CS 8803 DL Assigment 2

Jingdao Chen

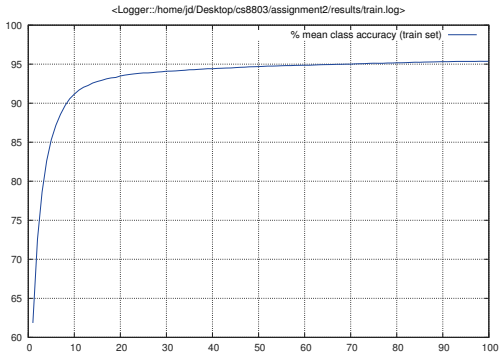
Feb 24

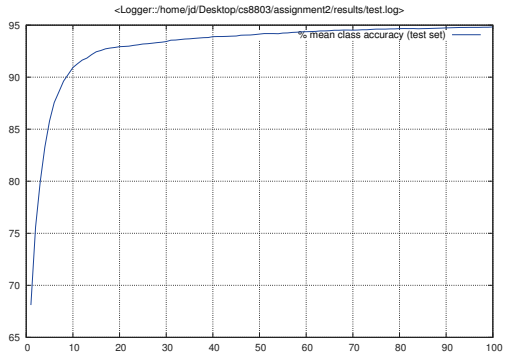
1 Optimization

The plots of training and test error for each method is shown below:

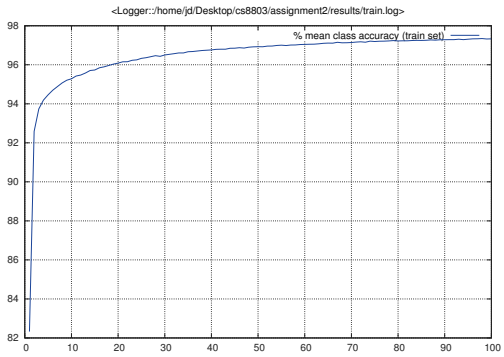
1.1 Stochastic Gradient Descent

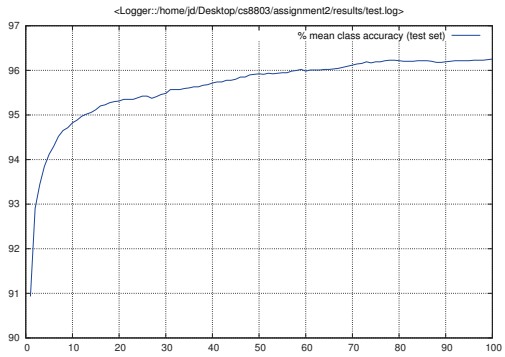
Learning rate = 1e-3



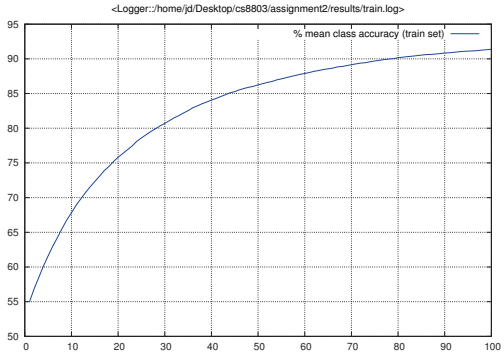


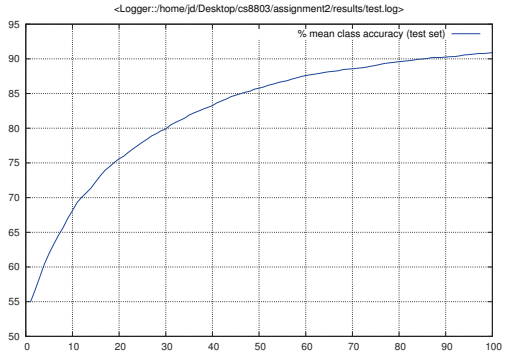
Learning rate = 1e-2



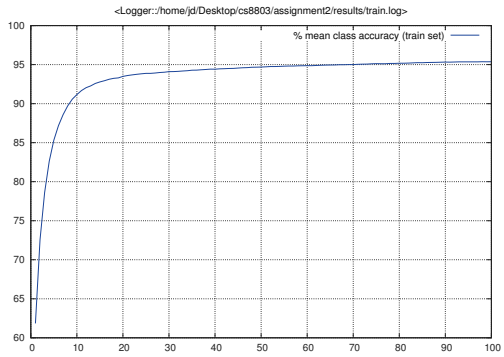


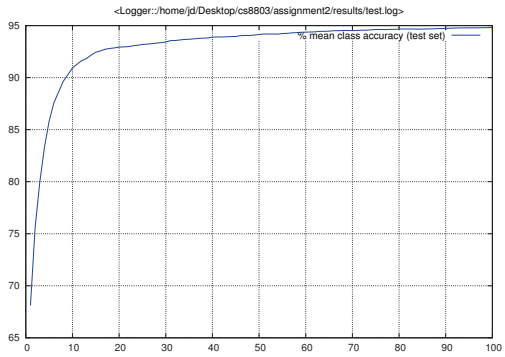
Learning rate = 1e-4



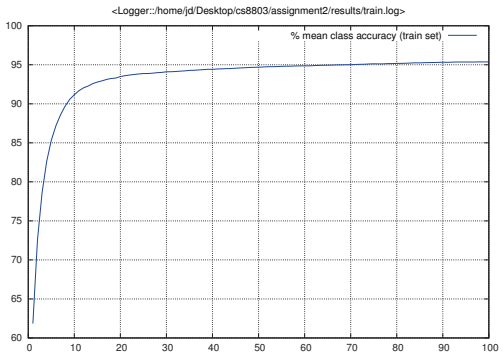


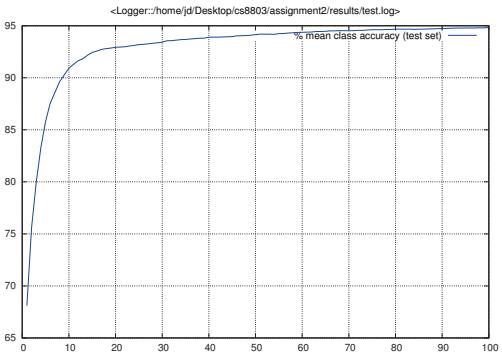
Learning rate decay = 1e-5





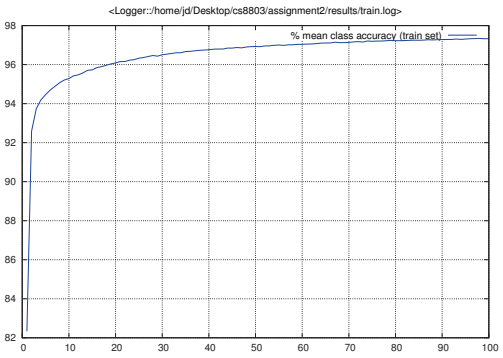
Learning rate decay = 1e-6

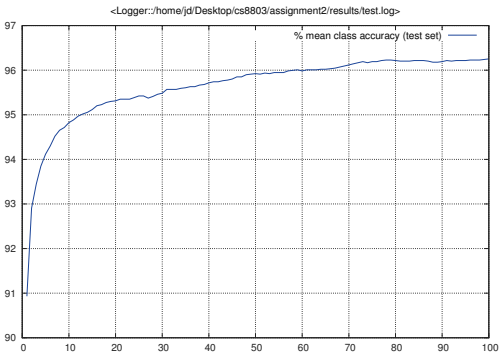




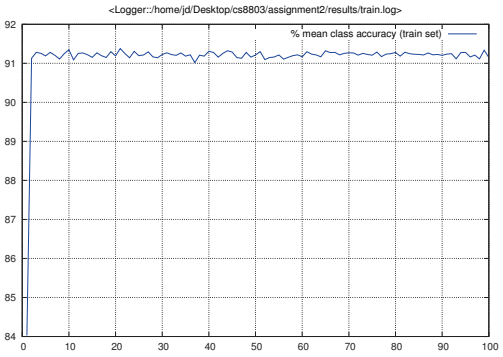
1.2 SGD + weight decay

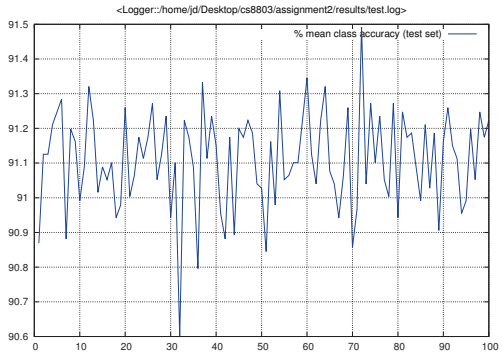
Weight decay = 1e-5



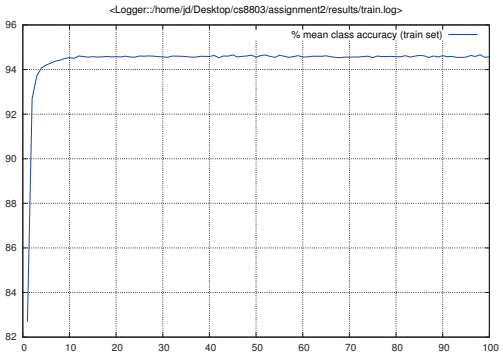


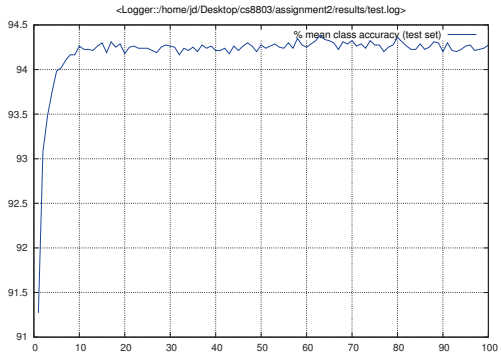
Weight decay = 1e0





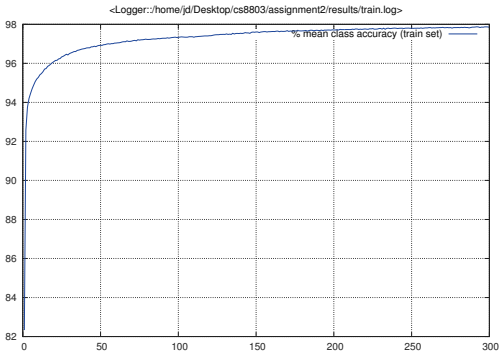
Weight decay = 1e-1

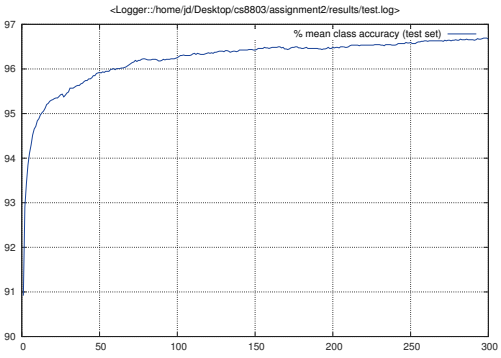




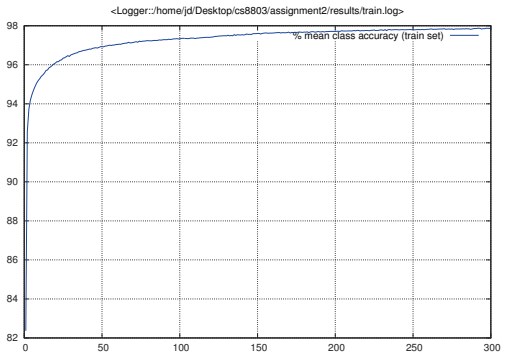
1.3 SGD + weight decay + momentum

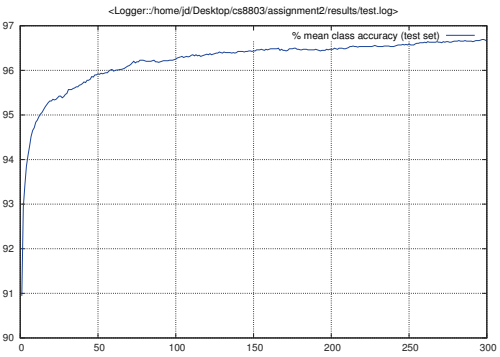
Momentum = 0.1



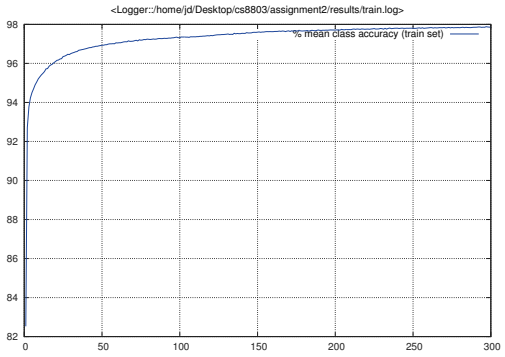


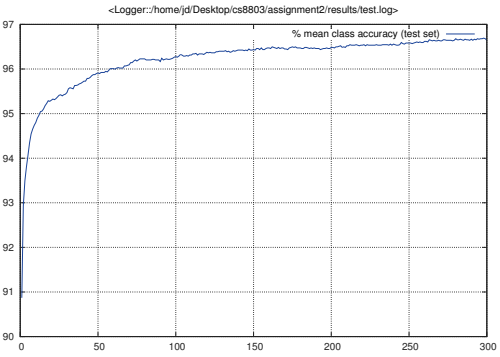
Momentum = 0.5





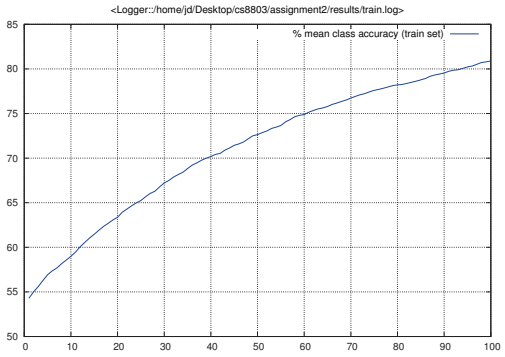
Momentum = 0.9

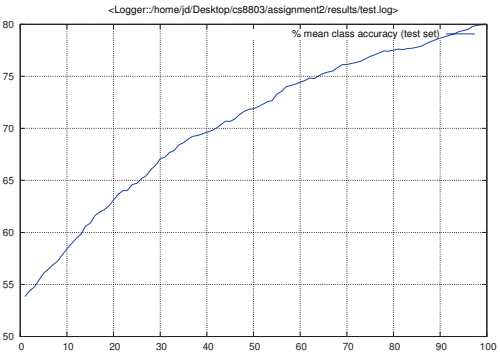




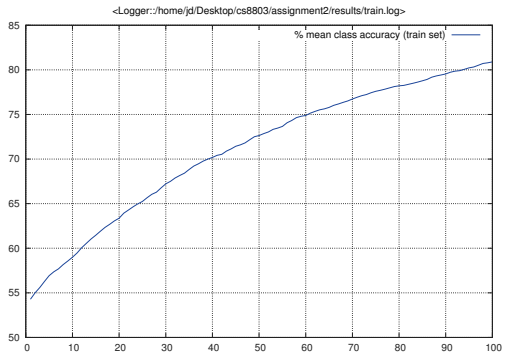
1.4 Coordinate Descent

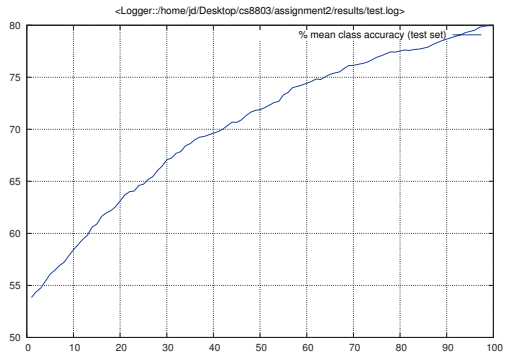
Learning rate = 1e-4



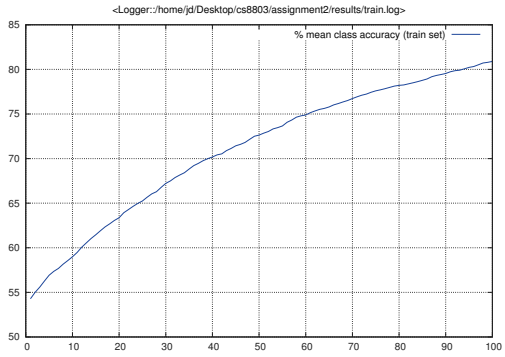


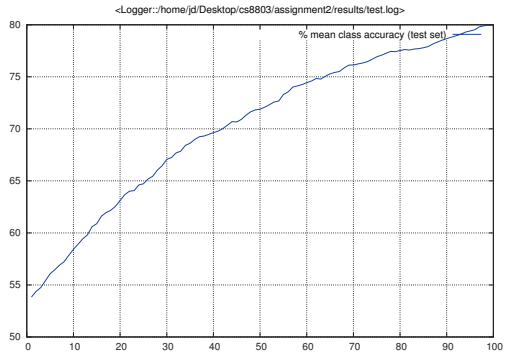
Learning rate = 1e-3





Learning rate = 1e-5





1.5 Analysis

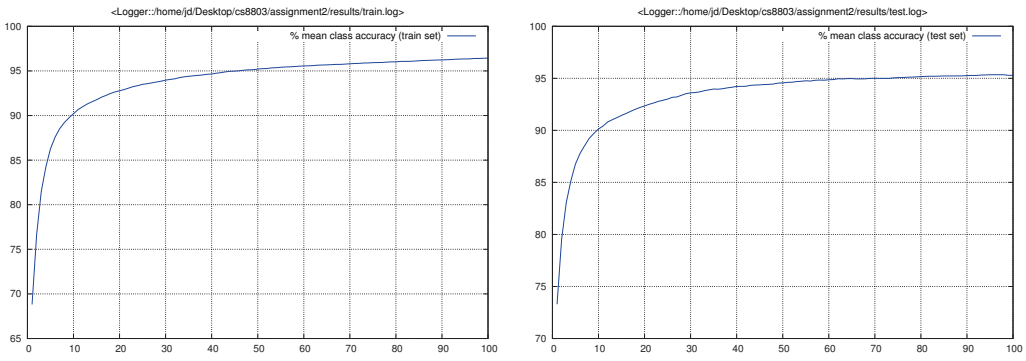
The above tests were conducted over 100 epochs with different gradient descent methods. Each method differed in terms of convergence rate with respect to their parameters. For basic stochastic gradient descent, an increase in learning rate increases convergence rate but leads to more oscillation, whereas learning rate decay causes convergence to slow down for later iterations. For stochastic gradient descent with weight decay, the regularization

term allows the optimization to generalize better to test data but oscillation may occur if the regularization term is set too high. For stochastic gradient descent with momentum, the momentum term reduces the effect of jumps or oscillations and increases the convergence rate. For coordinate descent, the convergence rate is slower than that of stochastic gradient descent. The effect of learning rate is similiar in that a higher learning rate leads to higher convergence rate.

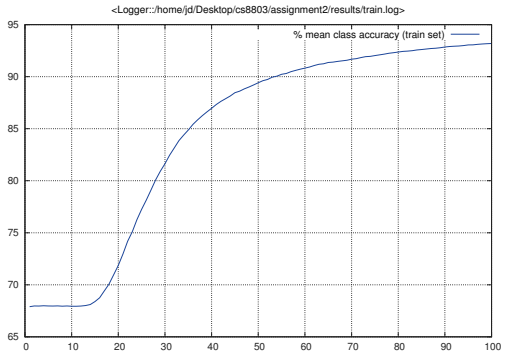
2 Activation

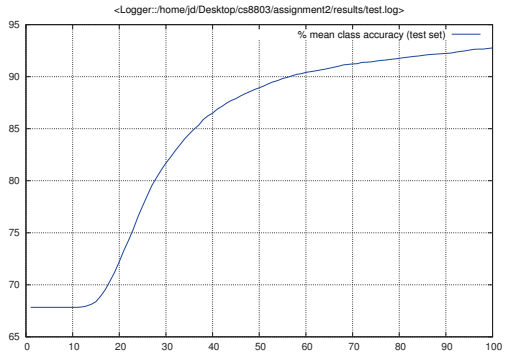
The plots of training and test error for each method is shown below:

2.1 Tanh

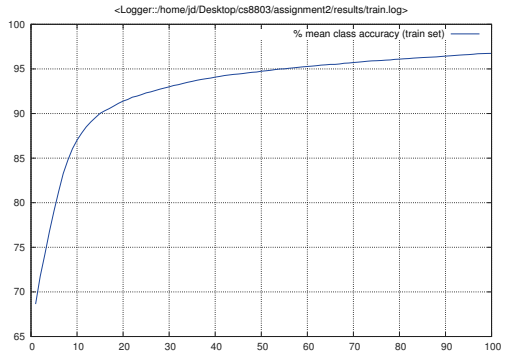


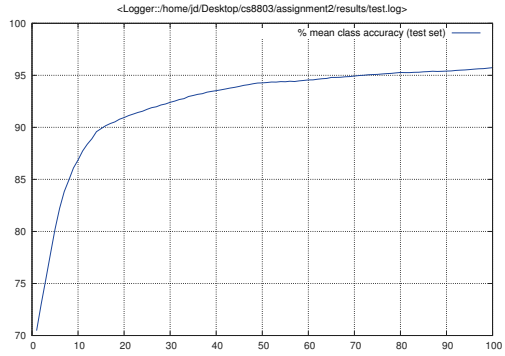
2.2 Sigmoid



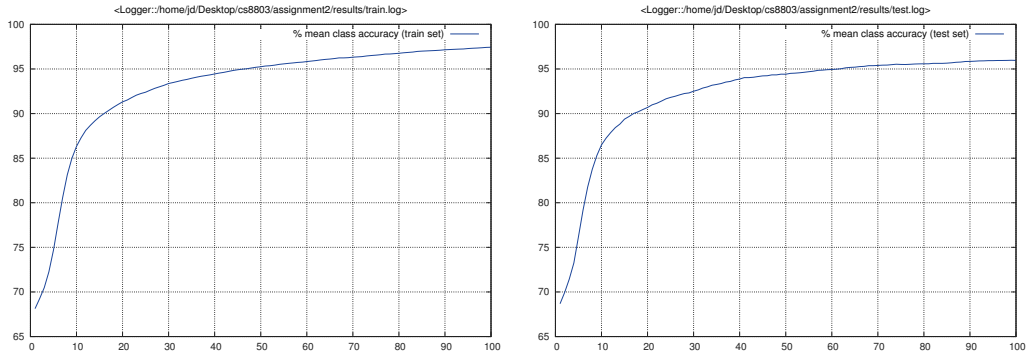


2.3 Rectified Linear Unit



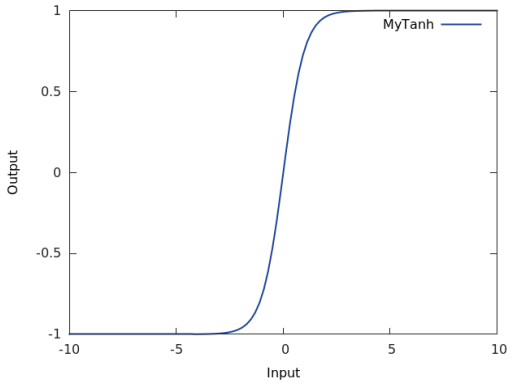


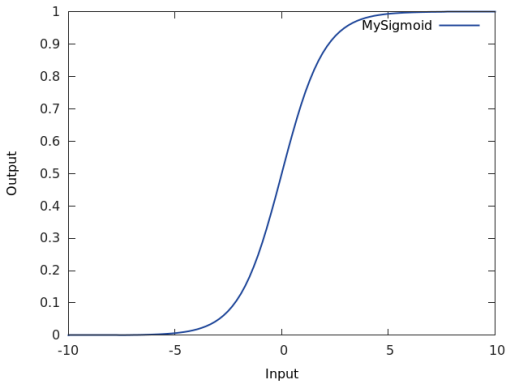
2.4 Rectified Quadratic Unit

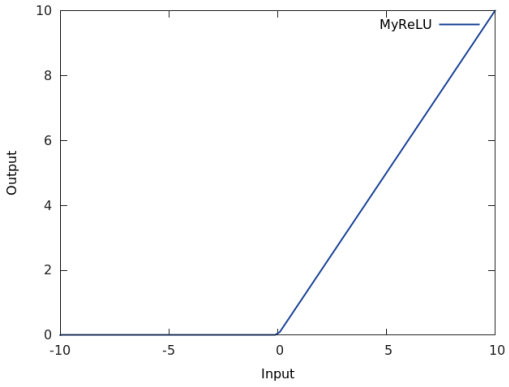


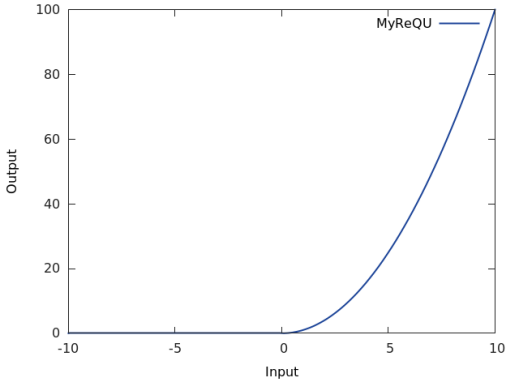
2.5 Comparison

The activation functions are plotted below. The graphical comparison shows that tanh is similiar in shape with sigmoid but with a range between -1 and 1. On the other hand, ReLU and ReQU are similiar in that negative values are set to zero but ReQU uses a quadratic instead of linear function for positive values.









From the plots of training and test error, tanh showed fast convergence rate whereas sigmoid demonstrated slow convergence at earlier iterations. The Rectified Linear and Quadratic Unit also showed slightly faster conver-

gence rate compared to tanh. A major difference between sigmoid (or tanh) and ReLU (ReQU) is that the former plateaus at later iterations whereas the latter continues to improve.