

Introduction

In this section of the course you will be building [ResNet](#), the winner of the 2015 [ImageNet Challenge](#) and one of the major innovations in deep learning. Specifically, you'll recreate [Version 2](#) of ResNet, which switches the ordering of certain layers from Version 1 for better performance.

A. Deep learning obstacles

While deep learning can be a great tool for solving modern day problems, there are still obstacles to overcome when building our models. Since deep models (models with many layers) tend to have numerous weight parameters, they take a long time to train. Furthermore, because deep models extract more hidden features than shallow models, they have a higher risk of overfitting the training data.

Nevertheless, the above two issues can be dealt with relatively seamlessly. Using modern day [GPUs](#) and distributed training (i.e. training a model in parallel across multiple processing units), models can be trained in significantly less time. We have also already gone over a few tricks to reduce overfitting, such as max pooling layers, dropout, and global average pooling.

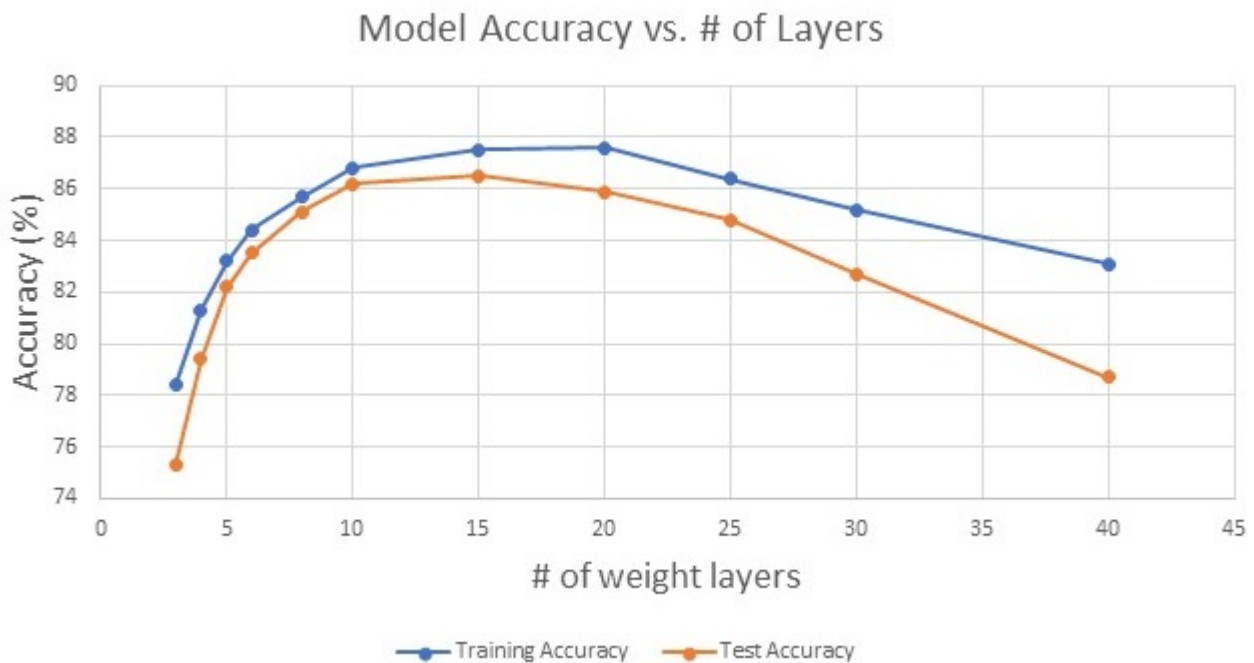
Although issues in deep learning continue to pop up, researchers are usually able to solve them. This problem solving process oftentimes leads to groundbreaking progress and revolutionary discoveries. In fact, the ResNet model that we will be building was created specifically to solve a problem in deep learning known as *degradation*.

B. Degradation

After the breakthrough made by AlexNet in 2012, people realized that adding more layers to their CNN improved the model's performance. Building off the AlexNet architecture, which used 5 convolution layers and 3 fully-connected layers, researchers began creating models with double digit layers. In 2014, the top two models in the ImageNet Challenge both had around 20 weight

layers. However, in the process of adding more and more layers to models, the degradation problem was discovered.

Degradation of deep neural networks refers to the plateau, followed by rapid decline, in model accuracy when increasing model depth beyond a certain point.



Example of model degradation. The blue line represents the model accuracy on the training set while the orange line represents accuracy on the test set.

This problem was identified by the creators of ResNet, and they realized it was not caused by overfitting. In overfitting, a model will have incredibly high accuracy on the training set but low accuracy on the test set. The problem of degradation, however, caused the model to have *lower accuracy in both training and testing*.

C. ResNet

The ResNet model architecture was developed specifically to solve the degradation problem. Similar to how the SqueezeNet model (see the **SqueezeNet** section) uses fire modules as its main building block, the ResNet model also has a main building block. Each of its building blocks incorporate *residual learning* to counteract degradation, which we'll discuss more in later chapters.

With the ResNet architecture, we're able to again see model improvement by adding more weight layers. In fact, the ResNet model that won the 2015 ImageNet Challenge had a staggering 152 layers!

