DD2424 Assignment2 Mandatory Part

Shuyuan Zhang shuyuanz@kth.se

April 2019

1 Brief Introduction

In this assignment, we were asked to implement a two layer network to classify cifar-10 data set.

2 Main Content

2.1 Gradient Check

To check whether the gradient was computed correctly, I compared my gradients with results of the provided function *ComputeGradsNumSlow* when $h = 1e-5$. The discrepancy between results (W1, b1, W2, b2) given by different functions is measured by:

$$\sum_{All-elements} \frac{|g_a - g_n|}{max(1e-15, |g_a| + |g_n|)}$$

I performed three independent tests on mini-batch gradients of 100 data points and the differences are shown in table below:

Test No.	W1 difference	b1 difference	W2 difference	b2 difference
1	1.4135e-09	1.1623e-09	5.5464e-10	5.4808e-10
2	1.4956e-09	1.2237e-09	5.6055e-10	6.0526e-10
3	1.5064e-09	1.3242e-09	5.6134e-10	4.8356e-10

Table 1: Gradient difference between two methods

From the table we can notice that discrepancies are rather small. So we may safely say that the gradients were calculated correctly and my gradient computations were bug free.

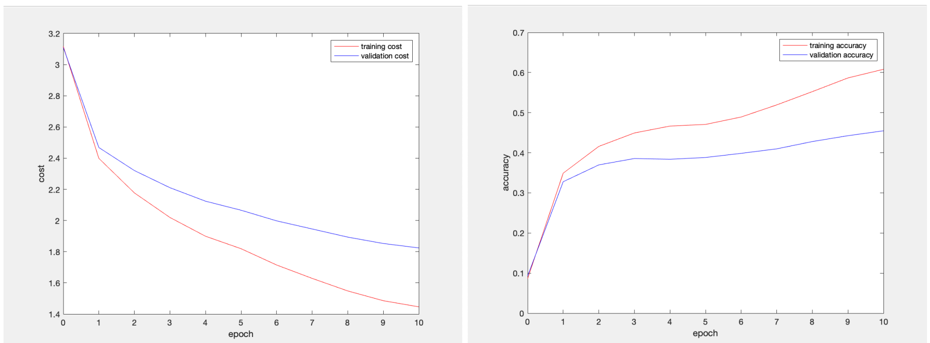
2.2 The curves for cyclical learning rates

In this part I would like to replicate figures given in the assignment description.

Parameters:

$lambda = 0.01, batch\_size = 100, eta\_min = 1e-5, eta\_max = 1e-1, n\_s = 500$

As the batch size is 100, one full cycle corresponds to 10 epochs of training.



(a) Cost plot (b) Accuracy plot

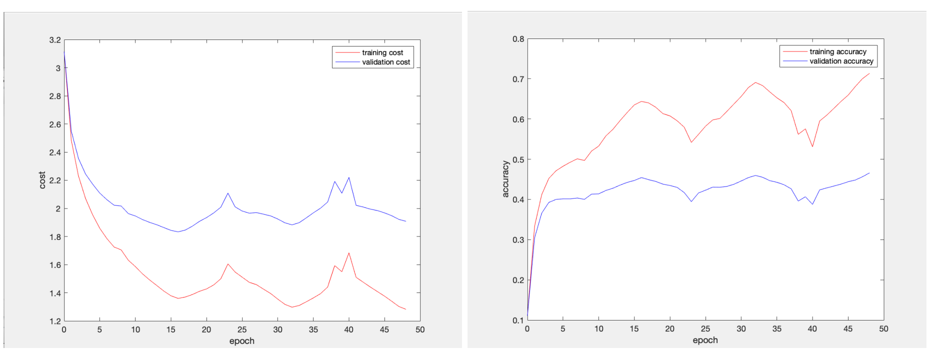
Figure 1: Training curves for one cycle of training (replicate Figure 3).

$Test\_accuracy = 0.4602$

Parameters:

$lambda = 0.01, batch\_size = 100, eta\_min = 1e-5, eta\_max = 1e-1, n\_s = 800$

As the batch size is 100, one full cycle corresponds to 16 epochs of training.



(a) Cost plot (b) Accuracy plot

Figure 2: Training curves for three cycles of training (replicate Figure 4).

$Test\_accuracy = 0.4756$

As the figures suggest, my results were quite similar with results given in the assignment description.
In figure 1, I trained for one cycle and the cost curve decreased fast in the first epoch, but slower in the following epochs. The accuracy plot shows a similar pattern, it grew quickly in the first epoch and kept growing but much slower in the following epochs. When the learning rate reached its highest (1e-1) at epoch 5, curves became a little bit smoother.
In figure 2, because I trained for 3 cycles, the cost curve reached its local minimum every time when a cycle finished and the learning rate reached its lowest (1e-5). Correspondingly, the accuracy curve reached its local maximum at the end of a cycle. When the learning rate was rather high, cost increased and accuracy decreased because a high learning rate may cause the parameters to leave its local optimum and thus degrade the network's performance.

More pages to follow.

2.3 Coarse Search

To search for a good lambda, firstly I did a coarse search. I put 45000 images in the training set and 5000 images in the validation set.
The batch size is 100. To figure out n_s, I used the equation:

$$n\_s = 2floor(n/n\_batch)$$

So n_s is set to 900 as $n = 45000$ and $n\_batch = 100$. As the number of batch in a epoch is $45000/100 = 450$, one full cycle corresponds to 4 epochs of training. I ran 2 cycles to train the network, that is, 8 epochs.
$eta\_min = 1e-5, eta\_max = 1e-1$
During the search $lambda$ was chosen randomly according to:

$$l = l\_min + (l\_max - l\_min) * rand(1, 1) \qquad lambda = 10^l$$

With $l\_min = -5, l\_max = -1$. and I tried 20 lambdas to find a good range of it.

Test No.	lambda	validation accuracy
1	1.8671e-04	0.5118
2	0.0059	0.5126
3	1.4667e-05	0.5158
4	0.0070	0.5178
5	0.0749	0.4072
6	4.6995e-05	0.5218
7	0.0080	0.5086
8	0.0683	0.4152
9	9.0878e-05	0.5112
10	6.6043e-04	0.5098
11	5.4351e-04	0.5108
12	0.0212	0.4894
13	6.2975e-05	0.5192
14	2.5401e-04	0.5188
15	0.0652	0.419
16	1.0324e-05	0.52
17	1.8608e-05	0.5148
18	1.2321e-05	0.5008
19	0.0086	0.5124
20	0.0280	0.4776

Table 2: Coarse Search Results

The three best networks are:
Gold Medal: $n\_batch = 100, n\_epoch = 8, n\_s = 900, lambda = 4.6995e-05$
Accuracy on validation set: 0.5218
Silver Medal: $n\_batch = 100, n\_epoch = 8, n\_s = 900, lambda = 1.0324e-05$

Accuracy on validation set: 0.52
Bronze Medal: $n\_batch = 100, n\_epoch = 8, n\_s = 900, lambda = 6.2975e-05$
Accuracy on validation set: 0.5192

2.4 Fine Search

According to the coarse search, we found out a good range for lambda. The lambdas of the 3 best networks are all smaller than 1e-4.
So I adjusted lambda to a narrower range: 1e-5 to 1e-3. To train for a few more cycles than before, I set $n\_epoch = 12$ to train for 3 cycles.
Meanwhile, all other hyper parameters were unchanged, with 45000 images in the training set and 5000 images in the validation set.

$$n\_batch = 100, n\_epoch = 12, n\_s = 900, eta\_min = 1e-5, eta\_max = 1e-1$$

Test No.	lambda	validation accuracy
1	2.9829e-05	0.5182
2	1.7074e-04	0.522
3	8.8433e-05	0.5162
4	1.3738e-05	0.5146
5	5.2409e-04	0.522
6	4.1407e-05	0.5216
7	1.7187e-04	0.513
8	4.7358e-04	0.5212
9	2.1083e-05	0.5162
10	1.1509e-05	0.5182
11	6.8378e-04	0.5212
12	1.4386e-04	0.518
13	1.1988e-04	0.5268
14	3.2082e-05	0.5192
15	4.6158e-05	0.5128
16	2.7608e-05	0.517
17	2.3665e-05	0.523
18	7.8728e-04	0.515
19	2.1171e-05	0.5238
20	8.7193e-04	0.5164

Table 3: Fine Search Results

The three best networks are:
Gold Medal: $n\_batch = 100, n\_epoch = 8, n\_s = 900, lambda = 1.1988e-04$
Accuracy on validation set: 0.5268
Silver Medal: $n\_batch = 100, n\_epoch = 8, n\_s = 900, lambda = 2.1171e-05$
Accuracy on validation set: 0.5238

Bronze Medal: $n\_batch = 100, n\_epoch = 8, n\_s = 900, lambda = 2.3665e-05$
Accuracy on validation set: 0.523

2.5 Final Test

I trained the network on all the training data, except for 1000 examples in a validation set. So the size of training set is 49000.
The batch size is 100, so there are 490 batches in a epoch. I set $n\_s = 980$ and one full cycle corresponds to 4 epochs of training. I ran 3 cycles to train the network, that is, 12 epochs.
The parameters are:

$n\_batch = 100, n\_epoch = 12, n\_s = 980, lambda = 1.1988e-04$

$eta\_min = 1e-5, eta\_max = 1e-1$

And training and validation costs are:

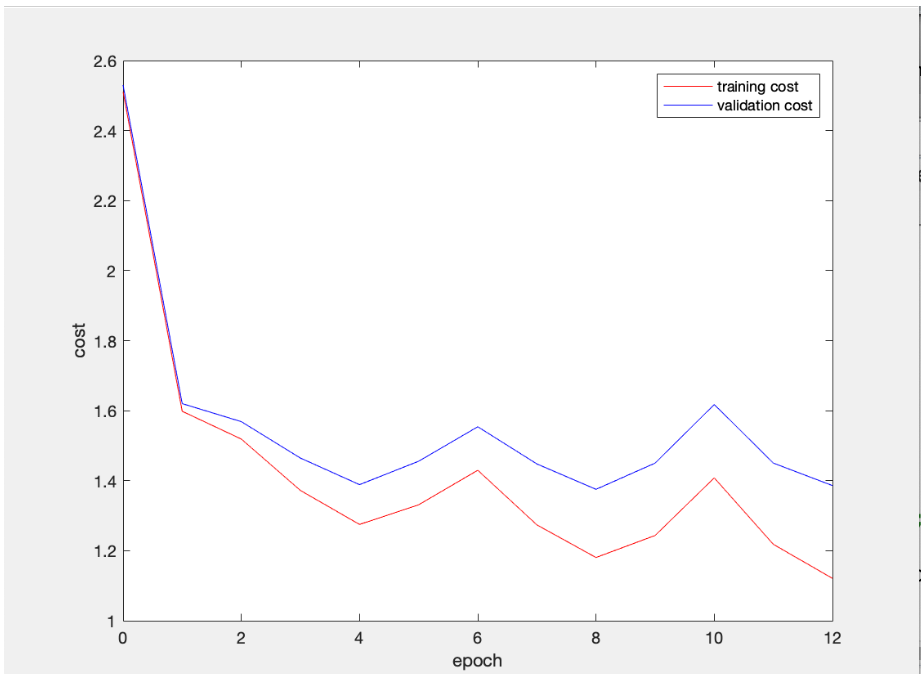


Figure 3: Train/validation cost of all data

Accuracy on test data: 0.5187