EEE418: Advanced Pattern Recognition **Spring 2020**

Lab 5: Neural Network

Lecturer: Xiaobo Jin *Unit: EEE Dept. of XJTLU*

Disclaimer:

1. Lab reports deadlines are strict. University late submission policy will be applied.

2. Collusion and plagiarism are absolutely forbidden (University policy will be applied).

3. Report is due 14 days from the date of running this lab

5.1 Objectives

- Implement the two-layer MLP (multilayer perceptron) algorithm, which is a classic neural network.
- In this experiment, we will use the publicly dataset to verify our algorithm. Download the UCI Breast dataset: http://archive.ics.uci.edu/ml/datasets/breast+cancer+wisconsin+(diagnostic)

5.2 Estimation of Classification Methods

- (**5 marks**) Read the dataset into a list and shuffle it with the **random.shuffle** method. Hint: fix the random seed (e.g. **random.seed(17)**) before calling **random.shuffle**
- (**5 marks**) Split the dataset as five parts to do cross-fold validation: Each of 5 subsets was used as test set and the remaining data was used for training. Five subsets were used for testing rotationally to evaluate the classification accuracy.

5.3 MLP Algorithm

- (**5 marks**) All input feature vectors are augmented with the 1 as follows

$$\hat{X} = \begin{bmatrix} X & \mathbf{1}_{N\times 1} \end{bmatrix},$$

since

$$w^T x + w_0 = \begin{bmatrix} w^T & w_0 \end{bmatrix} \begin{bmatrix} x \\ 1 \end{bmatrix}$$

- (**5 marks**) Scale linearly the attribute values x_{ij} of the data matrix $\hat{X}$ into $[-1, 1]$ for each dimensional feature as follows:

$$x_{ij} \leftarrow 2\frac{x_{ij} - \min_i x_{ij} + 10^{-6}}{\max_i x_{ij} - \min_i x_{ij} + 10^{-6}} - 1$$

where a small constant 10^{-6} is used to avoid that the number is divided by zero.

- (**10 marks**) The label l_n of the n-th example is converted into a K dimensional vector t_n as follows (K is the number of the classes)

$$t_{nk} = \begin{cases} +1, & k = l_n \\ 0, & k \neq l_n. \end{cases}$$

- (**10 marks**) Initialize all weight w_{ij} of MLP network such as $w_{ij} \in \left[-\sqrt{\frac{6}{D+1+K}}, \sqrt{\frac{6}{D+1+K}}\right]$ where D and K is the number of the input nodes and the output nodes (each node is related to a class), respectively.

- (**20 marks**) Choose randomly an input vector x to network and forward propagate through the network (H is the number of the hidden units)

$$\begin{aligned} a_j &= \sum_{i=0}^{D} w_{ji}^{(1)} x_i \\ z_j &= \tanh(a_j) \\ y_k &= \sum_{j=0}^{H} w_{kj}^{(2)} z_j \end{aligned} \tag{5.1}$$

to obtain the error rate $E = \frac{1}{2}\sum_{k=1}^{K}(y_k - t_k)^2$ of the example x. **Notice that the subscript n in the equations is omitted for the convinence.**

- (**10 marks**) Evaluate the δ_k for all output units

$$\delta_k = y_k - t_k$$

- (**10 marks**) Backpropagate the δ's to obtain δ_j for each hidden unit in the network

$$\begin{aligned} \delta_j &= \tanh(a_j)' \sum_{k=1}^{K} w_{kj}\delta_k \\ &= (1 - z_j^2) \sum_{k=1}^{K} w_{kj}\delta_k \end{aligned}$$

- (**10 marks**) The derivative with respect to the first-layer and the second-layer weights are given by

$$\frac{\partial E}{\partial w_{ji}^{(1)}} = \delta_j x_i, \quad \frac{\partial E}{\partial w_{kj}^{(2)}} = \delta_k z_j$$

- The framework of MLP algorithm is as follows, where $\eta = 0.001$. Note that η, T and H are the hyperparameters of the network.

Algorithm 1 Stochastic Backpropagation Algorithm

1: Initialize w, η
2: **for** $t = 1$ to T **do**
3: Shuffle the training data set randomly.
4: **for** $n = 1$ to N **do**
5: Choose the input x_n
6: Forward the input x_n through the network
7: Backward the gradient from the output layer through network to obtain $\frac{\partial E_n}{\partial w_{ji}^{(1)}}$ and $\frac{\partial E_n}{\partial w_{kj}^{(2)}}$
8: Update the weights of the network

$$w_{kj} = w_{kj} - \eta \frac{\partial E_n}{\partial w_{kj}^{(2)}}, \quad w_{ji} = w_{ji} - \eta \frac{\partial E_n}{\partial w_{ji}^{(1)}}$$

9: **end for**
10: **end for**
11: **return** w

- The algorithm may be terminated by setting the total iteration T except that setting the threshold θ of the gradient referred in the lecture slide.
- (**10 marks**) In the test stage, the test example x is forwarded into the network to obtain the output $y_{K\times 1}$ and then assigned to the label with the maximum output value.

5.4 Lab Report

- Write a short report which should contain a concise explanation of your implementation, results and observations.
- **Please insert the clipped running image into your report for each step with the mark.**
- Submit the report and the python source code electronically into ICE.
- The report must be written with the latex typesetting language.
- The report in pdf format and python source code of your implementation should be zipped into a single file. The naming of report is as follows:

  e.g. StudentID_LastName_FirstName_LabNumber.zip (123456789_Einstein_Albert_1.zip)

5.5 Hints

Please refer to the lecture slides.

- Latex IDE: texstudio
- Python IDE: pycharm
- Use the python numpy library flexibly.