Assignment 1

## 4. Explain what a residual network is, and the basic motivation for using it. Also explain what the main elements of resnet34 are and resnet50. How many layers, how many neurons total, how many weights; and then anything else you want to say.

Most convolutional neural nets demonstrate a high performance on image classification. Research shows that deeper the networks, better the performance. However, deeper networks take longer to train and sometimes don’t converge. Our discussion in class regarding Gradient Descent led me to believe that vanishing gradient could be one of the key reasons why deeper networks don’t converge. This is one of the main reasons that most CNNs use SGD and try to solve for the weights using back propagation.

Kaiming He and his colleagues at Microsoft Research Asia ran into similar problems and introduced the solution of residual networks. The key difference between residual neural nets and conventional CNNs is the way in which the output of previous layers connects to the output of new layers. As the name suggest, a residual network learns from “residuals” as opposed to only learning contributing features. In other words, ResNet50 subtracts feature information from the input of a layer to learn about the residuals. The image on the left shows how a 34-layer plain CNN varies from a 34-layer residual network

Another reason why ResNet gained popularity is because of over fitting. Larger networks can model more complex problems, but the risk of overfitting is higher too. Residual networks, however, are smaller networks with fewer parameters and don’t overfit as much. In a residual setup, you would not only pass the output of layer 1 to layer 2 and on, but you would also add up the outputs of layer 1 to the outputs of layer 2.

One major limitation of ResNets is the number of samples you can feed to it. Most cloud computing platforms limit the number of input samples to 80K for ResNets and hence other algorithms like VGG16 / VGG-19are more popular when

## 5. Transfer learning using Fast.ai and create\_cnn: Please explain how pretrained resnet34 is modified to get the network that the notebook ultimately trains (i.e., explain what are the last layers that are added).

## 6. Download a NOT pre-trained resnet34, and then by playing with the number of epochs and learning rates (possibly different learning rates across layers), see how low you can get the error. Can you get below 20%?

## 7. And for the main part of this HW: download (and label) your own data set of your choice, create a classification problem, and then use the main tools/ideas of this notebook to build a classifier. It does not need to be a multi-label classifier.

## For getting data, you may want to refer to the discussion here, for various tools that could be useful: https://forums.fast.ai/t/tips-for-building-large-image-datasets/26688/

## 36.

Data was pulled based on