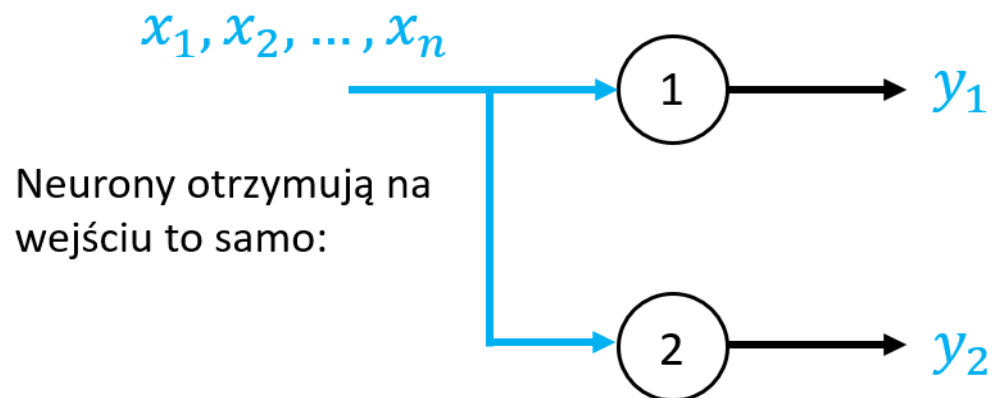


# Podstawy sztucznej inteligencji AI

Lab 6 – wskazówki do zadania  
z siecią neuronową  
rozpoznającą 1,4 i 7

# Architektura sieci

Sieć złożona jest z **dwóch neuronów sigmoidalnych\*** z **unipolarną funkcją aktywacji** w jednej warstwie:

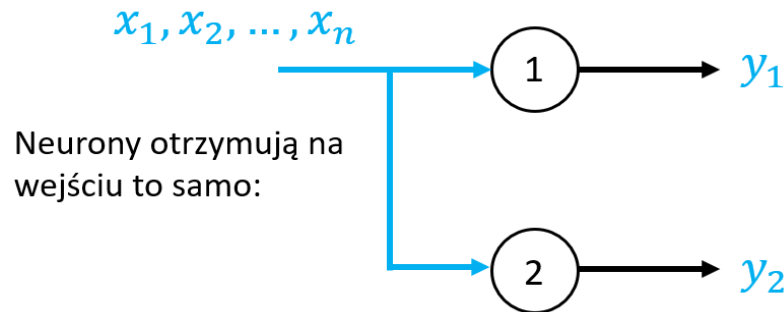


Każdy z dwóch neuronów ma **swój zestaw 13-tu wag**. A zatem sieć ma **26 wag**.

\* Szczegóły na temat neuronu sigmoidalnego w pliku: [PAI\\_lab\\_zadania\\_5.pdf](#)

# Wejście i wyjście sieci

Sieć na wejściu otrzymuje kolejno 3 tablice (każda 13 elementowa) reprezentujące cyfry: 1, 4 i 7.



Ponieważ **neurony w sieci są dwa**, zatem **na wyjściu sieci** otrzymujemy dwie liczby  $y_1$  (odpowiedź **neuronu 1**)  $y_2$  (odpowiedź **neuronu 2**).

Z definicji **unipolarnej funkcji aktywacji** wynika, że  $y_1$  i  $y_2 \in (0,1)$ .

# Wartości oczekiwane

Neurony są dwa, a zatem **wartości oczekiwane** mają postać:

$$d = (d_1, d_2)$$

gdzie  $d_1$  to **wartość oczekiwana na neuronie 1** i  $d_2$  to **wartość oczekiwana na neuronie 2**. Mamy zatem do dyspozycji następujące wartości oczekiwane dla sieci:

$$(0, 0), (1, 0), (0, 1), (1, 1)$$

W **ciągu uczącym** mamy **3 cyfry (1,4,7)** zatem możemy im przypisać następujące wartości oczekiwane:

$$\mathbf{1} \rightarrow \mathbf{d} = (1, 1) \quad \mathbf{4} \rightarrow \mathbf{d} = (1, 0) \quad \mathbf{7} \rightarrow \mathbf{d} = (0, 0)$$

## Pętla ucząca

Wybierz pierwszy element (tzn. **1**) z ciągu uczącego (zawierającego tablice reprezentujące **1**, **4** i **7**):

- A. Wylicz odpowiedzi neuronów  $y_1$  (odpowiedź neuronu 1) i  $y_2$  (odpowiedź neuronu 2).
- B. Zmodyfikuj wagi obu neuronów.
- C. Wylicz **błąd** na neuronie 1 ( $E_1$ ) i błąd na neuronie 2 ( $E_2$ ).
- D. Oblicz **błąd** dla sieci:  $E_1 + E_2$  po podaniu **1**.

**Powtórz kroki A-D dla każdego elementu z ciągu uczącego.**

Potem wylicz sumę błędów z punktu D uzyskanych dla cyfr **1**, **4** i **7**. Jeżeli suma ta będzie mniejsza od pewnej **wartości progowej** (np. 0.01) zakończ działanie pętli. Sieć jest nauczona.