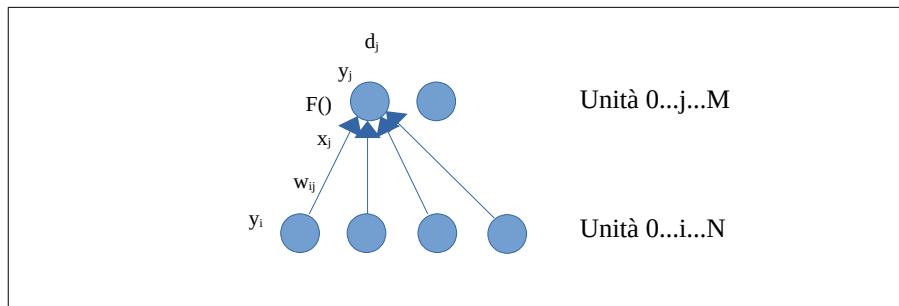


Reti neurali

Componenti della rete (a livelli)

- Unità: su livello i (precedente) e livello j (attuale)
- Attività dell'unità: y_j
- Pesi: w_{ij} del collegamento tra il nodo j ed i nodi i precedenti.



- Ingresso complessivo dell'unità j : $x_j = \sum_i w_{ij} \cdot y_i$ [1]
- Funzione di attivazione $F(x)$ e attività dell'unità; $y_j = F(x_j)$ [2]
- Derivata dell'attività all'ingresso complessivo; $\frac{dy_j}{dx_j} = F'(x_j)$ [3]
- Uscita desiderata: d_j [4]

Errore complessivo e dipendenze

- Errore complessivo delle unità di uscita: $E = \frac{1}{2} \cdot \sum_j (y_j - d_j)^2$ [5]
- Derivata dell'errore totale al variare dell'attività dell'unità di uscita j :

$$EA_j = \beta_j = \frac{\partial E}{\partial y_j} = y_j - d_j$$
 [6]
- Derivata dell'errore totale al variare dell'ingresso x_i dell'unità j , con la [6] e la [2]:

$$EI_j = \frac{\partial E}{\partial x_j} = \frac{\partial E}{\partial y_j} \cdot \frac{dy_j}{dx_j} = \beta_j \cdot F'(x_j)$$
 [7]
- Derivata dell'errore totale al variare del peso w_{ij} del collegamento tra l'unità j e l'unità i del livello precedente:

$$EW_{ij} = \frac{\partial E}{\partial w_{ij}} = \frac{\partial E}{\partial x_j} \cdot \frac{\partial x_j}{\partial w_{ij}} = \beta_j \cdot F'(x_j) \cdot y_i$$
 [8]

- Derivata dell'errore totale al variare dell'attività dell'unità i del livello precedente. L'unità i è connessa alle unità j del livello successivo, che contribuiscono tutte all'errore totale E . Con le [7] e [1]:

$$EA_i = \beta_i = \frac{\partial E}{\partial y_i} = \sum_j \frac{\partial E}{\partial x_j} \cdot \frac{\partial x_j}{\partial y_i} = \sum_j EI_j \cdot w_{ij} = \sum_j \beta_j \cdot F'(x_j) \cdot w_{ij} \quad [9]$$

Funzioni di attivazione

Funzioni di attivazione (https://it.wikipedia.org/wiki/Funzioni_di_attivazione), con derivate semplici

- Sigmoide: $F = \frac{1}{1+e^{-x}}$ [2a]

$$F' = \frac{e^{-x}}{(1+e^{-x})^2} = \frac{(1+e^{-x})-1}{(1+e^{-x})^2} = \left(1 - \frac{1}{1+e^{-x}}\right) \left(\frac{1}{1+e^{-x}}\right) = F \cdot (1-F) \quad [3a]$$

- Tangente iperbolica: $F = \tanh x = \frac{e^x - e^{-x}}{e^x + e^{-x}}$ [2b]

$$F' = (-1) \cdot \frac{(e^x - e^{-x}) \cdot (e^x - e^{-x})}{(e^x + e^{-x})^2} + \frac{1}{e^x + e^{-x}} \cdot (e^x - (-1) \cdot e^{-x}) = \frac{(e^x + e^{-x})^2 - (e^x - e^{-x})^2}{(e^x + e^{-x})^2} = \dots$$

$$\dots = 1 - \frac{(e^x - e^{-x})^2}{(e^x + e^{-x})^2} = 1 - F^2 \quad [3b]$$

- Rettificatore unità lineare ReLU: $F = \max(0, x)$ [2c]

$$F' = \begin{cases} 0 & \forall x < 0 \\ 1 & \forall x > 0 \end{cases} \quad [3c]$$

Algoritmo di back-propagation

- Costante di apprendimento δ
- Finché la rete non è addestrata, ripetere le successive operazioni per ogni configurazione.
- Calcolare, dagli ingressi del primo livello in poi, le uscite y di tutti i livelli (forward-propagation), con le formule [1] e [2].
- Calcolare le β_j delle unità di uscita, con la formula [6].
- Calcolare le β di tutti i livelli (dall'ultimo al primo), con la formula [9].
- Calcolare l'influenza $\frac{\partial E}{\partial w_{ij}}$ di ogni peso sull'errore totale, con la formula [8].
- Correggere ogni peso del valore: $\Delta w_{ij} = -\delta \cdot \frac{\partial E}{\partial w_{ij}}$ [10]