**PODSTAWY SZUCZNEJ INTELIGENCJI**

Scenariusz 4

Maria Podkalicka, IS grupa 3

**Temat sieci:** Uczenie sieci metodą Hebba.

**Cel ćwiczenia**: Poznanie działania reguły Hebba dla sieci jednowarstwowej na przykładzie grupowania liter alfabetu.

**Syntetyczny opis sieci jednowarstwowej**

**Sieci jednowarstwowe -** neurony w tej sieci ułożone są w jednej warstwie, zasilanej jedynie z węzłów wejściowych. Węzły wejściowe nie tworzą warstwy neuronowej, ponieważ nie zachodzi w nich proces obliczeniowy. Sieci tego rodzaju mogą stanowić zarówno perceptrony jednowarstwowe, jak i sieci Kohonena.

**REGUŁA HEBBA**

Jest to jedna z najpopularniejszych metod samouczenia sieci neuronowych. Polega ona na tym, że sieci pokazuje się kolejne przykłady sygnałów wejściowych, nie podając żadnych informacji o tym, co z tymi sygnałami należy zrobić**.** Sieć obserwuje otoczenie i odbiera różne sygnały, nikt nie określa jednak, jakie znaczenie mają pokazujące się obiekty i jakie są pomiędzy nimi zależności. Sieć na podstawie obserwacji występujących sygnałów stopniowo sama odkrywa, jakie jest ich znaczenie i również sama ustala zachodzące między sygnałami zależności.

Po podaniu do sieci neuronowej każdego kolejnego zestawu sygnałów wejściowych tworzy się w tej sieci pewien rozkład sygnałów wyjściowych - niektóre neurony sieci są pobudzone bardzo silnie, inne słabiej, a jeszcze inne mają sygnały wyjściowe wręcz ujemne. Interpretacja tych zachowań może być taka, że niektóre neurony „rozpoznają” podawane sygnały jako „własne” (czyli takie, które są skłonne akceptować), inne traktują je „obojętnie”, zaś jeszcze u innych neuronów wzbudzają one wręcz „awersję”. Po ustaleniu się sygnałów wyjściowych wszystkich neuronów w całej sieci - wszystkie wagi wszystkich neuronów są zmieniane, przy czym wielkość odpowiedniej zmiany wyznaczana jest na podstawie iloczynu sygnału wejściowego, wchodzącego na dane wejście (to którego wagę zmieniamy) i sygnału wyjściowego produkowanego przez neuron, w którym modyfikujemy wagi.

Dokładniejsza analiza procesu samouczenia metodą Hebba pozwala stwierdzić, że w wyniku konsekwentnego stosowania opisanego algorytmu początkowe, najczęściej przypadkowe „preferencje” neuronów ulegają systematycznemu wzmacnianiu i dokładnej polaryzacji. Jeśli jakiś neuron miał „wrodzoną skłonność” do akceptowania sygnałów pewnego rodzaju - to w miarę kolejnych pokazów nauczy się te sygnały rozpoznawać coraz dokładniej i coraz bardziej precyzyjnie. Po dłuższym czasie takiego samouczenia w sieci powstaną zatem wzorce poszczególnych typów występujących na wejściu sieci sygnałów. W wyniku tego procesu sygnały podobne do siebie będą w miarę postępu uczenia coraz skuteczniej grupowane i rozpoznawane przez pewne neurony, zaś inne typy sygnałów staną się „obiektem zainteresowania” innych neuronów W wyniku tego procesu samouczenia sieć nauczy się, ile klas podobnych do siebie sygnałów pojawia się na jej wejściach oraz sama przyporządkuje tym klasom sygnałów neurony, które nauczą się je rozróżniać, rozpoznawać i sygnalizować.

Zgodnie z regułą Hebba modyfikację wag przeprowadza się następująco:



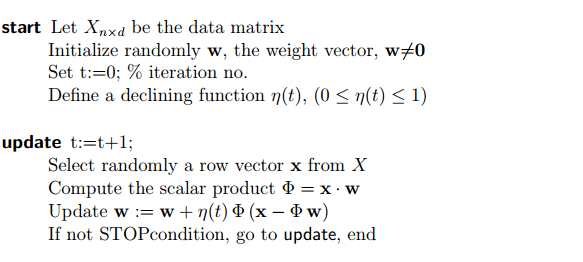
Oznaczenia:

* i-numer wagi neuronu
* t-numer iteracji w epoce
* y-sygnał wyjściowy neuronu
* x-wartość wyjściowa neuronu
* n-współczynnik uczenia(0,1)

Uczenie neuronu z zastosowaniem reguły Hebba może odbywać się również w trybie z nauczycielem.

Metoda uczenia z nauczycielem : modyfikacja algorytmu - wi(t+1) = wi(t) + n**d**xi,  
d – sygnał wzorcowy

Pseudokod obliczania pierwszej składowej głównej:



**Dane uczące i testujące**

Przygotowany przeze mnie zestaw uczący zawiera 20 dużych liter alfabetu łacińskiego utworzonych na matrycy 5x5 czyli na 25 polach.

Przykład : litera W

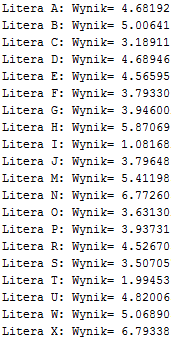
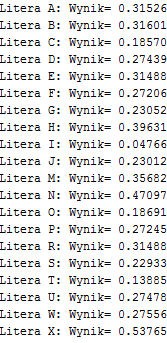
1 0 0 0 1  
1 0 0 0 1  
1 0 0 0 1  
1 0 0 0 1  
1 0 1 0 1  
1 0 1 0 1  
0 1 0 1 0  
Liniowym rozwinięciem tych matryc będą wektory wejściowe ciągu uczącego. Przykładowy wektory uczące :

**A={0,1,1,1,0,1,0,0,0,1,1,0,0,0,1,1,1,1,1,1,1,0,0,0,1,1,0,0,0,1,1,0,0,0,1}  
B={1,1,1,1,0,1,0,0,0,1,1,0,0,0,1,1,1,1,1,0,1,0,0,0,1,1,0,0,0,1,1,1,1,1,0}  
C={0,1,1,1,0,1,0,0,0,1,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,1,0,1,1,1,0}  
D={1,1,1,1,0,1,0,0,0,1,1,0,0,0,1,1,0,0,0,1,1,0,0,0,1,1,0,0,0,1,1,1,1,1,0}  
E={1,1,1,1,1,1,0,0,0,0,1,0,0,0,0,1,1,1,1,0,1,0,0,0,0,1,0,0,0,0,1,1,1,1,1}  
F={1,1,1,1,1,1,0,0,0,0,1,0,0,0,0,1,1,1,1,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0}  
G={0,1,1,1,0,1,0,0,0,1,1,0,0,0,0,1,0,1,1,1,1,0,0,0,1,1,0,0,0,1,0,1,1,1,0}  
H={1,0,0,0,1,1,0,0,0,1,1,0,0,0,1,1,1,1,1,1,1,0,0,0,1,1,0,0,0,1,1,0,0,0,1}**

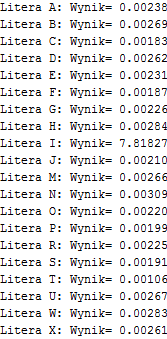
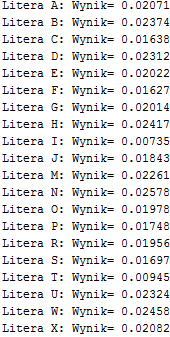
**Zestawienie otrzymanych wyników**

Testy były wykonywane dla różnych współczynników uczenia i zapominania.

Współczynnik uczenia: 0.1 Współczynnik uczenia: 0.3  
Współczynnik zapominania: 0.1 Współczynnik zapominania: 0.1

** **

Współczynnik uczenia: 0.1 Współczynnik uczenia: 0.1  
Współczynnik zapominania: 0.3 Współczynnik zapominania: 0.4

Dla lepszej analizy wyników przedstawiłam na wykresach wyniki dla liter A-F i zmieniających się współczynników uczenia i zapominania.

Dla współczynnika uczenia 0.3 wyniki są lepsze, natomiast dla współczynnika uczenia 0.1 wyniki są bardziej ustabilizowane.  
Podobnie jest ze współczynnikiem zapominania, dla 0.4 wyniki są większe ale mniej ustabilizowane niż dla współczynnika 0.3. Trzeba więc odpowiednio dobrać współczynniki uczenia i zapominania, ponieważ wyższe współczynniki powodują, że sieć lepiej się uczy, ale nie mogą być też zbyt wysokie, bo wyniki są bardziej zróżnicowane.

Wraz ze zwiększaniem wartości współczynnika uczenia zmniejsza się liczba iteracji, co ma wpływ na działanie programu i powoduje uzyskanie błędnych wyników. Aby zredukować liczbę błędnych wyników można zwiększyć liczbę iteracji.

**Wnioski**

Proces samouczenia ma wady. W porównaniu z procesem uczenia z nauczycielem samouczenie jest zwykle znacznie powolniejsze. Bez nauczyciela nie można z góry określić, który neuron wyspecjalizuje się w rozpoznawania której klasy sygnałów. Stanowi to pewną trudność przy wykorzystywaniu i interpretacji wyników pracy sieci. W dodatku nie można określić, czy sieć uczona w ten sposób nauczy się wszystkich prezentowanych jej wzorców. Dlatego sieć przeznaczona do samouczenia musi być większa niż sieć wykonująca to samo zadanie, ale trenowana w sposób klasyczny, z udziałem nauczyciela. - Szacunkowo sieć powinna mieć co najmniej trzykrotnie więcej elementów warstwy wyjściowej niż wynosi oczekiwana liczba różnych wzorów, które sieć ma rozpoznawać.

Bardzo istotną kwestią jest wybór początkowych wartości wag neuronów sieci przeznaczonej do samouczenia, w zależności od ich wartości zmienia się poprawność wyników i liczba iteracji.

Wartości te mają bardzo silny wpływ na ostateczne zachowanie sieci, ponieważ proces uczenia jedynie pogłębia i doskonali pewne tendencje istniejące w sieci od samego początku, zatem od jakości tych początkowych, „wrodzonych” właściwości sieci silnie zależy, do czego sieć dojdzie na końcu procesu uczenia.

Cały listing kodu został umieszony w repozytorium Git.