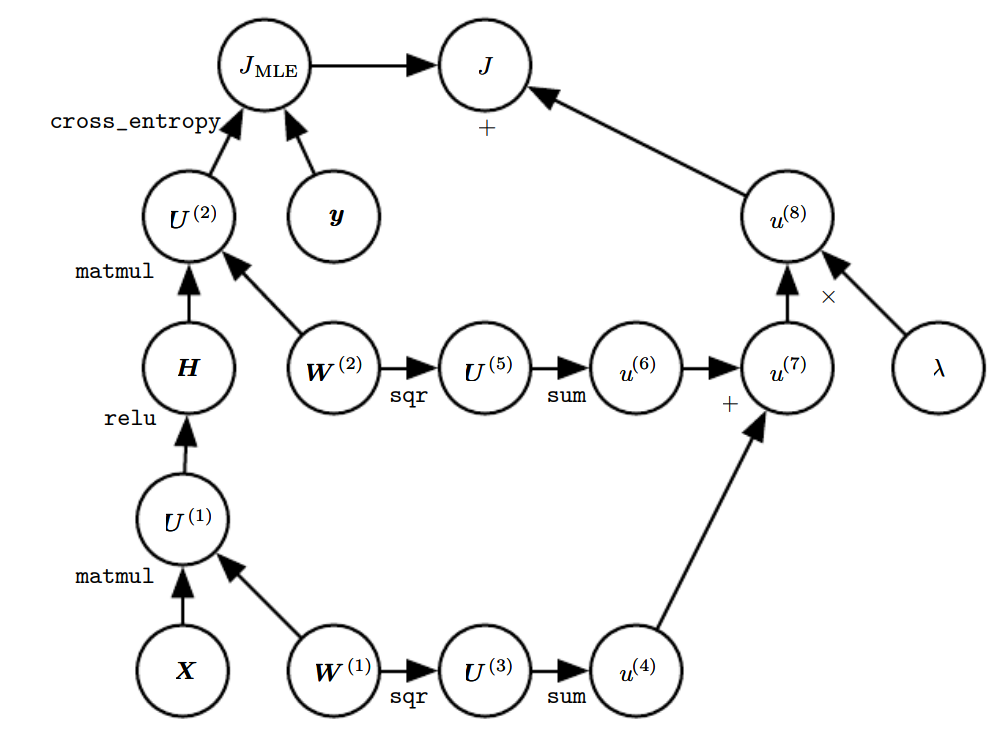
Homework1

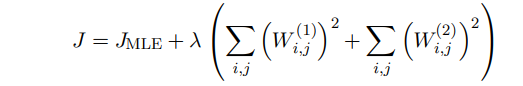
In this homework, I try to use a MLP on the Dataset MNIST.

The MLP have three layer, including one hidden layer. The design of the network follows the description in the book *Deep Learning* byIan Goodfellow, Yoshua Bengio and Aaron Courville.

The design and the back propagation are shown in the following graph. The samples, which are vectors, are multiplied by the W(1), which is a weight matrix to get U(1), a vector. After relu function, the U(1) will become the H, with the same shape as U(1). H is then multiplied by W(2), a weight matrix, and become U(2). U(2) is then processed by a softmax function before going into the cross entropy loss function. The loss function also contains a L2 regularization. The parameter of the regularization is 0.5. The computation graph can be found in pp.220 of the *Deep Learning* book.



Graph1: the computation graph of the MLP(pp.220)

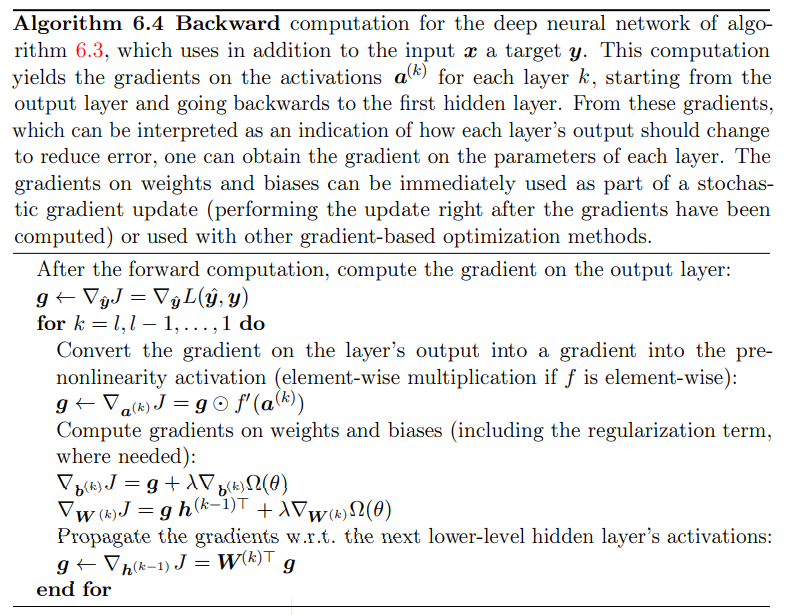


The loss function used in this homework

The backpropagation algorithm is as follows: For the JMLE part, define the gradient with respect to U(2) is G. The gradient of W(2) is then HTG. The backpropagation algorithm then computes ∇HJ, which is GW (2).Next, the relu operation uses its backpropagation rule to zero out components of GW (2) that were less than 0. Let the result be called G’. Then the gradient of W(1) is XTG’.

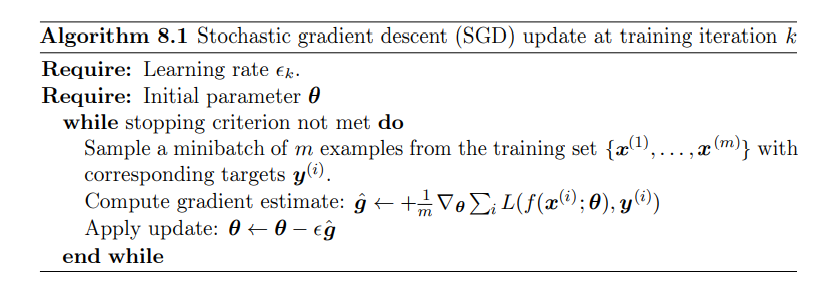
For the regularization part, its gradient with respect to W(1) is and for W(2) it is . So the total gradient with respect to W(1) is:

While the total gradient with respect to W(2) is:



Backpropagation algorithm used in the homework(pp.213)

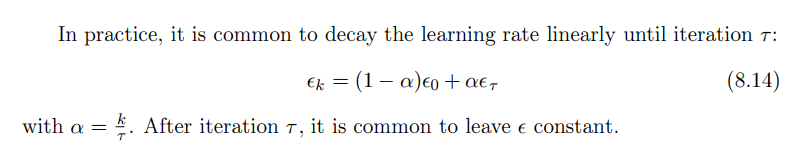
The SGD algorithm is as follows:



SGD algorithm, in pp.294

I use 50 as the max iteration, norm of the gradient of W(1) and W(2) both less than 0.1 as the stopping criterion, and 100 as the batch size.

For learning rate, I follows the following equation in pp.295:



The is the learning rate for k iteration, is the max number of iterations, is , is the initial learning rate, and is the final learning rate. I choose 0.001 as the initial learning rate, and 0.0005 as the final learning rate.

In the training process, after every iteration, I record the accuracy of the network, which is defined as the number of the truly predicted samples in the test set divided by the number of all samples in the test set. The network reaches the accuracy of approximate 0.38 after 23 iterations, and then become less accurate, at last, the accuracy of the network after 50 iterations is 0.34.