**Chapter 4**

# Results and Conclusions

1. The accuracy of the model which we have trained is close to 93%.
2. The size of the dataset which we use to train our model plays a major role in deciding the accuracy of the model. Larger the dataset, more experienced the model and hence more the accuracy.
3. Activation function such as ReLu function, plays an important role here as it adds the non-linearity to our model. The SoftMax function helps provide our predicted result on the scale of 0 to 1.
4. We can build more complex models and make our machines learn to identify anything and also tell the detail about it.
5. We have tested the app on 5 different pretrained models, the accuracy, inference/response time and model size are listed in the following table:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| *Model* | *Accuracy* | *Inference time*  *(Mobile CPU)* | *Inference time*  *(Mobile GPU)* | *Inference time*  *(Computer CPU)* | *Inference time*  *(Computer GPU)* | *Model size (before conversion to tflite)* | *Model size (After conversion to tflite)* |
| **VGG\_16** | 70.81% | 232 ms | 185 ms | 176 ms | 120 ms | 551 KB | 268 KB |
| **VGG\_19** | 72.13% | 230 ms | 182 ms | 173 ms | 126 ms | 552 KB | 268 KB |
| **Inception** | 81.10 % | 241 ms | 197 ms | 186 ms | 141 ms | 2148 KB | 1066 KB |
| **Xception** | 92.87% | 267 ms | 213 ms | 207 ms | 179 ms | 2148 KB | 1066 KB |
| **Resnet\_50** | 82.66 | 257 ms | 201 ms | 190 ms | 152 ms | 2148 KB | 1066 KB |

1. The app uses a pretrained model *Xception* which classifies an image if it contains a dog. If the picture provided has no dog in it,” No dog detected is displayed on the results screen else the breeds and their probability percentage is displayed on a Pie chart.