Institut for Theoretical Physics
University Leipzig

Prof. Dr. B. Rosenow
A. Afanah , M. Staats

Statistical Mechanics of Deep Learning - Problem set 3

Winter Term 2023/24

Hand in: Friday, 03.11 at 09:00 am, you can upload your solutions to the course webpage on Moodle platform.

5. Convolutional networks *6 Points*

(a) Train a Convolutional Neural Network on the Dataset MNIST. The architecture should contain two convolutional layers with max-pooling. The size of the Convolutional layers should be chosen such that after the second max-pooling, we have exactly 256 neurons (the flattened array has 256 entries). The max-pooling should be of dimension $(2, 2)$.

(b) Try to estimate the effect that new examples have on the performance of a neural network compared to showing the same example several times. Train the network on a subset of 10000 examples for 6 epochs and evaluate the test set afterward. Now train on 20000 and 30000 examples while keeping the number of training steps identical i.e. 3 epoch for 20000 and 2 epochs for 30000 examples. Plot the final test accuracies and describe the observed behavior.

(c) Pytorch offers functions that randomly rotate some of the images by chance. Repeat the experiment in b) while augmenting the dataset during training with random rotationsänd random cropping"(to dimension $(24, 24)$). Compare the results to the one obtained in part b).

6. Backpropagation in a convolutional network *6 Points*

The core equations of backpropagation in a network with fully-connected layers are

$$\boldsymbol{\delta}^l = \boldsymbol{\nabla}_a C \ \odot \sigma'(\boldsymbol{Z}^L) \tag{1}$$

$$\boldsymbol{\delta}^l = \left[\left(\boldsymbol{w}^{l+1}\boldsymbol{\delta}^{l+1}\right)^T \ \odot \sigma'(\boldsymbol{Z}^l)\right] \tag{2}$$

$$\frac{\partial C}{\partial b^l_j} = \delta^l_j \tag{3}$$

$$\frac{\partial C}{\partial w^l_{jk}} = a^{l-1}_{jk}\ \delta^l_j \tag{4}$$

Suppose we have a network containing a convolutional layer, a max-pooling layer, and a fully-connected output layer, How are the equations of backpropagation modified.

7. Unstable gradient with rectified linear units *2+2 Points*

In the lecture, the vanishing gradient problem was discussed for sigmoid neurons by analyzing the the derivative of the coast function

$$\frac{\partial C}{\partial b_1} = \sigma'(Z_1)w_2\sigma'(Z_2)w_3\sigma'(Z_3)w_4\sigma'(Z_4)\frac{\partial C}{\partial a_4}.$$

How does the analysis change for networks made up of rectified linear units, ReLU

(a) Calculate an explicit expression for the derivative of the cost function $\frac{\partial C}{\partial b_1}$ for a network with ReLU activation function, under what condition will the gradient be stable ?
Hint : search for the so-called "Dying ReLU " problem.

(b) The problem of "Dying ReLU" is a common problem in networks with rectified linear units, suggest two different methods to avoid the problem?
Hint : Think about the shape of ReLU function? can we use a different variation of the function to avoid the problem?