

[4]
```
!pwd
```

```
/content/drive/MyDrive/PI_AgroRiegos
```

[3]
```
cd '/content/drive/MyDrive/PI_AgroRiegos'
```

```
/content/drive/MyDrive/PI_AgroRiegos
```

[5]
```
data_path ='/content/drive/MyDrive/PI_AgroRiegos/Data/cats_vs_dogs_small'
```

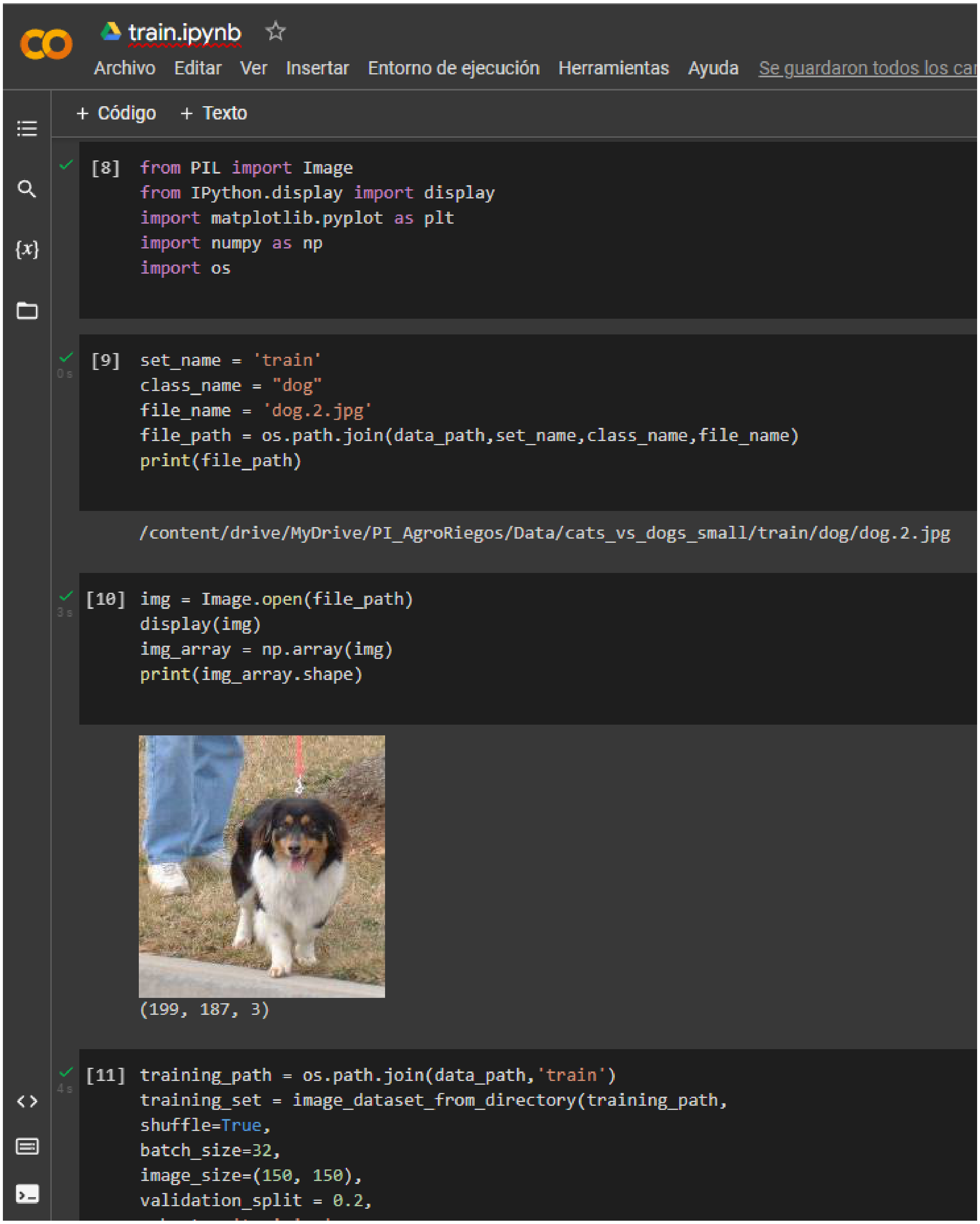
[6]
```
import tensorflow as tf
from tensorflow import keras
from tensorflow.keras.preprocessing import image_dataset_from_directory
```

[7]
```
print(tf.keras.__version__)
print(tf.__version__)
```

```
2.8.0
2.8.2
```

[8]
```
from PIL import Image
from IPython.display import display
import matplotlib.pyplot as plt
import numpy as np
import os
```

[9]
```
set_name = 'train'
class_name = "dog"
file_name = 'dog.2.jpg'
```



```
from PIL import Image
from IPython.display import display
import matplotlib.pyplot as plt
import numpy as np
import os
```

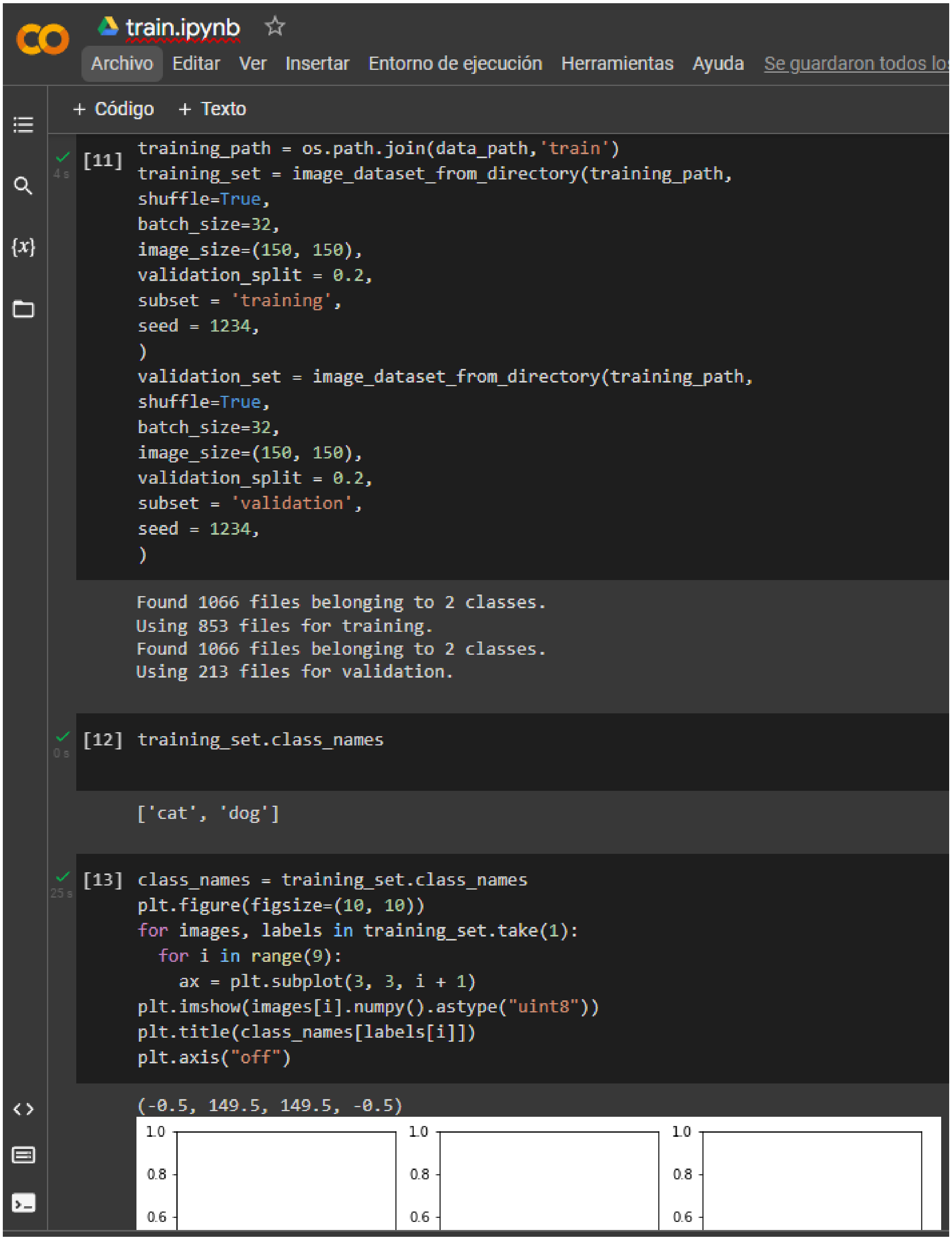
```
set_name = 'train'
class_name = "dog"
file_name = 'dog.2.jpg'
file_path = os.path.join(data_path,set_name,class_name,file_name)
print(file_path)
```

```
/content/drive/MyDrive/PI_AgroRiegos/Data/cats_vs_dogs_small/train/dog/dog.2.jpg
```

```
img = Image.open(file_path)
display(img)
img_array = np.array(img)
print(img_array.shape)
```

```
(199, 187, 3)
```

```
training_path = os.path.join(data_path,'train')
training_set = image_dataset_from_directory(training_path,
shuffle=True,
batch_size=32,
image_size=(150, 150),
validation_split = 0.2,
```



```
[11] training_path = os.path.join(data_path,'train')
     training_set = image_dataset_from_directory(training_path,
     shuffle=True,
     batch_size=32,
     image_size=(150, 150),
     validation_split = 0.2,
     subset = 'training',
     seed = 1234,
     )
     validation_set = image_dataset_from_directory(training_path,
     shuffle=True,
     batch_size=32,
     image_size=(150, 150),
     validation_split = 0.2,
     subset = 'validation',
     seed = 1234,
     )
```

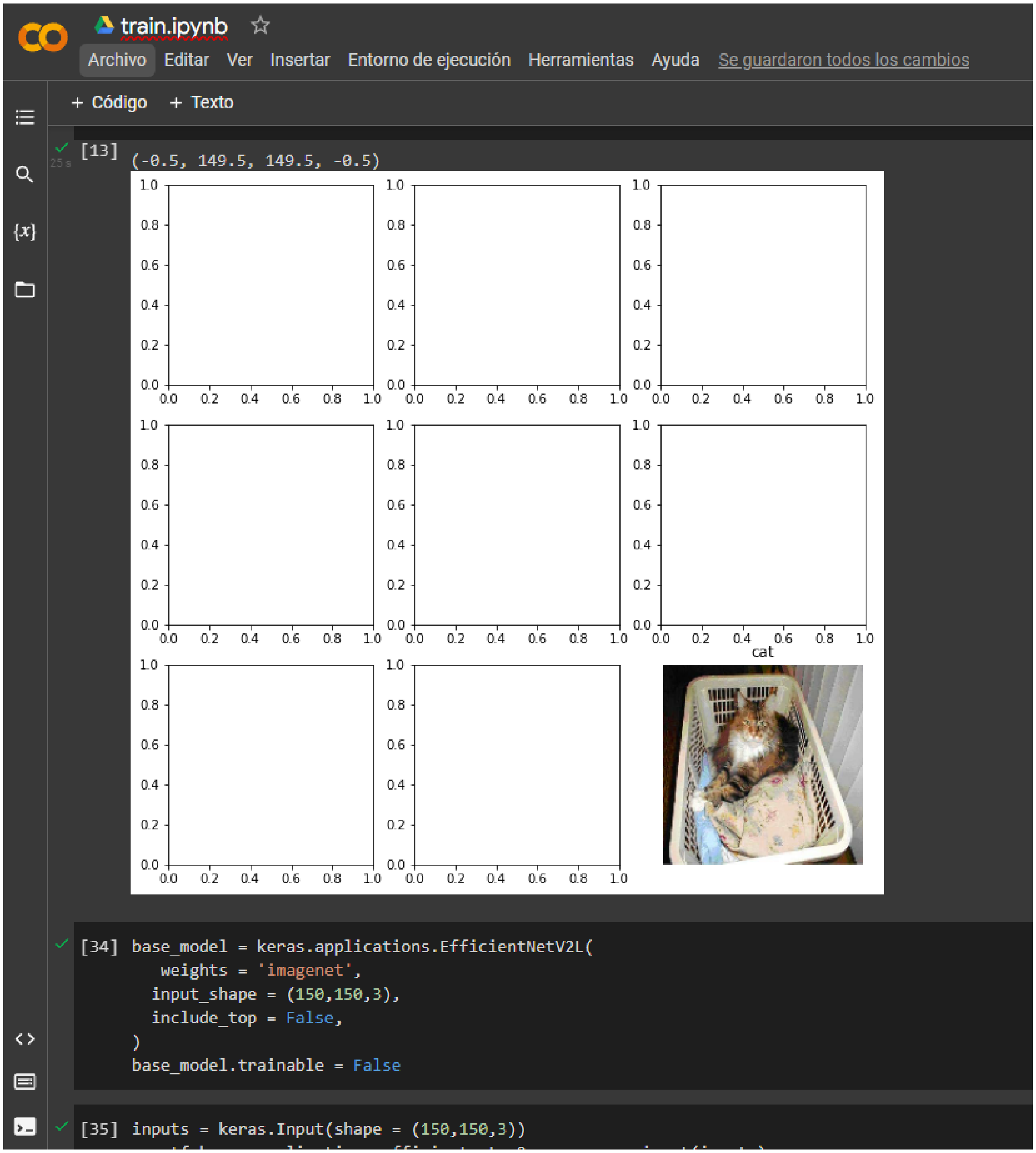
```
Found 1066 files belonging to 2 classes.
Using 853 files for training.
Found 1066 files belonging to 2 classes.
Using 213 files for validation.
```

```
[12] training_set.class_names
```

```
['cat', 'dog']
```

```
[13] class_names = training_set.class_names
     plt.figure(figsize=(10, 10))
     for images, labels in training_set.take(1):
       for i in range(9):
         ax = plt.subplot(3, 3, i + 1)
     plt.imshow(images[i].numpy().astype("uint8"))
     plt.title(class_names[labels[i]])
     plt.axis("off")
```

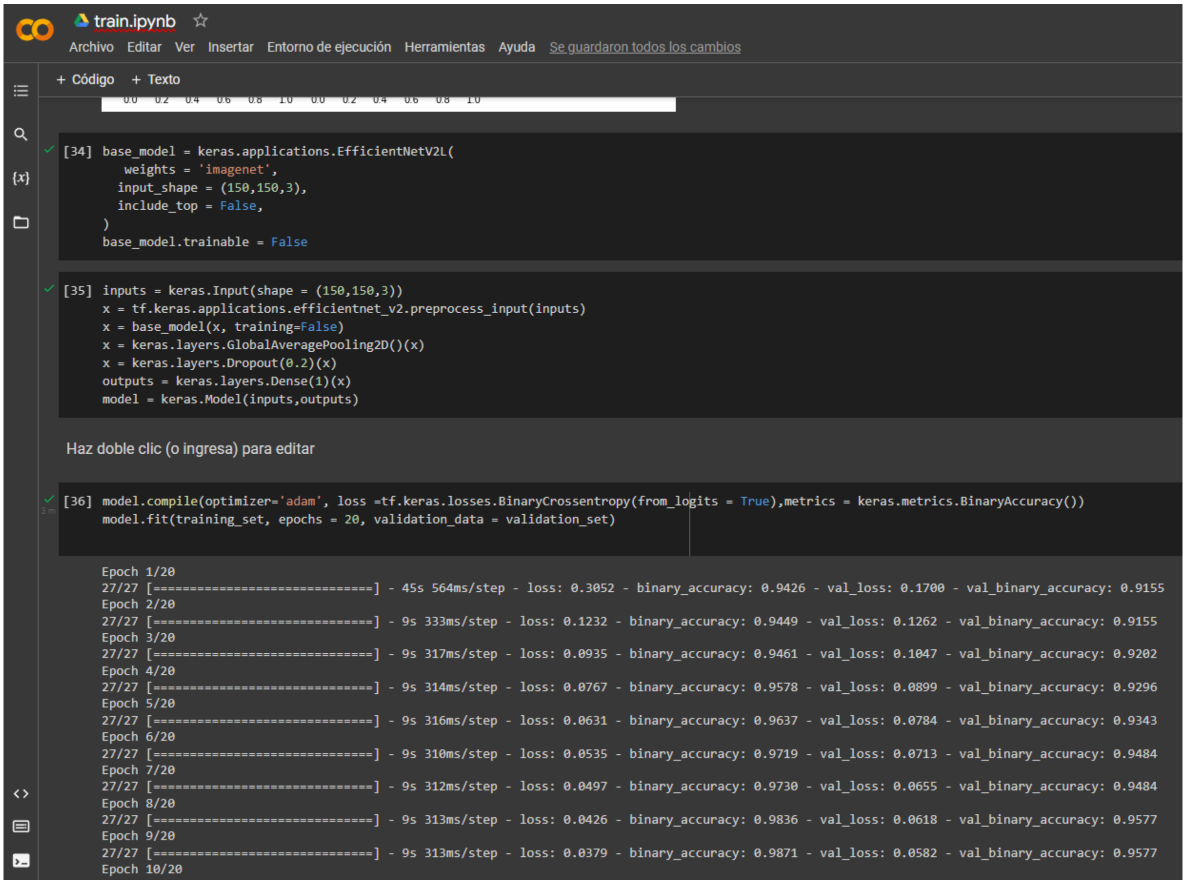
```
(-0.5, 149.5, 149.5, -0.5)
```



```
[13] (-0.5, 149.5, 149.5, -0.5)
```

```
[34] base_model = keras.applications.EfficientNetV2L(
        weights = 'imagenet',
       input_shape = (150,150,3),
       include_top = False,
     )
     base_model.trainable = False
```

```
[35] inputs = keras.Input(shape = (150,150,3))
```


CO
train.ipynb ☆
Archivo Editar Ver Insertar Entorno de ejecución Herramientas Ayuda Se guardaron todos los cambios
+ Código + Texto
0.0 0.2 0.4 0.6 0.8 1.0 0.0 0.2 0.4 0.6 0.8 1.0

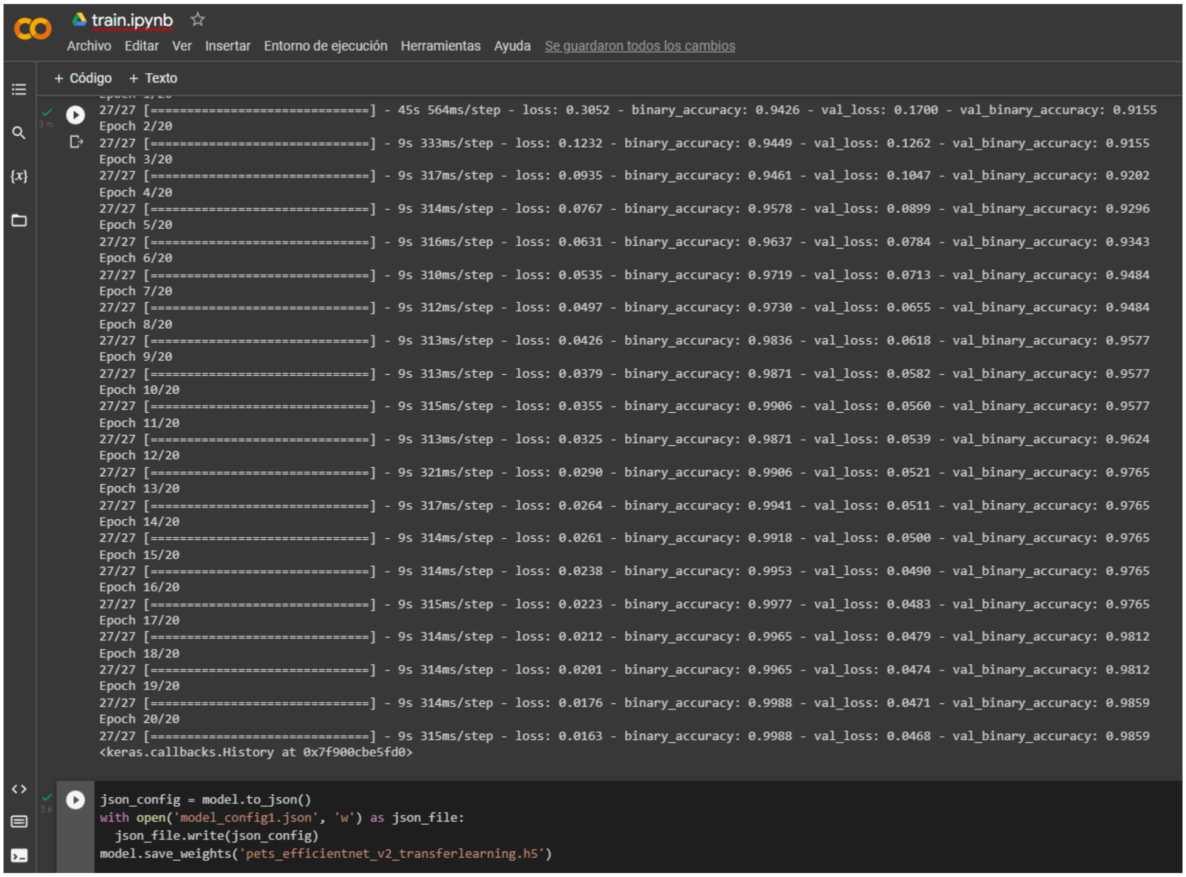
```
[34] base_model = keras.applications.EfficientNetV2L(
        weights = 'imagenet',
       input_shape = (150,150,3),
       include_top = False,
     )
     base_model.trainable = False
```

```
[35] inputs = keras.Input(shape = (150,150,3))
     x = tf.keras.applications.efficientnet_v2.preprocess_input(inputs)
     x = base_model(x, training=False)
     x = keras.layers.GlobalAveragePooling2D()(x)
     x = keras.layers.Dropout(0.2)(x)
     outputs = keras.layers.Dense(1)(x)
     model = keras.Model(inputs,outputs)
```

Haz doble clic (o ingresa) para editar

```
[36] model.compile(optimizer='adam', loss =tf.keras.losses.BinaryCrossentropy(from_logits = True),metrics = keras.metrics.BinaryAccuracy())
     model.fit(training_set, epochs = 20, validation_data = validation_set)
```

```
Epoch 1/20
27/27 [==============================] - 45s 564ms/step - loss: 0.3052 - binary_accuracy: 0.9426 - val_loss: 0.1700 - val_binary_accuracy: 0.9155
Epoch 2/20
27/27 [==============================] - 9s 333ms/step - loss: 0.1232 - binary_accuracy: 0.9449 - val_loss: 0.1262 - val_binary_accuracy: 0.9155
Epoch 3/20
27/27 [==============================] - 9s 317ms/step - loss: 0.0935 - binary_accuracy: 0.9461 - val_loss: 0.1047 - val_binary_accuracy: 0.9202
Epoch 4/20
27/27 [==============================] - 9s 314ms/step - loss: 0.0767 - binary_accuracy: 0.9578 - val_loss: 0.0899 - val_binary_accuracy: 0.9296
Epoch 5/20
27/27 [==============================] - 9s 316ms/step - loss: 0.0631 - binary_accuracy: 0.9637 - val_loss: 0.0784 - val_binary_accuracy: 0.9343
Epoch 6/20
27/27 [==============================] - 9s 310ms/step - loss: 0.0535 - binary_accuracy: 0.9719 - val_loss: 0.0713 - val_binary_accuracy: 0.9484
Epoch 7/20
27/27 [==============================] - 9s 312ms/step - loss: 0.0497 - binary_accuracy: 0.9730 - val_loss: 0.0655 - val_binary_accuracy: 0.9484
Epoch 8/20
27/27 [==============================] - 9s 313ms/step - loss: 0.0426 - binary_accuracy: 0.9836 - val_loss: 0.0618 - val_binary_accuracy: 0.9577
Epoch 9/20
27/27 [==============================] - 9s 313ms/step - loss: 0.0379 - binary_accuracy: 0.9871 - val_loss: 0.0582 - val_binary_accuracy: 0.9577
Epoch 10/20
```



```
27/27 [==============================] - 45s 564ms/step - loss: 0.3052 - binary_accuracy: 0.9426 - val_loss: 0.1700 - val_binary_accuracy: 0.9155
Epoch 2/20
27/27 [==============================] - 9s 333ms/step - loss: 0.1232 - binary_accuracy: 0.9449 - val_loss: 0.1262 - val_binary_accuracy: 0.9155
Epoch 3/20
27/27 [==============================] - 9s 317ms/step - loss: 0.0935 - binary_accuracy: 0.9461 - val_loss: 0.1047 - val_binary_accuracy: 0.9202
Epoch 4/20
27/27 [==============================] - 9s 314ms/step - loss: 0.0767 - binary_accuracy: 0.9578 - val_loss: 0.0899 - val_binary_accuracy: 0.9296
Epoch 5/20
27/27 [==============================] - 9s 316ms/step - loss: 0.0631 - binary_accuracy: 0.9637 - val_loss: 0.0784 - val_binary_accuracy: 0.9343
Epoch 6/20
27/27 [==============================] - 9s 310ms/step - loss: 0.0535 - binary_accuracy: 0.9719 - val_loss: 0.0713 - val_binary_accuracy: 0.9484
Epoch 7/20
27/27 [==============================] - 9s 312ms/step - loss: 0.0497 - binary_accuracy: 0.9730 - val_loss: 0.0655 - val_binary_accuracy: 0.9484
Epoch 8/20
27/27 [==============================] - 9s 313ms/step - loss: 0.0426 - binary_accuracy: 0.9836 - val_loss: 0.0618 - val_binary_accuracy: 0.9577
Epoch 9/20
27/27 [==============================] - 9s 313ms/step - loss: 0.0379 - binary_accuracy: 0.9871 - val_loss: 0.0582 - val_binary_accuracy: 0.9577
Epoch 10/20
27/27 [==============================] - 9s 315ms/step - loss: 0.0355 - binary_accuracy: 0.9906 - val_loss: 0.0560 - val_binary_accuracy: 0.9577
Epoch 11/20
27/27 [==============================] - 9s 313ms/step - loss: 0.0325 - binary_accuracy: 0.9871 - val_loss: 0.0539 - val_binary_accuracy: 0.9624
Epoch 12/20
27/27 [==============================] - 9s 321ms/step - loss: 0.0290 - binary_accuracy: 0.9906 - val_loss: 0.0521 - val_binary_accuracy: 0.9765
Epoch 13/20
27/27 [==============================] - 9s 317ms/step - loss: 0.0264 - binary_accuracy: 0.9941 - val_loss: 0.0511 - val_binary_accuracy: 0.9765
Epoch 14/20
27/27 [==============================] - 9s 314ms/step - loss: 0.0261 - binary_accuracy: 0.9918 - val_loss: 0.0500 - val_binary_accuracy: 0.9765
Epoch 15/20
27/27 [==============================] - 9s 314ms/step - loss: 0.0238 - binary_accuracy: 0.9953 - val_loss: 0.0490 - val_binary_accuracy: 0.9765
Epoch 16/20
27/27 [==============================] - 9s 315ms/step - loss: 0.0223 - binary_accuracy: 0.9977 - val_loss: 0.0483 - val_binary_accuracy: 0.9765
Epoch 17/20
27/27 [==============================] - 9s 314ms/step - loss: 0.0212 - binary_accuracy: 0.9965 - val_loss: 0.0479 - val_binary_accuracy: 0.9812
Epoch 18/20
27/27 [==============================] - 9s 314ms/step - loss: 0.0201 - binary_accuracy: 0.9965 - val_loss: 0.0474 - val_binary_accuracy: 0.9812
Epoch 19/20
27/27 [==============================] - 9s 314ms/step - loss: 0.0176 - binary_accuracy: 0.9988 - val_loss: 0.0471 - val_binary_accuracy: 0.9859
Epoch 20/20
27/27 [==============================] - 9s 315ms/step - loss: 0.0163 - binary_accuracy: 0.9988 - val_loss: 0.0468 - val_binary_accuracy: 0.9859
<keras.callbacks.History at 0x7f900cbe5fd0>
```

```
json_config = model.to_json()
with open('model_config1.json', 'w') as json_file:
  json_file.write(json_config)
model.save_weights('pets_efficientnet_v2_transferlearning.h5')
```