Deep Learning –HW#3

About the Assignment

The main aim of the assignment is to make you familiar with the linear classifiers, specifically with the perceptron algorithm. Please solve the problems individually and cheating will be harshly punished.

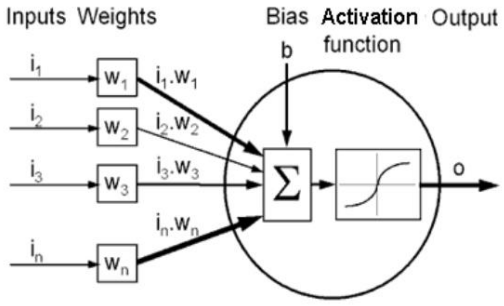

Fig. 1. General demonstration of perceptron algorithm [1].

Fig. 1 shows the general demonstration of perceptron algorithm. In the last head of activation, we will use the sigmoid function.

We will use the gradient descent algorithm in case of training our perceptron algorithm.

perceptron(training_samples, w_{init} , η)

Each training sample is a pair of the form $\langle \vec{x}, t \rangle$ where $\vec{x}$ is the vector of input values, and t is target output value. η is 0.0001.

- Initialize each weight, w_i to some small value.
- Until the all samples classified correctly or iteration condition met. Do
  - Initialize each $\triangle w_i$ $(called\ gradient)$ to zero
  - For each training samples, Do
    - Input the instance $\vec{x}$ to the unit and compute the output o

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

    - For each linear unit weight w_i Do

$$\triangle w_i = \eta (t-y) g'(y) x_i,$$

$(t-y)$ *referes to error*

$g'(y)$ *refers to* derivative of sigmoid function

$$g'(y) = g(y)(1-g(y))$$

  - For each linear unit weight w_i , Do

$$w_i = w_i + \triangle w_i$$

[1] http://aass.oru.se/~lilien/ml/seminars/2007_02_01b-Janecek-Perceptron.pdf

Submit the Assignment

Please write all answers on this document and forward this document as pdf format.

Ex: No_Name_Surname_HW#.pdf

Hint

You can look the implementation perceptron on notes of the course.

Step 1:

Implement a *trainPerceptron(inputs, t, weights, rho, iterNo):* function in PYTHON, in order to train the linear classifer, so called perceptron algorithm.

inputs **:** Feature vectors belonging to classes (0 and 1)

t **:** labels of classes (0 and 1)

weights: Initial weights for the linear discriminant function.

rho **:** learning rate is 0.0001

iterNo**:** refers to number of iterations in perceptron is 100

Step 2:

To implement your *testPerceptron(sample_test, weights)* function, you should return predicted value.

I have uploaded my python code for HW3 as HW3.py

You can open it, in your spyder or jupyter-notebook to fulfill the required tasks. Besides, the required python libraries are imported to use when it is needed.