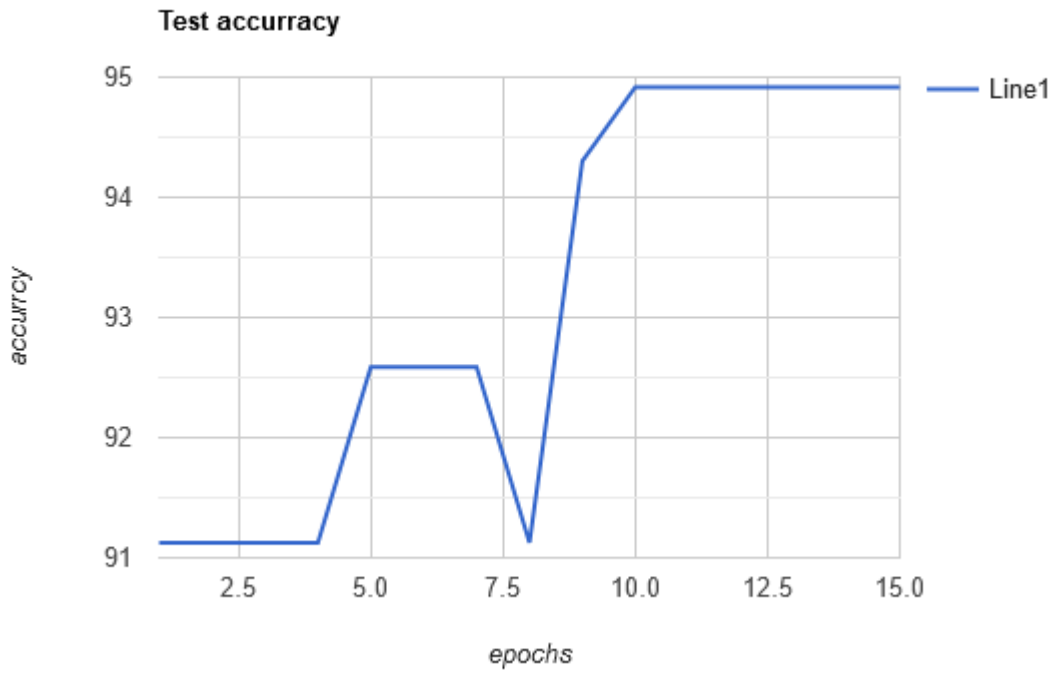
MLP2 writeup

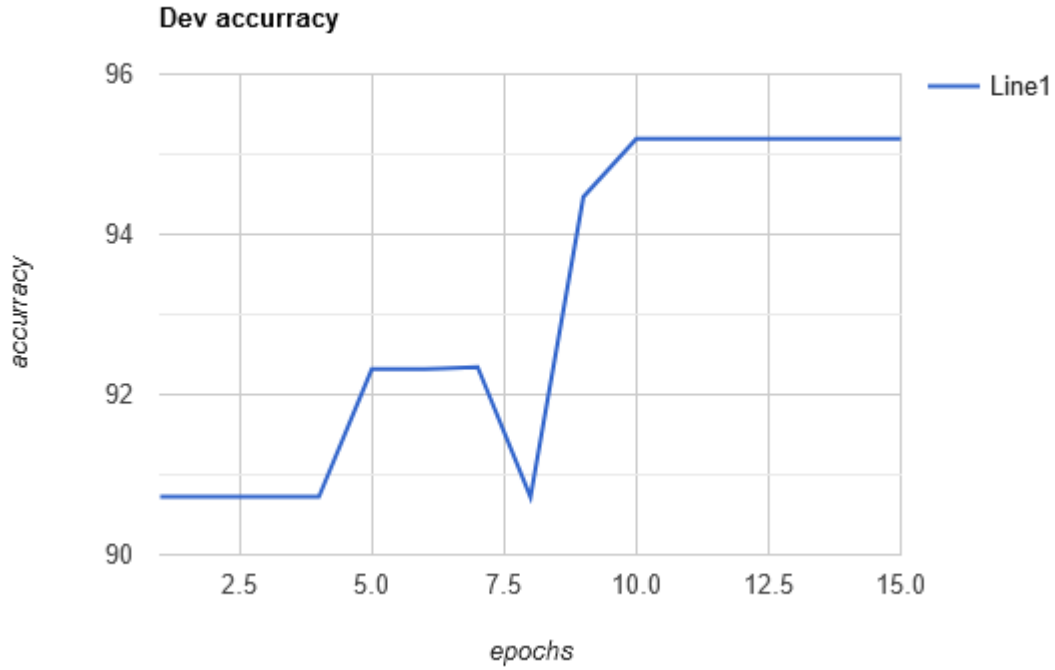
David Lau

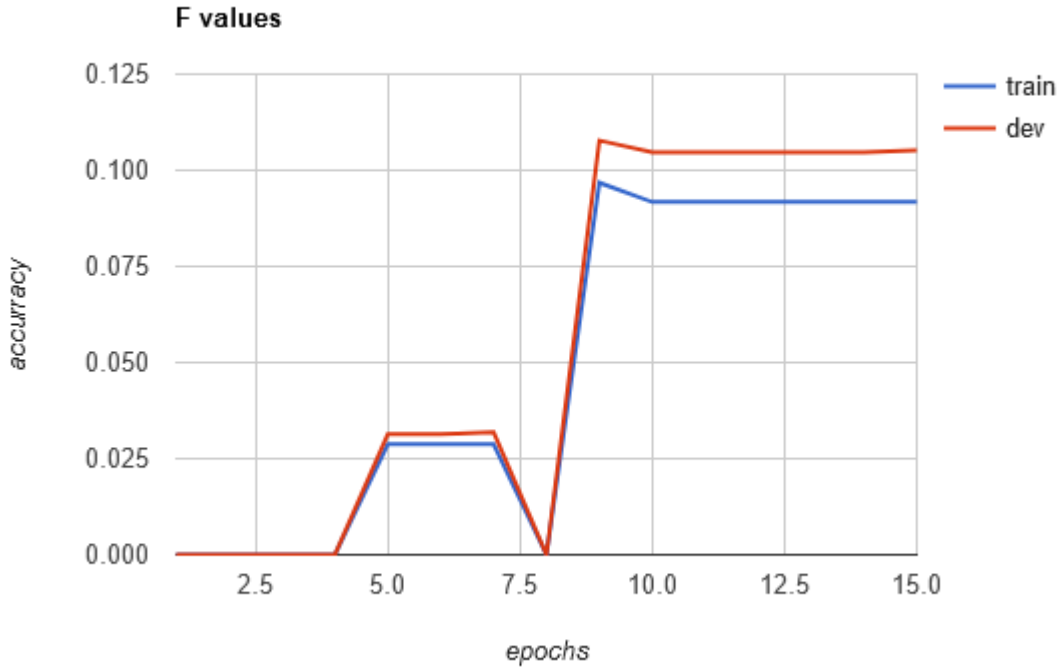
March 2021

1 Model recommendations

I would recommend that a model of 12 hidden units, a learning rate of .08, a batch size of 2, and at least 15 epochs of training. I would also recommend that the model be restarted occasionally when it gets stuck in bad local minima. I got a max accuracy of 94.916607, and a max F value of 0.091758. The graphs for the learning of the model, for the development set and the training set, are presented here.




Dev accurracy
96
94
92
90
accurracy
2.5
5.0
7.5
10.0
12.5
15.0
epochs
Line1



2 What I Learned

During this project, I learned that debugging this kind of machine learning program is at times very difficult, because when there's a problem in your math or code, its not very easy to suss out where the problem lies considering how you as the programmer doesn't really know what your code is doing. I had a bug that I spent 4 hours pouring over the math to try and fine when in reality it was just a typo. This tells me that the coder has to have very good practice when they make machine learning code, as a bug could be deep in the program and be impossible to find.

In addition, I learned how easy it is for neural networks to get trapped in local minima. There were multiple times where I was attempting to train the program, and although it was improving, it got to a point where it wouldn't go up and down anymore. I see now the necessity of manipulating the gradient descent procedure in the ways that we do. In addition, I made the conclusion that it is somewhat difficult to train on data that has discrete results, because there will be instances where the accuracy just stays at the same value for long periods of time.

It also became apparent to me that more hidden units isn't always good. When I added a bunch of extra units, the accuracy almost always permanently dropped off to zero. Additionally they are more computationally intensive. In general I think after this project I have a better idea of the choices and balances machine learning programs need in order to not take 2 years to run.