Abhishek Thurlapati(vxt160630)
Dan Crawford(dsc160130)

Summary of the 3 Methods for Image Classification

Comparing the three methods that we have implemented, we see that the first method we have done in R yielded an accuracy of 37.9%. The second method in python using no tensorflow libraries yielded 50.17% and the third method with CNN yielded a high of 74.44% accuracy. Clearly, we see that the third method was the best method at image classification. This is due to the usage of convolutional layers and deep learning.

Without CNNs, we were limited to vanilla neural networks, one with only one layer and a deep "dense" network with 3 layers. It was clear with the original one layer network in R that we struggled to produce acceptable results. At 37.9%, we know our model is not making a random decision, but it is still very weak. With deep networks we saw a big jump to 50.17% accuracy, which is very impressive. However, many different techniques such as regularization vai dropout and l2 were needed to control the model from overfitting. Therefore, we came onto 2 opposite problems, one where we did not have enough representational power to train a neural network and one where we had too much and it overfit too often. In the end, we saw that CNNs found the middle ground between these two results.

CNNs are very effective in reducing the number of parameters while not reducing the quality of the model. This allows for efficient and accurate image recognition. The reduction of parameters allows our model to be less complex, prevent overfitting, and be trained easier. CNNs also allow for sharing weights. This concept is what makes CNNs 'convolutional'. With this, we are able to learn the filters (features of each image) and predict based on these filters to recognize the images. The other methods were not as powerful to be able to reduce the number of parameters and learn particular features of the images.

We are satisfied with this result of 74.44%, but we are sure this value could be improved. With more fine tuning in the number of filters and convolutional/max-pooling layers, we should be able to increase the accuracy further.