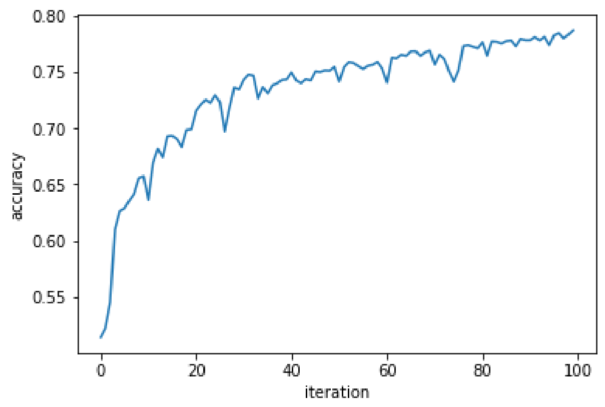
HW#2 - Neural Network for MNIST

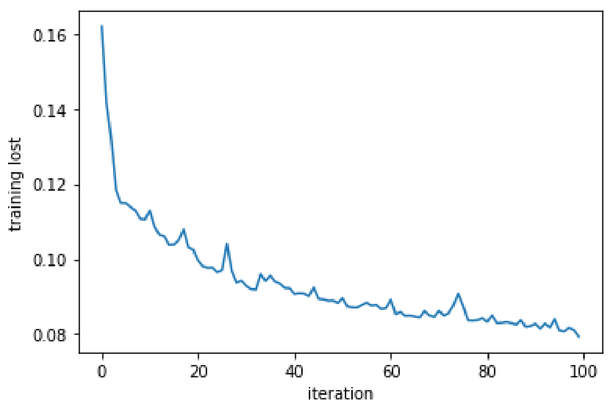
Yin-Yu Chang

1. (2 pts) Apply the normalization on the training and test data.

First, calculate the mean and standard deviation of training data.
Then, apply x = (x-mean)/std on both training and test data.

2. (2 pts) As a baseline, train a linear classifier yˆ = vT x and quadratic loss. Report its test accuracy.

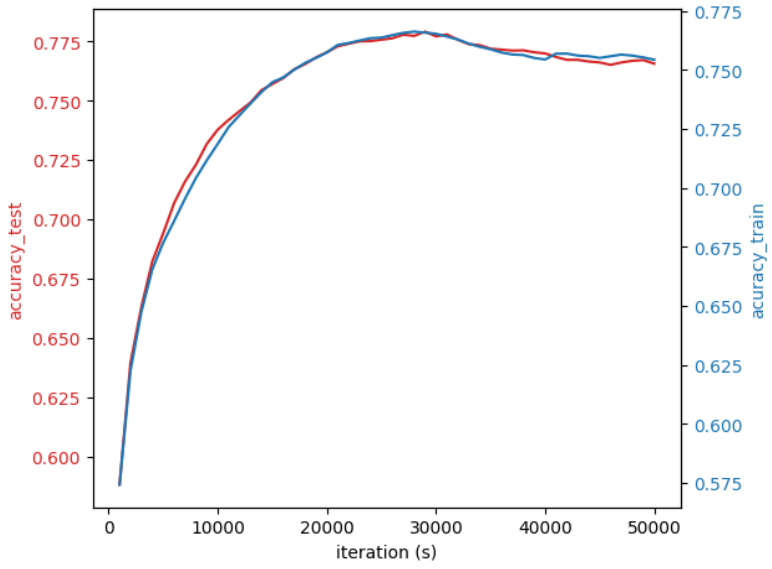




I train the linear classifier for 100 iterations with a learning rate = 0.01.
The test accuracy is 0.7868

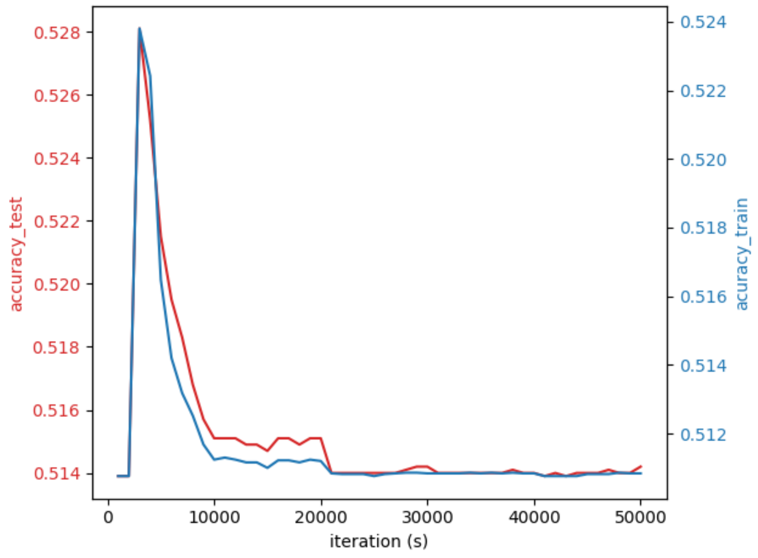
3. (7 pts) Train a neural network classifier with quadratic loss l(y,f(x))=(y−f(x))2. Plot the progress of the test and training accuracy (y-axis) as a function of the iteration counter t (x-axis)2. Report the final test accuracy for the following choices

- k=5



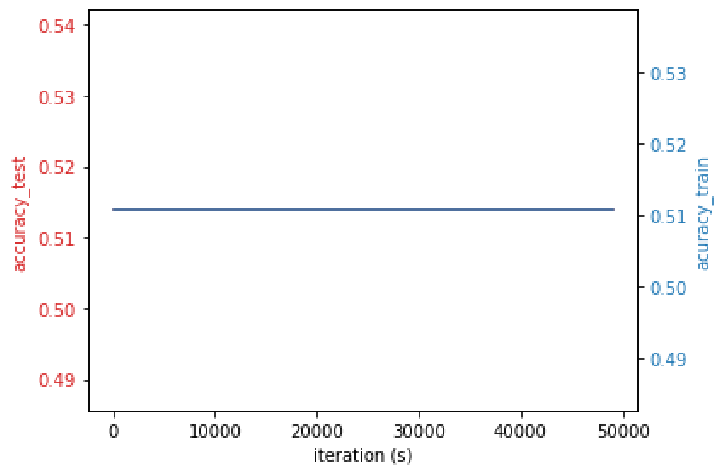
final test accuracy: 0.7558

- k=40



final test accuracy: 0.5152

• k=200



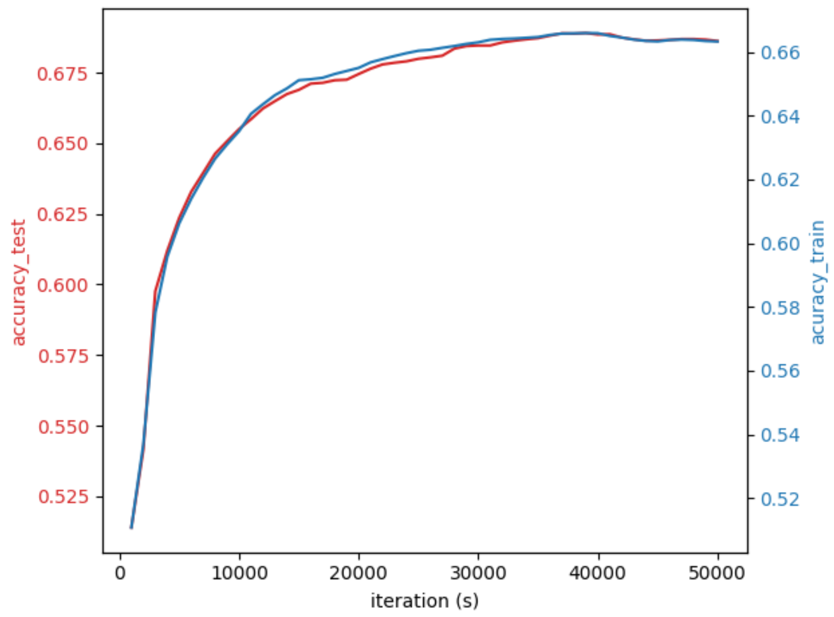
final test accuracy: 0.5139

• Comment on the role of hidden units k on the ease of optimization and accuracy.

Increasing k means to train with more neurons. When I increase the value of k, the accuracy decreases. It could be the problem of overfitting.

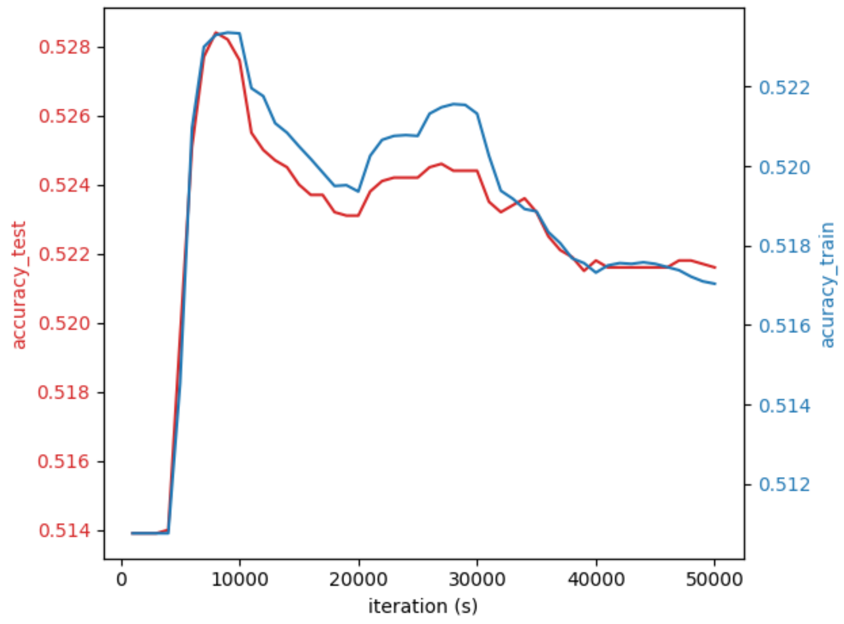
4. (7 pts) Train a neural network classifier with logistic loss, namely l(y, f (x)) = −y log(σ(f (x))) − (1 − y)log(1−σ(f(x))) where σ(x) = 1/(1+e−x) is the sigmoid function. In this case, the hard thresholding is applied on top of the sigmoid function, i.e., yˆ = 1σ(f(x))>0.5. Repeat step 3.

- k=5



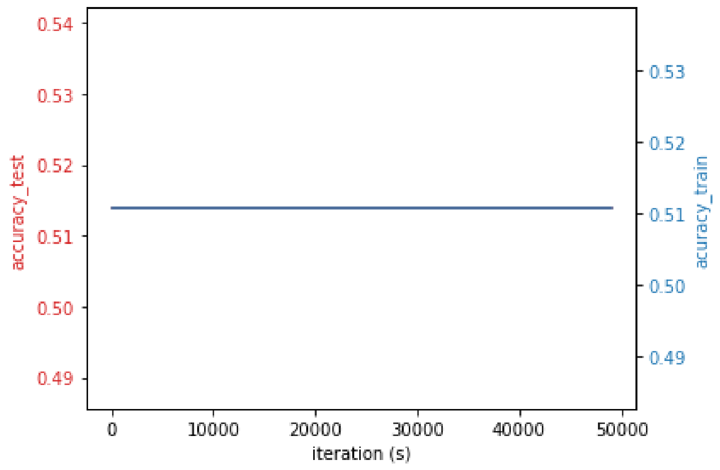
final test accuracy: 0.5496

- k=40



final test accuracy: 0.5179

- k=200



final test accuracy: 0.5139

- Comment on the role of hidden units k on the ease of optimization and accuracy.
  Increasing k means to train with more neurons. When I increase the value of k, the accuracy decreases. It could be the problem of overfitting.

5. (2 pts)

- Comment on the difference between linear model and neural net.

  The complexity of a linear model is simpler than a neural net. So, it is less possible for a linear model to overfit than a neural network.
  The time spent on training on linear model is less than neural net.

- Comment on the differences between logistic and quadratic loss in terms of optimization and test/train accuracy.

  Optimization:
  Logistic loss leads to better optimization in binary classification because it directly targets the probability distribution of the

outcomes. Updates are more meaningful and directly related to improving classification accuracy.
Quadratic loss can result in poor gradients when used for classification, especially when predictions are close to the decision boundary, leading to less efficient learning.

Accuracy:
Models trained with logistic loss should be more accurate on classification tasks.
Because they directly optimize for the probability of the correct class.
Models trained with quadratic loss might show a discrepancy between train and test accuracy if the loss does not align well using it for classification, which might be a sign of overfitting or underfitting the complexity of the decision boundary.