Jakub Sowiński

Temat ćwiczenia: Uczenie sieci regułą Hebba.

Cel ćwiczenia:

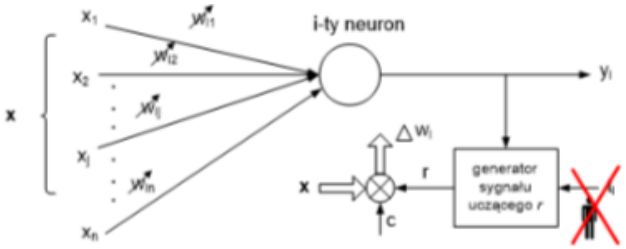
Celem ćwiczenia jest poznanie działania reguły Hebba dla sieci jednowarstwowej na przykładzie grupowania liter alfabetu.

Zadania do wykonania:

a) Wygenerowanie danych uczących i testujących, zawierających 20 dużych liter dowolnie wybranego alfabetu w postaci dwuwymiarowej tablicy np. 7x5 pikseli dla jednej litery.
b) Przygotowanie (implementacja lub wykorzystanie gotowych narzędzi) jednowarstwowej sieci oraz reguły Hebba z i bez współczynnika zapominania.
c) Uczenie sieci dla różnych współczynników uczenia i zapominania.
d) Testowanie sieci.

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.



Reguła uczenia bez nauczyciela

Sygnałem uczącym jest po prostu sygnał wyjściowy neuronu:

$$r = y_i = f(\mathbf{w}_i^T \mathbf{x})$$

Korekta wektora wag:

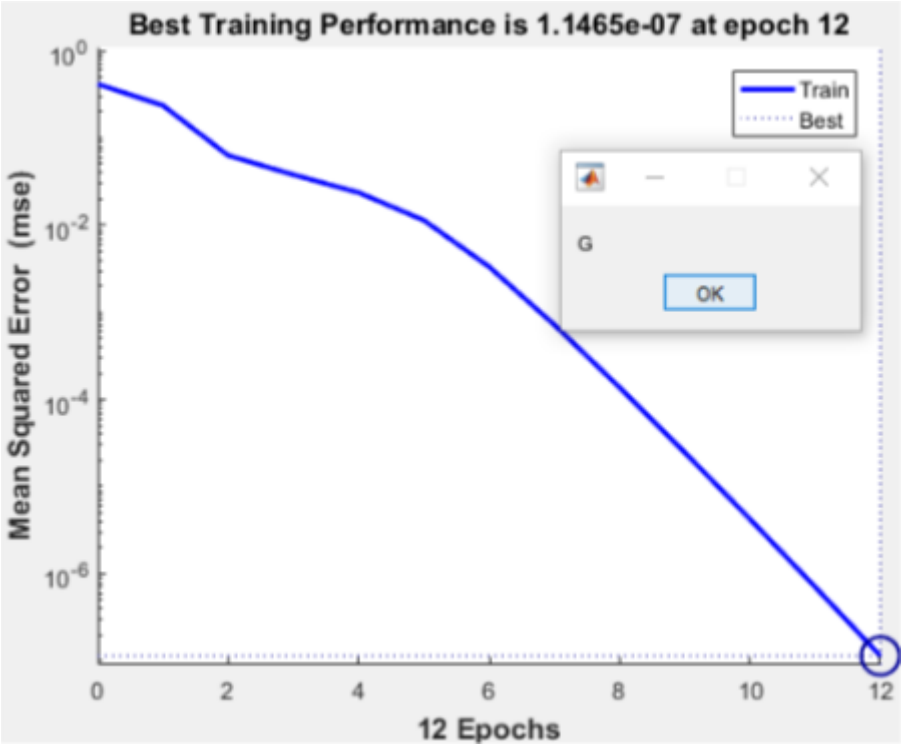
$$\Delta \mathbf{w}_i = c y_i \mathbf{x} = c f(\mathbf{w}_i^T \mathbf{x}) \mathbf{x}$$

$$\mathbf{w}_i(k+1) = \mathbf{w}_i(k) + \Delta \mathbf{w}_i = \mathbf{w}_i(k) + c y_i \mathbf{x}$$

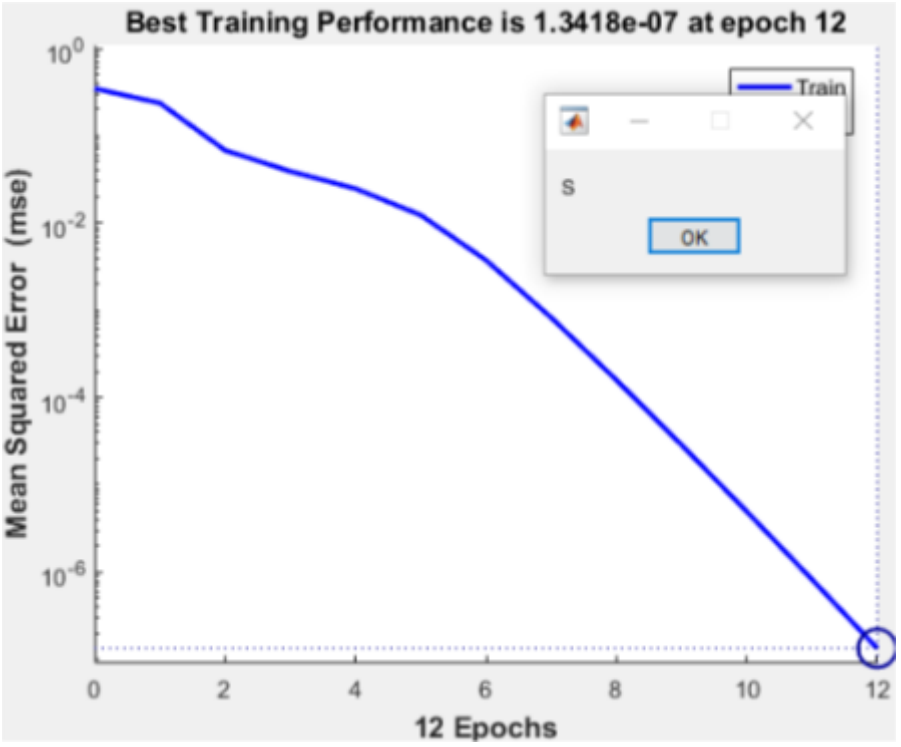
Po podaniu do sieci neuronowej każdego kolejnego zestawu sygnałów wejściowych tworzy się w tej sieci pewien rozkład sygnałów wyjściowych - niektóre neurony sieci są pobudzone bardzo silnie, inne słabiej, a jeszcze inne mają sygnały wyjściowe wręcz ujemne. Interpretacja tych zachowań może być taka, że 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. Łatwo zauważyć, że jest to właśnie realizacja postulatu Hebba - w efekcie opisanego wyżej algorytmu połączenia między źródłami silnych sygnałów i neuronami które na nie silnie reagują są wzmacniane.
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.
Proces samouczenia ma niestety wady. W porównaniu z procesem uczenia z nauczycielem samouczenie jest zwykle znacznie powolniejsze. Co więcej 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. Co więcej - 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.

1. Działanie sieci

Wprowadzona litera: G, sieć nauczyła się w 12 iteracji.



Wprowadzona litera: S, sieć nauczyła się po 12 epokach

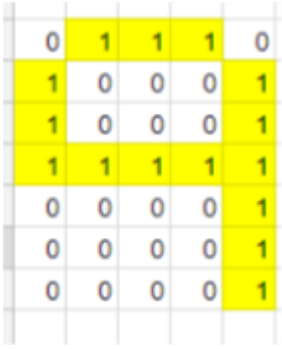


2. Grupowanie liter

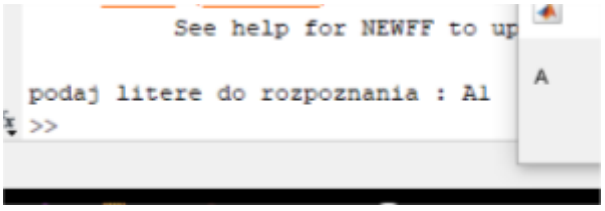
Wprowadzam zniekształcone litery A, patrząc po wynikach w zmiennej test, można określić jak bardzo jest podobna ta litera do innych wejściowych liter.

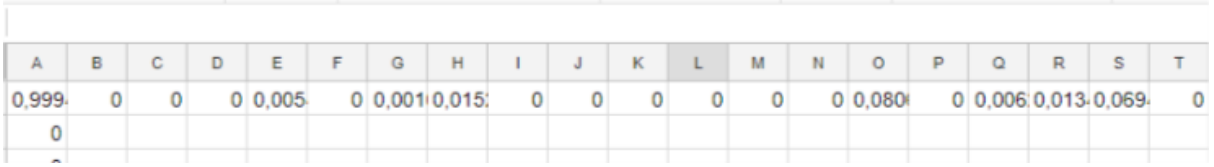
a)

Wprowadzona zniekształcona litera A, sieć wykryła że to litera A.



0	1	1	1	0
1	0	0	0	1
1	0	0	0	1
1	1	1	1	1
0	0	0	0	1
0	0	0	0	1
0	0	0	0	1



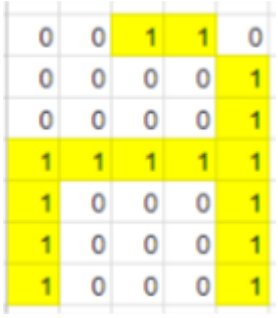


A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
0,999	0	0	0	0,005	0	0,001	0,015	0	0	0	0	0	0	0,080	0	0,006	0,013	0,069	0
0																			

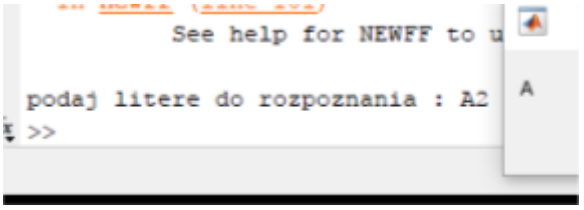
Sieć wskazała, że na 99% to litera A.

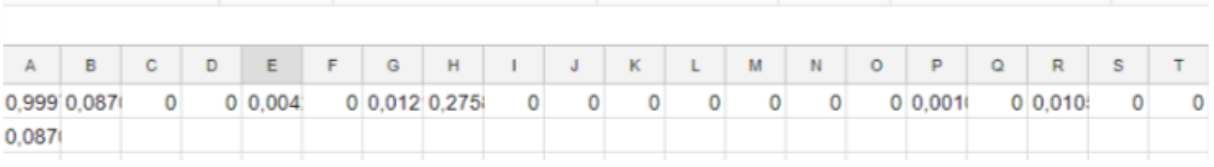
b)

Wprowadzona zniekształcona litera A, sieć wykryła że to litera A.



0	0	1	1	0
0	0	0	0	1
0	0	0	0	1
1	1	1	1	1
1	0	0	0	1
1	0	0	0	1
1	0	0	0	1



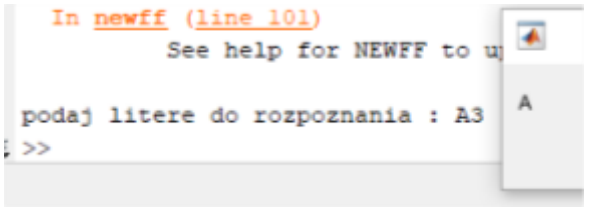


A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
0,999	0,087	0	0	0,004	0	0,012	0,275	0	0	0	0	0	0	0	0,001	0	0,010	0	0
0,087																			

Sieć wskazała, że na 99% to litera A, natomiast na 28% to litera H.

c)
Wprowadzona zniekształcona litera A, sieć wykryła że to litera A.

0	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	0	0	0	1

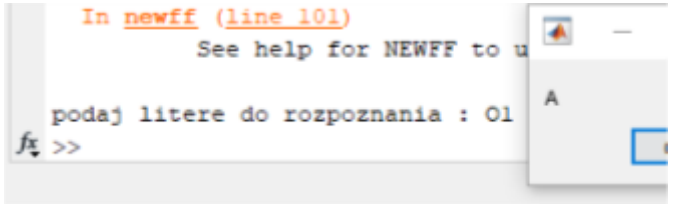


A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
0,997	0	0	0,502	0	0	0	0	0	0	0	0	0,013	0,005	0,214	0	0,009	0	0	0
0																			

Sieć wskazała, że na 99% to litera A, natomiast na 50% jest to D.

d)
Wprowadzona zniekształcona litera A, sieć wykryła że to litera A.

0	1	1	1	0
0	0	0	0	1
0	0	0	0	1
0	0	0	0	1
1	0	0	0	1
1	0	0	0	1
1	1	1	1	1



A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
0,514	0	0	0,067	0,002	0	0	0	0,006	0,448	0	0	0,001	0	0	0	0,146	0	0	0
0																			

Sieć wskazała, że na 51% to litera A, natomiast na 45% jest to J.

3. Listing kodu:

wyjscie = eye(20); **dane uczące**

litery=[A;B;C;D;E;F;G;H;I;J;K;L;M;N;O;P;Q;R;S;T]; **dane uczące**

net=newff(minmax(litery), 20, {'logsig', 'purelin', 'trainlm', 'learnh'}); **sieć wielowartstwowa zbudowana z 20 neuronów**

net.trainParam.epochs = 100;
net.trainParam.goal = 0.001; %**Cel wydajności**
net.trainParam.lr=0.08; % **wskaźnik uczenia sie**
lp.dr = 0.05;
lp.lr = 0.01;
dw=learnh([0],litery,[0],[0],wyjscie,[0],[0],[0],[0],[0],lp,[0]); **macierz wag dla reguły Hebba**
wynik=sim(net, litery);
litera=input('podaj litere do rozpoznania : ', 's');

net.trainFcn = 'trainbr'; **funkcja którą uczymy**
net.adaptFcn = 'adaptwb'; **funkcja do adaptacji wag i biasu**

net.inputWeights{:,:}.learnFcn = 'learnh'; % **każda waga jest korygowana za pomocą reguły Hebba**
net.layerWeights {:,:} .learnFcn = 'learnh'; **każda wartswa uczy się regułą Hebba**

net=train(net, litery,wyjscie); **uczenie sieci**
wynik=sim(net, litery);

switch litera **% Sprawdzenie czy jest taka literka którą podaliśmy**
….
…..
…...

4. Spostrzeżenia i wnioski

Można zauważyć, że grupowanie liter działa niezwykle poprawnie, nawet jak podamy niepełną literę, to i tak ją sieć rozpozna. Ma to kluczowe znaczenie w rozpoznawaniu pisma odręcznego, gdzie w różnych charakterach pisma, dana litera ma różny kształt.
Bardzo istotną kwestią jest wybór początkowych wartości wag neuronów sieci przeznaczonej do samouczenia. 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, przeto od jakości tych początkowych, „wrodzonych" właściwości sieci silnie zależy, do czego sieć dojdzie na końcu procesu uczenia. Nie wiedząc z góry, jakiego zadania sieć powinna się uczyć, trudno wprowadzać jakikolwiek zdeterminowany mechanizm nadawania początkowych wartości wag, jednak pozostawienie wszystkiego wyłącznie mechanizmom losowym może powodować, że sieć (zwłaszcza mała) może nie zdołać wystarczająco zróżnicować swego działania w początkowym okresie procesu uczenia i wszelkie późniejsze wysiłki, by znaleźć w strukturze sieci reprezentację dla wszystkich występujących w wejściowych sygnałach klas, mogą okazać się daremne.

Z reguły tej wynika, że dodatnia wartość składnika korelacyjnego $y_i x_j$ powoduje wzrost wagi w_{ij}, a w konsekwencji silniejszą reakcję neuronu przy kolejnej prezentacji tego samego obrazu wejściowego.
Często powtarzające się obrazy wejściowe dają więc silniejsza odpowiedź na wyjściu.