BME777: Emerging Topics in Biomedical Engineering (Fall 2017)
(Machine Learning for Health Analytics)

Lab 3: Multilayer Neural Networks

Objective

- Implement multi-layer neural networks using back propagation algorithm to classify linearly non-separable data.

Background

Multi-layer neural networks (MNN) implements the linear discriminant functions however in a feature space where the input patterns are mapped non-linearly. Neural networks are quite powerful and easily implemented using simple algorithms where the form of non-linearity can be learned from the training data. One of the most popular methods for training the MNN is based on the gradient descent in error commonly known as backpropagation algorithm. The below figure shows an example of a 3 layer neural network.

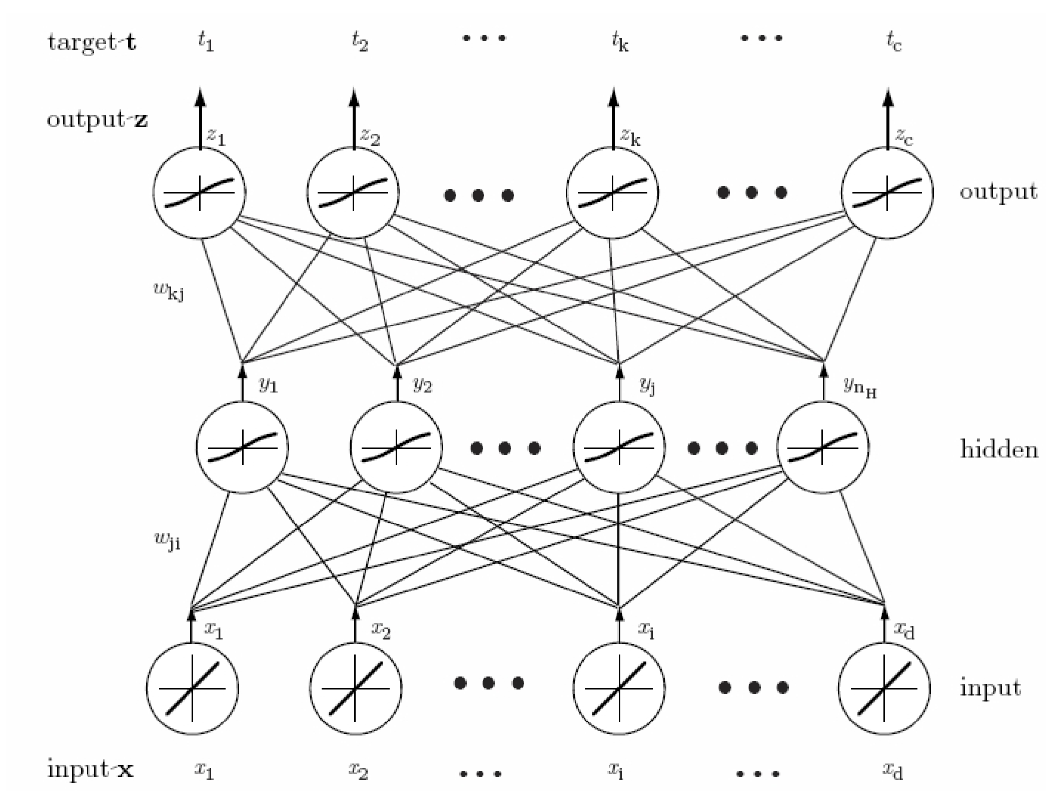

Figure 1: An $d - n_H - c$ fully connected 3 layer NN.

The task is to estimate the weight vectors (w_{ji} and w_{kj}) between the input-hidden layer and the hidden-output layer for a given activation function. The pseudo code of the batch backpropogation algorithm is given below:

begin **initialize** n_H, $\mathbf{w}$, criterion θ,η,$r \leftarrow 0$

do $r \leftarrow r + 1$ (increment epoch)

$m \leftarrow 0$; $\Delta w_{ji} \leftarrow 0$; $\Delta w_{kj} \leftarrow 0$

do $m \leftarrow m + 1$

$\mathbf{x}^m \leftarrow$ select pattern

$\Delta w_{ji} \leftarrow \Delta w_{ji} + \eta \delta_j x_i$; $\Delta w_{kj} \leftarrow \Delta w_{kj} + \eta \delta_k y_j$

until $m = n$

$w_{ji} \leftarrow w_{ji} + \Delta w_{ji}$; $w_{kj} \leftarrow w_{kj} + \Delta w_{kj}$

until $\|\nabla J(\mathbf{w})\| < \theta$

return $\mathbf{w}$

end

The sensitivity of unit k is given by

$$\delta_k = (t_k - z_k) f^{'}(net_k) \tag{1}$$

and the sensitivity for a hidden unit is given by

$$\delta_j = f^{'}(net_j) \sum_{k=1}^{c} w_{kj} \delta_k \tag{2}$$

The (net_j) and (net_k) are the net activation and $f(net_j)$ and $f(net_k)$ are the non-linear activation functions. The derivative of the function $f(.)$ is shown as $f^{'}(.)$.

Laboratory Exercises

The *lab3.zip* file contains this handout, *DataLab3.mat*, and a skeleton MATLAB program *lab3.m*.

- Construct a 2-2-1 neural network using the batch backpropogation algorithm for solving the classical **XOR** problem. The two inputs are given as $x_1 = [-1 \ -1 \ 1 \ 1]$ and $x_2 = [-1 \ 1 \ -1 \ 1]$. The targets (correct output) are $t = [-1 \ 1 \ 1 \ -1]$. Use a $\eta = 0.1$ and threshold $\theta = 0.001$. Assume the following sigmoid activation function (for both hidden and output units)

$$f(x) = a * tanh(b * x) \tag{3}$$

where $a = b = 1$.

- Verify that the computed final weight vectors satisfy the XOR operation. Plot the learning curve and note the number of epochs needed for convergence.
- Plot the decision surfaces in $x_1 - x_2$ and $y_1 - y_2$ spaces.
- Repeat all of the above steps using the given database "'DataLab3.mat"' and compute the classification accuracy. "DataLab3" is a Heart Disease database that contains two features [x_1 - Resting blood pressure and x_2 - ST depression induced by exercise relative to rest] from two different groups (ω_1 - Absence of heart disease and ω_2 - Presence of heart disease). The size of the data matrix is 100 X 3 (i.e. 100 rows and 3 columns). The first column contains the feature x_1 and the second column contains the feature x_2. The class labels [$\omega_1 = 1, \omega_2 = -1$] are given in the third column.

Lab Evaluation Information

- Submit via email the documented MATLAB code for Part I of the lab exercise before the demo. In the lab, the programs should be demonstrated to the TA/Instructor and each student will be evaluated on the components covered in the lab material. Marks will be provided based on the (i) demo and (ii) student′s response to the questions [3 + 4 marks].
- Submit via email a report addressing all the outcomes of the lab exercise, and your observations and conclusions. (3 marks)

Due date

- Due date for the report will be before the start of the next lab. Reports can be submitted via email till 5 pm on the due date, however demo/ evaluation in lab must be completed during scheduled lab sessions for this lab (i.e., Week 7-8).

Acknowledgment

We thankfully acknowledge UCI Machine Learning Repository for the dataset used in this lab exercise.
Lichman, M. (2013). UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science.