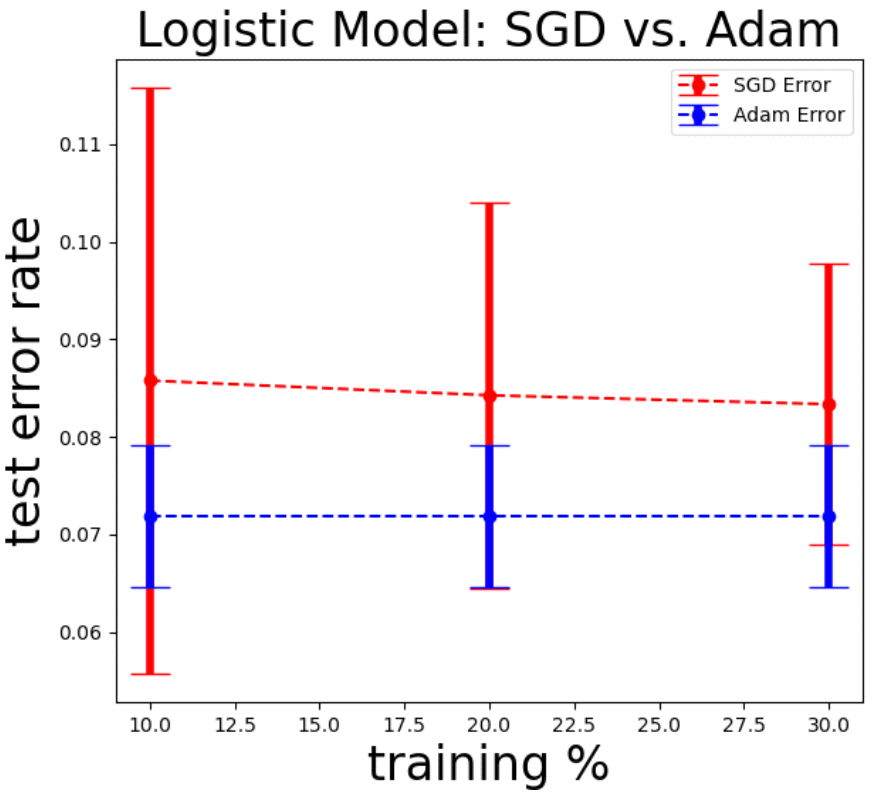
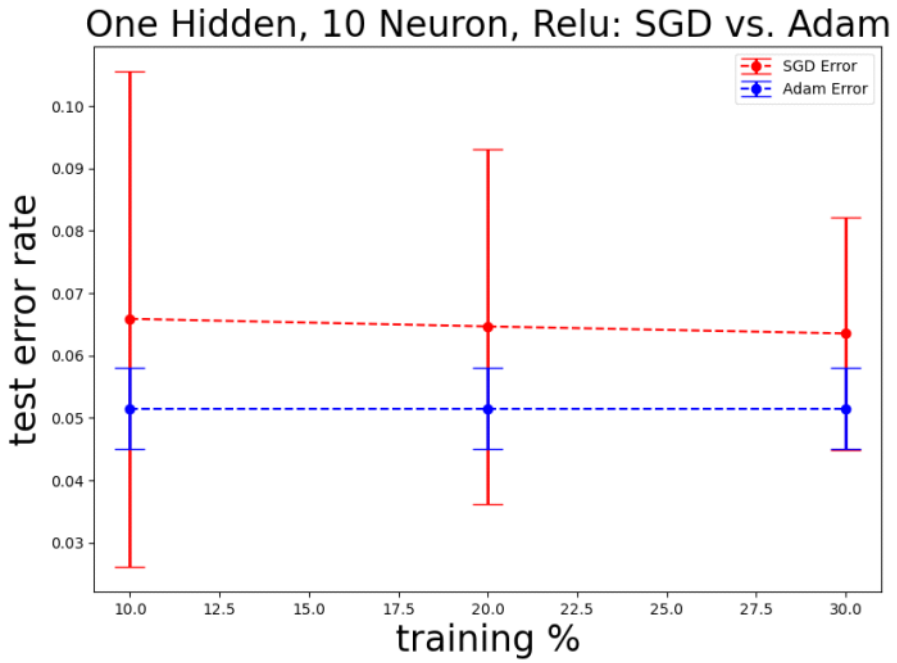
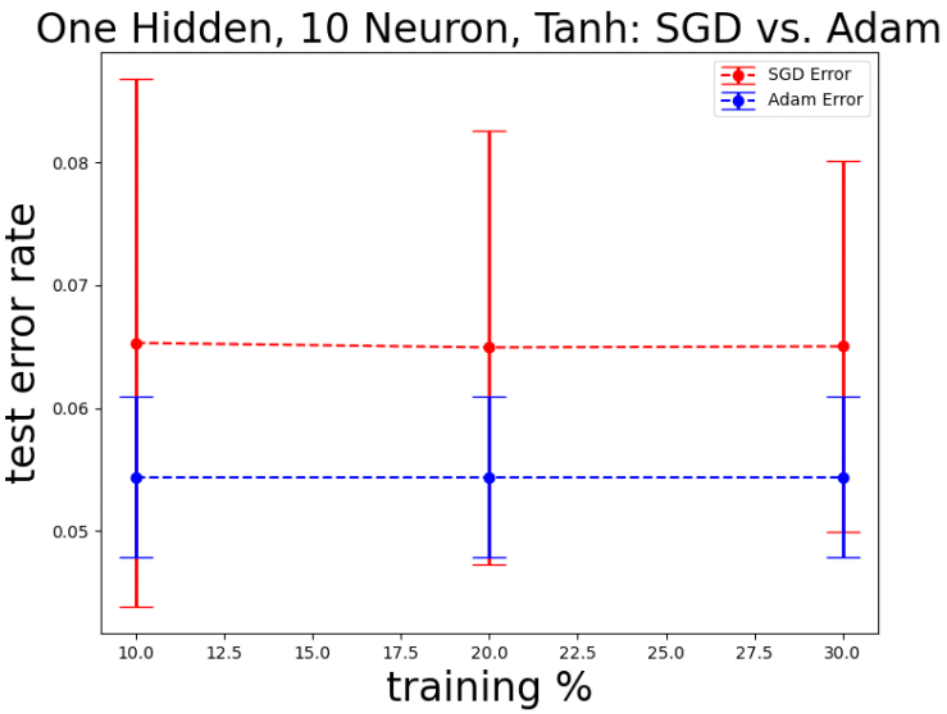
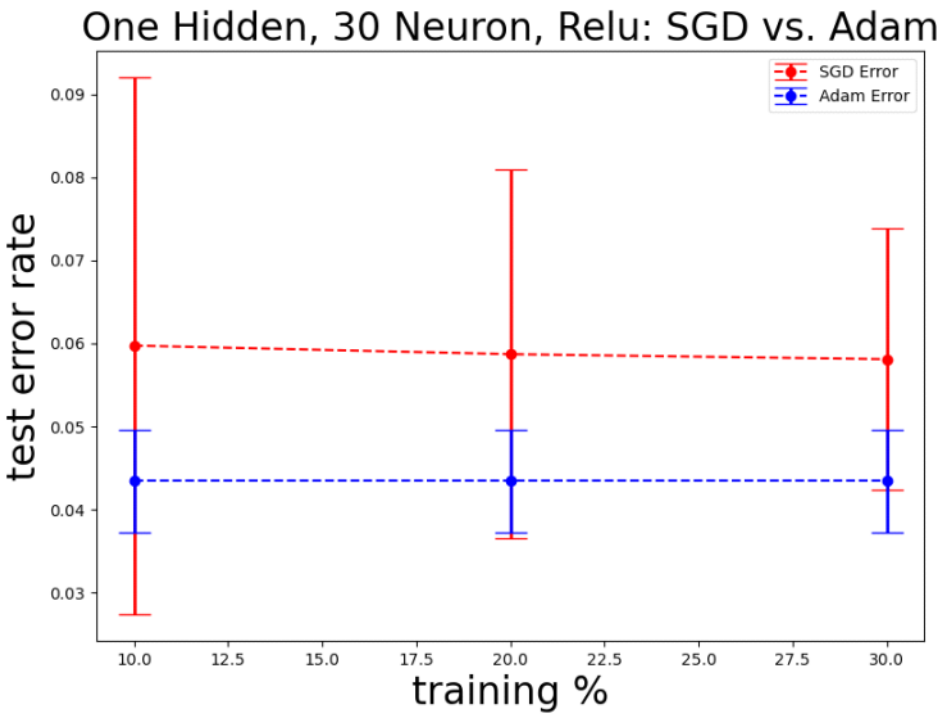
HW 6

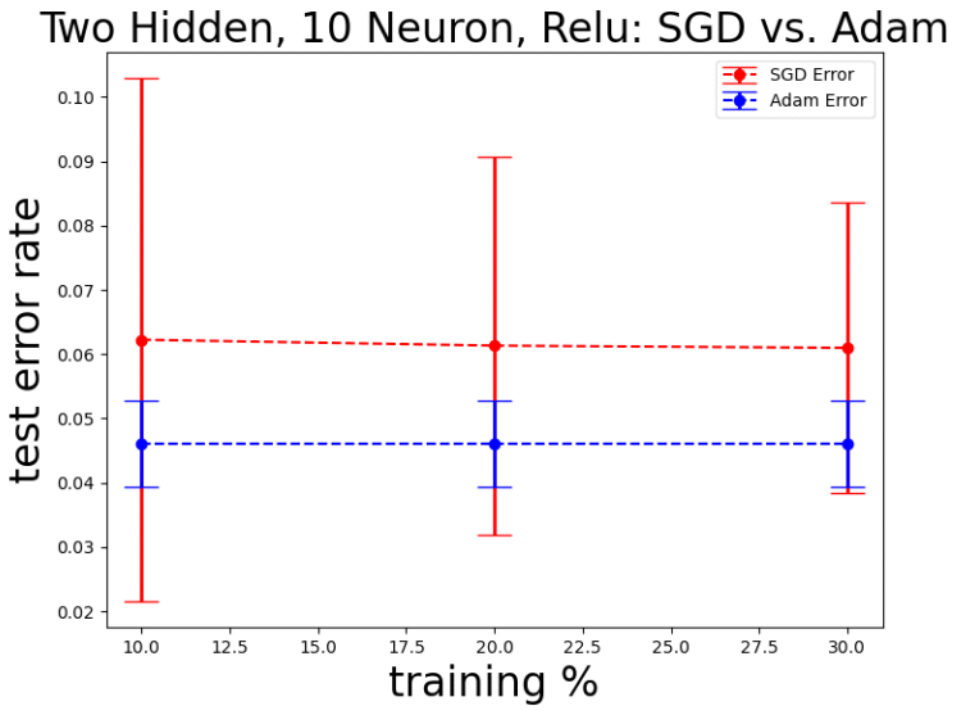
Tuesday, December 14, 2021 7:36 PM






One Hidden, 10 Neuron, Tanh: SGD vs. Adam
SGD Error
Adam Error
test error rate
0.08
0.07
0.06
0.05
10.0
12.5
15.0
17.5
20.0
22.5
25.0
27.5
30.0
training %


One Hidden, 30 Neuron, Relu: SGD vs. Adam
SGD Error
Adam Error
test error rate
0.09
0.08
0.07
0.06
0.05
0.04
0.03
10.0
12.5
15.0
17.5
20.0
22.5
25.0
27.5
30.0
training %



Summary:

The results are as follows. In terms of accuracy the order goes

$logisitic < 10_{relu} \approx 10_{tanh} < 30_{relu} \approx 10{,}10_{relu}$

Or in other terms

Logistic is the least accurate, followed by 10 relu and 10 tanh activiation, followed by 30 relu and 2 layer 10 relu activation.

On the spambase dataset all implementations were able to get reasonable accuracte after just a few epochs. And as expected they all got more accurate the more training data they received. Using the libraries like Tensorflow made running these algorithms very efficient. We did not record time but it was much faster than the first time analyzing the spambase dataset with our own implementation of logistic regression.

The neural network implementations were also fast. However the 2 hidden layer, and 30 neuron implementation were the slowest of the bunch. They however made up for that computation expense by being the most accuracte for both optimizers "SGD" and "Adam".

Adam optimizer was also consistently better than SGD. This gives reason as to why Adam optimizer is more standard in the field than SGD. However SGD is still a good way to ensure things are working.

The code used to generate the 80-20 splits and further training percentages from that were all from the hw3 solution. However the creation of the models and subsequent training were all written originally.