KTH Royal Institute of Technology

DD2424 Deep Learning in Data Science

Assignment Report 2

Two layer network with multiple outputs

Written by

Yutong Jiang

yutongj@kth.se

Date: Apr 9, 2022

1 introduction

In This assignment, I train and test a two layer network with multiple outputs to classify images from the CIFAR-10 dataset.

2 Gradients computation check

The analytical gradients computation is calculated as follows For the second layer

$$g = -(y - p)^T \tag{1}$$

$$\frac{\partial J}{\partial b_2} = g \tag{2}$$

$$\frac{\partial J}{\partial W_2} = g^T h^T + 2\lambda W_2 \tag{3}$$

Back propagate the gradients vector g to the first layer, we could find

$$g = gW_2 diag(Ind(s_1 > 0)) \tag{4}$$

$$\frac{\partial J}{\partial b_1} = g \tag{5}$$

$$\frac{\partial J}{\partial W_1} = g^T x^T + 2\lambda W_1 \tag{6}$$

After getting the gradients, we could update weight and bias iteratively.

In order to test the gradient is calculated correctly or not, the results is compared to the numerical results provided by ComputeGradsNum as it is a faster version.

The relative error could be computed as the absolute differences.

Initially, the sample size we choose is 20, and we could find the relative error is 2.9767e-8.

When increasing the sample size to 100, we could find the relative error is 1.5407e-8.

By increasing the sample size to 1000, we could find the relative error is 1.505e-8.

Hence, based on the comparison of analytical and numerical results shown as above, we could find that the gradients are calculated correctly.

2.1 training and validation loss/cost when using cyclical learning rates

The curves for the training and validation loss/cost could be shown as

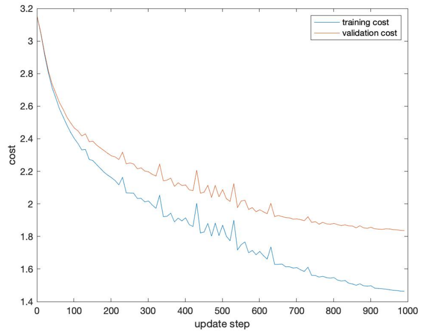

Figure 1: Cost when lambda = 0.01, n-s = 500, n-batch = 100, eta-max = 1e-1, eta-min = 1e-5

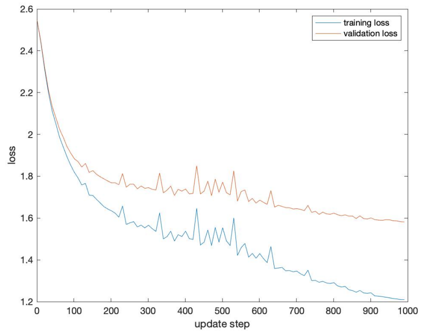

Figure 2: Loss when lambda = 0.01, n-s = 500, n-batch = 100, eta-max = 1e-1, eta-min = 1e-5

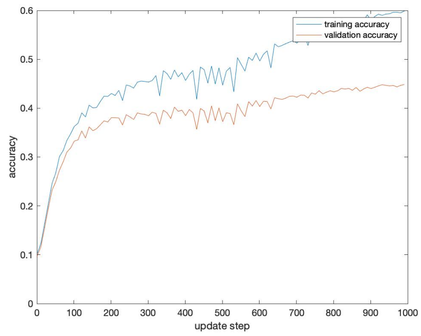

Figure 3: Accuracy when lambda = 0.01, n-s = 500, n-batch = 100, eta-max = 1e-1, eta-min = 1e-5

From the graph, we could find that the overall trend for cost and loss is to decrease. When update step is 500, there is a slight increase in loss function, and the reason is that at this update step, eta reaches eta-max and it starts to decrease, which has a slight influence on the updating of loss/cost function.

2.2 Coarse search for lambda

The basic parameter is set as follows. The range of coarse search is from 1e-1 to 1e-5, the number of cycle is 1, eta-max is 1e-5, eta-min is 1e-1.

$$n_s = 2 * floor(\frac{n}{n_{batch}}) \tag{7}$$

$$n_epochs = 2n_s n_{cycle} \frac{n_{batch}}{n} \tag{8}$$

The lambda I trained for the 3 best performing networks is 2.1478e-4 with validation accuracy 0.5124, 8.2469e-4 with validation accuracy 0.5102 and 8.5048e-4 with validation accuracy 0.5080.

2.3 Fine search for lambda

The range of Fine search for lambda is from 2.14e-4 - 8.25e-4. The basic parameter setting is the same as before. And finally lambda for 3 best performance is 3.04e-4 with validation accuracy 0.5143, 6.54e-4 with validation accuracy 0.5138, 7.34e-4 with validation accuracy 0.5128.

2.4 best found lambda setting

The curves for the training and validation loss/cost could be shown as

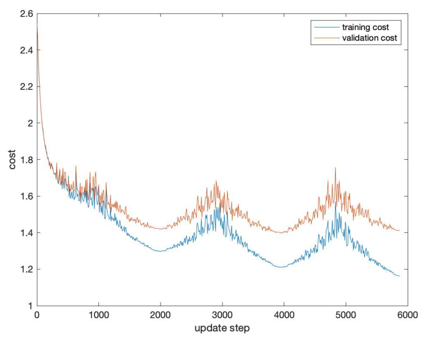

Figure 4: Cost when lambda = 3.04e-4, n-batch = 100, eta-max = 1e-1, eta-min = 1e-5

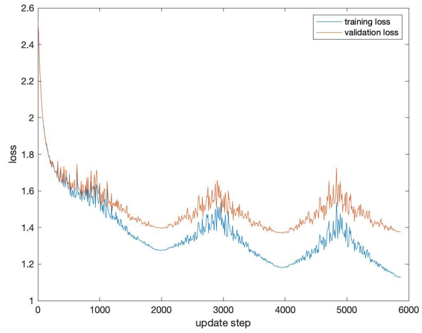

Figure 5: Loss when lambda = 3.04e-4,n-batch = 100, eta-max = 1e-1, eta-min = 1e-5

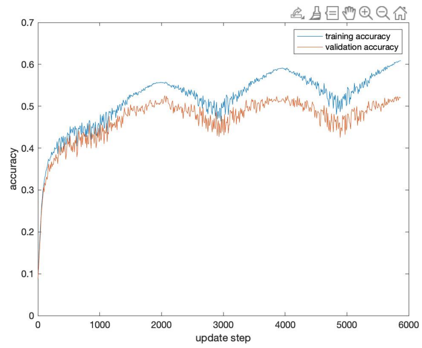

Figure 6: Accuracy when lambda = 3.04e-4, n-batch = 100, eta-max = 1e-1, eta-min = 1e-5

I train the network with 3 cycles and the final accuracy on test dataset is 0.5163.