Sprawozdanie z przeprowadzonego badania nad   
Siecią Hopfielda, wraz z jego implementacją w SciLabie.

Autor: Wojciech Sasiela

==================================================  
Spis treści:

I) Analityczne działanie sieci Hopfielda

II) Symulacyjne działanie sieci Hopfielda

III) Zastosowanie sieci Hopfielda do rozpoznawania znaków

IV) Podsumowanie

==================================================

1. Analityczne działanie sieci   
    x1 (w1)
2. x2 (w2) S y   
     
    x3 (w3)   
      
    d (pamięć)

OK?

Wejścia Wagi Blok sumujący Blok aktywacji Wyjście

Sieć Hopfielda działa na takiej zasadzie że dla każdego neurona przypisany jest wstępny sygnał, np. mamy 3 neurony i przypisujemy im przykładowe wartości: n1=1; n2=2;n3=-4 i 4 neuron (będący pamięcią sieci, potrafiącym się uczyć) o wartości d=1.

|  |  |  |  |
| --- | --- | --- | --- |
| n1 | n2 | n3 | d |
| 1 | 2 | -4 | 1 |

|  |  |  |
| --- | --- | --- |
| w1 | w2 | w3 |
| -0,5 | 1 | 0,5 |

Każdy sygnał przechodzi przez odpowiadającej mu wagi, w naszym przykładzie mamy 3 wagi, w1=-0.5, w2=1, w3=0,5.

Stąd dla n1 i w1 mamy (1\*-0,5=-0,5) -0,5,   
dla n2=2 i w2=1, mamy (2\*1=2) 2   
i dla n3=3 i w3=0,5 mamy (-4\*0,5=-2) -2.

|  |  |  |  |
| --- | --- | --- | --- |
| U(1) | U(2) | U(3) | U |
| -0,5 | 2 | -2 | -0,5 |

Powstałe nam wartości, przechodzą do bloku sumarycznego który sumuje nam wcześniejsze wyniki: -0,5+2-2=-0,5.  
Także dla tych 3ech neuronów powstaje nam sygnał U o wartości -0,5.  
  
Teraz pytanie jest w jakiej funkcji aktywacyjnej (g(U)) ma pracować nasza sieć? Może pracować w trybie bipolarnej bądź unipolarnej.   
  
Funkcja aktywacji unipolarnej. Funkcja aktywacji bipolarnej  
g(U) = g(U) =

W naszym przypadku dla U=-0,5, to dla funkcji unipolarnej mamy 0 (g(U)=0), a dla bipolarnej mamy -1 (g(U)=1).  
Sprawdzamy teraz czy g(U) == d.  
Dla unip, 0 != 1, a dla bipol 0 != 1, czyli dla obu przypadkach nie mamy punktu stałego ( który jest wtedy gdy g(U) == d).  
Ale zajmijmy się tylko funkcją aktywacji unipolarnym.  
Wtedy wartości d=1 przypisujemy g(U)=0   
(czyli robimy d=g(U)) i mamy d=0 (Ciekawostka: Jest to pierwszy krok nauki tej pamięci).  
  
Obliczamy jeszcze raz to samo co wyżej, i znowu sprawdzamy czy d == g(U), ponieważ wynik znowu wyjdzie nam g(U)=1, to d będzie nam pasował i mamy wtedy zbieżność punktu stałego po drugim kroku (a to oznacza że pamięć poprawnie jest nauczona).  
  
Co by było, jakbyśmy wykorzystali jakiś obraz o macierzy 3 x 3 piksele?  
Mielibyśmy wtedy coś takiego:

|  |  |  |  |
| --- | --- | --- | --- |
| n1 | n2 | n3 | d |
| -0,5 | 1 | -0,5 | 1 |
| 1 | -0,5 | 1 | 1 |
| -0,5 | 1 | -0,5 | 1 |

|  |  |  |
| --- | --- | --- |
| w1 | w2 | w3 |
| 0 | 1 | 0 |
| 1 | 1 | 1 |
| 0 | 1 | 0 |

To w stosowanym przez nas „algorytmie”   
to liczylibyśmy dla 1 wiersza.

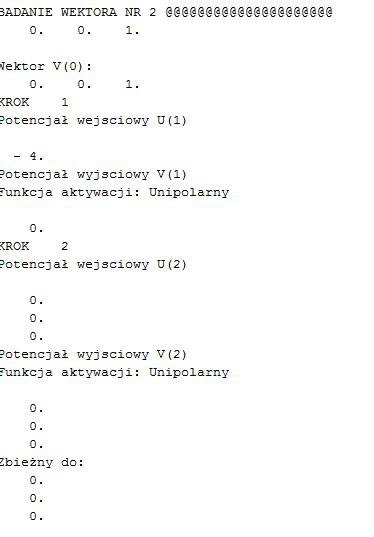
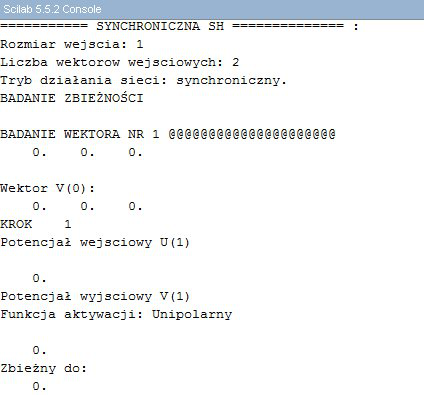
|  |  |  |  |
| --- | --- | --- | --- |
| n1 | n2 | n3 | d |
| -0,5 | 1 | -0,5 | 1 |

|  |  |  |
| --- | --- | --- |
| w1 | w2 | w3 |
| 0 | 1 | 0 |

Gdzie w bloku sumarycznym obliczylibyśmy sygnał S=  
i dla funkcji unipolarnej, mielibyśmy y=   
co dałoby/nie dałoby nam punktu stałego.  
Kroki te powtarzamy dla wszystkich wierszy (w naszym wypadku dla 3 wierszy), i tak po (szacunkowo) maksimum 30 krokach, dojdziemy do takiego etapu że wszystkie wiersze będą miały punkt stały, co pozwoliłoby nam powiedzieć że sieć w końcu nauczyła się prawidłowo rozpoznawać obraz, pomimo jego wcześniejszych zniekształceń danych.

II ) Symulacyjne działanie sieci Hopfielda

Sprawdźmy teraz czy powyższe wyniki działania algorytmu sieci Hopfielda, mają realne przełożenie w praktyce, poprzez zwertyfikowaniu wyników z poprzedniego rozdziału, w programie matematycznym SciLabie.

* Inicjalizacja:  
  
* Wywołanie metody:  
  
* Kroki wykonywane przez SciLaba  
  
* Zwrócenie tablicy ze zbieżnościami  
  