Artem Maevskiy
amaevskij@hse.ru

Machine Learning and Data Mining
Assignment 2

2019/11/13

Overview

The below assignment consists of theoretical and practical parts. Please prepare the solutions for the theoretical part as PDF scans of your derivations on paper (alternatively you can use any software, e.g. LaTeX, to generate PDF with formulas). For the practical part, please prepare the solution in jupyter notebook format. The solutions should be clear and easy to follow.

Please send your PDF and .ipynb files to amaevskij@hse.ru under the following subject:
MLDM-2019-HW2-<YOUR LASTNAME>.

The deadline is: **23:59, 28.11.2019**.

1 Theory

1.1 XOR problem (0.1 points)

Design a neural network to solve the XOR problem:

x_1	x_2	y
0	0	0
0	1	1
1	0	1
1	1	0

Draw the network architecture and specify the weights and activation functions.

1.2 Counting parameters (0.1 points)

How many trainable weights does a 3D convolutional layer have, if the kernel size is 3x4x4, padding is 'VALID', strides and dilations are all unit, and the numbers of input and output channels are 10 and 5, respectively? Please, explain your answer.

1.3 Weight initialization (0.3 points)

Given a layer of neurons $\mathbf{x}^{(a)} \in \mathbb{R}^m$ distributed (before activation) as $\mathbf{x}^{(a)} \sim \mathcal{N}(\mathbf{0}, \mathbb{I})$, derive the initialization rule for weights $\mathbf{W} \in \mathbb{R}^{n \times m}$ such that each component of the next layer $\mathbf{x}^{(b)} = \mathbf{W} \cdot \text{ReLU}(\mathbf{x}^{(a)})$ has unit variance: $\text{Var}(x_i^{(b)}) = 1, \forall i \in \{1, \dots, n\}$. Use random i.i.d. initialization with 0 mean.

2 Practice

2.1 FizzBuzz problem (0.5 points)

Implement and train a neural network that classifies positive integers into one of the following four categories:

1. the number is not divisible by 3 or 5;
2. the number is divisible by 3, but not by 5;
3. the number is not divisible by 3, but is divisible by 5;
4. the number is divisible by both 3 and 5.

Use binary representation of numbers (each bit being a separate feature). Use numbers from 101 to 1023 as training set and numbers from 1 to 100 inclusively as the test set.