Assignment 3 Bonus: Convolutional network

DD2424 - Deep Learning
Svenja Räther

April 2020

1 Introduction

This assignment 3 in the course DD2424 Deep Learning in Data Science trains a ConvNet to predict the language of a surname from its spelling. The bonus task covered in this report is: (2) generalizing the code so networks with L convolutional layers can be defined.

2 Network with L convolutional layers

To be able to define multiple convolutional layers, the code structure had to be generalized for this purpose. This was specifically done in the forward and backward pass as well as adjusting the relevant data structures.
After this was done, it is interesting to see how the network behaves once more layers get added. Also the numbers of filters is a parameter that influences the performance of the network in this case. To elaborate this behavior a small search was constructed increasing the number of layers for certain number of filters. However, mixed numbers of filters were not tested. This would require another randomized search.

In order to keep the runtime relatively short, the balanced dataset was used with *n_epochs = 100 (equivalent to 1000 update steps), eta = .01 (to see faster convergence), rho=.9, n_batch = 100, n_update = 500 and k = 3 (with k=5 it was not possible to train 6 or more layers).*

Looking at Table 1, is seems like increasing the number of layers does not lead to a better accuracy on Validation while it does on Training in some cases. Furthermore, for increasing the number of filters, the networks with 2 and 4 layers show an increased accuracy on Validation while the value for network with 6 and 8 layers decreases. Therefore, it is hard to generalize the effects of changing those parameters. Furthermore, differences might be influenced by the parameters that have not been changed (e.g. eta or k). Finding different values for those might change the outcome. The best performance was achieved with a 2 layer network with 40 filters at each layer.

Layers	**Filters**	**Training**	**Validation**
2	20	1.5691 0.5071	2.1476 0.3849
2	30	1.5225 0.5335	2.1559 0.3849
2	40	1.4015 0.5772	**2.1349 0.4246**
4	20	1.4258 0.5660	2.3128 0.3730
4	30	1.4912 0.5479	2.1091 0.3888
4	40	1.2701 0.6263	2.4528 0.4087
6	20	1.5476 0.5125	2.3545 0.3769
6	30	1.2403 0.6332	2.5837 0.3888
6	40	1.4613 0.5547	2.5591 0.3333
8	20	1.5409 0.5293	2.3144 0.3492
8	30	1.5526 0.5108	2.3299 0.3928
8	40	1.6104 0.5077	2.7111 0.2817

Table 1: Cost and Accuracy for various numbers of layers and sizes