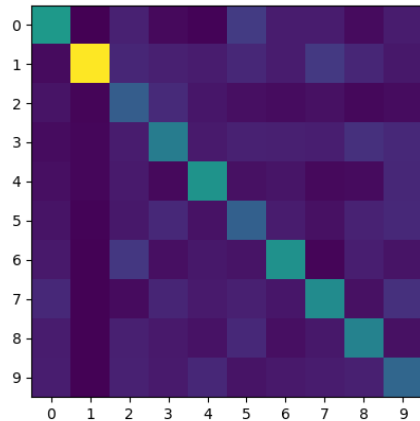


After implementing a single-layer linear perceptron, I use learning rate = 0.02, decay rate = 0.95, and loop 2000 times. I get the result like:

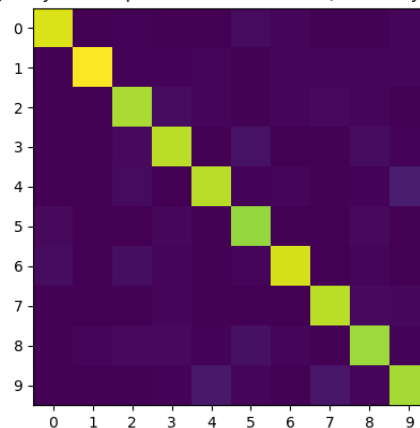
Single-layer Linear Perceptron Confusion Matrix, accuracy = 0.452



It has only 45% accuracy since it has only one layer. The average result should be about 30%.

Secondly, for the Single-layer Perceptron, I use learning rate = 0.1, decay rate = 0.95,

Single-layer Perceptron Confusion Matrix, accuracy = 0.885

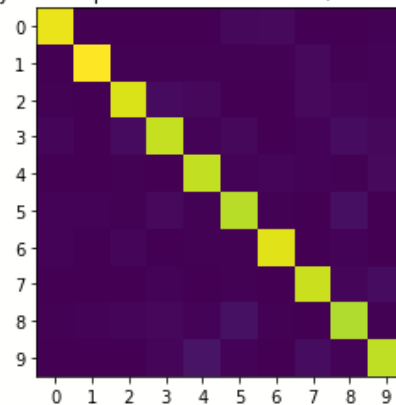


and loop 2000 times. I get:

It has about 88% accuracy which is more than 85%. We can see the diagonal is yellowish.

Thirdly, for the Multi-layer Perceptron, I use learning rate = 0.22, decay rate = 0.92, and

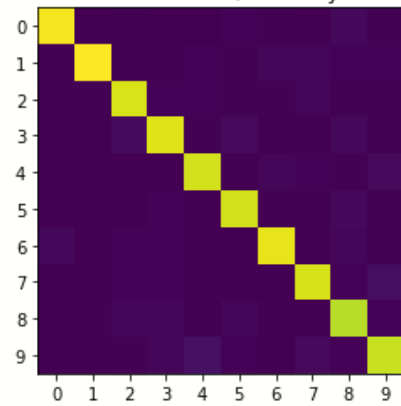
Multi-layer Perceptron Confusion Matrix, accuracy = 0.912



loop 20000 times. I get:

One thing needs to mention is that this result fluctuates a lot between 0.895 to 0.919. But overall, as the result of 91.2%, the diagonal gets more yellow than the Single-layer Perceptron.

CNN Confusion Matrix, accuracy = 0.933



Finally, the Convolutional Neural Network:

I use learning rate = 0.22, decay rate = 0.9, and loop 11000 times. I run a ton of time and the best result is 0.936. This model is also fluctuating a lot from 0.889 to 0.936. For a more stable result, please iterate over 11000 times. For details please see my comments in `train_cnn()`. We can see that the diagonal are much more yellow and other grids are in deep blue.