Assignment 2: Multilayered Neural Networks

Problems:

1. *Weights initialization for a single neuron*

   Let us have a simple neural network with N inputs and a single output neuron **without bias** (threshold). The neuron uses sigmoidal transfer function

   $$f(x) = \frac{1}{1 + e^{-x}} \ .$$

   Its potential is

   $$\xi = \sum_{i=1}^{N} w_i x_i \, ,$$

   where w_i and x_i are the weight and value of i-th input, respectively, for $i = 1, \ldots, N$.

   For fast learning of this neuron, the absolute value of the potential should not be too high. The input attributes are real values without any restriction, but we know they have a normal probability distribution with mean value $\mu = 0$ and standard deviation σ, $\sigma > 0$. We will initialize the weights of the neuron with values from the uniform distribution on an interval $\langle -a, a \rangle$. How should we set the value a with respect to σ if we require that the potential on the neuron should have zero expected value and standard deviation $A = 1$?

   Hint: A similar problem when the input values were from a uniform distribution was solved within the lecture on multilayered neural networks.

2. *Manual design of a neural network for computing a function*

   Suggest weights of a multilayered neural network computing the function $f(x_1, x_2) = 1 + x_1 + x_2$, where x_1, x_2 are input bits (of value 0 or 1 each). The neurons of the network should use the sigmoidal transfer function with the slope 1 and **they have biases**. The topology of the network must be the following:

   (a) two input neurons – inputs are bits (with value 0 or 1),

   (b) two neurons in a single hidden layer, and

   (c) two neurons in the output layer.

   Outputs of the network (at the output layer only!) will be interpreted as two-bit binary number in the following way:

   - output greater or equal to 0.5 will be considered as logical 1,
   - output less than 0.5 will be considered as logical 0.

   Submit a list of weights and biases of all neurons and also a table with the actual outputs of the network (before rounding) for all four combinations of the input bits.

3. We have a multilayered neural network with the topology 2-2-2, i.e., it has two input neurons, one hidden layer containing two neurons, and two output neurons. All neurons use the sigmoidal transfer function with the slope $\lambda = 2.0$. The extended weight matrix of weights between the input and the hidden layer is

```
np.array([[ 1.1, -2.2],
          [ 0.5,  0.9],
          [ 0.0, -0.4]])
```

and the extended weight matrix between the hidden and the output layer is

```
np.array([[ 2.0,  0.9],
          [-1.0,  1.1],
          [ 0.5, -0.5]])
```

Compute the weights of the network after the first iteration of the back-propagation algorithm for input

```
p = np.array([[-1,1]])
```

with the desired output

```
d = np.array([[0.2, 0.4]])
```

and the learning rate $\alpha = 1.5$.

For solving this assignment, you should not use any library for learning neural networks!

What to submit for full score:

1. A complete derivation of the proper value of the variable a with respect to σ and A. The derivation can be written by hand and submitted as a scanned picture.
2. A list of parameters (weights and thresholds) for all neurons in the network and table showing actual outputs (before rounding) of the network for all four combinations of input bits.
3. A list of the new weights and biases together with a Python script that computes them (e.g., in a Jupyter notebook).