

UNIVERSITY OF MISSOURI KANSAS CITY

# CS5590 APS – DEEP LEARNING PROGRAMMING

ASSIGNMENT I

MOHAMMED, MAHMOOD UDDIN (UMKC-STUDENT)

11-2-2017

## I. INTRODUCTION

This report is based on Logistic Regression. The main difference between linear regression and logistic regression is the variable dependency. In linear regression the output is continuous and independent. Where as the output of the logistic regression is finite and dependent.

Linear regression provides a linear functionality. However, the logistic regression is not linear but provide a change in X and its ratio as an exponential function. The purpose of logistic regression is to provide probabilities with linear regression. Since predicting the class probabilities is better than prediction of classes itself. Linear regression output as probabilities may give an error since, the output can have a negative value and value greater than 1. But this is not possible in case of probability. Since, regression can provide probabilities, logistic regression was introduced.

In this program, we are using MNIST (Mixed National Institute of Standards and Technology database) dataset of handwritten digits. It consists of 60,000 and 10,000 testing and training sets. The image size of digits is 28x28 pixels. In addition to this, the image is converted to one dimension numpy array of 784 features (28\*28).

## II. OBJECTIVE

To implement logistic regression with new data set which is not used in class.

## III. APPROACH/METHOD

The dataset is taken from Yann LeCun database. It consists of handwritten digits. Linear regression is carried on the dataset. In Logistic Regression, the probability for given the features x is given as:

$$P(y = 1|x) = \frac{1}{1 + \exp(-(W^T x + b))}$$

where W are the model weights for the features x. b is the **bias** of the model.

The logistic function is employed to change the outcome of the linear model  $y=mx+b$  from any number into the range of 0 and 1 that can be consider as probability.

We are using GradientDescentOptimizer which means that our update rule is gradient descent.

#### IV. WORK-FLOW

Following steps were performed in the program:

1. Import the data from Yann LeCun database.
2. Import tensorflow
3. 30 training epochs were used and batch size = 50.
4. Define placeholders x and y, which are graphs input.
5. Set weight and bias for the regression.
6. Define model and cost.
7. Define optimizer with GradientDescentOptimizer. Selected learning rate is 0.01
8. Start tensor session.
9. Write summary of events, in order to get graphs on tensor board.
10. For and if loop for simulating different epochs and batches.

#### V. DATASET

The MNIST database (Mixed National Institute of Standards and Technology database) is used. It is a database of handwritten digits.



## VI. PARAMETERS

Following parameters were set:

learning\_rate=0.01

training\_epochs = 25

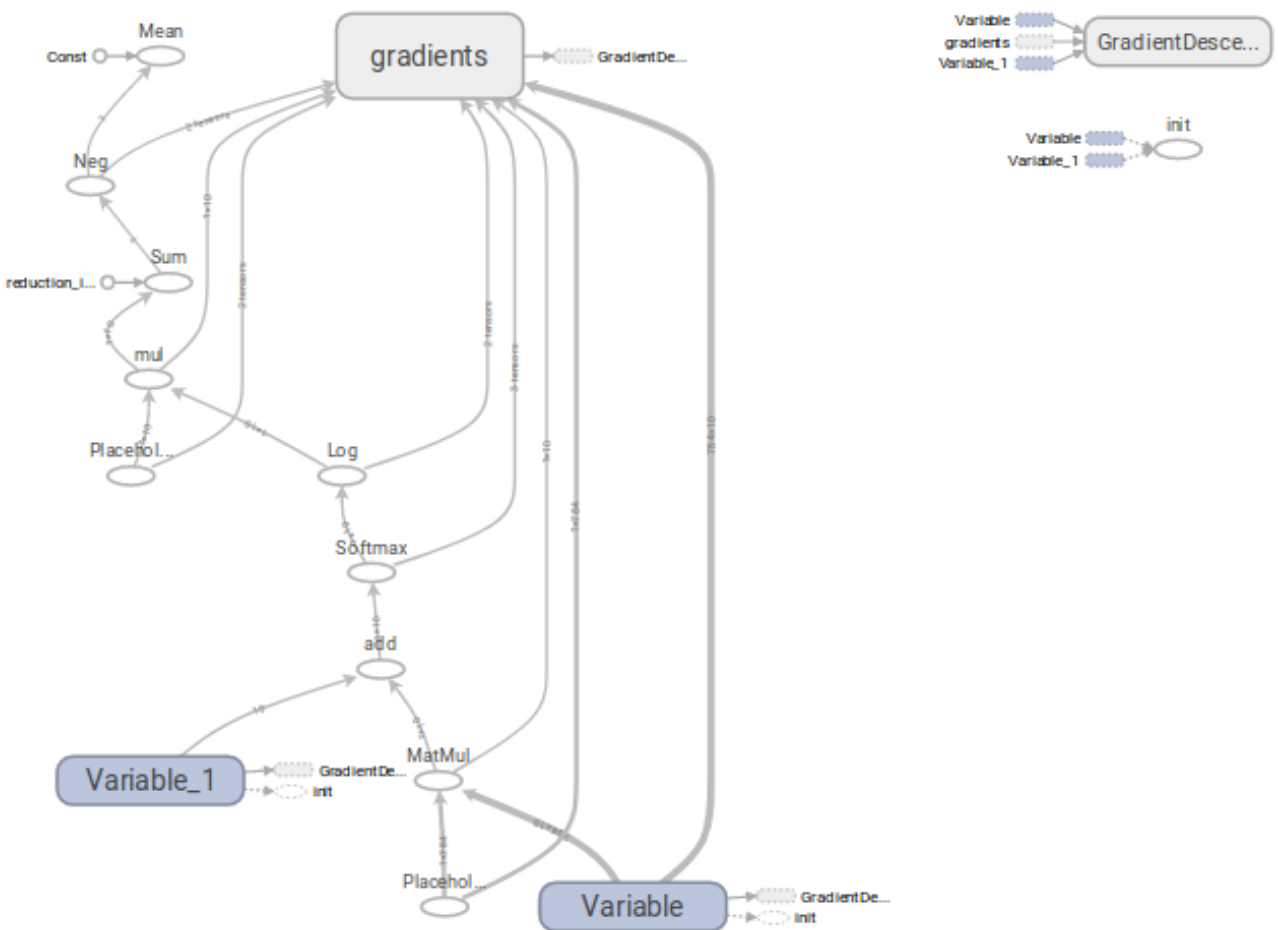
batch\_size = 100

display\_step = 1

reduction\_indices=1

## VII. Evaluation & Discussion

Following graph is obtained on the tensor board after execution of the program. Summary of events are required to get the graph on the tensorboard. For this we create a new directory logistic\_reg inside the graph directory.



Following steps are involved to get the graph:

1. Run python program on your terminal .
2. Type below command to get http link.  
`tensorboard --logdir="./graphs" --port 6006`
3. Open the link in browser.

Total epochs are 30 so the for loop in the program iterates 30 time to get different weights and bias. This optimizer is improved for 60 batch size. GradientDescentOptimizer is used in the logistic regression python program.

## VIII. Conclusion

The logistic regression program with MNIST datasets is performed. The total runtime of program in Ubuntu 17.04, i7 processor is 47.16 seconds.

## IX. REFERENCES

1. <https://www.tensorflow.org/tutorials/wide>
2. [http://www.machinelearningguru.com/deep\\_learning/tensorflow/machine\\_learning\\_basics/logistic\\_regression/logistic\\_regression.html](http://www.machinelearningguru.com/deep_learning/tensorflow/machine_learning_basics/logistic_regression/logistic_regression.html)
3. <http://yann.lecun.com/exdb/mnist/>