**DEEP LEARNING**

## LAB ASSIGNMNET – 1

**INTRODUCTION:**

In this assignment logistic regression is implemented for the desired dataset using tensor flow.

**Objective:**

The key objective here is to focus on logistic regression using tensor flow and to learn how to plot the graph using tensor board.

**METHODS:**

For the logistic regression I have considered iris data and used pandas to set into the data frame and also installed sklearn package to load the iris data. Here I have used one hot coding method for easier computation. After the data framing I have implemented logistic regression and also calculated loss function to analyze the accuracy. In this data is divided into training and testing set such as 80% of training data and 20% of testing data. Softmax cross entropy is used for computation and also I have used optimizer to minimize the loss. At last tensor flow session is created for the logistic regression model and also calculated the accuracy.

# WORKFLOW

* Installed required packages
* Fitting into data frame
* Labels extraction
* Divided into training and testing
* One hot coding
* Implemented logistic regression
* Optimizer to minimize the loss
* Calculated accuracy

**DATASETS:**

Iris data set is used to implement the logistic regression model. This dataset has three kinds of colors for petal and sepal length to store in n dimensional array.

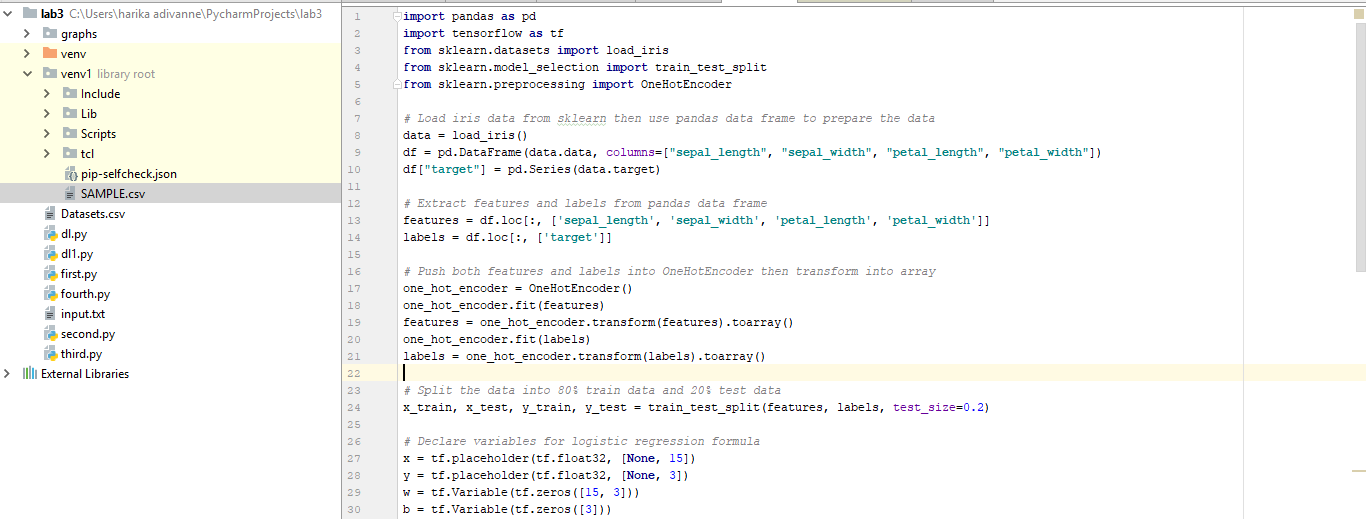
**PARAMETERS:**

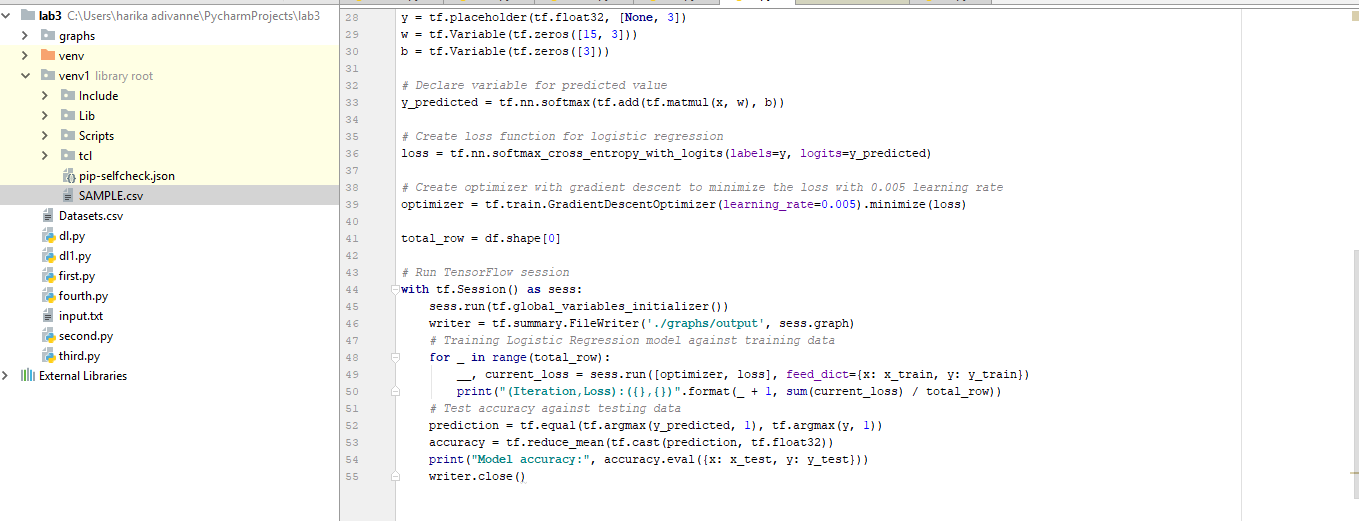
Number of iterations = 150

Optimizer rate=0.005

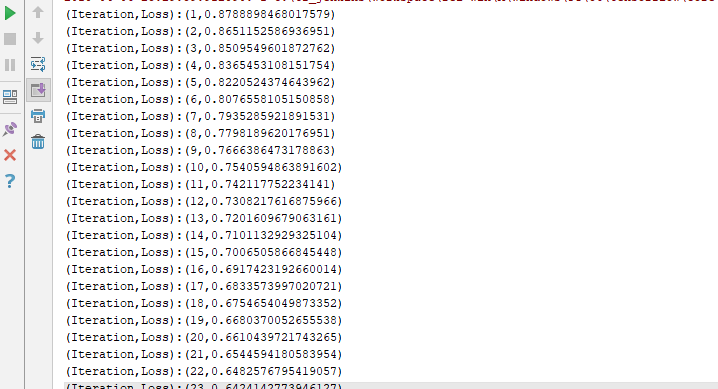
**EVALUATION:**

**Source code:**

****

****Here I have used packages like pandas and sklearn to fit the data into data frame and divided the data into training and testing and used one hot encoding to make computation easier. To implement logistic regression here I have used iris data and calculated accuracy.

**OUTPUT:**

****

****

**CONCLUSION:**

The accuracy results in 0.9 for the parameter of 0.003 if the parameter differs, accuracy differs and for the less parameter value accuracy increases suppose for 0.0001 the accuracy is 0.9666. In this I have used 150 iterations.