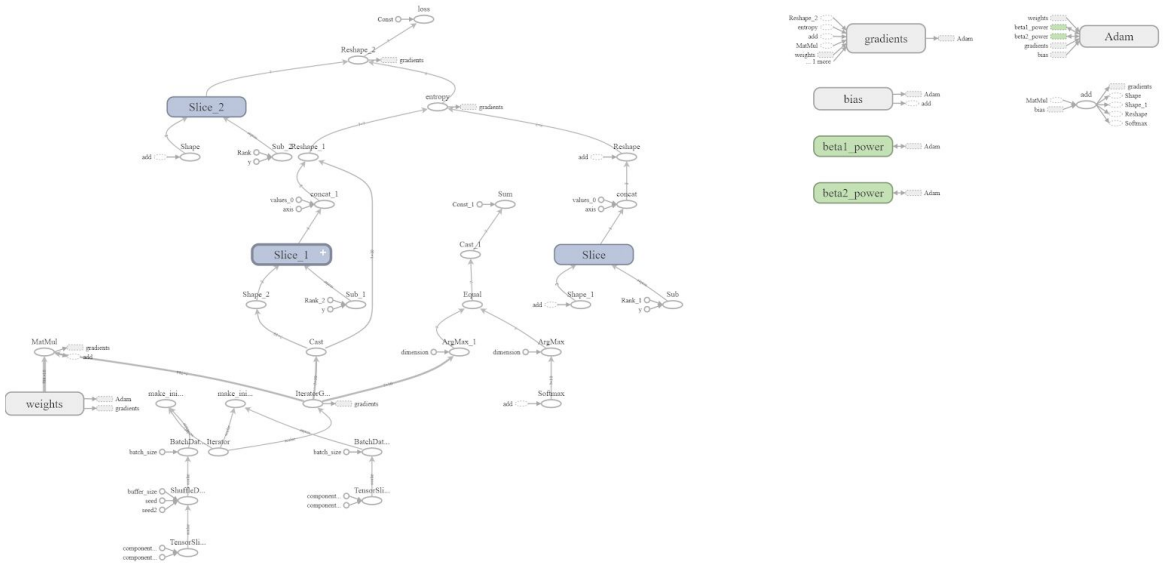
CSE 598 - Data Intensive Systems for Machine Learning
Assignment 1 - Report
Parth Doshi - pdoshi4@asu.edu

Task 1

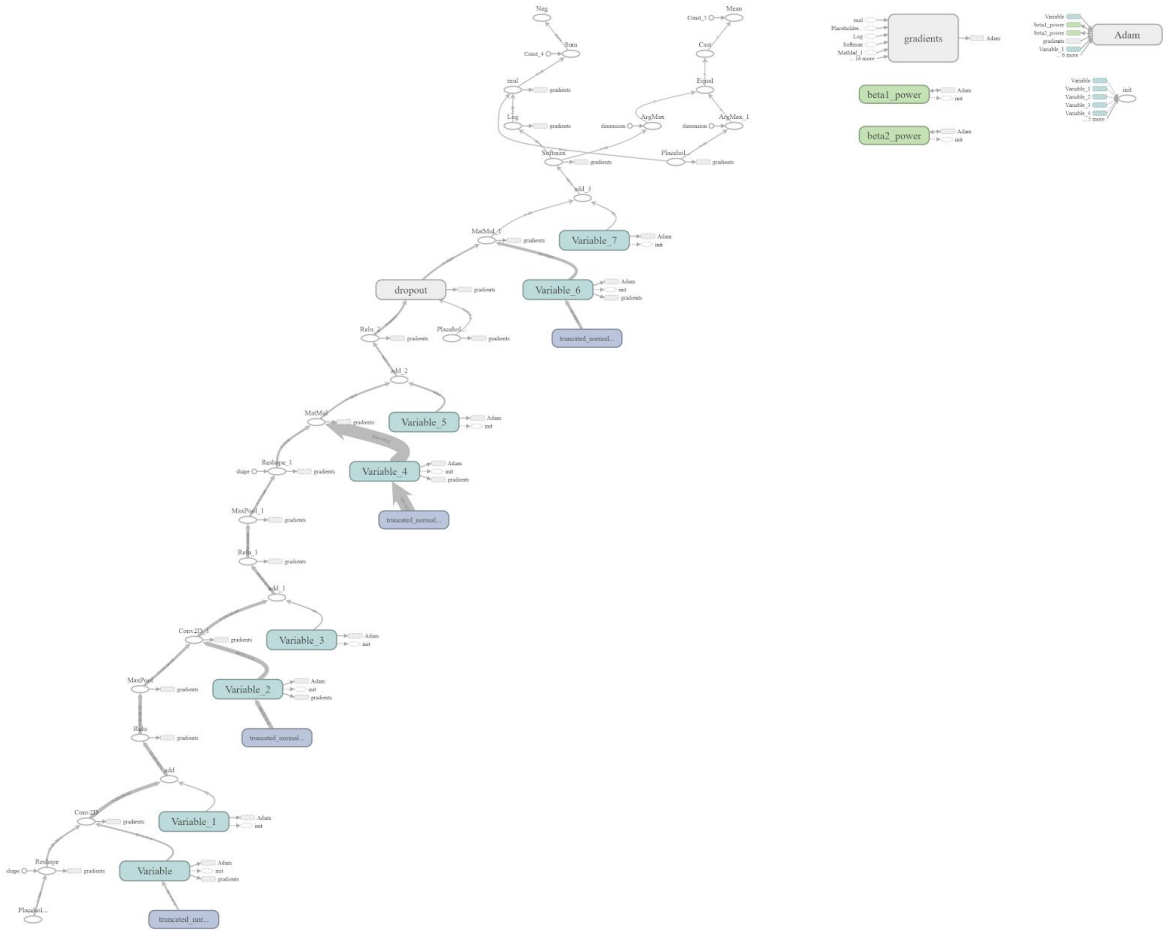
- **Description**: Logistic Regression model is essentially a single layer perceptron where the input is combined with weights and added to the bias while softmax cross entropy loss function is used with Adam optimizer to minimize loss.
- **Observed TensorBoard Graph**:



- **Observed Accuracy**: 92.11%
- **Time spent in task 1**: 8 hours
- Challenges encountered: I encountered problems mainly with the installation of Tensorflow on my local device but once I was able to figure it out, I only had some difficulty with the syntax of the framework. Luckily lots of resources and tutorials associated with tensorflow are available online for learners and I was able to figure out the resulting syntax through them.

Task 2

- **Description**: From the logistic regression model, we see that 92% accuracy is not good enough for the MNIST dataset. Convolutional Neural Networks(CNN) are the best models for image classification currently with many of them currently delivering over 99% accuracies for the MNIST dataset when combined with data processing and augmentation techniques. So, after looking up various CNN architectures online and on Kaggle competitions, I decided to build a simple model with 2 convolutional layers with 5x5 filter supplemented with max pooling. After these layers, I have added two fully-connected layers with drop out of 50%. Applying an entropy loss function with Adam Optimizer once again, I trained the model for 3000 epoches with a batch size of 50.
- **Observed TensorBoard Graph**:



- **Observed Accuracy**: ~98.23%
- **Time spent in task 2**:42-46 hours
- **Challenges encountered**: For this task, I had to understand the convolution neural networks architectures soundly. How to make your model perform better with different types of parameters hypertuning, different architectures of CNN being used was something I found challenging. Also, the tensorflow framework syntax proved to be a difficulty once again but I'm sure with more exposure to this framework, I'll not find it a challenge in the future. I referred various tutorials on Github and Kaggle to understand the best approach for the CNN architecture. Playing around with epoches, batch size and learning rates was an interesting problem.
- **Referrences** -https://github.com/floydhub/tensorflow-notebooks-examples