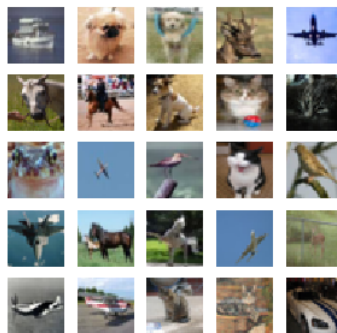# DD2424 Lab 1

Artem Los (arteml@kth.se)

June 22, 2018

Introduction

In this lab, we will implement a simple 1-layer network and train it on the CIFAR-10 dataset (sample from the data set shown below).

Tests for analytical gradient

In order to test the correctness of analytical gradients, numerical gradients were used. We developed a function called `checks`, which computes both the numerical and analytical gradients. For analytical gradients, batch size and the number of epochs are set to 1 with no regression (i.e. $\lambda = 0$).

To compare these gradients, we used the function `grad_diff` (which computes the relative error between the numerical and analytical gradients). We concluded that the relative error was small (in the order of 10^{-8} for W and 10^{-1} for b) and this did not change significantly as we increased the sample size.

Moreover, with the right configuration of hyperparameters, similar error values were observed as those presented in the lab instructions.

Results

A summary of the results is found in Table 1,2,3. From the results, especially Table 2, one can see that increasing the amount of regularization makes the training and test loss more similar and smoother, when comparing the the top (without regularization) vs. bottom (with regularization) images. From the same table, we can observe that having a very large learning rate will increase the number of spikes in the graph and in some cases may not lead to convergence (as seen in top-left image).

Lambda	Eta	Training error	Test error
0	0.1	0.1844	0.1705
0	0.01	0.4149	0.3685
0.1	0.01	0.3417	0.3338
1	0.01	0.2223	0.2173

Figure 1: Training and test error for each set of hyperparameters used during training (with $epochs = 40, n\_batch = 100$).
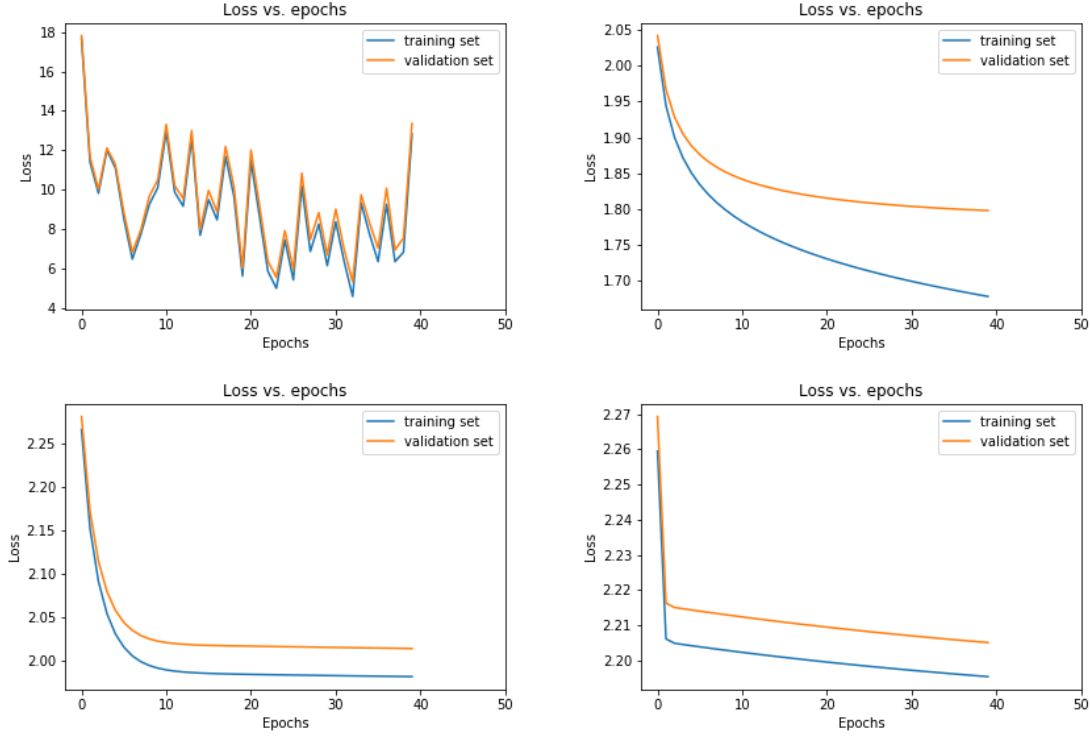

Figure 2: The loss vs. the number of epochs for each set of hyperparameters in 1. Row 1-2 are the top images (from left to right in ascending order), and row 3-4 are at the bottom.
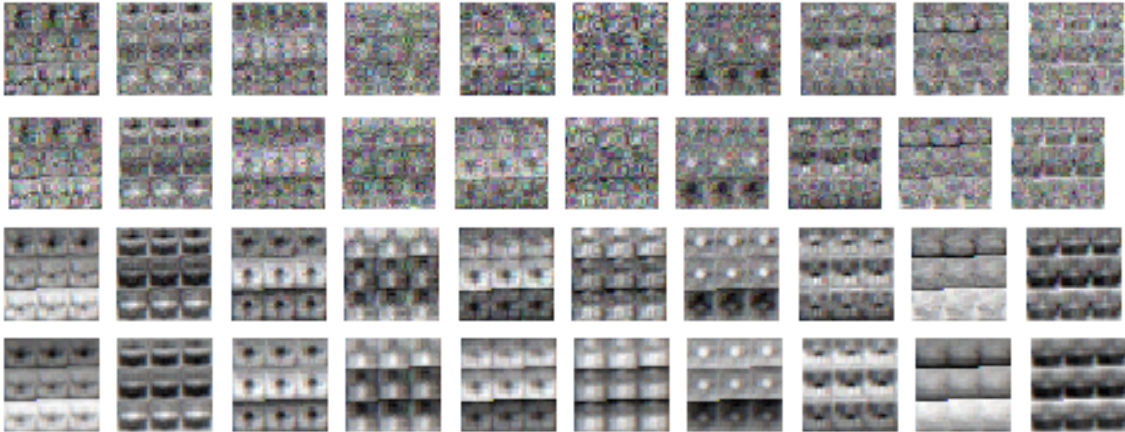

Figure 3: The learned weights for each of the parameter sets in Table 1. Each row corresponds to a row in this figure.