

NPTEL ONLINE CERTIFICATION COURSES

Course Name: Fuzzy logic and Neural Networks

Faculty Name: Prof. D.K. Pratihar

Department: Mechanical Engineering

Week 5

NPTEL



NPTEL ONLINE CERTIFICATION COURSES

Course Name: Fuzzy logic and Neural Networks

Faculty Name: Prof. D.K. Pratihar

Department: Mechanical Engineering

NPTEL

Topic

Lecture 06: Some Examples of Neural Networks

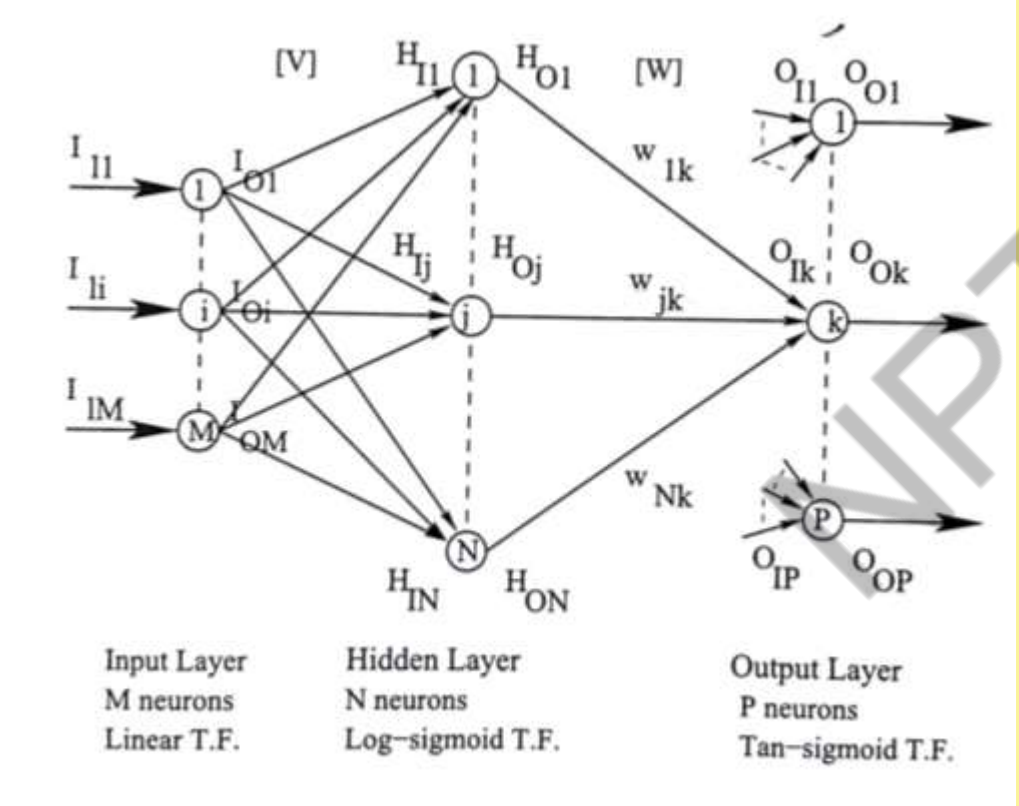
CONCEPTS COVERED

Concepts Covered:

- Multi-Layer Feed-Forward Neural Networks
- Radial Basis Function Networks
- Recurrent Neural Networks
- Self-Organizing Map
- Counter-Propagation Neural Networks

NPTEL

Multi-Layer Feed-Forward Neural Network (MLFFNN)



I_{I1}: Input of 1-st neuron lying on input layer

I_{O1}: Output of 1-st neuron lying on input layer

NPTEL

$$[V] = \begin{bmatrix} V_{11} \ldots . V_{1j} \ldots . V_{1N} \\ . \quad . \quad . \\ . \quad . \quad . \\ V_{i1} \ldots . V_{ij} \ldots . V_{iN} \\ . \quad . \quad . \\ . \quad . \quad . \\ V_{M1} \ldots . V_{Mj} \ldots . V_{MN} \end{bmatrix}$$

$$[W] = \begin{bmatrix} W_{11} \ldots . W_{1K} \ldots . W_{1P} \\ . \quad . \quad . \\ . \quad . \quad . \\ W_{j1} \ldots . W_{jK} \ldots . W_{jP} \\ . \quad . \quad . \\ . \quad . \quad . \\ W_{N1} \ldots . W_{Nk} \ldots . W_{NP} \end{bmatrix}$$

NPTEL

Forward Calculations

- **Step 1:** Determination of the outputs of input layer

$$I_{Oi} = I_{Ii},$$

where i = 1, 2, ..., M

- **Step 2:** Calculation of inputs of hidden layer

$$H_{Ij} = v_{1j} I_{O1} + ...+ v_{ij} I_{Oi} + ……+ v_{Mj} I_{OM},$$

where j = 1, 2, ... , N

Determination of the outputs of hidden neurons

NPTEL

Step 3: Determination of the outputs of hidden neurons

$$H_{O_j} = \frac{1}{1 + e^{-a_1 H_{Ij}}}$$

where a_1 is the coefficients of TF

Step 4: Determination of the inputs of output layer

$O_{Ik} = w_{1k} H_{O1} + \ldots\ldots + w_{jK} H_{Oj} + \ldots\ldots + w_{NK} H_{ON}$,

where k= 1, 2, ... , P

NPTEL

FREE ONLINE EDUCATION swayam

Step 5: Estimation of the outputs of Output layer O

$$O_{Ok} = \frac{e^{a_2 O_{Ik}} - e^{-a_2 O_{Ik}}}{e^{a_2 O_{Ik}} + e^{-a_2 O_{Ik}}}$$

where a_2 is the coefficient of TF

Step 6: Determination of error in prediction

Error in prediction at k-th output neuron

$$E_k = \frac{1}{2}(T_{Ok} - O_{Ok})^2$$

NPTEL

FREE ONLINE EDUCATION swayam

Error in prediction considering all output neurons

$$E = \sum_{k=1}^{P} \frac{1}{2} (T_{Ok} - O_{Ok})^2$$

where P: No. of output neurons

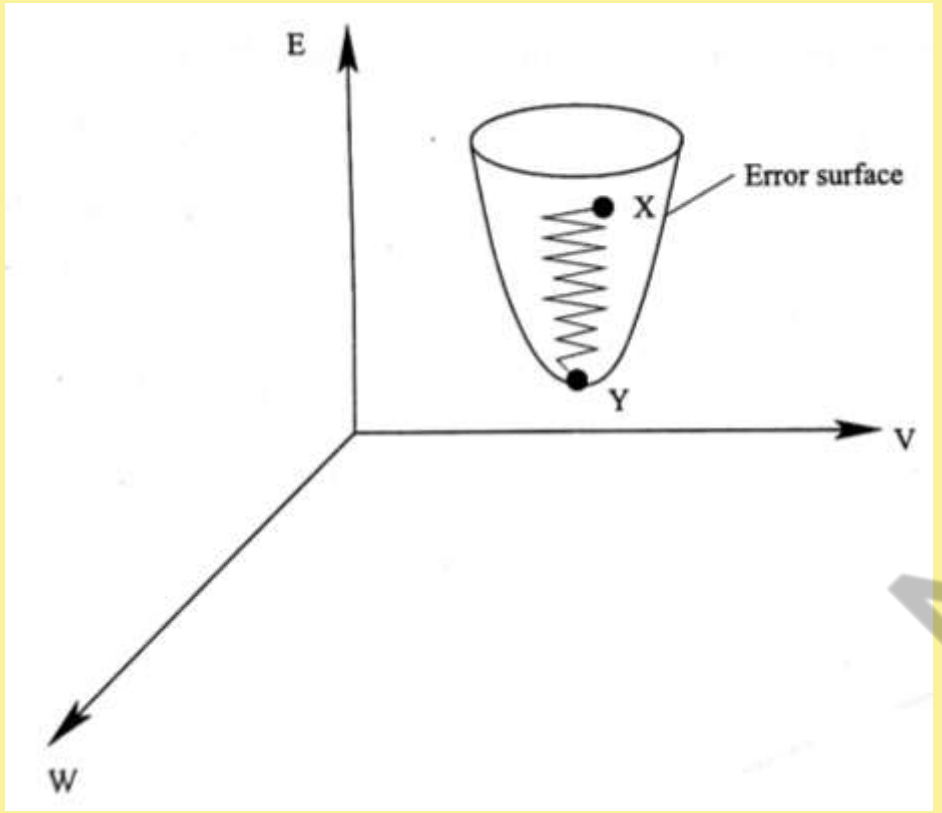
Total error in prediction considering all L training scenarios

$$E_{total} = \sum_{l=1}^{L} \sum_{k=1}^{P} \frac{1}{2} (T_{Okl} - O_{Okl})^2$$

where L: No. of training scenarios

NPTEL

Back-Propagation Algorithm



- **Delta Rule**

$$E = f(V, W)$$

$$\Delta V = -\eta . \frac{\partial E}{\partial V}$$

$$\Delta W = -\eta . \frac{\partial E}{\partial W}$$

NPTEL

Incremental Mode of Training

Updating of [W]

$$w_{jk,\ updated} = w_{jk,\ previous} + \Delta w_{jk}$$

where $\Delta w_{jk} = -\eta . \frac{\partial E_k}{\partial W_{jk}}$

Now, $\frac{\partial E_k}{\partial w_{jk}} = \frac{\partial E_k}{\partial O_{Ok}} . \frac{\partial O_{Ok}}{\partial O_{Ik}} . \frac{\partial O_{Ik}}{\partial w_{jk}}$

$$\frac{\partial E_k}{\partial O_{Ok}} = -(T_{Ok} - O_{Ok})$$

NPTEL

FREE ONLINE EDUCATION swayam

$$\frac{\partial O_{Ok}}{\partial O_{Ik}} = a_2(1 + O_{Ok})(1 - O_{Ok})$$

$$\frac{\partial O_{Ik}}{\partial w_{jk}} = H_{Oj}$$

$$\frac{\partial E_k}{\partial w_{jk}} = -(T_{Ok} - O_{Ok})a_2(1 + O_{Ok})(1 - O_{Ok})H_{Oj}$$

$$\Delta w_{jk} = \eta a_2(T_{Ok} - O_{Ok})(1 + O_{Ok})(1 - O_{Ok})H_{Oj}$$

NPTEL

FREE ONLINE EDUCATION swayam

To update the connecting weight v_{ij} between i-th neuron of input layer and j-th neuron of hidden layer, that is,

$$v_{ij,\ updated} = v_{ij,\ previous} + \Delta v_{ij}$$

where $\Delta V_{ij} = -\eta \left\{ \frac{\partial E}{\partial V_{ij}} \right\}_{av}$

where $\left\{ \frac{\partial E}{\partial V_{ij}} \right\}_{av} = \frac{1}{p} \sum_{k=1}^{p} \frac{\partial E_k}{\partial V_{ij}}$

NPTEL

FREE ONLINE EDUCATION swayam

Now, $$\frac{\partial E_k}{\partial v_{ij}} = \frac{\partial E_k}{\partial O_{Ok}} \cdot \frac{\partial O_{Ok}}{\partial O_{Ik}} \cdot \frac{\partial O_{Ik}}{\partial H_{Oj}} \cdot \frac{\partial H_{Oj}}{\partial H_{Ij}} \cdot \frac{\partial H_{Ij}}{\partial v_{ij}}$$

where, $$\frac{\partial E_k}{\partial O_{Ok}} = -(T_{Ok} - O_{Ok})$$

$$\frac{\partial O_{Ok}}{\partial O_{Ik}} = a_2(1 + O_{Ok})(1 - O_{Ok})$$

NPTEL

FREE ONLINE EDUCATION swayam शिक्षित भारत, उन्नत भारत

$$\frac{\partial O_{Ik}}{\partial H_{Oj}} = w_{jk}$$

$$\frac{\partial H_{Oj}}{\partial H_{Ij}} = a_1 H_{Oj}(1 - H_{Oj})$$

$$\frac{\partial H_{Ij}}{\partial v_{ij}} = I_{Oi} = I_{Ii}$$

We get

$$\frac{\partial E_k}{\partial v_{ij}} = -a_1 a_2 (T_{Ok} - O_{Ok})(1 + O_{Ok})(1 - O_{Ok})(1 - H_{Oj}) w_{jk} H_{Oj} I_{Ii}$$

NPTEL

FREE ONLINE EDUCATION swayam

Batch Mode of Training:

Let us consider L training scenarios. Mean Squared Deviation in prediction for k-th output neuron

$$E' = \frac{1}{2} \cdot \frac{1}{L} \sum_{l=1}^{L} (T_{Okl} - O_{Okl})^2$$

The change in w_{jk}, that is, Δw_{jk} is determined as follows:

$$\Delta w_{jk} = -\eta \cdot \frac{\partial E'}{\partial w_{jk}}$$

Now,

$$\frac{\partial E'}{\partial w_{jk}} = \frac{\partial E'}{\partial E_l} \cdot \frac{\partial E_l}{\partial E_k} \cdot \frac{\partial E_k}{\partial O_{Ok}} \cdot \frac{\partial O_{Ok}}{\partial O_{Ik}} \cdot \frac{\partial O_{Ik}}{\partial w_{jk}}$$

NPTEL

Similarly, Δv_{ij} can be calculated as follows: $\Delta v_{ij} = -\eta . \left\{ \frac{\partial E'}{\partial v_{ij}} \right\}_{av}$

Where $\left\{ \frac{\partial E'}{\partial v_{ij}} \right\}_{av} = \frac{1}{p} \sum_{k=1}^{p} \frac{\partial E'_k}{\partial v_{ij}}$

Now,

$$\frac{\partial E'_k}{\partial v_{ij}} = \frac{\partial E'_k}{\partial E_{kl}} . \frac{\partial E_{kl}}{\partial O_{Ok}} . \frac{\partial O_{Ok}}{\partial O_{Ik}} . \frac{\partial O_{Ik}}{\partial H_{Oj}} . \frac{\partial H_{Oj}}{\partial H_{Ij}} . \frac{\partial H_{Ij}}{\partial v_{ij}}$$

NPTEL

Momentum Constant (α')

Generalized Delta Rule:

$$\Delta w(t) = -\eta \frac{\partial E}{\partial w}(t) + \alpha' \Delta w(t-1)$$

η: (0.0 to 1.0)
α': (0.0 to 1.0)

α' is used to ensure a stable network even at a higher value of learning rate η

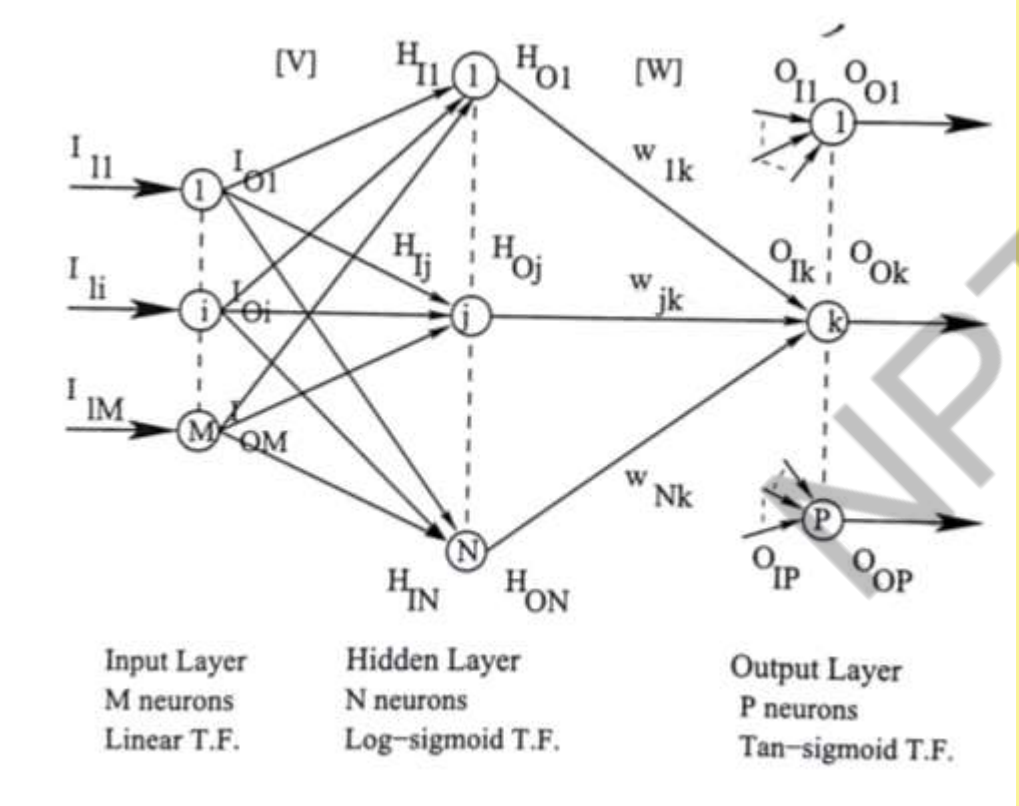
NPTEL

Notes:

- BP algorithm → there is a chance of local minima problem
- BP algorithm → transfer functions are to be differentiable in nature
- BPNN may not be able to capture the dynamics of a highly dynamic process
- Inputs are normalized
- Connecting weights lie in the range of either (0.0, 1.0) or (-1.0,1.0)
- Convergence Criterion: Absolute value of the rate of change of error in prediction becomes less than or equal to a pre-specified value
- A neural network can be either fully-connected or partially-connected one

NPTEL

Multi-Layer Feed-Forward Neural Network (MLFFNN)



I_{I1}: Input of 1-st neuron lying on input layer

I_{O1}: Output of 1-st neuron lying on input layer

NPTEL

Advantages of BPNN

- Can handle a problem having many variables
- May not require so much problem information as that is needed for an FLC

Disadvantages of BPNN

- Solutions of BPNN may get stuck at the local minima
- Training of NN is more involved computationally compared to that of FLC
- It works like a black box

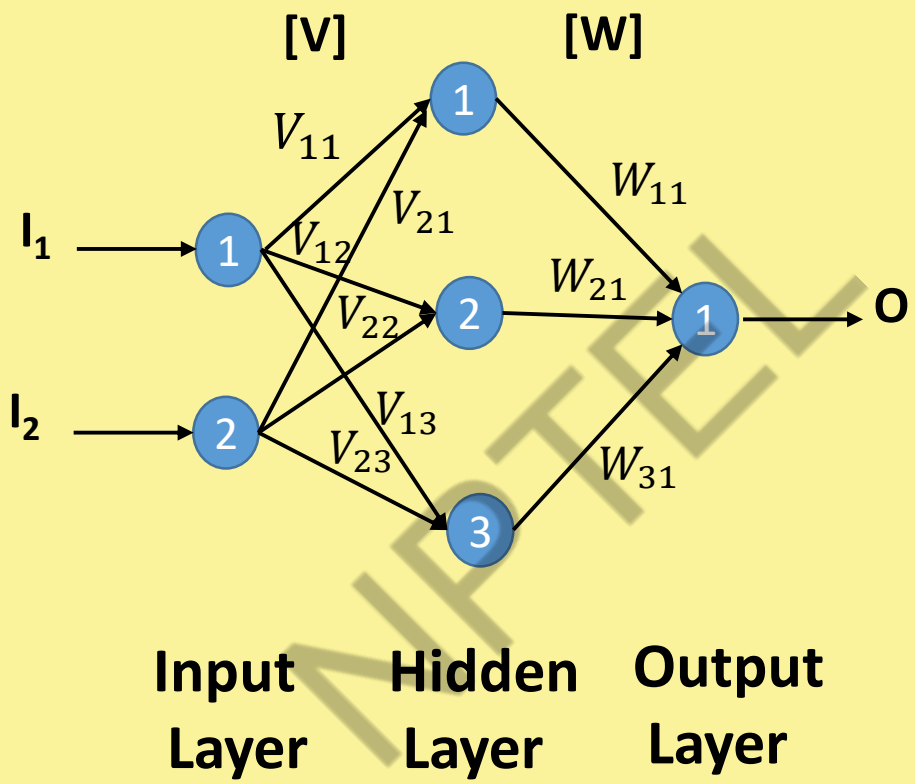
NPTEL

Numerical Example

Figure below shows the schematic view of an NN consisting of three layers, such as input, hidden and output layers. The neuron lying on the input, hidden and output layers have the transfer function represented by $y = x, y = \frac{1}{1+e^{-x}}, y = \frac{e^{x}-e^{-x}}{e^{x}+e^{-x}}$, respectively. There are two inputs, namely I_1 and I_2 and one output, that is, O. The connecting weights between the input and hidden layers are represented by [V] and those between hidden and output layers are denoted by [W]. The initial values of the weights are assumed to be as follows:

NPTEL

FREE ONLINE EDUCATION swayam


[V]
[W]
V_{11}
V_{21}
V_{12}
V_{22}
V_{13}
V_{23}
W_{11}
W_{21}
W_{31}
I_1
I_2
1
2
3
1
2
1
O
Input Layer
Hidden Layer
Output Layer

NPTEL

$$\begin{bmatrix} v_{11} & v_{12} & v_{13} \\ v_{21} & v_{22} & v_{23} \end{bmatrix} = \begin{bmatrix} 0.2 & 0.4 & 0.3 \\ 0.1 & 0.6 & 0.5 \end{bmatrix};$$

$$\begin{bmatrix} w_{11} \\ w_{21} \\ w_{31} \end{bmatrix} = \begin{bmatrix} 0.1 \\ 0.2 \\ 0.1 \end{bmatrix}$$

Sl. No.	I_1	I_2	T_O
1	0.5	-0.4	0.15
2	-	-	-
.	.	.	.
.	.	.	.
.	.	.	.

Using an incremental mode of training for the case shown in the Table, calculate the changes in V (that is, ΔV) and W (that is ΔW) values during back-propagation of error, the learning rate η is assumed to be equal to 0.2. Show only one iteration.

NPTEL

Solution:

Forward Calculations:

The neurons of input layer have linear transfer function (y=x). Therefore, the output of each input layer neuron is made equal to its corresponding input value.

$$I_{O1} = I_{I1} = 0.5$$
$$I_{O2} = I_{I2} = -0.4$$

The inputs of different neurons of the hidden layer are calculated as follows:

$$H_{I1} = I_{O1}v_{11} + I_{O2}v_{21} = 0.06$$
$$H_{I2} = I_{O1}v_{12} + I_{O2}v_{22} = -0.04$$
$$H_{I3} = I_{O1}v_{13} + I_{O2}v_{23} = -0.05$$

NPTEL

The neurons of the hidden layer are assumed to have log-sigmoid transfer function ($y = \frac{1}{1+e^{-x}}$). The outputs of different hidden neurons are determined like the following:

$$H_{O1} = \frac{1}{1 + e^{-H_{I1}}} = 0.514995$$

$$H_{O2} = \frac{1}{1 + e^{-H_{I2}}} = 0.490001$$

$$H_{O3} = \frac{1}{1 + e^{-H_{I3}}} = 0.487503$$

NPTEL

Now, the input of the output neuron can be calculated as follows:

$$O_{I1} = H_{O1}w_{11} + H_{O2}w_{21} + H_{O3}w_{31} = 0.198249$$

As the output neuron has a tan-sigmoid transfer function, its output can be determined like the following:

$$O_{O1} = \frac{e^{O_{I1}} - e^{-O_{I1}}}{e^{O_{I1}} + e^{-O_{I1}}} = 0.195692$$

The squared error in prediction is found to be as follows:

$$E = \frac{1}{2}(T_O - O_{O1})^2 = 0.001044$$

NPTEL

Back-propagation Algorithm:

The change in w_{11} can be determined using the procedure below.

$$\Delta w_{11} = -\eta \frac{\partial E}{\partial w_{11}},$$

where $$\frac{\partial E}{\partial w_{11}} = \frac{\partial E}{\partial O_{O1}} \frac{\partial O_{O1}}{\partial O_{I1}} \frac{\partial O_{I1}}{\partial w_{11}}.$$

Now, $$\frac{\partial E}{\partial O_{O1}} = -(T_O - O_{O1})$$

$$\frac{\partial O_{O1}}{\partial O_{I1}} = \frac{4}{(e^{O_{I1}} + e^{-O_{I1}})^2}$$

$$\frac{\partial O_{I1}}{\partial w_{11}} = H_{O1}.$$

NPTEL

FREE ONLINE EDUCATION swayam

Substituting the values of $T_O, O_{O1}, O_{I1}, H_{O1}$ in the last expression of $\frac{\partial E}{\partial w_{11}}$ we get

$$\frac{\partial E}{\partial w_{11}} = 0.022630$$

Now, substituting the values of $\frac{\partial E}{\partial w_{11}}$ and η in the expression of Δw_{11}, we get

$$\Delta w_{11} = -0.004526$$

Similarly, we can determine Δw_{21} and Δw_{31} and these are found to be as follows:

$$\Delta w_{21} = -0.004306$$
$$\Delta w_{31} = -0.004284$$

NPTEL

The necessary change in v_{11} can be obtained as follows:

$$\Delta v_{11} = -\eta \frac{\partial E}{\partial v_{11}}$$

where $$\frac{\partial E}{\partial v_{11}} = \frac{\partial E}{\partial O_{O1}} \frac{\partial O_{O1}}{\partial O_{I1}} \frac{\partial O_{I1}}{\partial H_{O1}} \frac{\partial H_{O1}}{\partial H_{I1}} \frac{\partial H_{I1}}{\partial v_{11}}$$

Now, $$\frac{\partial E}{\partial O_{O1}} = -(T_O - O_{O1})$$

$$\frac{\partial O_{O1}}{\partial O_{I1}} = \frac{4}{(e^{O_{I1}} + e^{-O_{I1}})^2}$$

NPTEL

$$\frac{\partial O_{I1}}{\partial H_{O1}} = w_{11}$$

$$\frac{\partial H_{O1}}{\partial H_{I1}} = \frac{e^{-H_{I1}}}{(1 + e^{-H_{I1}})^2}$$

$$\frac{\partial H_{I1}}{\partial v_{11}} = I_{O1}$$

Substituting the values of $T_O, O_{O1}, O_{I1}, w_{11}, H_{I1}$ and I_{O1} in the last expression of $\frac{\partial E}{\partial v_{11}}$, we obtain

$$\frac{\partial E}{\partial v_{11}} = 0.000549$$

NPTEL

Now, substituting the values of $\frac{\partial E}{\partial v_{11}}$ and η, we get $\Delta v_{11} = -0.000110$

Similarly, the values of $\Delta v_{21}, \Delta v_{12}, \Delta v_{22}, \Delta v_{13}$ and Δv_{23} are determined and found to be as follows:

$$\Delta v_{21} = 0.000088$$
$$\Delta v_{12} = -0.000220$$
$$\Delta v_{22} = 0.000176$$
$$\Delta v_{13} = -0.000110$$
$$\Delta v_{23} = 0.000088$$

NPTEL

Therefore, the updated values of the weights are coming out to be as follows:

$$\begin{bmatrix} v_{11} & v_{12} & v_{13} \\ v_{21} & v_{22} & v_{23} \end{bmatrix} = \begin{bmatrix} 0.199890 & 0.399780 & 0.299890 \\ 0.100088 & 0.600176 & 0.500088 \end{bmatrix};$$

$$\begin{bmatrix} w_{11} \\ w_{21} \\ w_{31} \end{bmatrix} = \begin{bmatrix} 0.095474 \\ 0.195694 \\ 0.095716 \end{bmatrix}$$

NPTEL

Radial Basis Function Network (RBFN)

Radial Basis Function

A special type of function, whose response decreases or increases monotonically with its distance from a central point

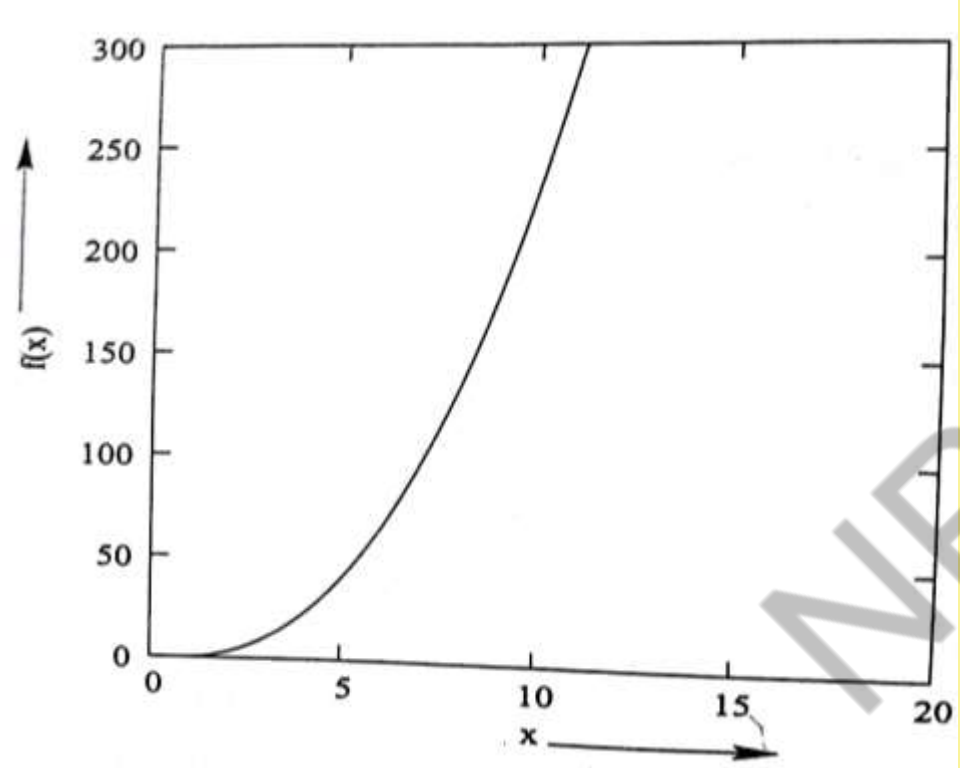
NPTEL

Various types of Radial Basis Function:

- Thin plate spline function: f(x) = $x^2 \log x$
- Gaussian function: f(x) = $exp\left(-\frac{\|x-\mu\|^2}{2\sigma^2}\right) = exp\left(-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right)$
- Multi-quadratic function: f(x) = $\sqrt{x^2+\sigma^2}$
- Inverse multi-quadratic function: f(x) = $\frac{1}{\sqrt{x^2+\sigma^2}}$

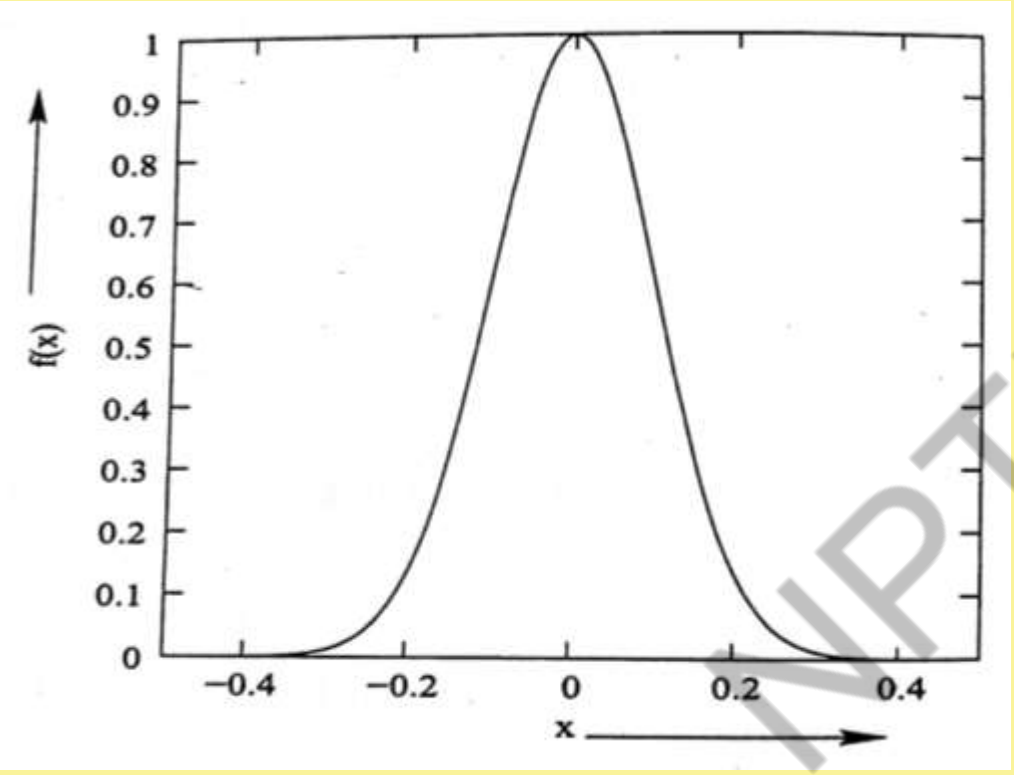
NPTEL

Radial Basis Functions



Thin Plate Spline

NPTEL

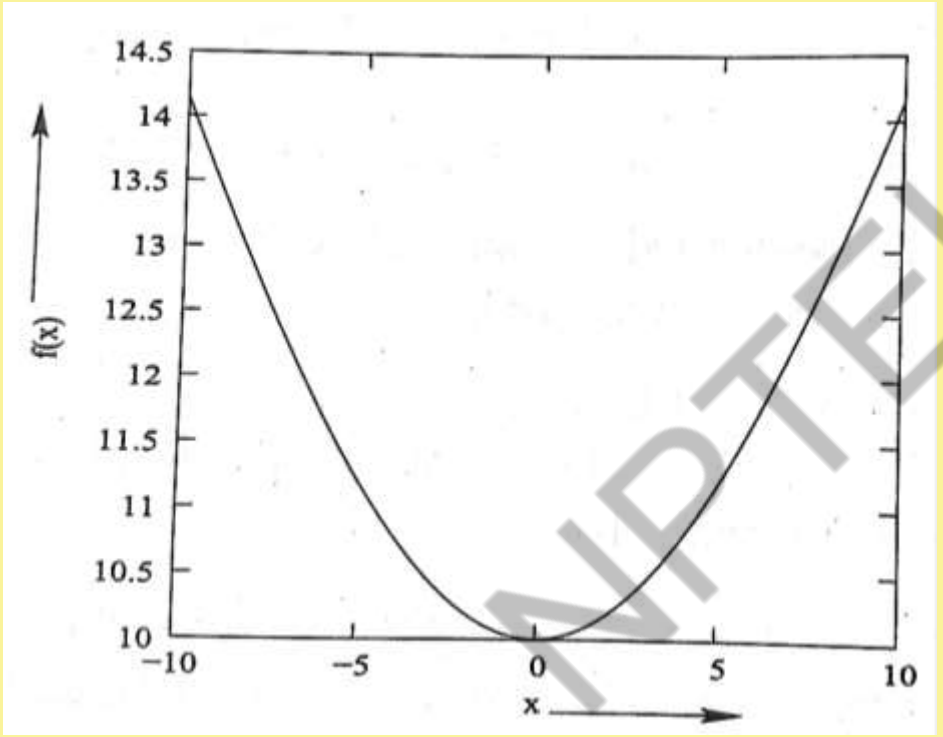


NPTEL

Gaussian Function

FREE ONLINE EDUCATION
swayam
शिक्षित भारत, उन्नत भारत

Radial Basis Functions

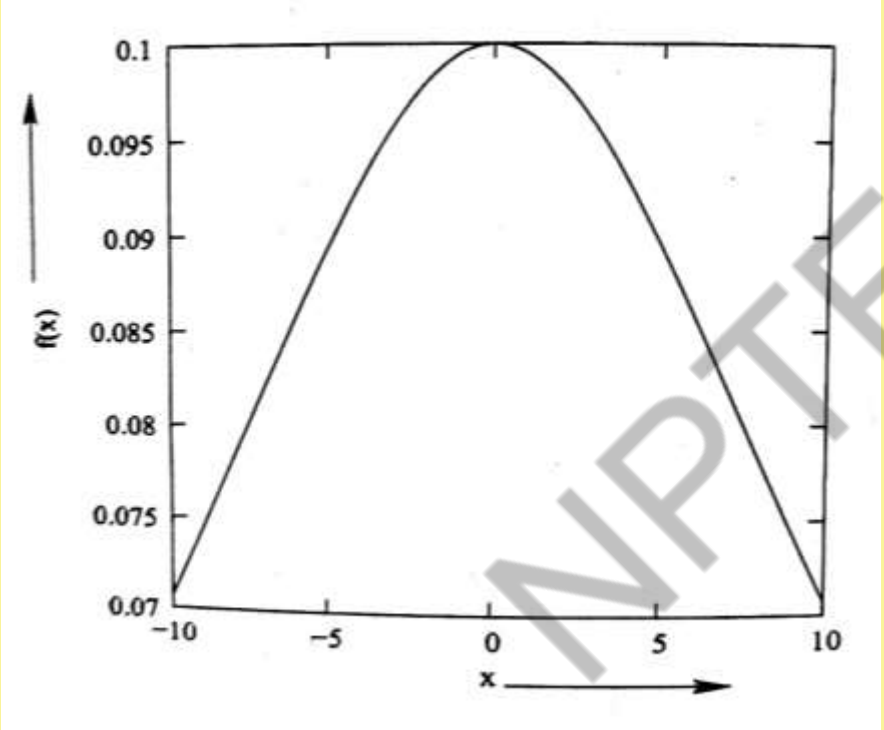


Multi-quadratic function

Multi-quadratic function

Inverse Multi-quadratic function

Radial Basis Functions

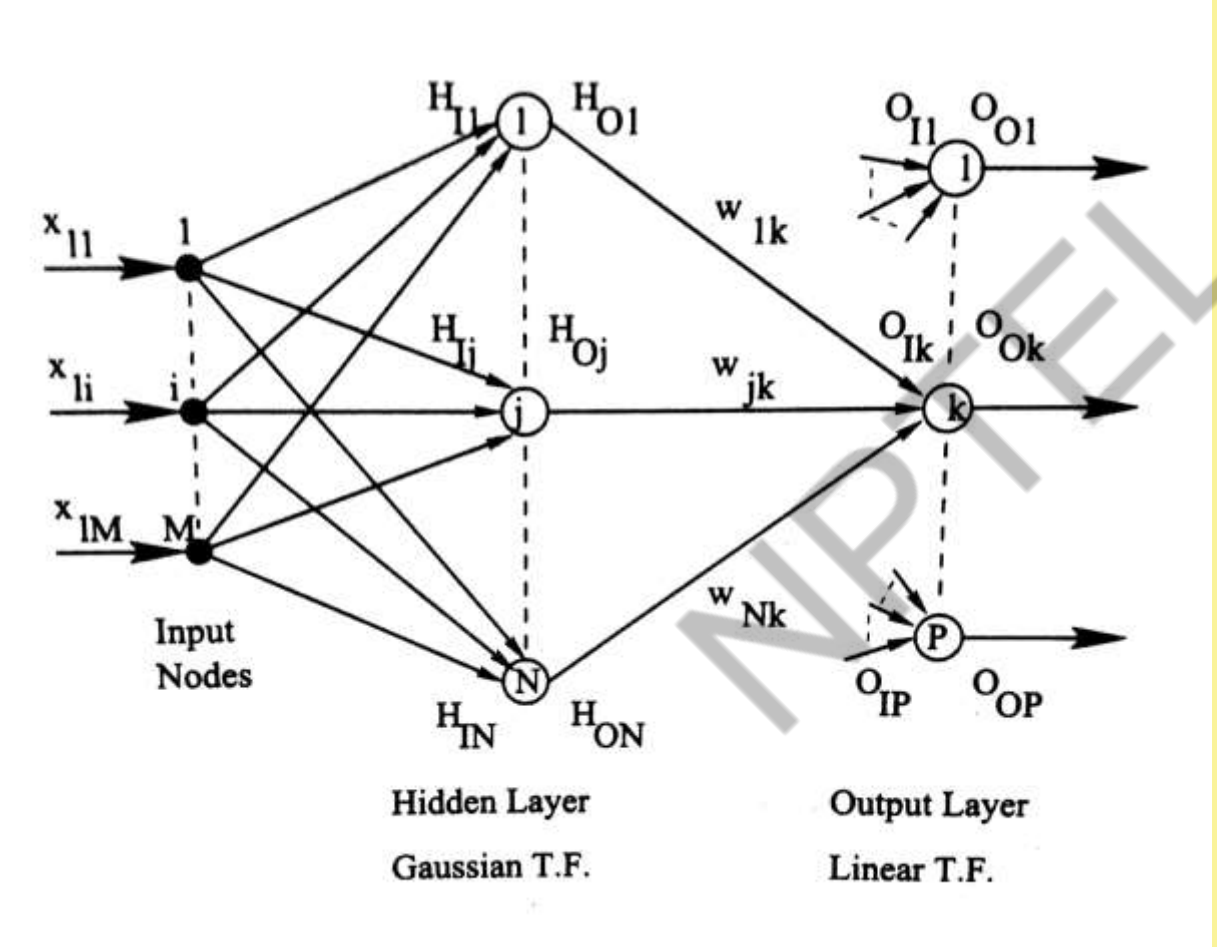


Inverse Multi-quadratic function

Multi-quadratic function

Inverse Multi-quadratic function

Radial Basis Function Neural Network (RBFNN)



NPTEL

Forward Calculations

- **Step 1: Determination of the outputs of input nodes**

  Let us consider L training scenarios.

  Input vector

$$X = \begin{bmatrix} x_1 \\ x_2 \\ . \\ . \\ x_l \\ . \\ . \\ x_L \end{bmatrix} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1i} & \cdots & x_{1M} \\ x_{21} & x_{22} & \cdots & x_{2i} & \cdots & x_{2M} \\ . & . & \cdots & . & \cdots & . \\ . & . & \cdots & . & \cdots & . \\ x_{l1} & x_{l2} & \cdots & x_{li} & \cdots & x_{lM} \\ . & . & \cdots & . & \cdots & . \\ . & . & \cdots & . & \cdots & . \\ x_{L1} & x_{L2} & \cdots & x_{Li} & \cdots & x_{LM} \end{bmatrix}$$

NPTEL

Let us assume that *l*-th training scenario (that is, x_{l1}, x_{l2}, . . . , x_{li}, ..., x_{lM}) is passed through the network. Outputs of different nodes of the input layer are nothing but the inputs of corresponding nodes.

- **Step 2:** Determination of the outputs of hidden layer
  Output of j-th hidden neuron

$$H_{Oj} = \exp\left[-\frac{\left\|x_l - \mu_j\right\|^2}{2\sigma_j^2}\right]$$

NPTEL

FREE ONLINE EDUCATION swayam

- **Step 3: Determination of the inputs of output layer**

  Input of k-th neuron lying on output layer

$$O_{Ik} = \sum_{j=1}^{N} W_{jk} H_{Oj}$$

- **Step 4: Determination of the outputs of the output layer**

$$O_{Ok} = O_{ik}$$

Error in prediction of k-th output neuron

$$E_k = \frac{1}{2} (T_{Ok} - O_{Ok})^2$$

NPTEL

FREE ONLINE EDUCATION swayam

Tuning of RBFNN Using BP Algorithm

Incremental Mode of Training

- **Step 1: Weight Updating**

$$w_{updated} = w_{previous} + \Delta w$$

Now, $$\Delta w_{jk}(t) = -\eta \frac{\partial E_k}{\partial w_{jk}}(t) + \alpha' \Delta w_{jk}(t-1)$$

Where $$\frac{\partial E_k}{\partial w_{jk}} = \frac{\partial E_k}{\partial O_{Ok}} \cdot \frac{\partial O_{Ok}}{\partial O_{Ik}} \cdot \frac{\partial O_{Ik}}{\partial w_{jk}}$$

NPTEL

Here $\frac{\partial E_k}{\partial O_{Ok}} = -\left(T_{Ok} - O_{Ok}\right)$

$$\frac{\partial O_{Ok}}{\partial O_{Ik}} = 1$$

$$\frac{\partial O_{Ik}}{\partial w_{jk}} = H_{Oj}$$

NPTEL

FREE ONLINE EDUCATION swayam शिक्षित भारत, उन्नत भारत

- **Step 2: Mean Updating**

$$\mu_{j,updated} = \mu_{j,previous} + \Delta\mu_j$$

$$\text{Now, } \Delta\mu_j(t) = -\eta\left\{\frac{\partial E}{\partial \mu_j}(t)\right\}_{av} + \alpha'\Delta\mu_j(t-1)$$

$$\text{Where, } \left\{\frac{\partial E}{\partial \mu_j}\right\}_{av} = \frac{1}{P}\sum_{k=1}^{P}\frac{\partial E_k}{\partial \mu_j}$$

$$\text{Now, } \frac{\partial E_k}{\partial \mu_j} = \frac{\partial E_k}{\partial O_{Ok}} \cdot \frac{\partial O_{Ok}}{\partial O_{Ik}} \cdot \frac{\partial O_{Ik}}{\partial H_{Oj}} \cdot \frac{\partial H_{Oj}}{\partial \mu_j}$$

NPTEL

Now, $\frac{\partial E_k}{\partial O_{Ok}} \cdot \frac{\partial O_{Ok}}{\partial O_{Ik}} = -\left(T_{Ok} - O_{Ok}\right)$

$$\frac{\partial O_{Ik}}{\partial H_{Oj}} = w_{jk}$$

$$\frac{\partial H_{Oj}}{\partial \mu_j} = H_{Oj}\left\{\frac{\left(x_{l1} + x_{l2} + \ldots\ldots + x_{li} + \ldots\ldots + x_{lM}\right) - M\mu_j}{\sigma_j^2}\right\}$$

NPTEL

- **Step 3: Standard Deviation Updating**

$$\sigma_{j,updated} = \sigma_{j,previous} + \Delta\sigma_j$$

Now, $$\Delta\sigma_j(t) = -\eta\left\{\frac{\partial E}{\partial \sigma_j}(t)\right\}_{av} + \alpha'\Delta\sigma_j(t-1)$$

where $$\left\{\frac{\partial E}{\partial \sigma_j}\right\}_{av} = \frac{1}{P}\sum_{k=1}^{P}\frac{\partial E_k}{\partial \sigma_j}$$

NPTEL

Now, $$\frac{\partial E_k}{\partial \sigma_j} = \frac{\partial E_k}{\partial O_{Ok}} \cdot \frac{\partial O_{Ok}}{\partial O_{Ik}} \cdot \frac{\partial O_{Ik}}{\partial H_{Oj}} \cdot \frac{\partial H_{Oj}}{\partial \sigma_j}$$

$$\frac{\partial E_k}{\partial O_{Ok}} \cdot \frac{\partial O_{Ok}}{\partial O_{Ik}} = -\left(T_{Ok} - O_{Ok}\right)$$

$$\frac{\partial O_{Ik}}{\partial H_{Oj}} = w_{jk}$$

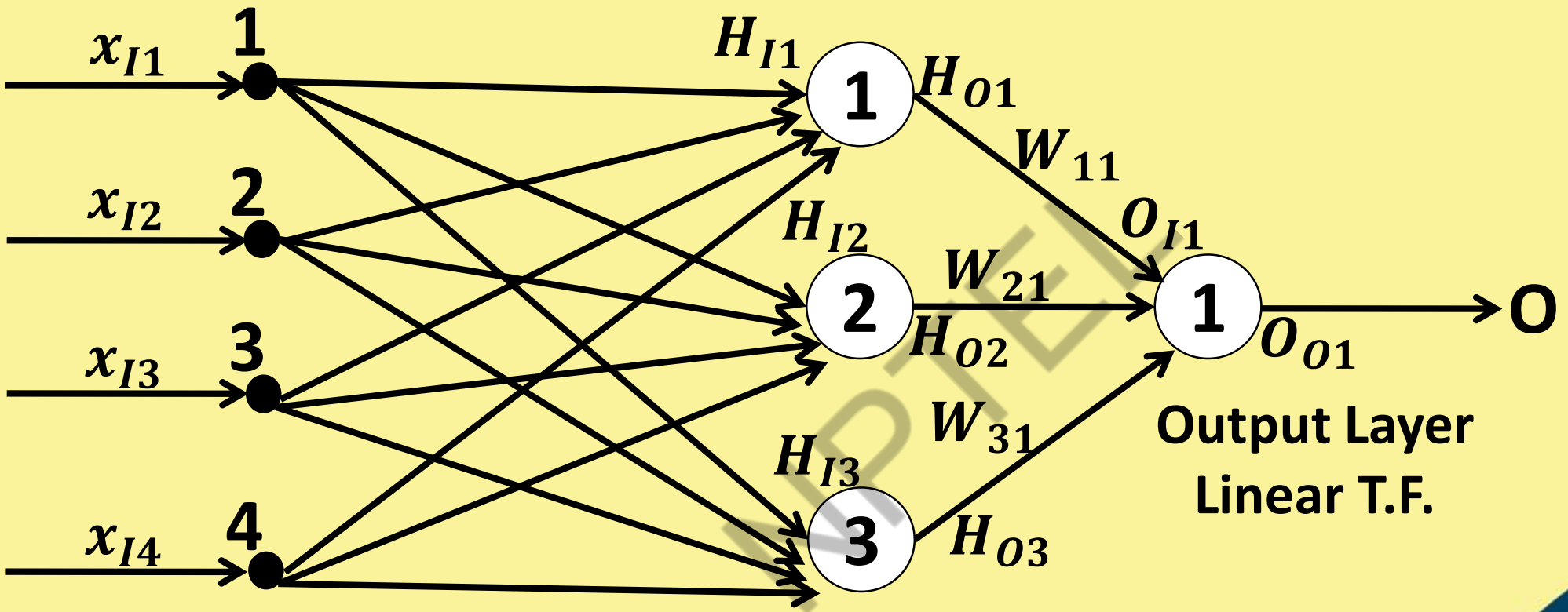
$$\frac{\partial H_{Oj}}{\partial \sigma_j} = H_{Oj}\left\{\frac{\left(x_{l1} - \mu_j\right)^2 + \left(x_{l2} - \mu_j\right)^2 + ... + \left(x_{lM} - \mu_j\right)^2}{\sigma_j^3}\right\}$$

NPTEL

Numerical Example

•A radial basis function neural network (RBFNN) is to be used to model input-output relationships of an engineering process having four inputs and one output, as shown in Figure.

NPTEL


x_{I1}
1
x_{I2}
2
x_{I3}
3
x_{I4}
4
Input Nodes
H_{I1}
1
H_{I2}
2
H_{I3}
3
Hidden Layer
Inverse multi-quadratic function
H_{O1}
W_{11}
O_{I1}
W_{21}
H_{O2}
W_{31}
H_{O3}
1
O_{O1}
O
Output Layer
Linear T.F.

There are three neurons on the hidden layer, which are assumed to have inverse multi-quadratic function of the form $y = f(x) = \frac{1}{\sqrt{x^2+\sigma^2}}$. Take σ_1, σ_2 and σ_3 for the first, second and third hidden neurons as 0.2, 3.0 and 4.0, respectively. Assume initial weights: $w_{11} = 0.2, w_{21} = 0.4,\ w_{31} = 0.5$. Use incremental training scheme with the help of a scenario: $x_{I1} = 1.5, x_{I2} = 2.0, x_{I3} = 1.7, x_{I4} = 2.5$, and output $O = 0.14$. Use back-propagation algorithm with a learning rate of $\eta = 0.2$. Calculate the updated values of w_{11} , σ_1. Show only one iteration.

NPTEL

Solution

Given $x_{I1} = 1.5, x_{I2} = 2.0, x_{I3} = 1.7, x_{I4} = 2.5$

- $H_{I1} = H_{I2} = H_{I3} = 1.5 + 2.0 + 1.7 + 2.5 = 7.7$
- $H_{O1} = \frac{1}{\sqrt{x^2 + {\sigma_1}^2}} = \frac{1}{\sqrt{7.7^2 + (0.2)^2}} = 0.129$
- $H_{O2} = \frac{1}{\sqrt{x^2 + {\sigma_2}^2}} = \frac{1}{\sqrt{7.7^2 + (3.0)^2}} = 0.121$

NPTEL

- $H_{O3} = \frac{1}{\sqrt{x^2+\sigma_3{}^2}} = \frac{1}{\sqrt{7.7^2+(4.0)^2}} = 0.115$

- $O_{I1} = H_{O1} \times w_{11} + H_{O2} \times w_{21} + H_{O3} \times w_{31}$

$= 0.129 \times 0.2 + 0.121 \times 0.4 + 0.115 \times 0.5 = 0.1317$

$\therefore\ O_{O1} = 0.1317$

NPTEL

Now, w_{11} (updated) $= w_{11}$ (previous) + Δw_{11}

•$\Delta w_{11} = -\eta \frac{\partial E}{\partial w_{11}}$

Now, $\frac{\partial E}{\partial w_{11}} = \frac{\partial E}{\partial O_{O1}} \times \frac{\partial O_{O1}}{\partial O_{I1}} \times \frac{\partial O_{I1}}{\partial w_{11}}$

$= -(T_{O1} - O_{O1}) \times 1 \times H_{O1}$

$= -(0.14 - 0.1317) \times 1 \times 0.129 = -0.00107$

NPTEL

$\therefore\ \Delta w_{11} = -0.2 \times (-0.00107) = 0.000214$

w_{11} (updated) $= 0.2 + 0.000214 = 0.200214$

Similarly, w_{21} (updated) and w_{31} (updated) can be determined.

NPTEL

FREE ONLINE EDUCATION swayam

σ_1 (updated) $=\sigma_1$ (previous) + $\Delta\sigma_1$

- $\Delta\sigma_1 = -\eta\frac{\partial E}{\partial \sigma_1}$

Now, $\frac{\partial E}{\partial \sigma_1} = = \frac{\partial E}{\partial O_{O1}} \times \frac{\partial O_{O1}}{\partial O_{I1}} \times \frac{\partial O_{I1}}{\partial H_{O1}} \times \frac{\partial H_{O1}}{\partial \sigma_1}$

$$\frac{\partial H_{O1}}{\partial \sigma_1} = \frac{\partial\left(\frac{1}{\sqrt{x^2+{\sigma_1}^2}}\right)}{\partial \sigma_1} = -\frac{1}{2}\left(x^2 + {\sigma_1}^2\right)^{-\frac{3}{2}} \times 2\sigma_1$$

$$= -\left(x^2 + {\sigma_1}^2\right)^{-\frac{3}{2}} \times \sigma_1$$

NPTEL

Now, $\frac{\partial E}{\partial \sigma_1} = -(T_{O1} - O_{O1}) \times 1 \times w_{11} \times \left\{-\left({H_{I1}}^2 + {\sigma_1}^2\right)^{-\frac{3}{2}} \times \sigma_1\right\}$

$= -(0.14 - 0.1317) \times 1 \times 0.2 \times \left\{-\left(7.7^2 + 0.2^2\right)^{-\frac{3}{2}} \times 0.2\right\}$

$= 7.26 \times 10^{-7}$

$\therefore \ \Delta\sigma_1 = -\eta\frac{\partial E}{\partial \sigma_1} = -0.2 \times 7.26 \times 10^{-7} = -1.45 \times 10^{-7}$

NPTEL

$\therefore \sigma_1$ (updated) $= \sigma_1$ (previous) + $\Delta\sigma_1$

$= 0.2 - 1.45 \times 10^{-7}$

$= 0.199999$

Similarly, the updated values of σ_2 and σ_3 can be determined.

NPTEL



FREE ONLINE EDUCATION
swayam
शिक्षित भारत, उन्नत भारत

NPTEL ONLINE CERTIFICATION COURSES

Thank you!

NPTEL