**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. **Project Plan and Progress**

|  |  |  |
| --- | --- | --- |
| **Target Date** | **Work Item** | **Status** |
| 5/8 | Tensorflow installation and getting machine ready | Completed |
| 5/12 | Learn about Neural Networks and Back propagation | Completed. |
| 5/15 | Learning Numpy  Tensorflow basics and simple code semantics | Completed |
| 5/18 | Simple machine learning models with Tensorflow (Linear Regression) with basic data sets | Completed |
| 5/22 | Learn about implementing Neural Networks and theory behind Convolution Neural Networks | Not Started |
| 5/25 | Experimenting with CIFAR to implement Logistic Regression model | Not Started |
| 5/26-5/29 | Travel | Not Started |
| 5/31 | Implementing Tensorflow CNN model on MNIST data. | Not Started |
| 6/6 | Implement Tensorflow CNN model with CIFAR dataset. | Not Started |
| 6/7 | Create Project Report | In-Progress |

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.

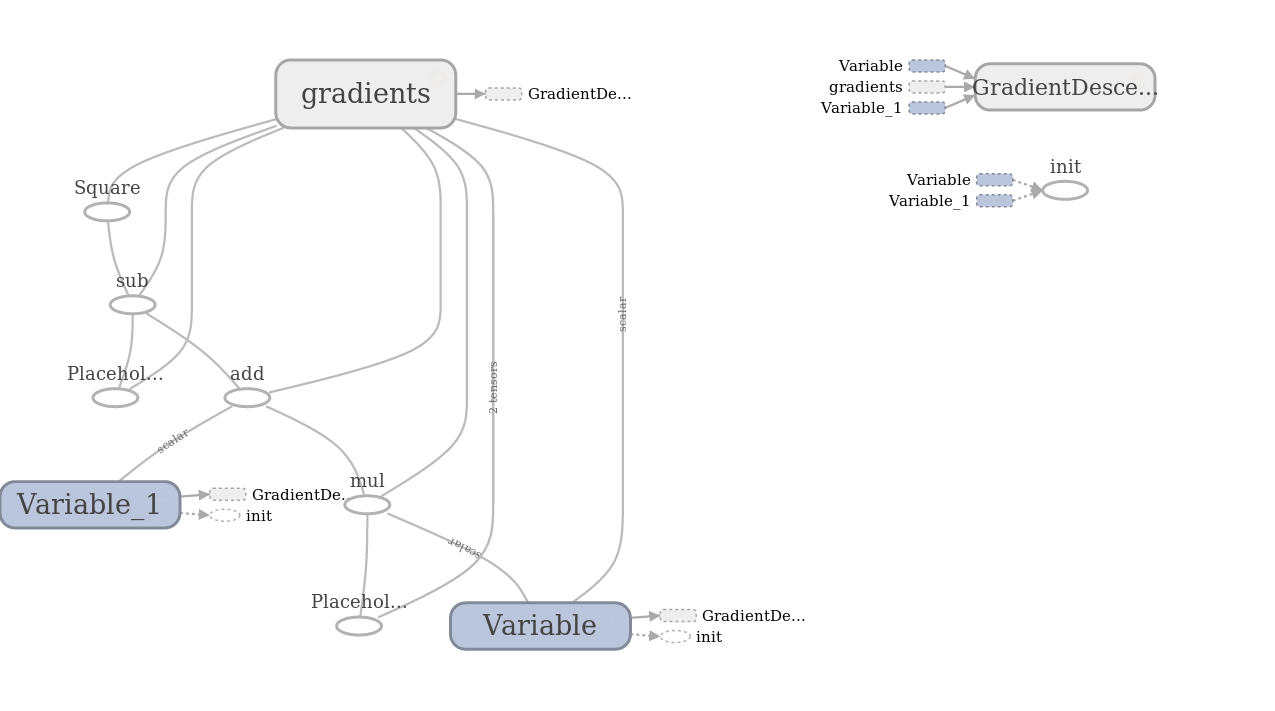
I will continue the lectures to learn about Convolutional Neural networks after some experimentation with Tensorflow.

* 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.

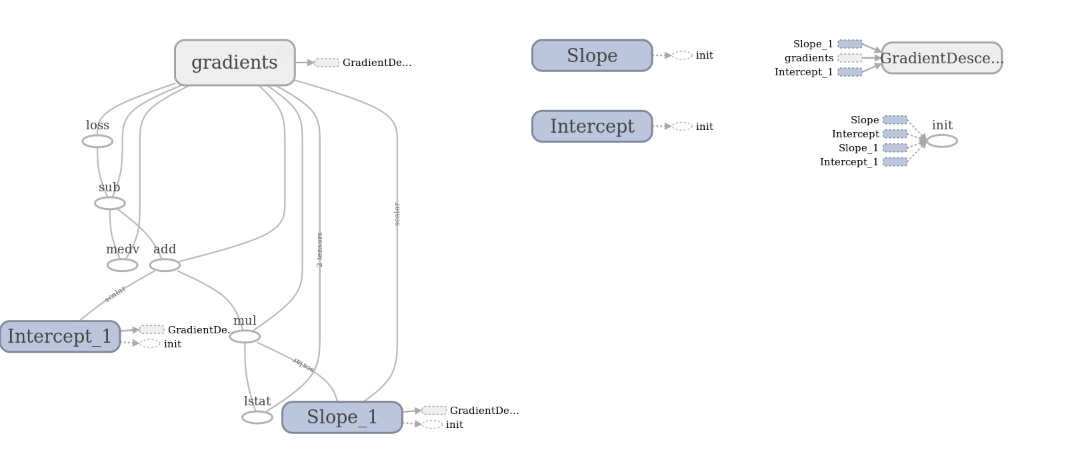
* + 1. **Experiment 1: Linear Regression on Synthetic Data**

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:

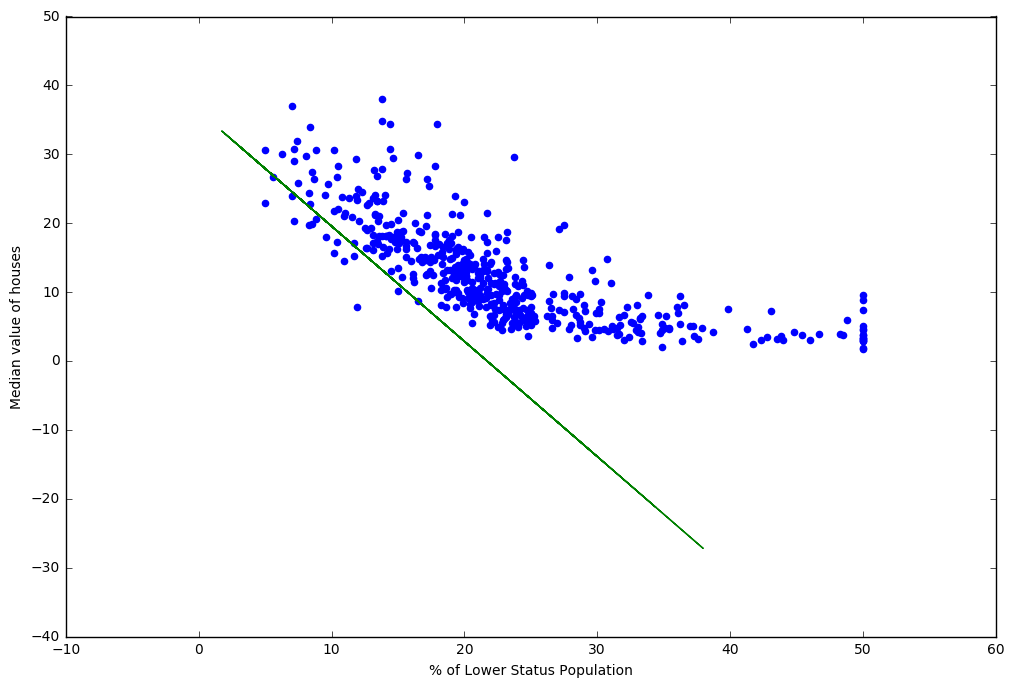


* + 1. **Experiment 2: Linear Regression on Boston Dataset**

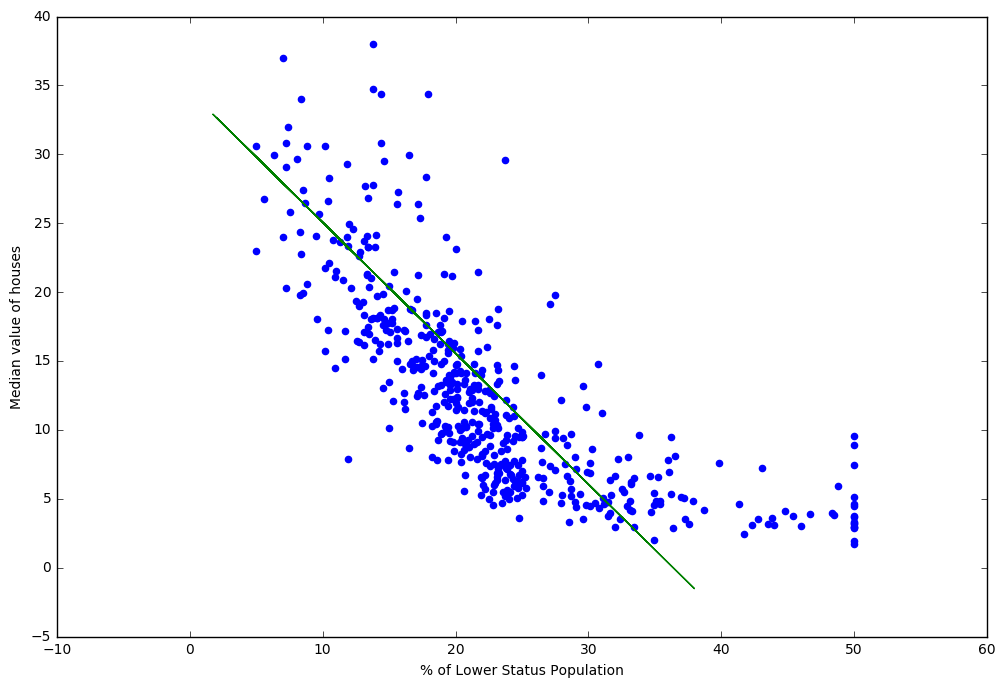
For this experiment, I experimented with the Boston dataset. In this dataset, I tried to model the median house value in a neighborhood based on the % of lower status population in the neighborhood. Here also, I used Tensorflow and gradient descent to find the optimal value of slope and intercept of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

1. **Reference:**
   1. [https://www.cs.toronto.edu/~kriz/cifar.html](https://www.cs.toronto.edu/%7Ekriz/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/>