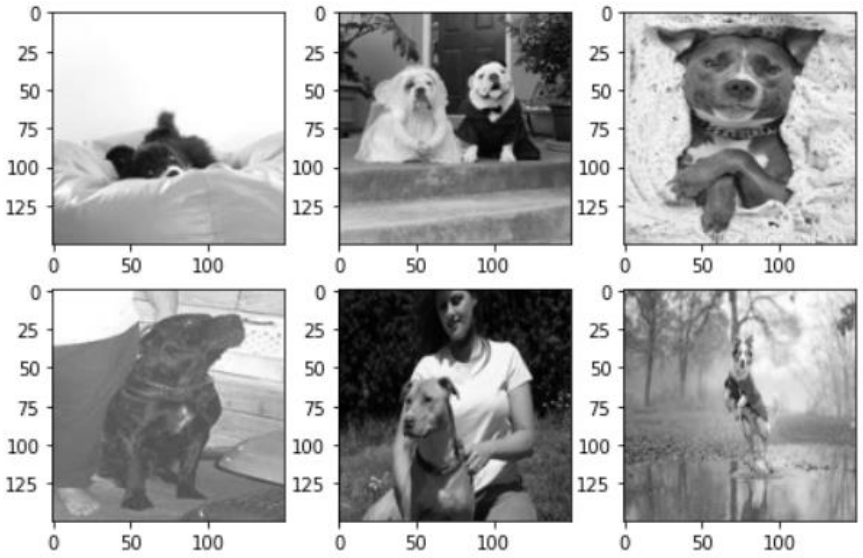
DOGS AND CATS RECOGNITION

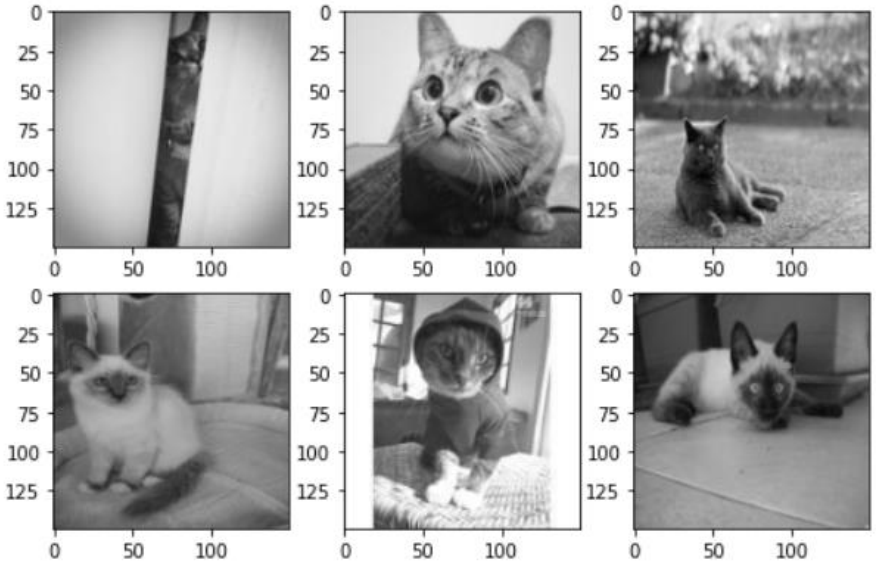
A project by Andrea Nappi

The goal of this project is to build a neural network able to distinguish dog images from cat images with a decent degree of accuracy.
The model should be able to correctly classify the animal by spotting and learning to recognize some of their unique features.
The dataset that will be used for this project is “30k Cats and Dogs 150x150 Greyscale” from “Kaggle”, composed of 15002 images of dogs and 15060 images of cats. All the images are 150x150 pixels in dimension, grayscale (monochrome).

Sample images of dogs



Sample images of cats



As it is already possible to see from the above samples, the own nature of the pictures will make it very hard for the neural network to achieve top-tier levels of accuracy, as different breeds of dogs and cats are present in the dataset(and they can be very different from each other); the animals are in very different poses; sometimes there is more than one animal in the same picture; sometimes humans are present in the image; other times the animals are hiding/hidden, dressed, and more.

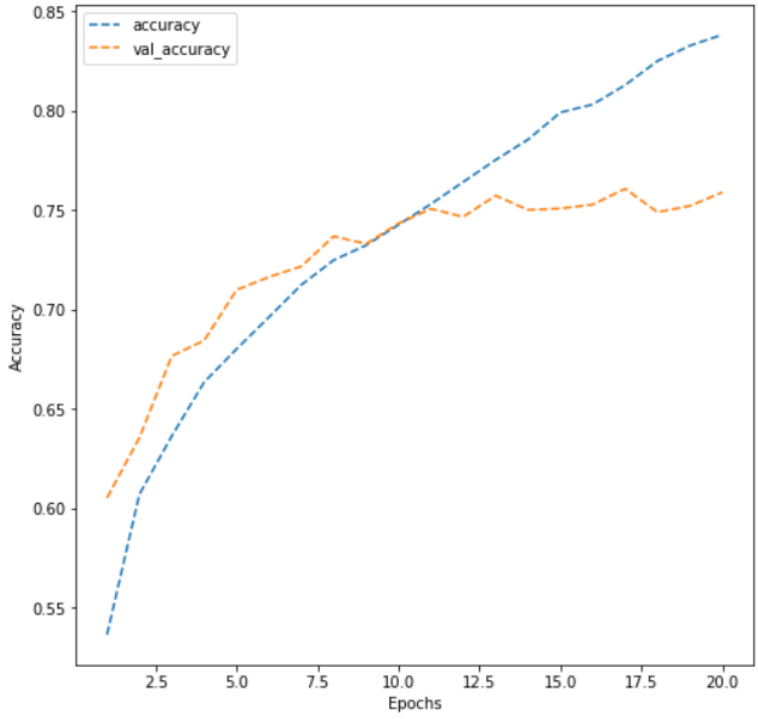
To proceed with the development of a neural network, the dataset has been divided into 3 sections: Training (containing 70% of the images), validation (containing 20% of the images), and test(containing about 10% of the images).

For the model, it has been chosen to use a CNN (Convolutional Neural Network) instead of an ANN (artificial Neural network) because of the nature of the problem faced, as CNN are neural networks that, unlike ANNs, are meant to work on image data. The model built is a sequential model, composed like so:

- Three convolutional blocks, composed of one conv2D layer with kernel size 2x2, "relu" activation, with 64 neurons in the first block and 32 neurons in the second and third block, a max-pooling 2x2 layer and a dropout layer with ratio 0.2;
- A Flatten layer to flatten the output, which allows to pass on to the Dense layers;
- A Dense layer with 512 neurons and "relu" activation
- A drop-out layer with ratio 0.4
- The final output layer with 2 neurons 8as the number of classes) and "SoftMax" activation.

The model has 5 322 626 parameters and all of them are trainable, it is compiled with the optimizer "Adam" (learning rate = 0.001), the loss chosen is "categorical crossentropy" and the metric chosen is "accuracy". The model is trained for 20 epochs.

Accuracy trend of the model:



Training values after 20 epochs:

```
loss: 0.3641 - accuracy: 0.838
```

Validation values after 20 epochs:

```
val_loss: 0.5264 - val_accuracy: 0.7590
```

Test values:

```
loss: 0.5837 - accuracy: 0.7393
```

After completing the 20 epochs training, the model shows a validation accuracy of about 75%, which matches pretty well the score obtained on the test set. As it is possible to see from the accuracy trend above, the model shows signs of overfitting starting at about the 10th epoch, with the final training accuracy being about 10% higher than the validation one. This means that the model is starting to go too deep on the training features, thinking some of them could be useful to recognize cats and dogs in general when they are actually not. The validation accuracy trend has been pretty stable in the second half of the training.

Overall, the performance of the model can be considered good, even considering that a random guessing model would have got the classification right about 50% of the times, because it has to be taken into account the nature of the images which has been previously discussed.

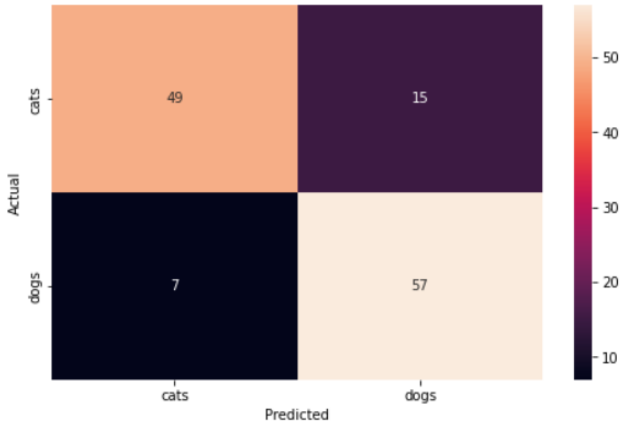
Classification report

```
Found 3007 images belonging to 2 classes.
4/4 [==============================] - 0s 7ms/step
              precision    recall  f1-score   support

           0       0.88      0.77      0.82        64
           1       0.79      0.89      0.84        64

    accuracy                           0.83       128
   macro avg       0.83      0.83      0.83       128
weighted avg       0.83      0.83      0.83       128
```

Confusion Matrix



The classification report and the confusion matrix above show how the model performed o a batch of 128 images. In this case, the model did overperform, with an accuracy ratio of about 84%, showing how there is not a class in which the model struggles particularly, even if it did perform slightly better on dog pictures.

After this analysis, it is possible to say that the model build is considered good enough to satisfy the needs of this project.

Andrea Nappi, Rome, Italy, 02-03-2023