**Data Science 450**

Project: Deep learning on CIFAR-10 Dataset

Anish Mohan

**Checkpoint 2:**

1. **Introduction and Goals**

Deep learning is one of the exciting new areas in Machine learning. My goal is to use this Capstone project as a motivation to learn about Deep learning. I intend to use Tensorflow as the platform for deep learning experiments as there is a good online community support for it.  For the Deep learning project, I am planning to use the CIFAR-10 dataset. I have some background in Computer Vision and Image processing, so I am hoping to leverage some of the knowledge for pre-processing image data.

1. **CIFAR Dataset**

CIFAR Dataset [1] is subset of the Tiny Images dataset [2] . Tiny images data set contains about 80 Million images of size 32x32. Each image in the original dataset is labelled with non-abstract nouns [3] obtained from Wordnet lexical database. CIFAR-10 consists of 60,000 images of size 32x32 pixels. These images have objects/content that belong to the following classes: Airplanes, automobiles, bird, cat, deer, dog, frog, horse, ship and truck. Goal would be to train Deep Neural networks and be able correctly classify a test set.

1. **Progress Description:**
   1. **Tensorflow Installation**

I installed Tensorflow on my machine following instructions from here: <https://www.tensorflow.org/install/>

* 1. **Learning about Neural Networks/Back propagation.**

To learn about Convolutional Neural networks, I am using the notes and lectures taught in CS31N Course: <http://cs231n.stanford.edu/>

The video lectures are available [here:](https://www.youtube.com/watch?v=NfnWJUyUJYU&list=PLkt2uSq6rBVctENoVBg1TpCC7OQi31AlC)

So far, I have completed 4 lectures and notes. In the lectures, so they taught about K-Nearest neighbors and an introduction to Neural networks. They also taught the concept of backpropagation for neural networks.

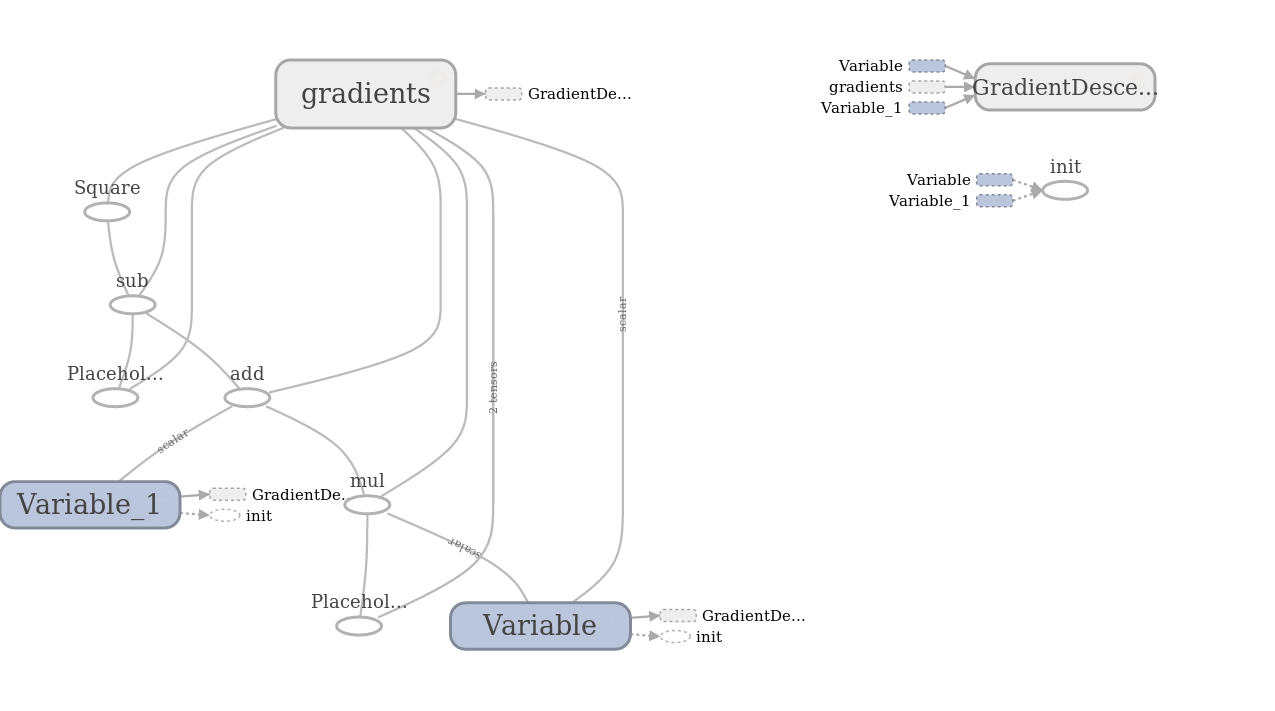
I will continue the lectures to learn about Convolutional Neural networks after some experimentation with Tensorflow.

* 1. **Linear Regression with Tensorflow**

I learnt that Tensorflow supports running many machine learning algorithms. Since, I am starting to learn from scratch, I thought it will be useful to run some basic algorithm like Linear Regression with Tensorflow. Here are the results from two experiments I have run so far.

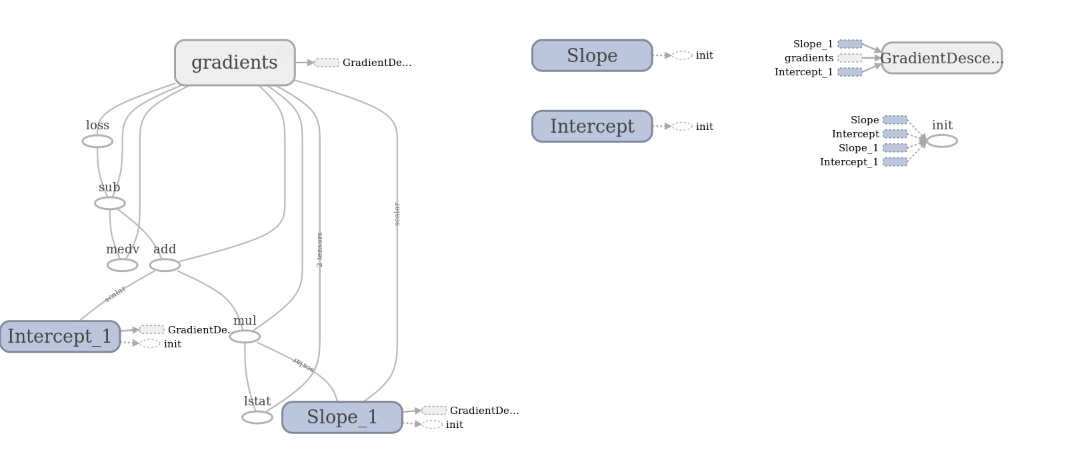
* + 1. **Experiment 1: Linear Regression on Synthetic Data**

For this experiment, I generated synthetic x and y data and tried to learn the slope and intercept of the linear regression line using gradient descent. Here is the graph generated from Tensorflow:

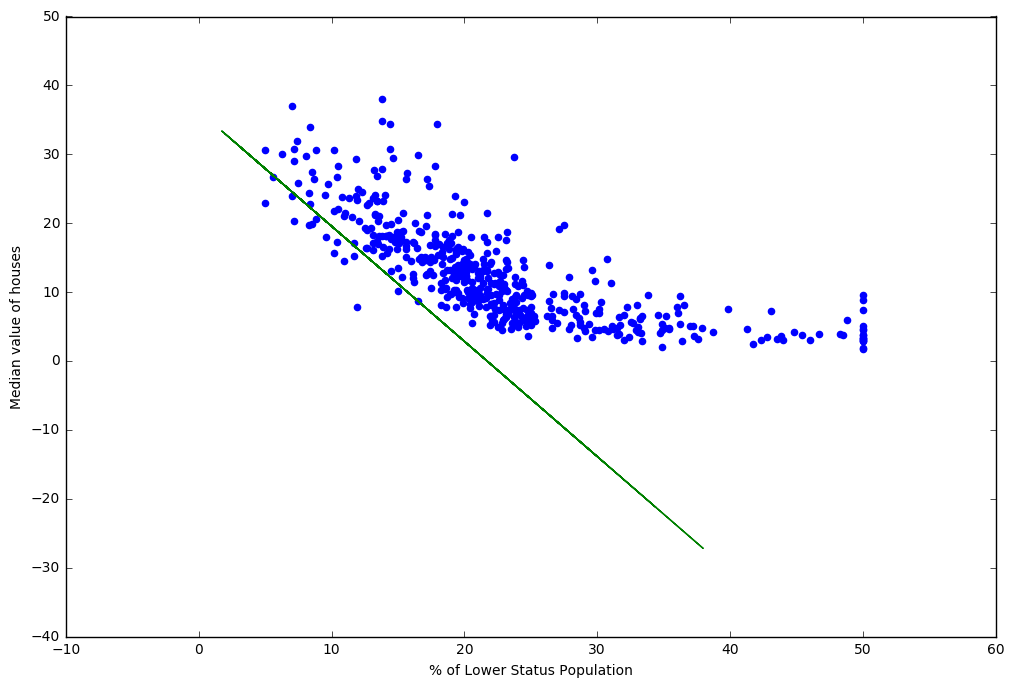


* + 1. **Experiment 2: Linear Regression on Boston Dataset**

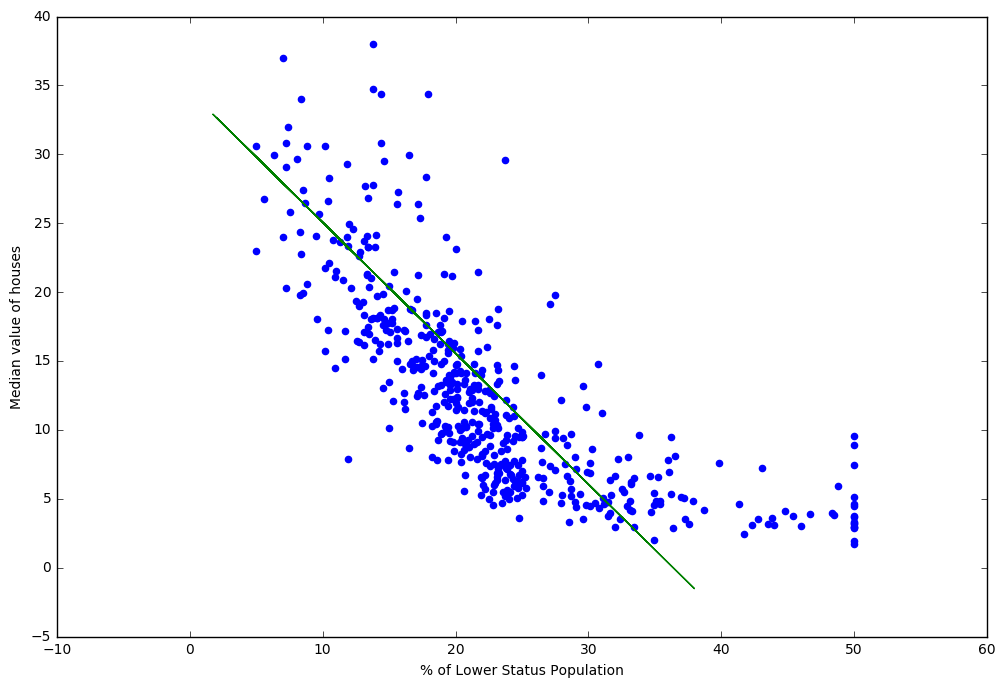
For this experiment, I experimented with the Boston dataset. In this dataset, I tried to model the median house value in a neighborhood based on the % of lower status population in the neighborhood. Here also, I used Tensorflow and gradient descent to find the optimal value of slope and intercept of the line. Here is the Tensorflow graph generated for this work:



Interestingly, I experimented with various learning rates and different number of epochs, however, the regression line fit does not seem to be very good. Here is the result generated using gradient descent:



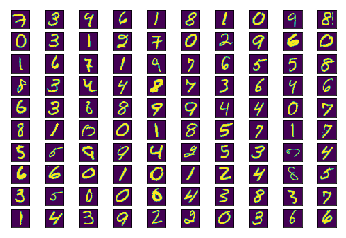
To compare, I generated another fit using Ordinary Least Square regression:



The fit from the OLS seems to be much better than the one generated with Gradient descent. Over the next 2 weeks, I will to investigate more and learn/apply Convolution Neural Networks on CIFAR data

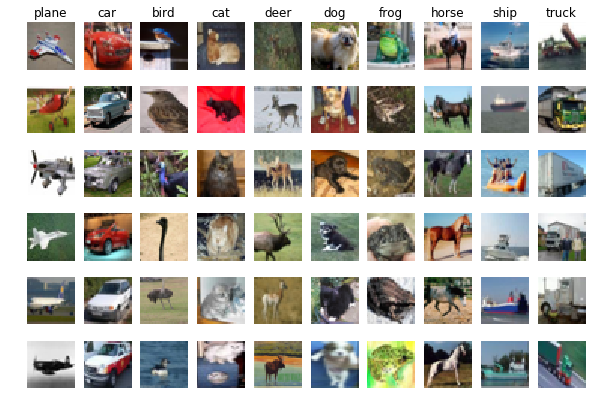
**5. MINST With Tensorflow**

To get a working understanding of the Deep learning, I built a neural network model for classifying MINST data.



With the MINST data and simple neural network model, I could get an accuracy of about 92% on the training and test data.

**6. CIFAR With Tensor Flow**

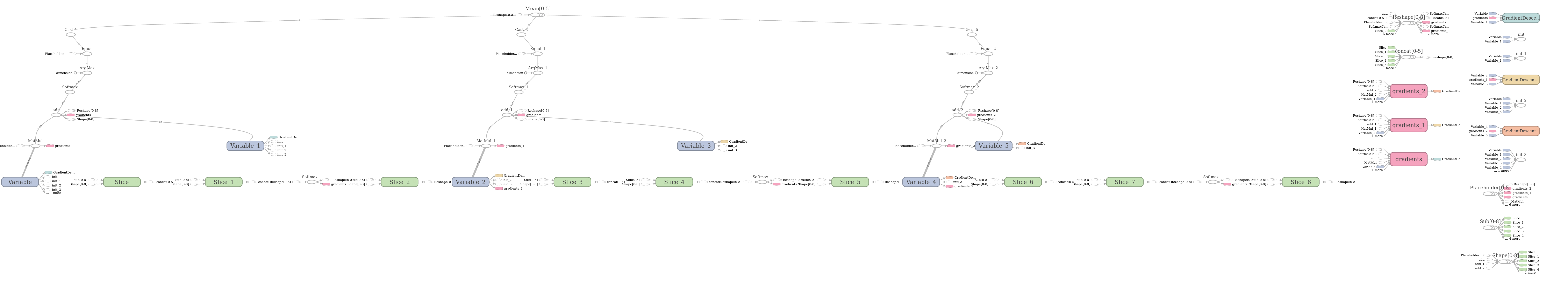


I built 3 model with Tensor flow for this:

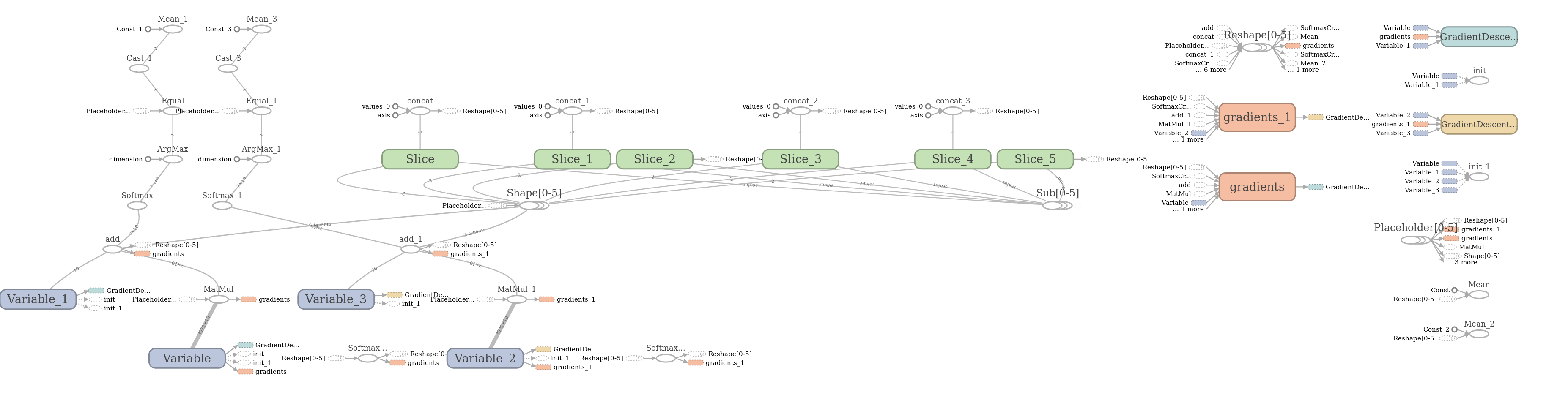
1. Single Layer Neural Network Model
2. Multi Layer Neural Network Model
3. Convolution Neural Network Model

Here is the Tensor Flow graph for all the three modes

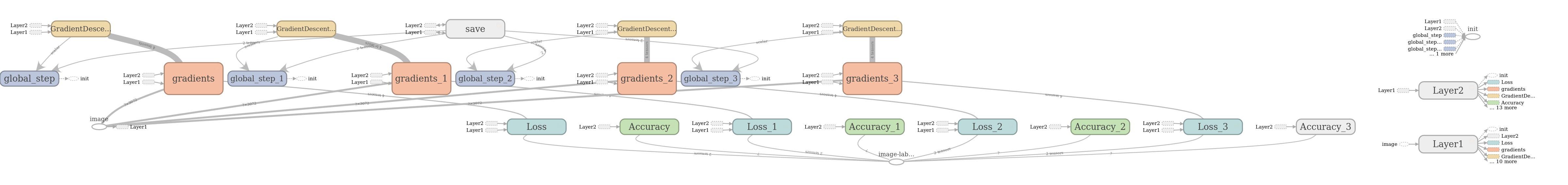
**Single Layer Neural Network Model**

******

**Multi Layer Neural Network Model**

******

**Convolution Neural Network Model**

******

1. **Accuracy Results:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Simple Neural Network** | **Multi Layer Neural Network** | **CNN Model** |
| **Training Accuracy** | 23% | 23% | 14% |
| **Test Accuracy** | 24% | 10% | 10% |

1. **Learnings:**

* Linear Model with Stochastic Gradient proved to be a workable solution for CIFAR data set.
* Convolution Network has many parameters that can impact the performance of the Deep network.
* My implementation of CNN is faulty as the results seem to be low even after 200-800 iterations.
* CNN is expensive and running beyond 800 iterations on a local machine is not practical

1. **Reference:**
   1. <https://www.cs.toronto.edu/~kriz/cifar.html>
   2. <http://groups.csail.mit.edu/vision/TinyImages/>
   3. <http://people.csail.mit.edu/torralba/publications/80millionImages.pdf>
   4. <http://cs231n.stanford.edu/>
   5. Lectures: <https://www.youtube.com/watch?v=NfnWJUyUJYU&list=PLkt2uSq6rBVctENoVBg1TpCC7OQi31AlC>
   6. <https://www.tensorflow.org/install/>