

Wstęp do teorii sztucznej inteligencji

Wykład II

Uczenie sztucznych neuronów.

- powtórzyć o klasyfikacji:

- **Sieci liniowe I nieliniowe**
- **Sieci rekurencyjne**
- **Uczenie z nauczycielem lub bez**
- **Jednowarstwowe I wielowarstwowe**

Podać przykład elektronicznej realizacji neuronu

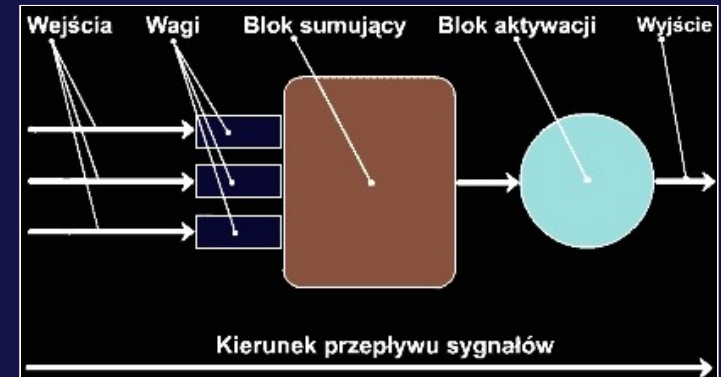
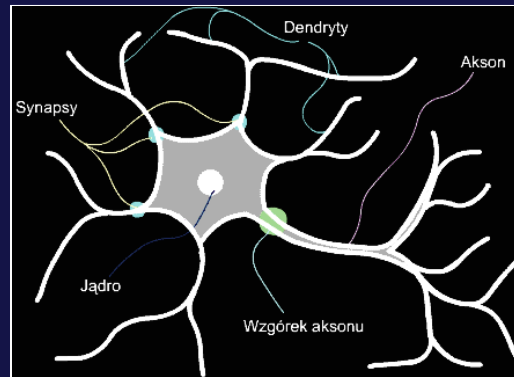
Podać przykład programowej(?) realizacji neuronu??? (Python?)

Zastosowanie sztucznych sieci neuronowych

- w rozpoznawaniu dźwięków i obrazów (mowy i pisma);
- do prognozowania, klasyfikacji i rozpoznawania stanów obiektów ekonomicznych;
- do analizy, kojarzenia i optymalizacji w podsystemach doradczych w procesach zarządzania;
- przy sterowaniu robotami;
- do kompresji i kodowania danych;
- do rozpoznawania obrazów;
- do rozpoznawania i syntezy mowy;
- prognozowanie sprzedaży, giełdy, wyścigów konnych;

- interpretacja badań biologicznych i medycznych;
- diagnostyka układów elektronicznych;
- dobór pracowników;
- selekcja celów śledztwa w kryminalistyce.

Neuron a jego sztuczny odpowiednik



Dendryt - „wejście” neuronu. Tędy trafiają do jądra sygnały. Mające być w nim później poddane obróbce. Biologiczne neurony mają ich tysiące.

Synapsa - „furtka” do neuronu (poprzedza dendryt).

Może ona zmienić moc sygnału napływającego poprzez dendryt.

Jądro - „centrum obliczeniowe” neuronu. To tutaj zachodzą procesy kluczowe dla funkcjonowania neuronu.

Wzgórek aksonu - stąd wysyłany jest sygnał wyjściowy, który wędruje dalej poprzez akson.

Akson - „wyjście” neuronu. Za jego pośrednictwem neuron powiadamia świat zewnętrzny o swojej reakcji na dane wejściowe. Neuron ma tylko jeden akson.

Wejścia to dendryty, lub ściślej - sygnały przez nie nadchodzące.

Wagi stanowią cyfrowe odpowiedniki modyfikacji dokonywanych na sygnałach przez synapsy.

Blok sumujący to odpowiednik jądra.

Blok aktywacji to wzgórek aksonu.

Wyjście odpowiada aksonowi.

Sieci neuronowe a komputery



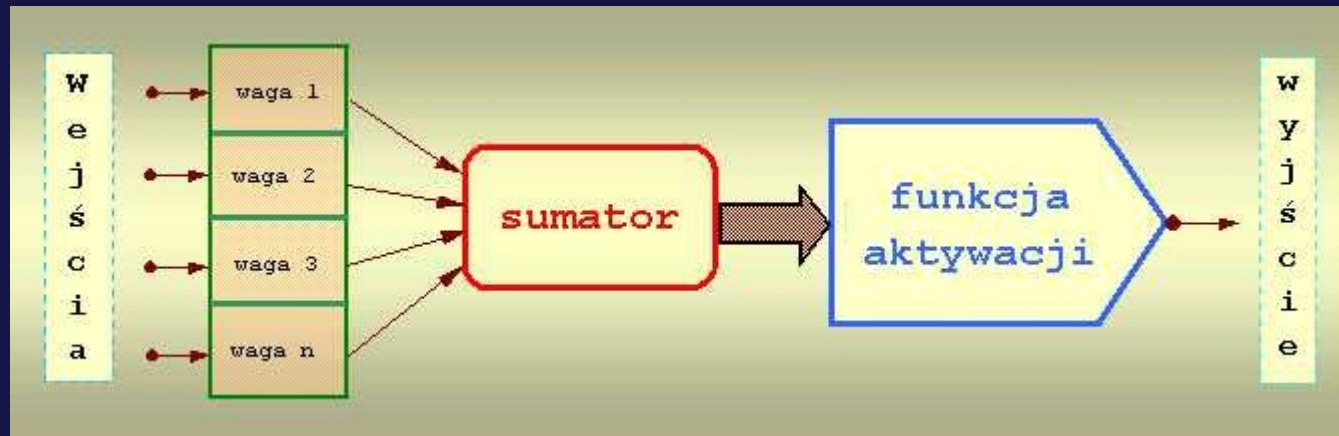
- Wiele (10^{11}) prostych elementów przetwarzających (neuronów)
- Przetwarzanie równoległe, rozłożone
- Rozłożona w całej strukturze pamięć adresowana zawartością
- Duża odporność na uszkodzenia

- Wiele (10^{11}) prostych elementów przetwarzających (neuronów)
- Przetwarzanie równoległe, rozłożone
- Rozłożona w całej strukturze pamięć adresowana zawartością
- Duża odporność na uszkodzenia

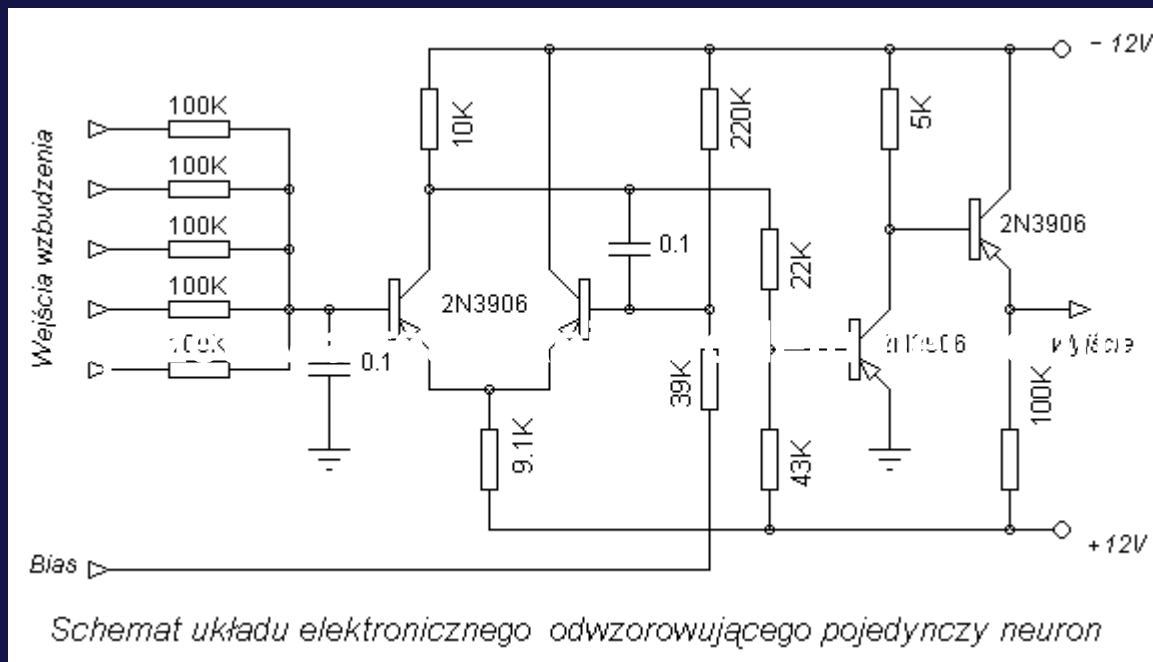
Podział sztucznych sieci neuronowych

Sztuczne sieci neuronowe można podzielić ze względu na:

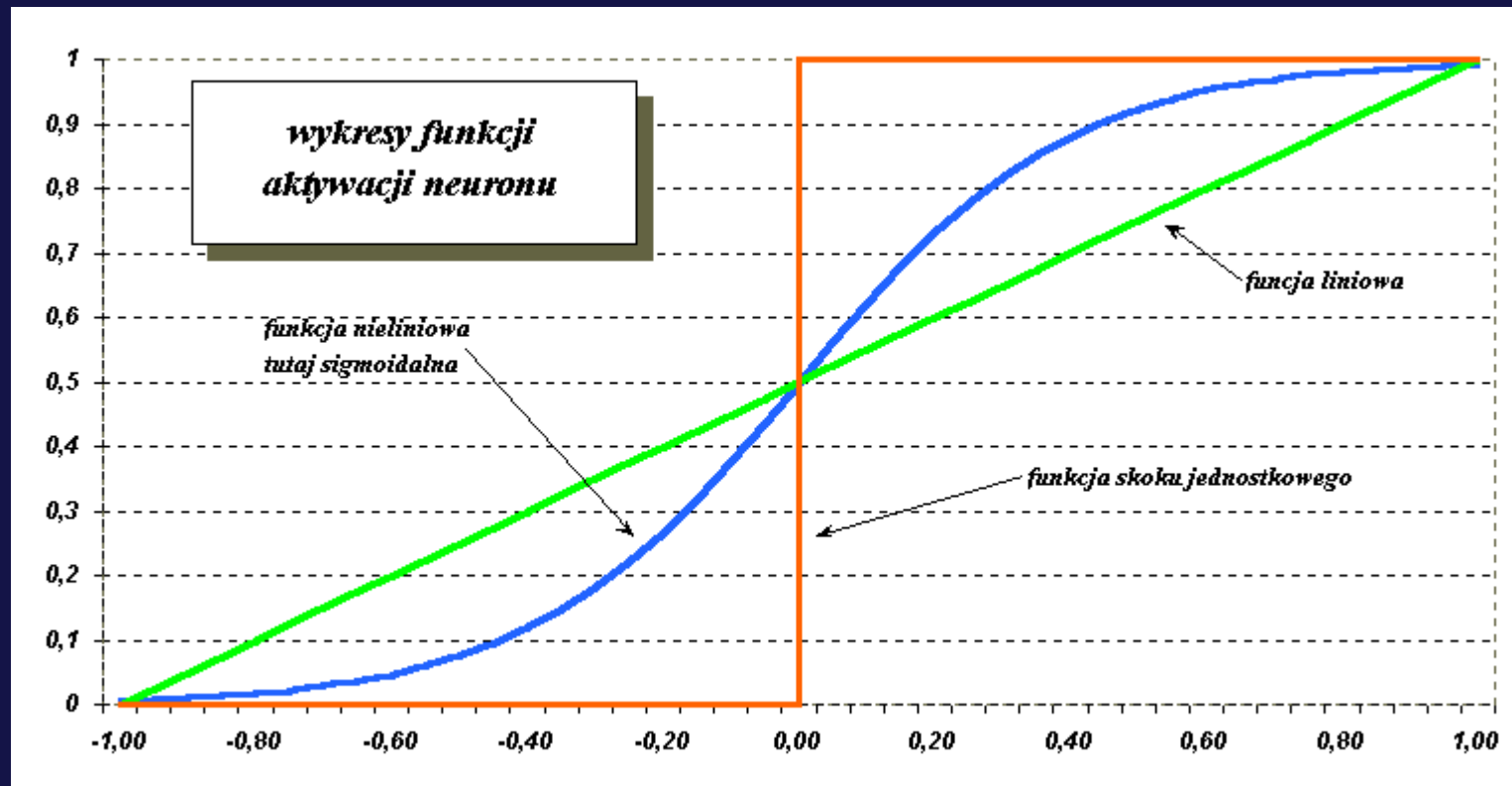
- **ilość warstw** (tutaj nie bierze się pod uwagę warstwy wejściowej)
 - wyróżniamy sieci jednowarstwowe oraz sieci wielowarstwowe;
- **kierunek przepływu sygnałów**
 - wyróżnić możemy sieci jednokierunkowe, w których sygnał płynie zawsze w jednym kierunku
 - sieci rekurencyjne, w których występują tzw. *sprzężenia zwrotne*. Za ich pośrednictwem sygnały mogą po przejściu danej warstwy wracać na jej wejście, zmieniając przy tym swoje wartości, co powtarza się wiele razy, aż do osiągnięcia pewnego ustalonego stanu.



Schemat neuronu opracowany przez McCullocha i Pittsa w 1943 roku.



Sprzętowa realizacja modelu neuronu.

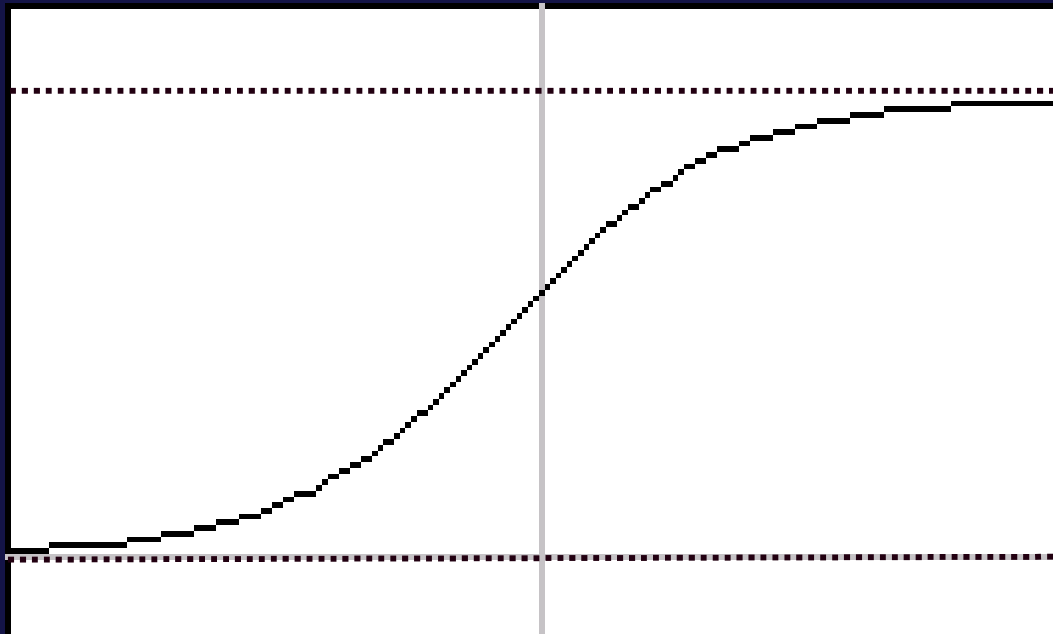


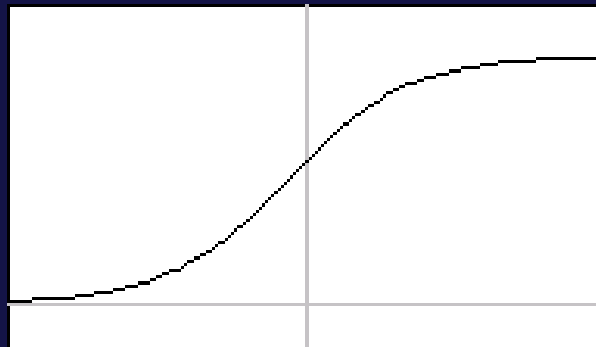
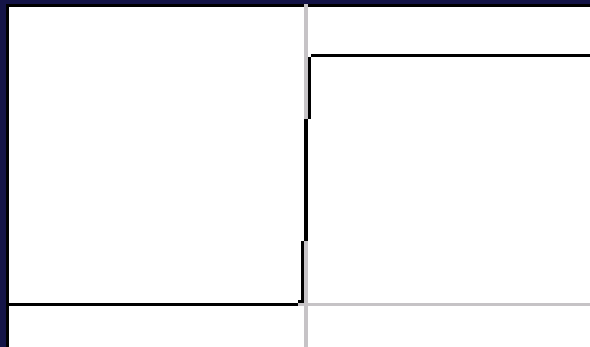
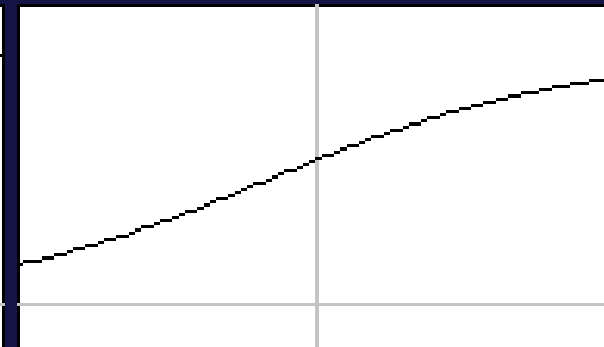
Funkcje aktywacji neuronu.

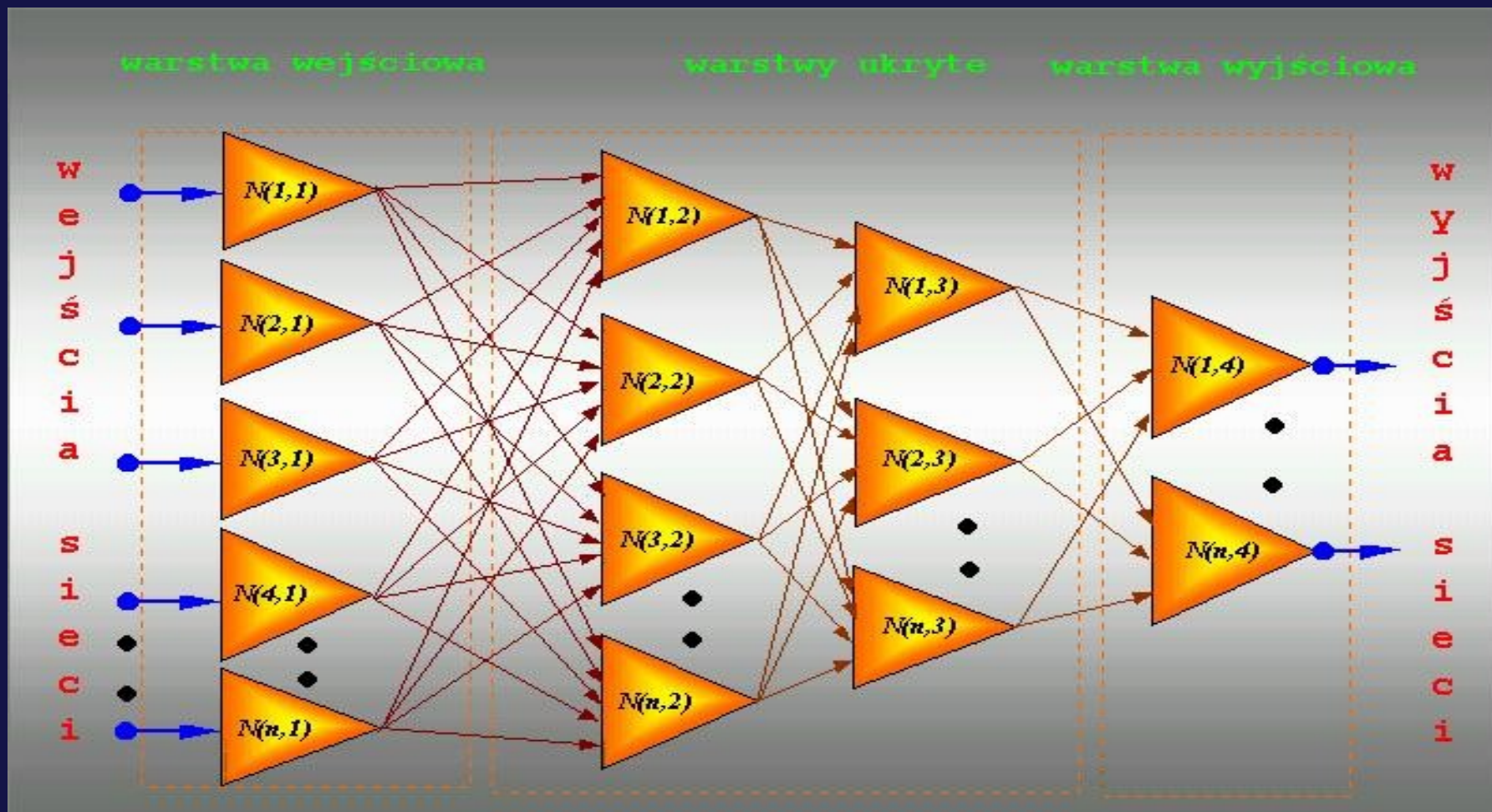
Przykład funkcji aktywacyjnej

Tzw. funkcja sigmoidalna, np.:

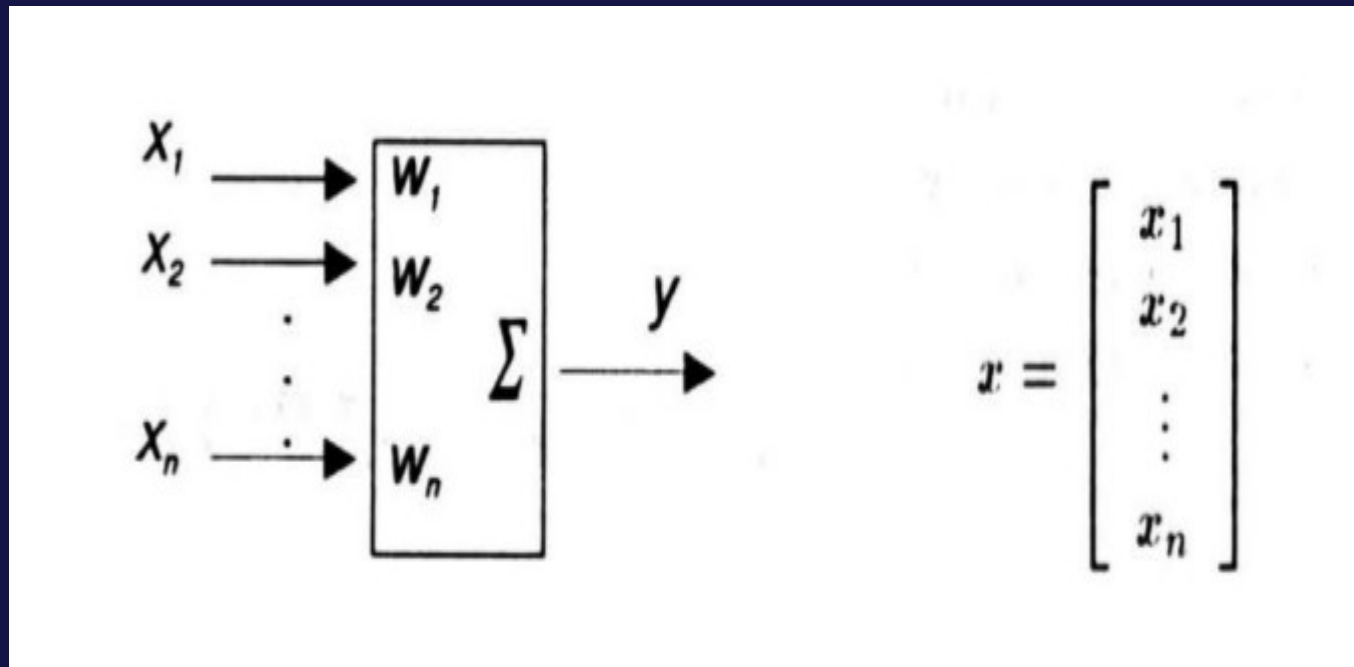
$$f(z) = \frac{1}{1 + e^{-\beta z}}$$



$\beta=1$  $\beta=100$  $\beta=0.4$ 

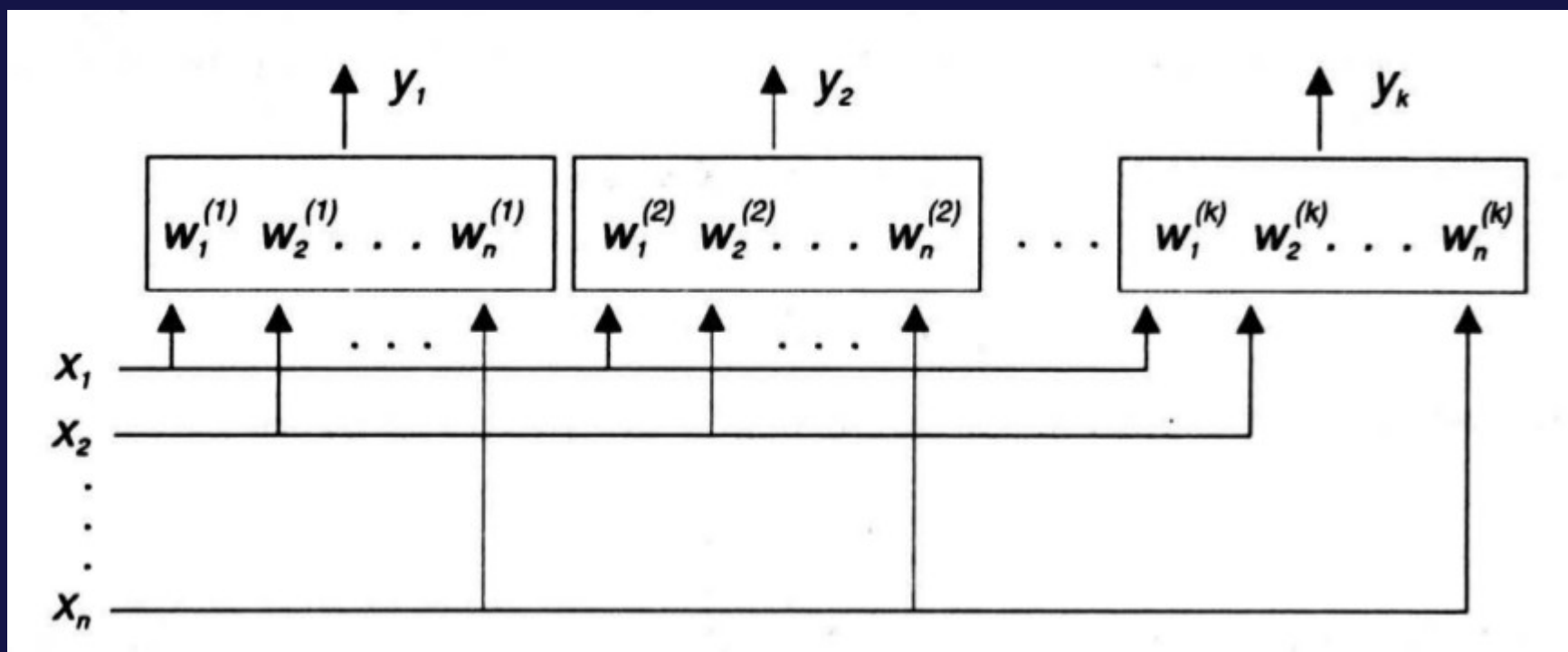


Schemat sieci wielowarstwowej.



Wagi i wejścia jako wektory dla pojedynczego neuronu.

Jeśli na wejściu x rozważanego neuronu X , t.o. wyjście neuronu y będzie miało tym większą wartość, im bardziej podobny sygnał X będzie podobny do "wzorcowego" sygnału, który neuron pamięta w postaci swojego zestawu wag W



**Wagi i wejścia dla sieci
neuronowej:
reprezentacja macierzowa**

$$W_k = \begin{bmatrix} w_1^{(1)} & w_2^{(1)} & \dots & w_n^{(1)} \\ w_1^{(2)} & w_2^{(2)} & \dots & w_n^{(2)} \\ \vdots & \vdots & & \vdots \\ w_1^{(k)} & w_2^{(k)} & \dots & w_n^{(k)} \end{bmatrix}$$

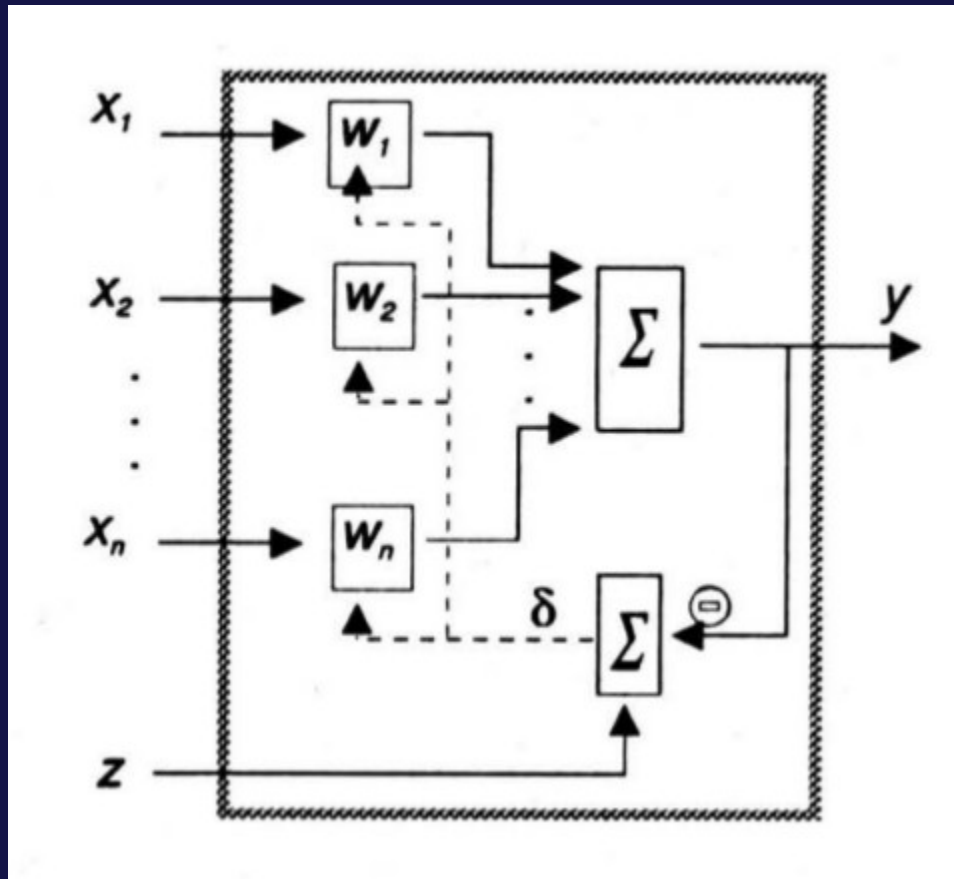
$$Y = W * X$$

**Wzór ten ujawnia jedną z podstawowych własności
rozważanej tu sieci neuronowej: Otóż
macierz W zadaje określone odwzorowanie sygnału X
w sygnał Y**

Na szczęście istnieje możliwość zastąpienia jednorazowego aktu zaprogramowania sieci iteracyjnym, wieloetapowym procesem jej uczenia.

Aby zapewnić możliwość uczenia, trzeba wprowadzony wyżej model neuronu uzupełnić o dwa dodatkowe elementy: procesor zmiany wag i detektor błędu.

Tak uzupełniony neuron nazywany bywa ADALINE
(ADaptive LINear Element)



$$\delta = y - z$$

$$W' = W + \eta \delta X$$

**Metoda ADALINE;
Uczenie z nauczycielem (supervised learning)**