

Na podstawie zagadnienia oraz algorytmu omówionego podczas ćwiczeń proszę napisać program, który będzie „odtworzał literki”.

Zbiór uczący jest postaci

$$P = \{p_1, p_2, p_3, p_4\},$$

$p_1 = 11111 \ 1-1-1-11 \ 1-1-1-11 \ 11111 \ 1-1-1-11 \ 1-1-1-11 \ 1-1-1-11$  (odpowiada literce A)

$p_2 = 11111 \ 1-1-1-1-1-1 \ 1-1-1-1-1-1 \ 1-1-1-1-1-1 \ 1-1-1-1-1-1 \ 1-1-1-1-1-1 \ 11111$  (odpowiada literce C)

Analogicznie przedstawiają się literki X oraz I.

Algorytm:

- 1) Podanie na wejście sieci wektora  $x$ . Wektor  $x$  podawany jest na wejście sieci tylko raz, aby zainicjalizować jej działanie (np. wyliczenie wag). W kolejnych krokach rolę sygnału wejściowego pełnić będą sygnały sprzężenia zwrotnego.
- 2) Obliczenie wartości sygnałów wyjściowych. Asynchroniczna aktualizacja stanów neuronów. Sygnały wyjściowe stają się nowym sygnałem wejściowym.
- 3) Porównanie poprzedniego i obecnego sygnału wejściowego (sprawdzamy stabilność sieci). Jeśli są one identyczne to kończymy działanie sieci (przechodzimy do „odtworzenia”). W przeciwnym razie powracamy do punktu 2.

## Algorytm zapisu

Jak we wszystkich sieciach neuronowych, cała wiedza również w tej sieci ukryta jest w wartościach wag. Istnieje kilka algorytmów pozwalających wyznaczyć wartości wag dla sieci Hopfielada mającej pracować jako pamięć autoasocjacyjna; przyjrzymy się tutaj najprościej z nich.

Zapisujemy do pamięci wektory wzorcowe  $s^m$ ,  $m = 1, \dots, p$  o składowych  $-1$  lub  $+1$ . Wagi wyznaczamy według wzoru:

$$w_{ij} = (1 - \delta_{ij}) \sum_{m=1}^p s_i^m s_j^m,$$

gdzie delta jest deltą Kroneckera:

$$\delta_{ij} = \begin{cases} 1 & \text{gdy } i = j \\ 0 & \text{gdy } i \neq j. \end{cases}$$

## Algorytm odczytu

1. Ustalenie stanu początkowego przez podanie sygnału na wejścia neuronów; najczęściej jest to właśnie zaszumiona informacja.
2. Ustalenie losowej kolejności w jakiej neurony będą obliczały swoje sygnały wyjściowe.
3. Obliczenie sygnałów wyjściowych dla wszystkich neuronów (w kolejności ustalonej w poprzednim punkcie).
4. Jeśli dla żadnego neuronu nie nastąpiła zmiana sygnału wyjściowego, to kończymy algorytm. W przeciwnym razie wracamy do 2.

Należy zaimplementować zaprezentowany algorytm dla zadania odtwarzania liter. Przyjmujemy postać liter analogiczną jak w ćwiczeniu 1. Do celów dydaktycznych wystarczy jak sieć zapamięta, powiedzmy 3 - 5 liter. Program ma wczytywać zniekształconą literę, na jej podstawie odtworzyć zapamiętany obraz i go zwrócić.