

## ĆW 2 Deep Neural Network (DNN)

### DNN

- Wejścia (input):  $u(p) = (u_1(p), u_2(p), u_3(p)) \in \{0, 1\}^3 \subset \mathbb{R}^3$ ,  $u_3(p) \equiv 1$  ( $1 \leq p \leq 4$ )  
 $u(1) = (0, 0, 1), u(2) = (1, 0, 1), u(3) = (0, 1, 1), u(4) = (1, 1, 1)$

- $f(x) = \frac{1}{1+e^{-\beta x}}$  ( $\beta > 0$  stałe)

- Sygnały z środkowej warstwy:  $x(p) = (x_1(p), x_2(p), x_3(p)) \in \mathbb{R}^3$ ,  $x_3(p) \equiv 1$  ( $1 \leq p \leq 4$ )

$$x_i(p) = f\left(\sum_{j=1}^3 w_{ij}u_j(p)\right) \quad (1 \leq i \leq 2)$$

- Wyjścia (output):  $y(p) \in \mathbb{R}$  ( $1 \leq p \leq 4$ )

$$y(p) = f\left(\sum_{i=1}^3 s_i x_i(p)\right)$$

### Zadanie.

Ustalić  $\beta > 0$  (np.  $\beta = 1.0 \sim 3.0$ ).

Implementować powyżej podany algorytm dla DNNa.

(1) Niech

$$\begin{aligned} w_{11} &= 2.0, & w_{12} &= 2.0, & w_{13} &= -3.0, \\ w_{21} &= 2.0, & w_{22} &= 2.0, & w_{23} &= -1.0, \\ s_1 &= -2.0, & s_2 &= 2.0, & s_3 &= -1.0. \end{aligned}$$

Wyświetlić  $y(p) \in \mathbb{R}$  ( $1 \leq p \leq 4$ ).

(Próbować i porównać różne parametry  $\beta$ .)

(2) Niech

$$\begin{aligned} w_{11} &= 0.0, & w_{12} &= 1.0, & w_{13} &= 2.0, \\ w_{21} &= 0.0, & w_{22} &= 1.0, & w_{23} &= 2.0, \\ s_1 &= 0.0, & s_2 &= 1.0, & s_3 &= 2.0. \end{aligned}$$

Wyświetlić  $y(p) \in \mathbb{R}$  ( $1 \leq p \leq 4$ ).

(Próbować i porównać różne parametry  $\beta$ .)

*Notacja.*

$$u_j(p) \rightsquigarrow u[p][j], \quad x_i(p) \rightsquigarrow x[p][i], \quad y(p) \rightsquigarrow y[p]$$

$$w_{ij} \rightsquigarrow w[i][j], \quad s_i \rightsquigarrow s[i]$$

$$f(x) \rightsquigarrow f, \quad \beta \rightsquigarrow beta$$

$$e^{-\beta x} \rightsquigarrow \text{math.exp}((-1) * beta * x) \quad (\text{Python})$$