KTH Royal Institute of Technology

DD2424 Deep Learning in Data Science

Assignment Report 1 bonus

One layer network with multiple outputs bonus part

Written by

Yutong Jiang

yutongj@kth.se

Date: Mar 29, 2022

1 introduction

This bonus assignment could be divided into two parts. For the first part, I try to improve the performance of the network by trying four different methods separately. In the second part, multiple binary cross-entropy loss is used to replace the softmax operation.

2 Improvement performance of the network

2.1 Method 1: increase size of training data set

In method 1, all available data is used for training(49000), and the size of validation set has been decreased to 1000. The configuration of parameters is lambda = 0, eta = 0.001, n-batch = 100 and n-epochs = 40.

The final test accuracy is 0.4101. It is obvious to notice that there is an improvement when comparing with the original accuracy 0.3893.

2.2 Method 2: shuffle training data

In method 2, I try to shuffle the training data before each epoch by applying the function called *randperm*. The training set is 'data-batch-1.mat', the validation set is 'data-batch-2.mat' and the test set is 'test-batch.mat', which is the same as data set in assignment 1. The configuration of parameters is lambda = 0, eta = 0.001, n-batch = 100 and n-epochs = 40.

The final test accuracy is 0.3931, and there is a slight increase when comparing with the original accuracy 0.3893.

2.3 Method 3: decay the learning rate

In method 3, I try to decay the learning rate by a factor of 10 after every 8 epochs. All the other parameters remain the same as that in basic part.

The final test accuracy is 0.3934, and there is a slight increase when comparing with the original accuracy 0.3893.

2.4 Method 4: increase number of epochs

In method 4, I try to increase epoch to 50 and the other parameters remain the same as that in basic part.

The final test accuracy is 0.3916. The improvement is not obvious as above. The main reason is that the training set and validation set is almost converged when epochs are 40. Hence, when increasing it to 50, it doesn't help much. However, for the situation where the training set and validation set doesn't converge due to the small number of epochs, it is an efficient way to increase the number of epochs.

2.5 conclusion

The best accuracy is 0.4101, which is achieved by increasing size of training data.

3 network with multiple binary cross-entropy loss

3.1 mathematical deduction

The basic structure of the network is linear score function + Sigmoid function + multiple binary cross-entropy loss + Regularization. multiple binary cross-entropy and Sigmoid function could be shown as follows

$$\sigma(s) = \frac{exp(s)}{exp(s)+1} \tag{1}$$

$$l_{multiple}(x,y) = -\frac{1}{K}\sum_{k=1}^{K}[(1-y_k)log(1-p_k) + y_k log(p_k)] \tag{2}$$

By applying chain rule, for each sample, it could be shown that

$$\frac{\partial l_{multiple}}{\partial s} = \frac{\partial J}{\partial P}\frac{\partial P}{\partial S} = -(\frac{y_i}{p_i} - \frac{1-y_i}{1-p_i})(p_i(1-p_i)) = -(y_i - p_i) \tag{3}$$

Hence, for multiple binary cross-entropy, G could be shown as

$$G = -(Y - P) \tag{4}$$

3.2 results

The final test accuracy I achieve training with sigmoid + multiple binary cross-entropy loss is 0.3846, which is smaller than softmax + cross-entropy under the condition of lambda = 0.1, eta = 0.001, n-batch = 100, n-epochs = 40. And the loss and cost function could be shown as follows.

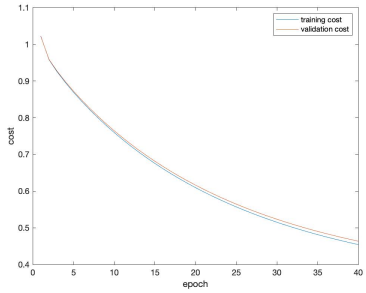

Figure 1: cost value versus epochs (sigmoid + multiple binary cross-entropy)

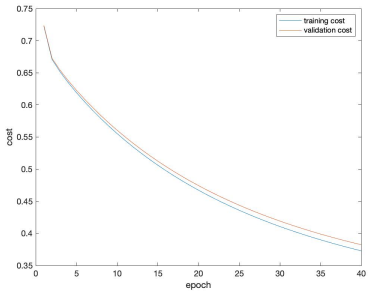

Figure 2: loss value versus epochs (sigmoid + multiple binary cross-entropy)

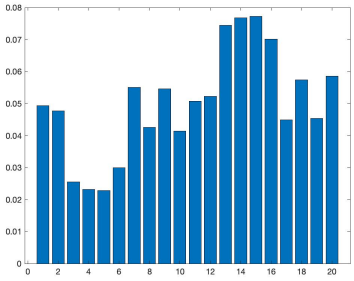

Figure 3: histogram of softmax + cross-entropy

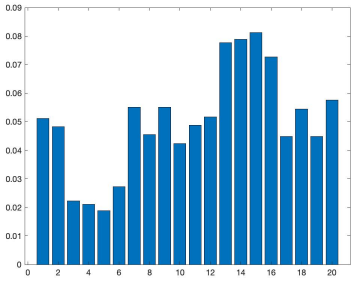

Figure 4: histogram of sigmoid + multiple binary cross-entropy

3.3 conclusion

From the loss function, we could find there is a qualitative difference between these histograms. And by looking at the loss function, there is no overfitting in both loss function.

The histogram of probability for the ground truth could be shown as above. The first 10 points shows probability of correctly categorized from label 1 - 10. And the last 10 shows probability of incorrectly categorized from label 1 - 10. The basic trend is similar, but we could find that there is an increase on the probability of correctly categorised label 3, 4 and 5. And there is an decrease in incorrectly categorised label 3, 4 and 5.