Deep Learning in Data Science, DD2424

Short report on lab assignment 3

Bonus points

Anna Sánchez Espunyes

April 12, 2021

1 Main objectives and scope of the assignment

In this assignment you will train and test k-layer networks with multiple outputs to classify images (once again) from the CIFAR-10 dataset. You will upgrade your code from Assignment 2 in two significant ways. Firstly, generalizing your code so that you can train and test k-layer networks. Secondly, incorporating batch normalization into the k-layer network both for training and testing.

2 Results and discussion

2.1 Bonus 1:More exhaustive search for lambda

A wider exploration of the regularization term will be performed in order to find a narrower range. I have chosen 10 values between 10e-7 and 10e-1 drawn from a uniform distribution. When training each network, I have used 2 cycles of training. The validation accuracy for the 10 different lambda values can be found in Table 1

Lambda	Validation accuracy
1e-07	52.12
4.64e-07	51.76
2.15e-06	51.88
1e-05	52.7
4.64e-05	52.92
0.00021	51.94
0.001	52.86
0.004	54.22
0.0215	51.139
0.1	43.78

Table 1: Validation accuracy for different regularization terms

Now that we have detected the most promising regularization terms, we can do a more narrow search. The range will now be between 10e-3.5 and 10e-1.5, using 2 cycles for training. The validation accuracy for the 20 different lambda values can be found in Table 2.

The final step will be to train the top networks with 3 cycles of training. The test accuracies obtained with the top network can be seen in Table 3.

Lambda	Validation accuracy
0.00031	52.26
0.0004	52.3
0.0005	52.52
0.0006	52.52
0.0008	53.1
0.00106	52.46
0.0013	53.16
0.0017	52.98
0.0022	54.2
0.0028	53.54
0.0035	54.54
0.0045	**54.94**
0.0058	54.54
0.0074	**54.9**
0.0094	**54.74**
0.012	54.48
0.015	54.26
0.019	53.62
0.025	52.44
0.031	52.54

Table 2: Validation accuracy for different regularization terms

Lambda	Test accuracy
0.0045	54.13

Table 3: Test accuracy for the top network

2.2 Bonus 2. Network architecture

Number of layers	Validation accuracy
2-layers	53.26
3-layers	53.98
4-layers	**54.34**
5-layers	54.2
6-layers	53.14
7-layers	53.2
8-layers	52.88
9-layers	52.52

Table 4: Validation accuracy for different number of hidden layers

The next bonus that will be implemented is searching the best network architecture. In order to find the most appropriate one, a first search regarding the number of layers will be performed. Once the optimal number of layers is found, a search regarding the number of hidden nodes will be executed.

The number of hidden nodes in the first layer will now be optimized between the following range [30,40,50,60,70,80]. The validation accuracy for the different number of hidden nodes can be found in Table 5. The optimal number of hidden nodes is 80 for the first network. Fixing this number of hidden nodes, the same experiment will be done for the second and third layer. The results can be seen in Figure 6 and Figure 7. The results show that the network that yields better validation accuracy is a 4-layer network with number of hidden nodes [80,50,50]. **The test accuracy for this network is 55.34** (Figure 8).

Number of hidden nodes	Validation accuracy
30	53.3
40	53.76
50	54.34
60	55.46
70	55.6
80	**55.94**

Table 5: Validation accuracy for the first hidden layer

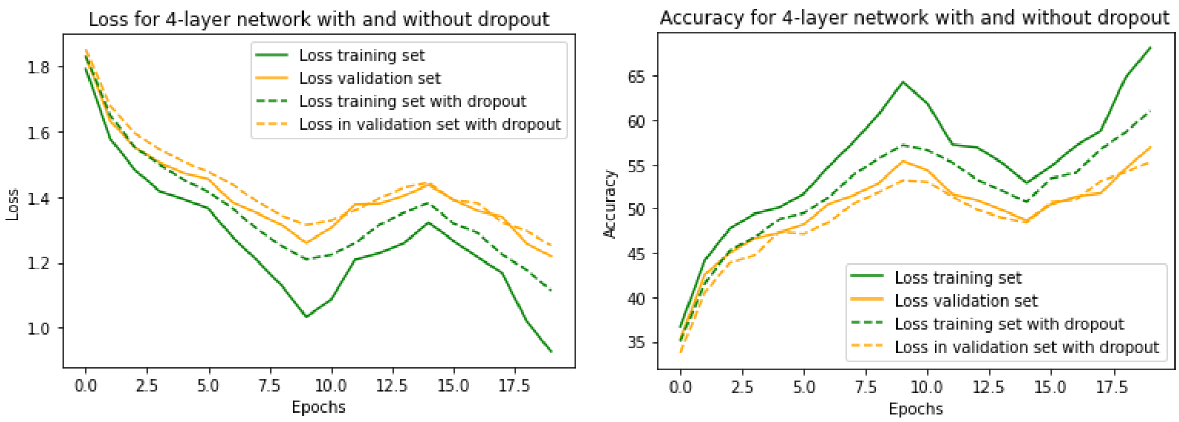
Number of hidden nodes	Validation accuracy
30	55.94
40	55.8
50	**56.54**
70	56.37

Table 6: Validation accuracy for the second hidden layer

2.3 Bonus 3: Dropout

Dropout helps us with regularization. It allows training deep networks and makes them less prone to over-fitting. During the training phase, it sets a certain amount of neurons to 0. Intuitively, we are making the network learn different types of neuron subsets. As dropout is a useful technique when using a lot of hidden nodes. The experiment has been performed with the optimal parameters found in the previous sections (4-layer network with 80, 50, 50 hidden nodes).

The results can be seen in Figure 1. It can be seen that we do not achieve better validation accuracy or test accuracy. Nevertheless, we see that with dropout we generalize better (the difference between the training error and the validation error is much more narrow).



(a) Loss with and without dropout

(b) Accuracy with and without dropout

Figure 1: 4-layer neural network

2.4 Bonus 4: Jitter

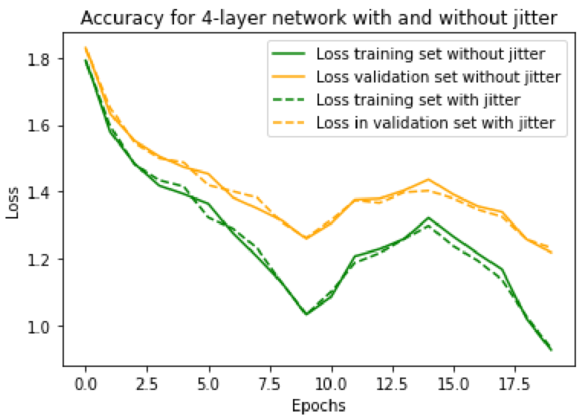
Adding jitter also helps with regularization. If the available data set is too small, the network may memorize the samples. By adding Gaussian noise when training it is more unlikely to over-fit. The mean of the noise added is 0 and the variance 0.1. The results can be seen in Figure 2. Better validation accuracy and training accuracy is achieved when using jitter. **Furthermore, the test accuracy is also higher, achieving 55.46**.

Number of hidden nodes	Validation accuracy
20	56.4
30	56.5
40	56.3
50	**56.92**

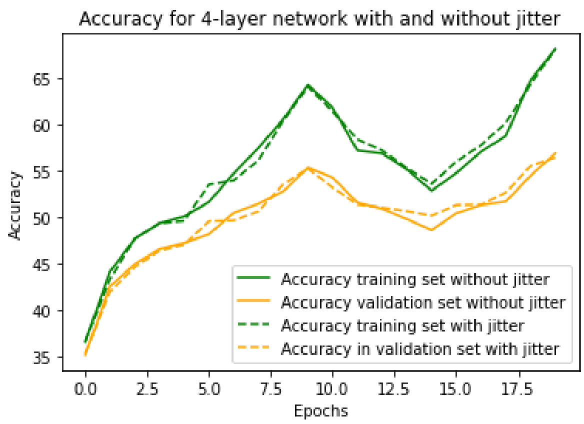
Table 7: Validation accuracy for the third hidden layer

Number of hidden nodes	Test accuracy
[80,50,50]	**55.34**

Table 8: Final test accuracy for the top network



(a) Loss with and without jitter



(b) Accuracy with and without jitter

Figure 2: 4-layer neural network

2.5 Final model

The improvements that have been tested are: an exhaustive search for lambda, search for the network architecture, dropout and jitter. The improvement that brought largest gains was the network architecture search. On the other hand, searching for the best lambda and adding jitter also improved the accuracy, whereas using dropout did not help in improving accuracy. **The final test accuracy is 55.46**. It was achieved by using jitter, a lambda of 0.0045, an architecture of 4-layers with hidden nodes [80,50,50] and without using dropout.