Assignment – 2 Report

Run the codes using python3 ML2_parta.py for part a and ML2_partb.py for part b

In part a an extra input for chosing the function as sigmoid and tanh to be given

Output is produced after every 1000 SGD iterations to ensure that code is running fine

Statistics as asked are produced after every complete pass over dataset

By our analysis

Softmax function performs the best as it reaches high accuracy very fast and gives low values of mse

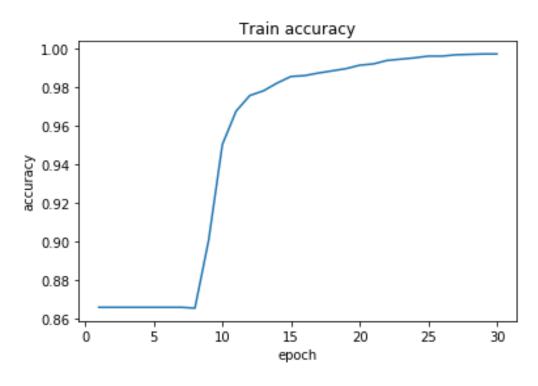
After that tanh fuction performs best and gives near comparable results to softmax in our simulation

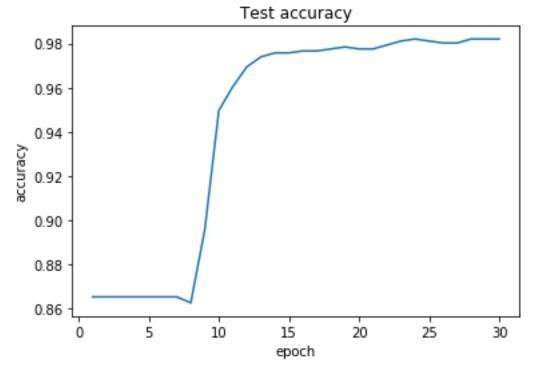
Sigmoid function performs worst in our dataset and takes atleast about 15 epoch to give results comparable to other functions. Its worst performance can be attributed to use of squared error loss than crossentropy.

Part a)

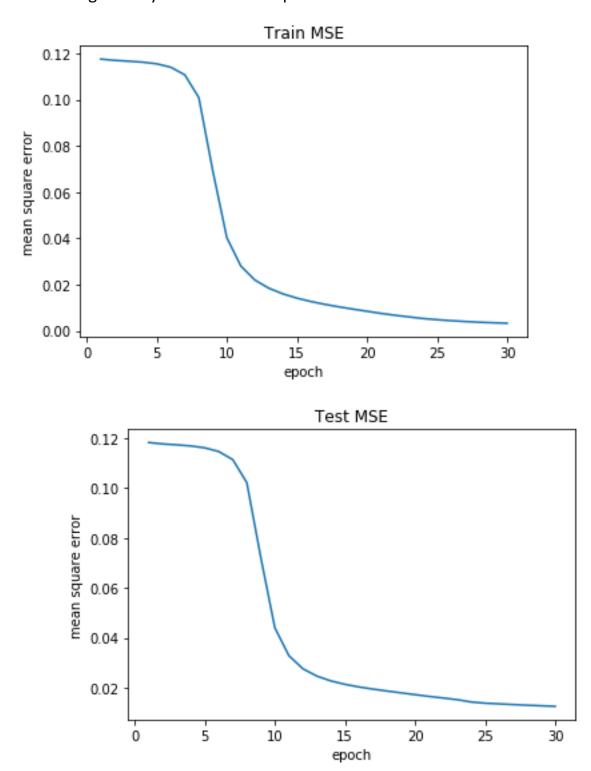
SIGMOID

Sigmoid function with suared error loss was slow to implement (popular implementation includes crossentropy function). Thus it remained at low accuracy in first few epochs but rose steeply once the learning slowdown stopped.





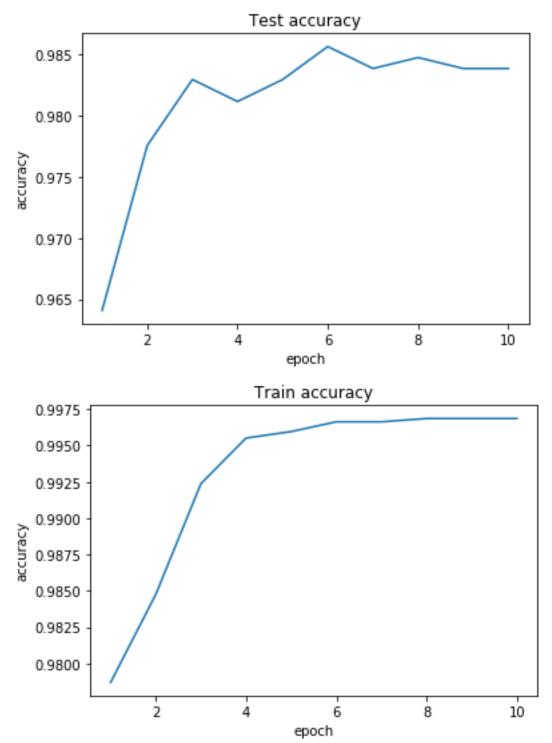
The error gradually decreased as expected



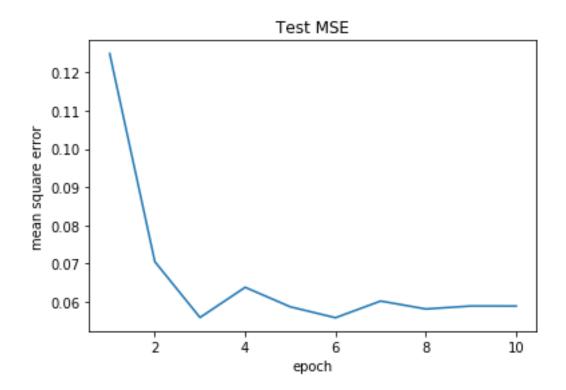
The optimal value of iteration in this case can be around 20 , as after that accuracy remained near about constant and MSE remains unchanged too $\,$

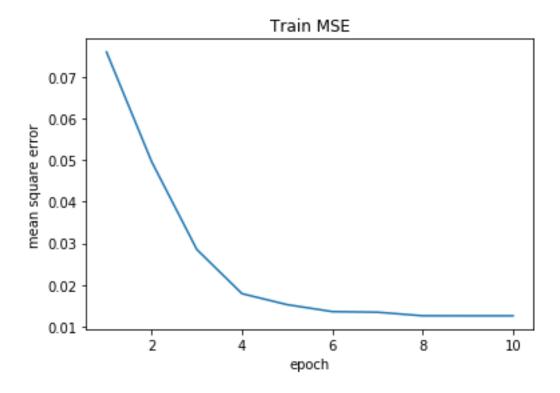
TANH

Tanh was faster to learn than sigmoid due to use of larger variance cost function which gives output between -1 and 1



The error is plotted as follows





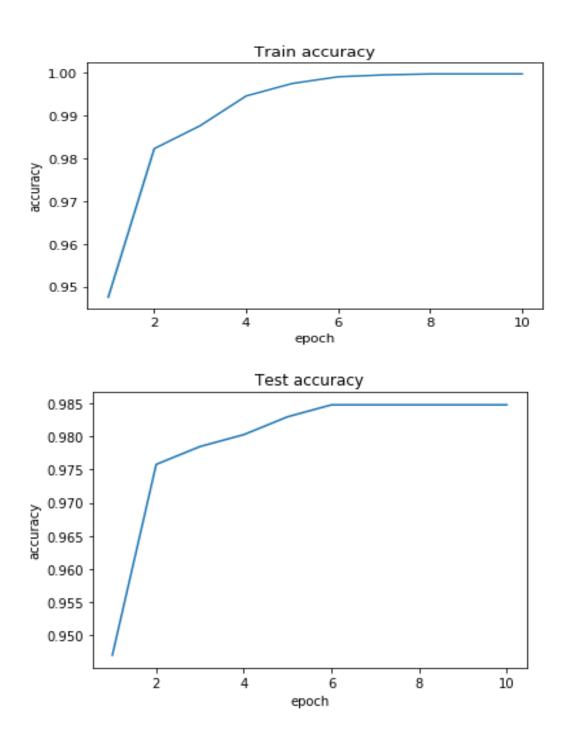
Optimal iterations can be 5 as accuracy and MSE saturated after that

Part B)

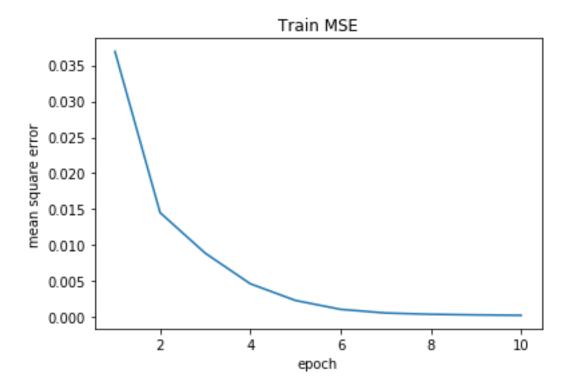
SOFTMAX

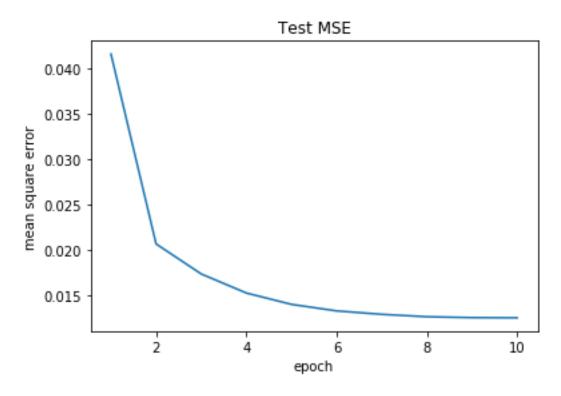
Softmax function gave the best results with reaching highest accuracy in least number of epochs.

This can be due to increased nonlinearity in final layer and categorical classification , which is needed for the particular problem



The MSE decreased as expected





The optimal value can be 6 iterations as after that both MSE and accuracy saturated