Assignment 4:

Syed Alle Mustafa

Task 1:

To train a binary classifier and try it out with pre trained model and apply freeze and unfreeze training. Visualize training and activation layers.

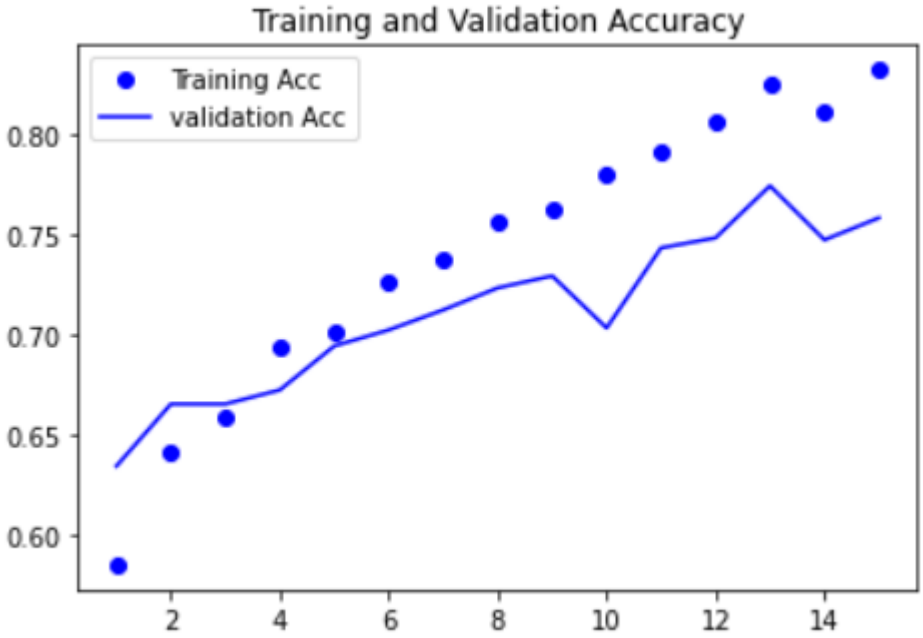
Implementation:

Drive was mounted on collab, and unzipped on the directory, and then training and validation sets were created as a directory, using train and test generator. Quantities of test, train and validation sets are determined by the question. i.e total of 2000 images. Then a model was defined and trained using the image generators for 30 epochs and overfitting was noticed and was trained for 15 epochs and the results are discussed below. Then visualization was done between the activation layers and the images were saved and are also discussed below, then the pretrained model was loaded, and put to a model with freezed weights and then training was done and results were noticed, then those results were optimized using the unfreeze weights afterwards. Then the data augmentation was done on the input to increase the outputs.

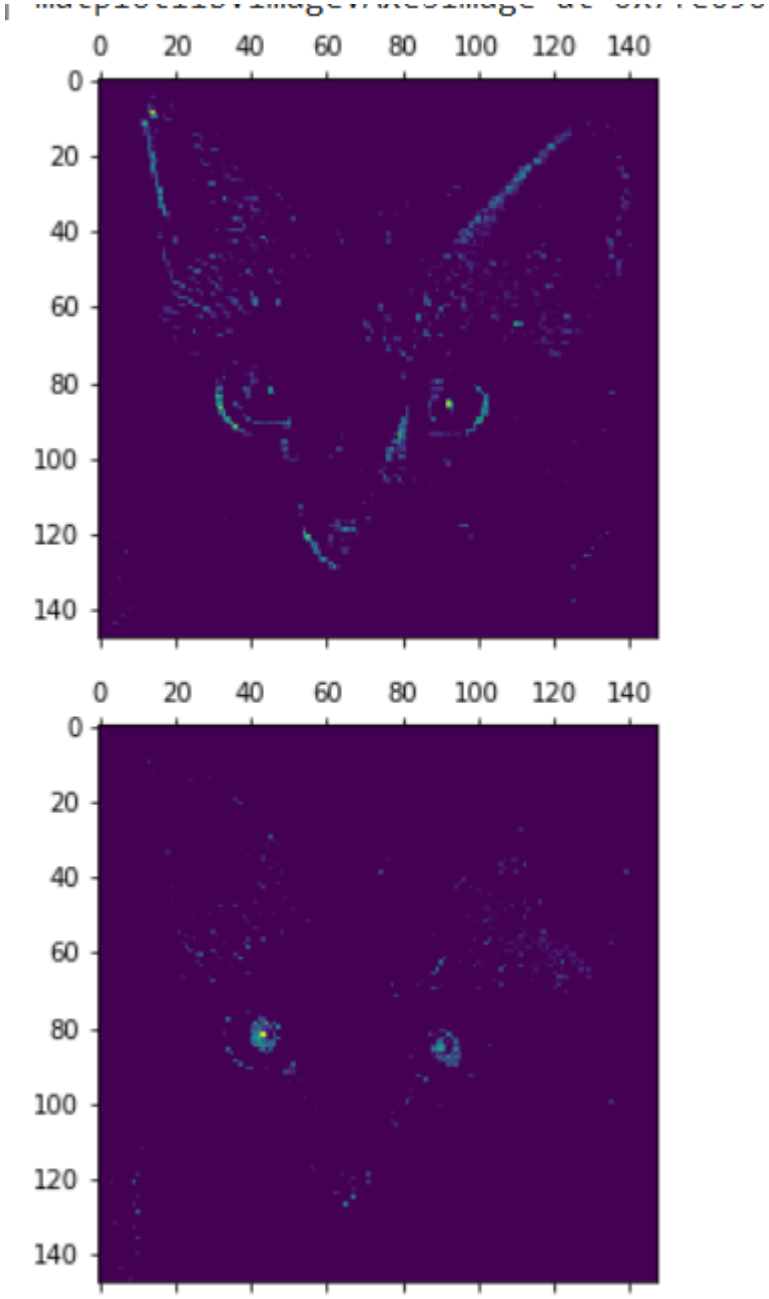
Results:

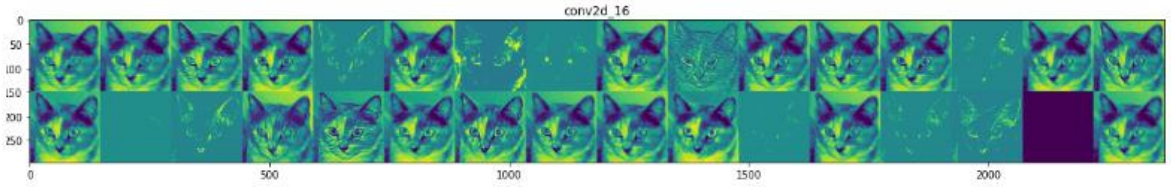
Model Trained for 15 epochs and 100 steps per epoch. The first training was done for 30 epochs with same layers, but overfitting was detected.

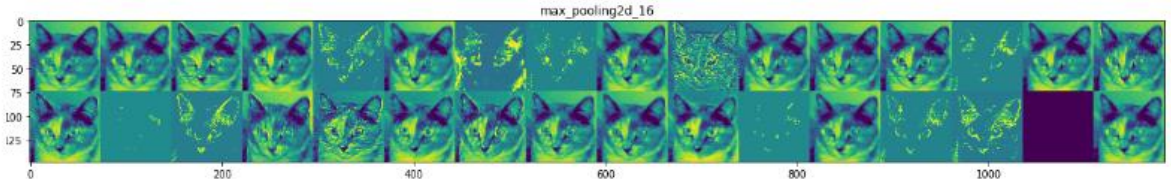
Following are the results:

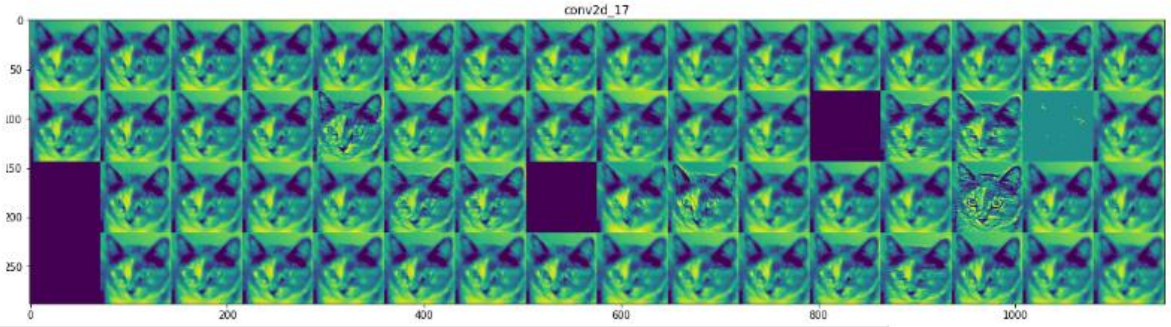


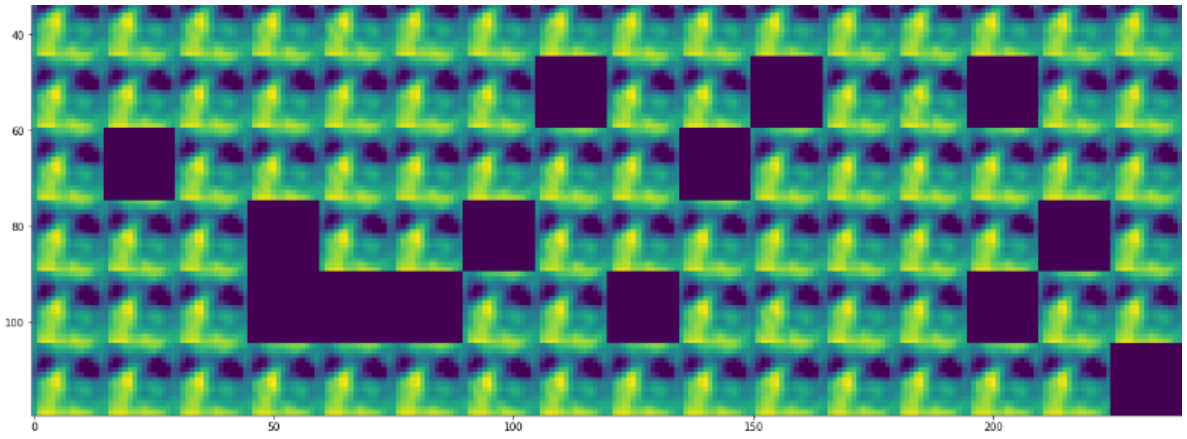
Following are the results of visualization of activations, this shows that the first layer makes the face structure of the cat prominent and the other one detects the eyes of the cat.

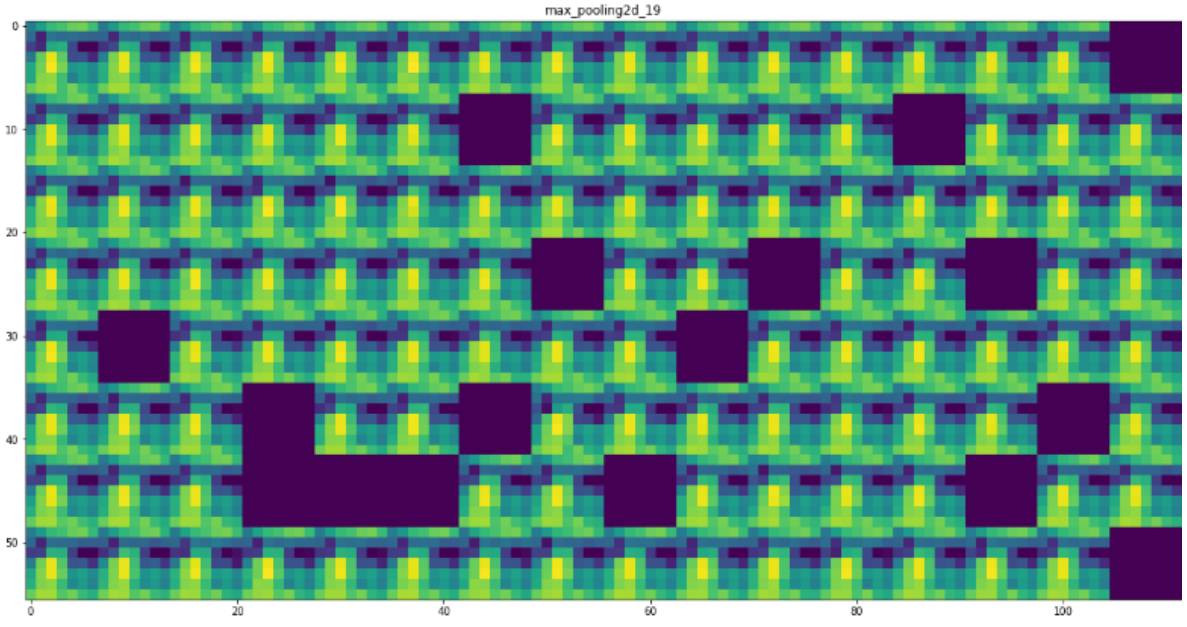

0
20
40
60
80
100
120
140
0
20
40
60
80
100
120
140


conv2d_16
0
50
100
150
200
250
0
500
1000
1500
2000


max_pooling2d_16
0
25
50
75
100
125
0
200
400
600
800
1000

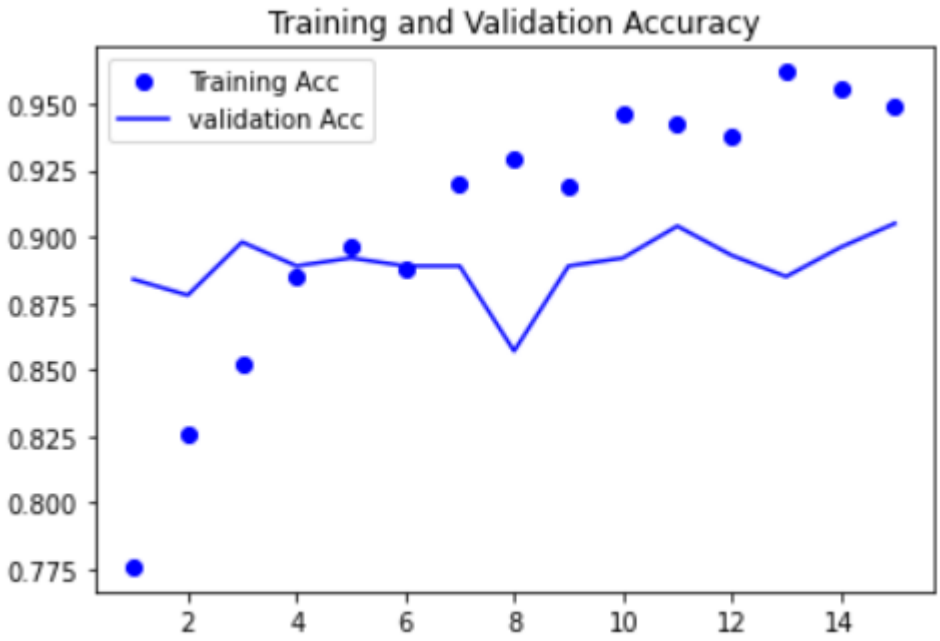

conv2d_17
0
50
100
150
200
250
0
200
400
600
800
1000


40
60
80
100
0
50
100
150
200

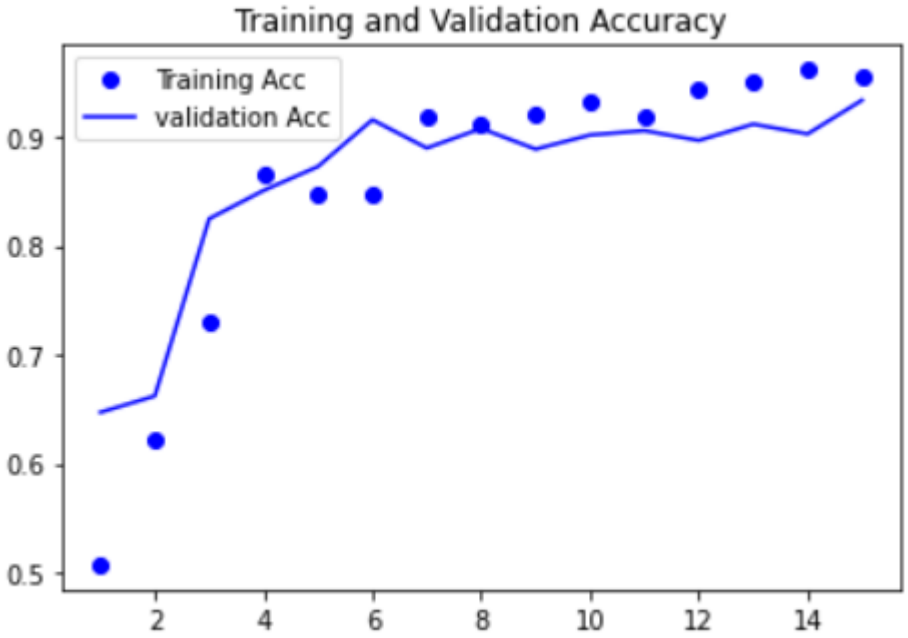

max_pooling2d_19
0
10
20
30
40
50
0
20
40
60
80
100

We can observe that as the layer goes deep, the main structure gets prominent, i.e the high level features of cat gets visual.

Now lets visualize the accuracy of pretrained model used as a layer:



The results seem quite promising but gets more better when retrained using unfreeze weights.



The data augmentation, increases the data and prevents overfitting the data, i.e training as well as validation accuracy is good.

Task 2:

To train the multiclass classification using image data and use convolution neural networks.

Implementation:

Data was extracted way it was extracted was previous model. Then, we created a model for multiclass classification, the details can be found on python notebook.

Results:

Following results were obtained:

```
tep - loss: 0.7351 - accuracy: 0.7417 - val_loss: 0.7027 - val_accuracy: 0.7558
tep - loss: 0.6853 - accuracy: 0.7592 - val_loss: 0.6163 - val_accuracy: 0.7851
tep - loss: 0.6422 - accuracy: 0.7736 - val_loss: 0.5920 - val_accuracy: 0.8005
tep - loss: 0.6246 - accuracy: 0.7824 - val_loss: 0.5799 - val_accuracy: 0.8021
tep - loss: 0.5768 - accuracy: 0.7996 - val_loss: 0.5247 - val_accuracy: 0.8179
tep - loss: 0.5515 - accuracy: 0.8061 - val_loss: 0.4869 - val_accuracy: 0.8324
tep - loss: 0.5054 - accuracy: 0.8205 - val_loss: 0.4285 - val_accuracy: 0.8556
tep - loss: 0.4707 - accuracy: 0.8368 - val_loss: 0.3989 - val_accuracy: 0.8625
- loss: 0.4414 - accuracy: 0.8491
```

Inception and Residual Blocks were tested alternatively but the results weren’t saved, The code for them are in the python notebook.