ARTIFICIAL NEURAL NETWORK

Unit-2: Perceptron

Ms. Swetha R.
Department of Electronics and Communication Engineering
PES University

BPA: Heuristics to make performance better

- **Sequential v/s Batch mode:**
  - For large data sequential processing is faster
  - For smaller data and network, batch processing is faster
- **Maximizing the information content**
- Every traning example presented to the BPA should be choosen on the basis that its information content is the largest possible for the task at hand
- This can be achieved in 2 ways

  a. use of an example that results in the largest traing error

  b. the use of example that is radially different from all those previously used

BPA: Heuristics to make performance better

Activation Function:

- A M.L.P trained with the back-propagation algorithm may, in general,learn faster when the asymmetric sigmoidal activation function are used in neuron model,for example tanh(.).

Target Value: it is important to choose the target value within the activation function range i.e +1 and -1

Normalization of the inputs:

The training samples must be preprocessed before presenting to the neural network. the preprocessing steps are

BPA: Heuristics to make performance better

a. subtracting each input from mean

b.decorrelate the training samples

c. covariance: it ensures the different synaptic weight in the networks learn at same speed.

Learning Rate

Initialization

Artificial Neural Network-Perceptron

BPA-Output representation and decision rule

Consider a M-class classification problems, which means each input pattern belong to one of the M-distinct classes.

Therefore, we will have M number outputs in the network.

Xj denotes the jth input pattern. and it is m-dimensional input vector

The output is M-dimensional vector. ykj is the kth ouput of network in correspondence to the jth input

BPA-Output representation and decision rule

Therefore, block diagram can be repressented as

$$y_{kj} = F_k(x_j)$$

Let the output vector y_j

$$y_j = \left[y_{1j} \quad y_{2j} \quad \ldots\ldots. \quad y_{kj} \quad \ldots.. \quad y_{Mj}\right]^T$$

$$y_j = \left[F_1(x_j) \quad F_2(x_j) \quad \ldots\ldots. \quad F_k(x_j) \quad \ldots.. \quad F_M(x_j)\right]^T$$

$$y_j = F(x_j)$$

BPA-Output representation and decision rule

F(.) is continuous function and minimizes the emprical risk function

$$R = \frac{1}{2N}\sum_{j=1}^{N}\left\| d_j - F(x_j) \right\|^2$$

Now train the network with binary values as follows:

- The output space has M dimension
- An input Xj belongs to class Ck
- let d_k=1 when Xj belongs to class C_k otherwise 0

Output Decision Rule stated as follows:

Classify the random vector X as belonging to C_k if

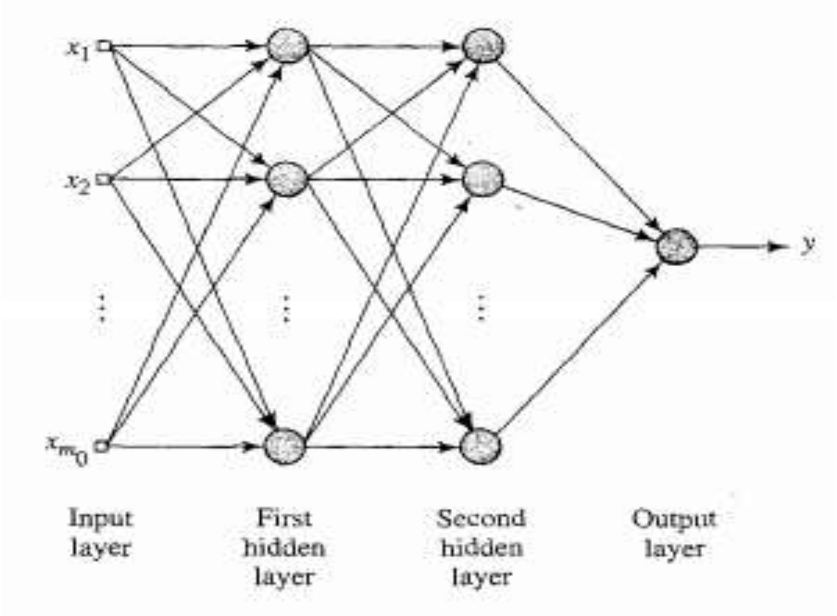
$$F_k(X) > F_j(X) \quad \text{for all } k \neq j$$

BPA-Feature detection

- Hidden layer neurons play a important role in the operation of MLP with BPA because they act as a feature detectors.
- As the laerning process progresses, the hidden neurons begin to discover the salient features that characterizes the traing data.
- Performing non linear transformation from the input space to hidden space.
- For example,non linearly separable classes will be easily separated in the hidden space

Artificial Neural Network-Perceptron

Back-propagation and differentiation



The above MLP is parameterised by an architecture A (representing a discrete parameter) and a weight vector W

Artificial Neural Network-Perceptron

Back-propagation and differentiation

Let Alj denote the part of the architecture extending from the input layer to node j in layer (l=1,2,3). Accordingly we may write

$$F(W,X)=\varphi(A_{31})$$

$$\frac{\partial F(W,X)}{\partial w_{3lk}}=\varphi'(A_{3l})\varphi(A_{2k})$$

$$\frac{\partial F(W,X)}{\partial w_{2kj}}=\varphi'(A_{3l})\varphi'(A_{2k})\varphi(A_{1j})w_{3lk}$$

$$\frac{\partial F(W,X)}{\partial w_{1ji}}=\varphi'(A_{3l})\varphi'(A_{1j})X_j\left[\sum_k w_{3lk}\varphi'(A_{2k})w_{2kj}\right]$$

Back-propagation and differentiation

The sensitivit of F(w,x) is

$$S_w^F = \frac{\partial F \; / \; F}{\partial w \; / \; w}$$

Home Work:

1. Jacobian Matrix
2. Hessian Matrix