P04 – Deep learning concepts and building blocks

In this assignment you will go over basic concepts and building blocks related to Deep Learning.

This lab description collects some tutorials to get acquainted with PyTorch, supervised learning, neural networks, activation functions, and the basis of learning (optimization of a loss function and parameter adjustment of the neural network via backpropagation). The first two tasks are to get acquainted with some tutorials demonstrating the above, where you can change parameters and study the change in performance. The last task contains a set of problems, testing your understanding of the concepts from the lectures.

1. Introduction to PyTorch

1.1 Get acquainted with PyTorch by following the tutorial `P04a_Introduction_to_PyTorch.ipynb`. Note that the examples here might be useful for the problems from task 4 – it is worth going through them carefully.

1.2 Run the tutorial `P04b_Learning_by_Example.ipynb`. Note that the number of epochs in model training is set always to 9. Run it like this, then increase the number of epochs and observe the changes in performance. What is the performance improvement increase if you go to 20 or 50 (or 100 if your PC allows it) epochs?

2. Neural networks: Activation functions, optimization, initialization

2.1 Neural networks: Activation functions

Run the tutorial `P04c1_Comparing_Activation_Functions.ipynb` to see how just changing the activation function (ReLu or Sigmoid) affects the performance of a neural network. Then run `P04c_Activation_Functions.ipynb` to understand in more detail more why the choice of activation functions is important and how it affects learning.

2.2 Optimization and initialization

Run the tutorial P04d_Optimization_and_Initialization.ipynb to test the effect of different optimizers and initialization. You can also try changing different parameters on your own and comparing the performance.

2. Deep learning concepts and building blocks problems

The following tasks follow the UDL book problems of chapters 3-5.

3.1 Take a simple feedforward neural network with only one input layer consisting of 2 neurons, one hidden layer with 3 neurons, and an output layer with 1 neuron. All neurons use a ReLU activation function. How would you manually calculate the output of this network given an input and the weights and biases for each neuron?

Hint: Denote the input vector as x = [x1, x2], the weights from the input layer to the hidden layer as W1 = [[w11, w12], [w21, w22]], and denote similarly the biases according to their dimensionality. What do they look like for each layer?

The definition of the ReLU function is f(x) = max(0, x).

3.2 Take a simple feedforward neural network with only one input layer consisting of 2 neurons, one hidden layer with 1 neuron, and an output layer with 1 neuron. All neurons use a sigmoid activation function. How would you manually calculate the output of this network given an input and the weights and biases for each neuron?

3.3 Consider a network with Di = 1 input, Do = 1 output, and K = 10 layers, with D = 10 hidden units in each. Would the number of weights increase more if we increased the depth by one or the width by one? Provide your reasoning.

3.4 Consider a multivariate regression problem in which we predict the height of a person in meters and their weight in kilos from data x. Here, the units take quite different ranges. What problems do you see this causing? Propose some solutions to these problems.

3.5 Consider the task of training a deep neural network for a classification problem. You have the choice of using either Stochastic Gradient Descent (SGD) or Adam as the optimizer.

a) Briefly explain how each of these optimizers works, and how they update the model's weights differently during training.

b) Discuss the potential advantages and disadvantages of each optimizer, and under what circumstances you might choose to use each one.

c) Suppose you chose to use Adam for this task. What are the roles of the hyperparameters beta, gamma, and epsilon in the Adam optimizer, and how might changing these values affect the training process?