NPTEL ONLINE CERTIFICATION COURSES

Course Name: Deep Learning

Faculty Name: Prof. P. K. Biswas
Department : E & ECE, IIT Kharagpur

Topic

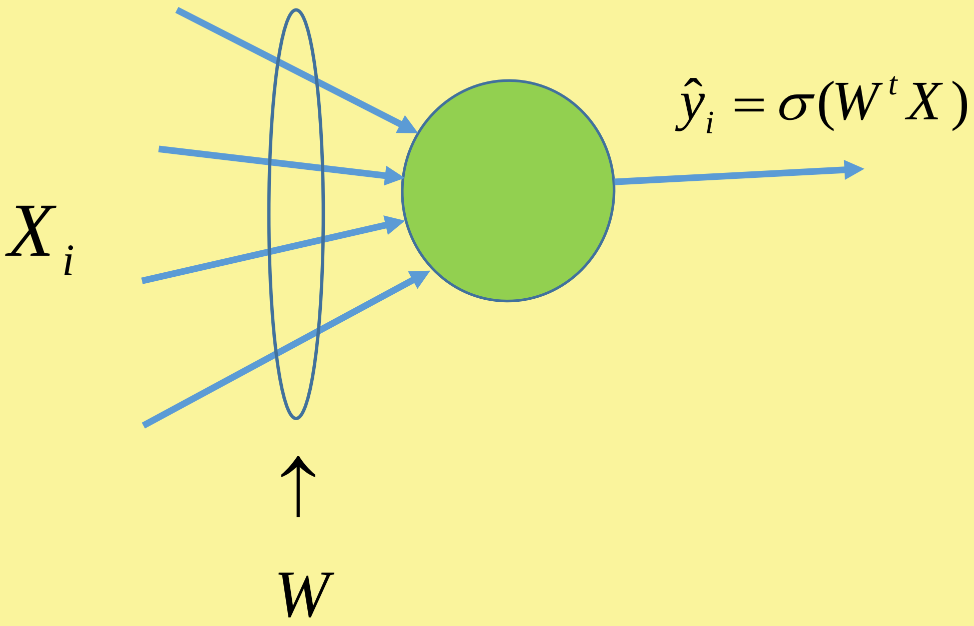
Lecture 23: Back Propagation Learning

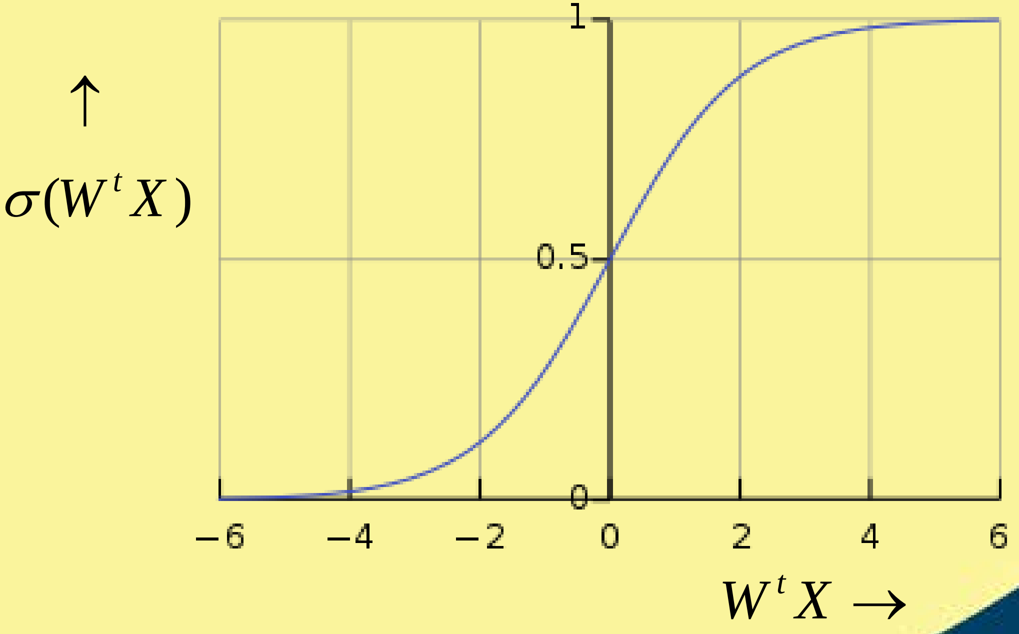
CONCEPTS COVERED

Concepts Covered:

- Learning in Single Layer Perceptron
- Back Propagation Learning in MLP

Single Layer Network- Single Output with nonlinearity





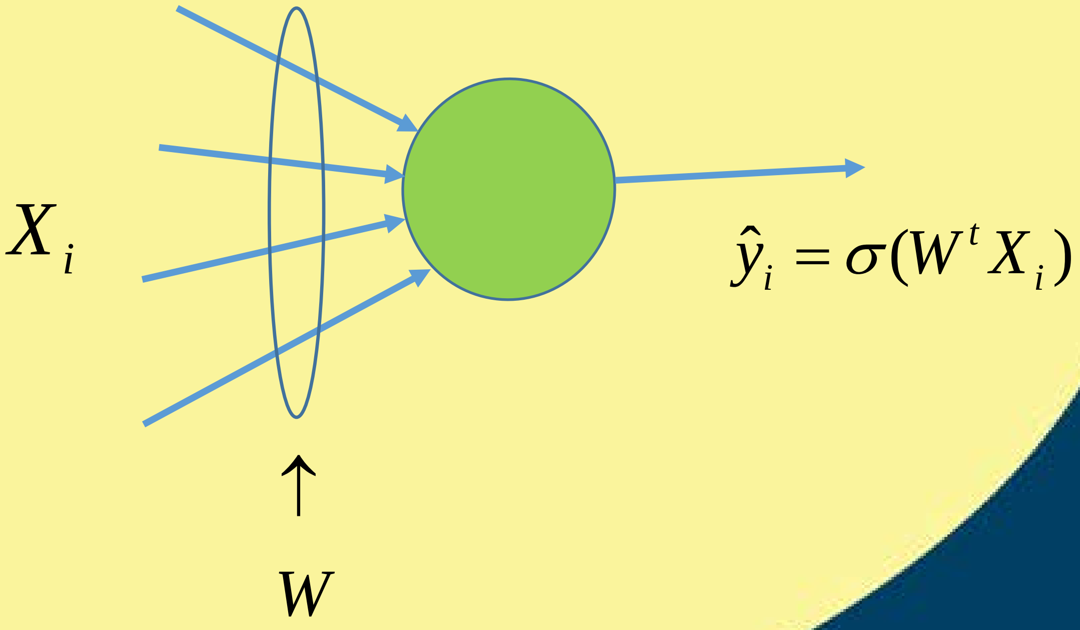
Single Layer Network- Single Output with nonlinearity

$$E = \frac{1}{2}(\hat{y}_i - y_i)^2 = \frac{1}{2}(\sigma(W^t X_i) - y_i)^2$$

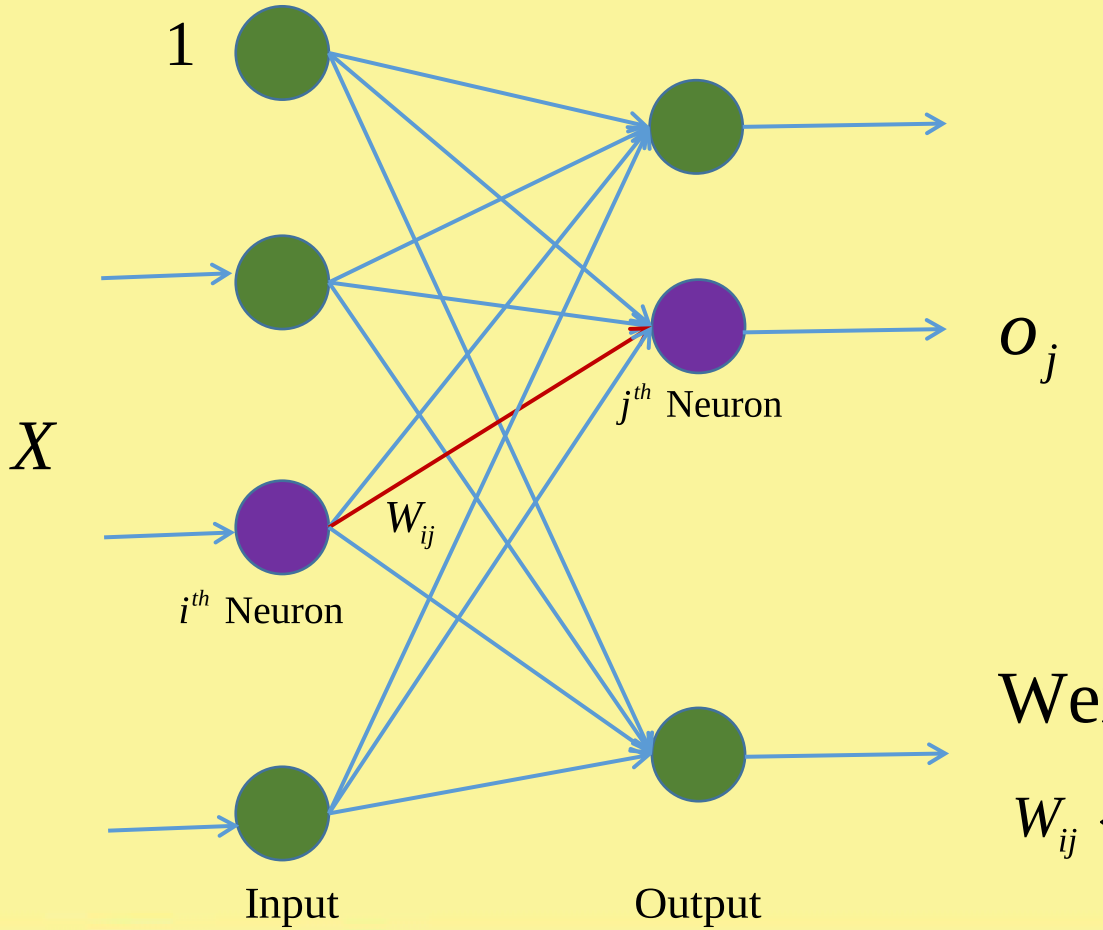
$$\nabla_W E = \hat{y}_i (1 - \hat{y}_i)(\hat{y}_i - y_i) X_i$$

Weight updation rule $\Rightarrow$

$$W \leftarrow W - \eta \hat{y}_i (1 - \hat{y}_i)(\hat{y}_i - y_i) X_i$$



Back Propagation Learning:- Single Layer Multiple Output



$$o_j = \frac{1}{1+e^{-\theta_j}} \qquad \theta_j = \sum_{i=1}^{D} W_{ij} x_i$$

$$E = \frac{1}{2}\sum_{j=1}^{M}\left(o_j - t_j\right)^2$$

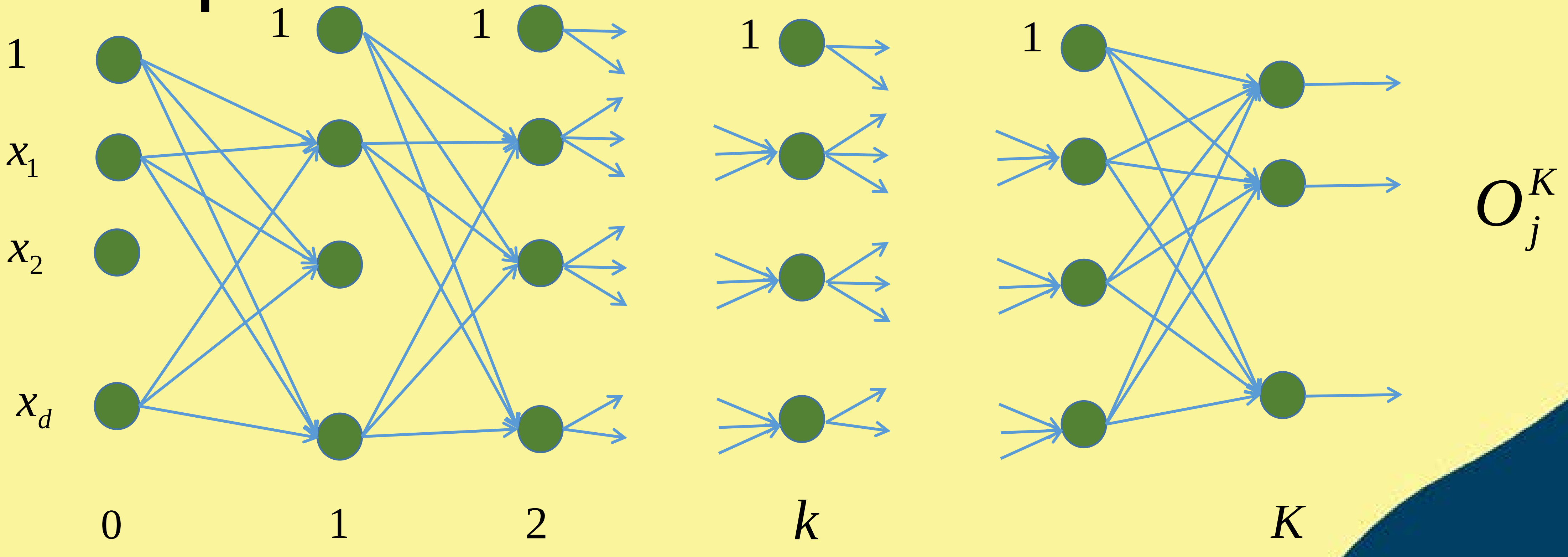
$$\frac{\partial E}{\partial W_{ij}} = \frac{\partial E}{\partial o_j} \cdot \frac{\partial o_j}{\partial \theta_j} \cdot \frac{\partial \theta_j}{\partial W_{ij}}$$

$$= (o_j - t_j) o_j (1 - o_j) x_i$$

Weight updation rule $\Rightarrow$

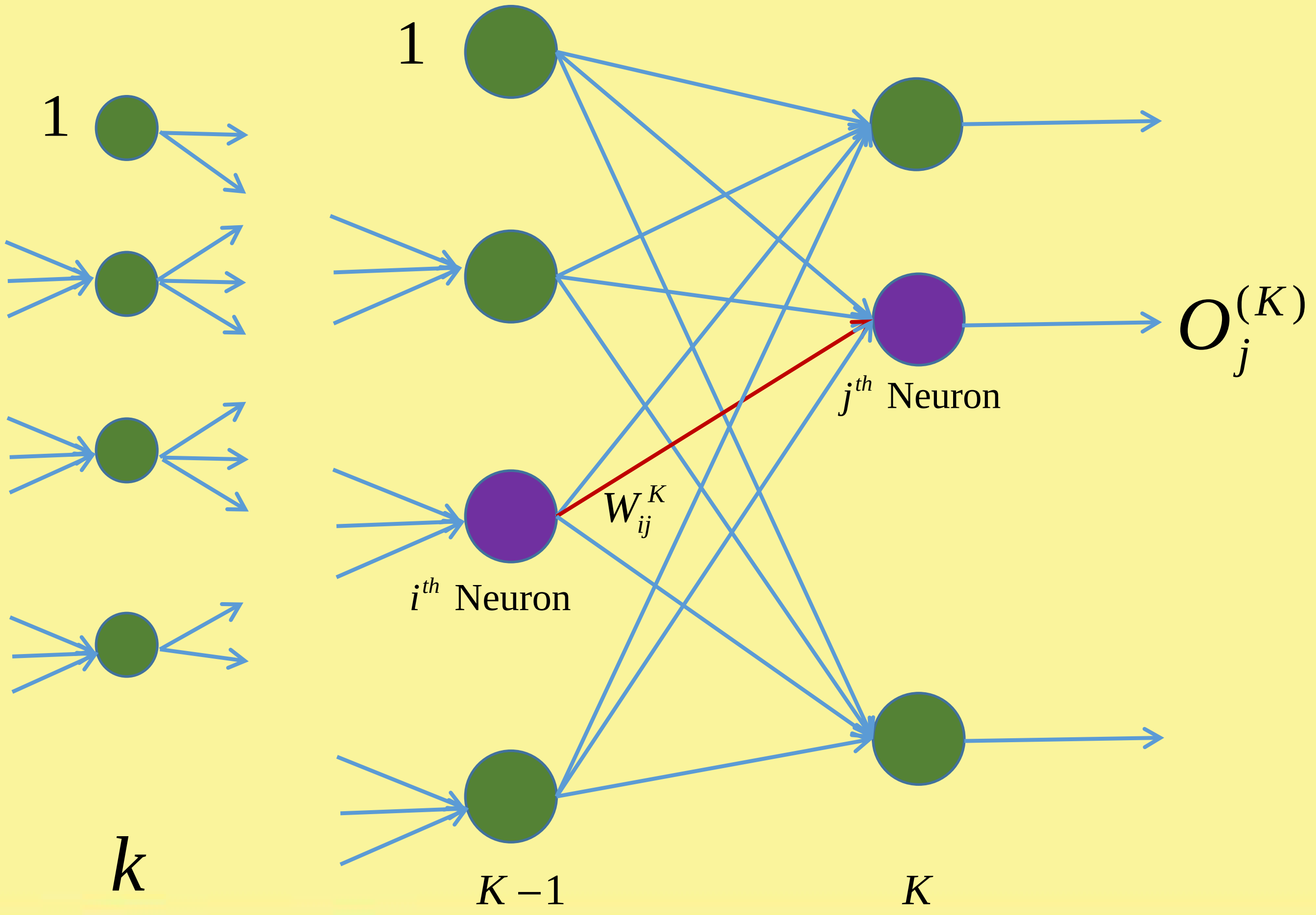
$$W_{ij} \leftarrow W_{ij} - \eta (o_j - t_j) o_j (1 - o_j) x_i$$

Multilayer Perceptron



$M_k \rightarrow$ No. of nodes in k^{th} layer

Back Propagation Learning:- Output Layer



$$O_j^K = \frac{1}{1+e^{-\theta_j^K}} \qquad \theta_j^K = \sum_{i=1}^{M_{K-1}} W_{ij}^K x_i^{K-1}$$

$$E = \frac{1}{2}\sum_{j=1}^{M_K}\left(O_j^K - t_j\right)^2$$

Back Propagation Learning:- Output Layer

Find W_{ij}^{K} that minimizes $E = \frac{1}{2}\sum_{j=1}^{M_K}\left(O_j^K - t_j\right)^2$

Gradient Descent $\frac{\partial E}{\partial W_{ij}^{K}}$

Back Propagation Learning:- Output Layer

$$\frac{\partial E}{\partial W_{ij}^K} = \frac{\partial E}{\partial O_j^K} \cdot \frac{\partial O_j^K}{\partial \theta_j^K} \cdot \frac{\partial \theta_j^K}{\partial W_{ij}^K}$$

$$= (O_j^K - t_j) O_j^K (1 - O_j^K) O_i^{K-1}$$

Let $\delta_j^K = O_j^K (1 - O_j^K)(O_j^K - t_j)$

$$\Rightarrow \frac{\partial E}{\partial W_{ij}^K} = \delta_j^K O_i^{K-1}$$

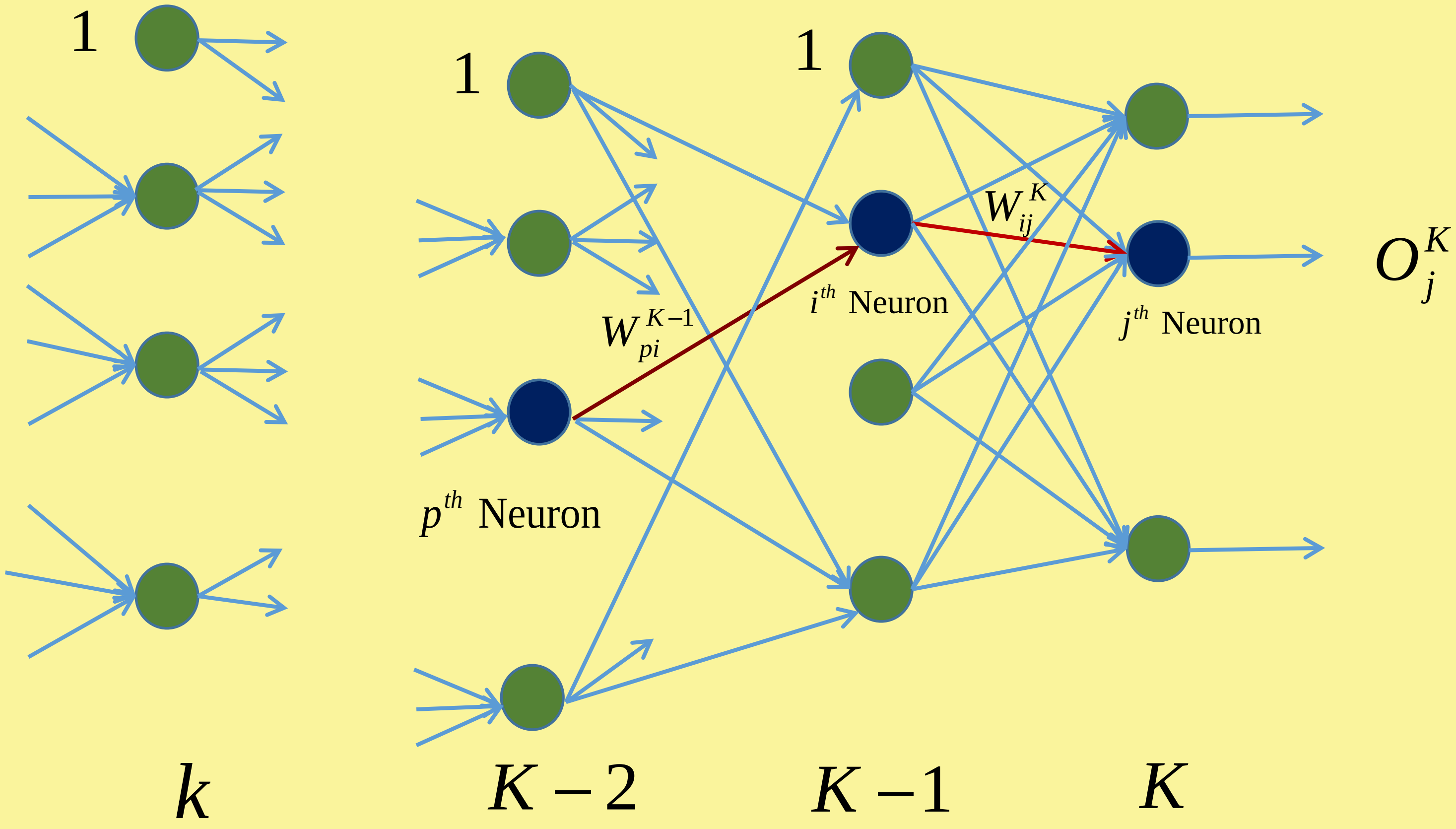
$$O_j^K = \frac{1}{1 + e^{-\theta_j^K}} \qquad \theta_j^K = \sum_{i=1}^{M_{K-1}} W_{ij}^K O_i^{K-1}$$

Weight updation rule

Output Layer

$$W_{ij}^K \leftarrow W_{ij}^K - \eta \delta_j^K O_i^{K-1}$$

Back Propagation Learning:- Hidden Layer



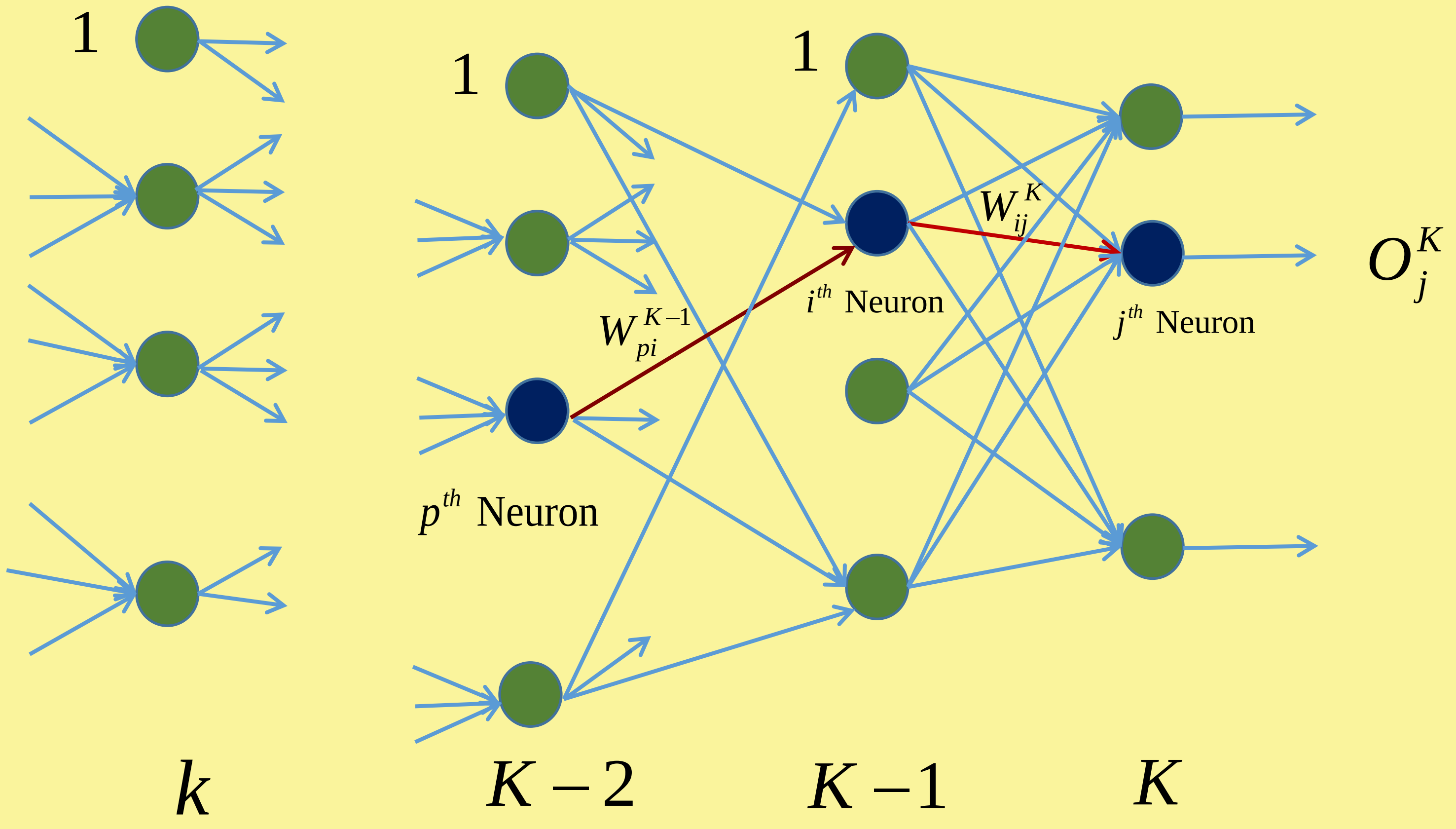
$$E = \frac{1}{2}\sum_{j=1}^{M_K}\left(O_j^K - t_j\right)^2$$

Back Propagation Learning:- Hidden Layer

Find W_{pi}^{K-1} that minimizes $E = \frac{1}{2}\sum_{j=1}^{M_K}\left(O_j^K - t_j\right)^2$

Gradient Descent $\Rightarrow \dfrac{\partial E}{\partial W_{pi}^{K-1}}$

Back Propagation Learning:- Hidden Layer



$$O_i^{K-1} = \frac{1}{1+e^{-\theta_i^{K-1}}}$$

$$\theta_i^{K-1} = \sum_{p=1}^{M_{K-2}} W_{pi}^{K-1} O_p^{K-2}$$

Back Propagation Learning:- Hidden Layer

$$\frac{\partial E}{\partial W_{pi}^{K-1}} = \frac{\partial E}{\partial O_i^{K-1}} \cdot \frac{\partial O_i^{K-1}}{\partial W_{pi}^{K-1}}$$

$$= \frac{\partial E}{\partial O_i^{K-1}} \cdot \frac{\partial O_i^{K-1}}{\partial \theta_i^{K-1}} \cdot \frac{\partial \theta_i^{K-1}}{\partial W_{pi}^{K-1}}$$

$$= \frac{\partial E}{\partial O_i^{K-1}} . O_i^{K-1}(1 - O_i^{K-1}) . O_p^{K-2}$$

$$O_i^{K-1} = \frac{1}{1 + e^{-\theta_i^{K-1}}}$$

$$\theta_i^{K-1} = \sum_{p=1}^{M_{K-2}} W_{pi}^{K-1} O_p^{K-2}$$

Back Propagation Learning:- Hidden Layer

$$\frac{\partial E}{\partial O_i^{K-1}} = \frac{\partial E}{\partial O_j^K} \cdot \frac{\partial O_j^K}{\partial \theta_j^K} \cdot \frac{\partial \theta_j^K}{\partial O_i^{K-1}}$$

$$= \sum_{j=1}^{M_K} (O_j^k - t_j) O_j^K (1 - O_j^K) W_{ij}^K$$

$$= \sum_{j=1}^{M_K} \partial_j^K W_{ij}^K$$

$$E = \frac{1}{2} \sum_{j=1}^{M_K} \left(O_j^K - t_j\right)^2$$

$$O_j^K = \frac{1}{1 + e^{-\theta_j^K}} \qquad \theta_j^K = \sum_{i=1}^{M_{K-1}} W_{ij}^K O_i^{K-1}$$

$$\delta_j^K = O_j^K (1 - O_j^K)(O_j^K - t_j)$$

Back Propagation Learning:- Hidden Layer

$$\frac{\partial E}{\partial W_{pi}^{K-1}} = \frac{\partial E}{\partial x_i^{K-1}} .O_i^{K-1}(1-O_i^{K-1}).O_p^{K-2} = O_i^{K-1}(1-O_i^{K-1}).O_p^{K-2}\sum_{j=1}^{M_K}\partial_j^K W_{ij}^K$$

Putting $\delta_i^{K-1} = O_i^{K-1}(1-O_i^{K-1})\sum_{j=1}^{M_K}\partial_j^K W_{ij}^K$

Weight updation rule

Last but Output Layer

$$W_{pi}^{K-1} \leftarrow W_{pi}^{K-1} - \eta\delta_i^{K-1}O_p^{K-2}$$

Back Propagation Learning:- any Hidden Layer

For any hidden layer weight W_{ij}^{k}

Putting $\delta_i^k = O_i^k(1 - O_i^k)\sum_{j=1}^{M_{k+1}} \partial_j^{k+1} W_{ij}^{k+1}$

Weight updation rule

$$W_{ij}^{k} \leftarrow W_{ij}^{k} - \eta \delta_j^k O_i^{k-1}$$

NPTEL ONLINE CERTIFICATION COURSES

Thank you