

## Sieć Kohonena

(kierunek studiów: *Zarządzanie i Inżynieria Produkcji*)

### 1 Cel laboratorium

Celem laboratorium jest zapoznanie się z zasadą działania oraz algorytmem strojenia sieci Kohonena. Sieć należy zastosować do rozwiązania problemu klasyfikacji próbek pomiarowych, dla przykładowego zbioru o nazwie *Iris* z repozytorium UCI (<http://archive.ics.uci.edu/ml/>). Problem dotyczy klasyfikacji irysów na trzy typy: *Setosa*, *Versicolour*, *Virginica*, na podstawie czterech atrybutów wejściowych prezentujących odpowiednie wymiary w cm:

1. długość podziałki kielicha;
2. szerokość podziałki kielicha;
3. długość płatk;
4. szerokość płatk.

Zbiór zawiera 150 próbek, po 50 dla każdej z klas.

### 2 Przebieg laboratorium

Do obsługi obliczeń należy zdefiniować m-plik w formie skryptu lub funkcji, który będzie realizował następujące zadania:

1. Przygotowanie danych.
  - (a) Pobranie pliku z próbkami pomiarowymi (`load iris.txt -ascii`)
  - (b) Podział na część z atrybutami wejściowymi  $X=iris(:,1:4)$  oraz wyjściem  $Y=iris(:,5)$ , pamiętając o transpozycji macierzy  $X$  do postaci wierszowej:  $X=X'$ .
  - (c) Wizualizacja danych:

```
figure(1);  
plot3(X(1,Y==1),X(2,Y==1),X(3,Y==1),'ob',X(1,Y==2),X(2,Y==2),X(3,Y==2),  
'or',X(1,Y==3),X(2,Y==3),X(3,Y==3),'og');  
figure(2);  
plot3(X(1,Y==1),X(2,Y==1),X(4,Y==1),'ob',X(1,Y==2),X(2,Y==2),X(4,Y==2),  
'or',X(1,Y==3),X(2,Y==3),X(4,Y==3),'og');
```

2. Przygotowanie sieci neuronowej.
  - (a) Inicjacja sieci Kohonena: `net=newc(minmax(X),3)`.
  - (b) W celu wizualizacji procesu adaptacji wartości współczynników wagowych, należy ustawić liczbę kroków uczenia na wartość równą 10 (`net.trainParam.epochs = 10`).

(c) Proszę zwrócić uwagę na wstępne wartości współczynników wagowych. Macierz z wartościami wag neuronów (`net.IW{1}`) oraz typ funkcji użytej do inicjacji wag (`net.inputWeights{1}`).

(d) Strojenie sieci:

```
for i = 1:10
    net = adapt(net,X);
    figure(1);
    hold on;
    plot3(net.IW{1}(1,1:3),net.IW{1}(2,1:3),net.IW{1}(3,1:3),'.k');
    figure(2);
    plot3(net.IW{1}(1,[1,2,4]),net.IW{1}(2,[1,2,4]),net.IW{1}(3,[1,
        2,4]),'.k');
end
```

3. Analiza wyników.

(a) W ramach analizy wyników należy obliczyć odpowiedzi sieci na zadaną macierz z próbkami wejściowymi: `YSN = sim(net,X)`. W jakiej formie sieć udzieliła odpowiedzi, z czego to wynika?

Funkcja do konwersji wyników: `YS = vec2ind(YSN);`.

(b) Należy porównać odpowiedzi sieci z wartościami zadanymi, w celu ustalenia dokładności treningu.

4. Obliczenia dodatkowe.

(a) Zbadać wzajemną zależność zmiennych wejściowych, licząc ich współczynniki korelacji (funkcja `R = corrcoef(X)`, dla dla poszczególnych kolumn (traktowanych jako zmienne wejściowe) macierzy  $X$  zwraca macierz  $R$  z wartościami korelacji).

(b) W celu weryfikacji wpływu sposobu inicjacji wag neuronów należy zmienić ich wartości. Przykładowo, wpisując nowe dane bezpośrednio do macierzy wag (przyp. `net.IW{1}`) lub zmieniając typ funkcji inicjującej (w celu uzyskania listy możliwych funkcji: `help mninit`)

(c) Sprawdzić standardową wartość współczynnika uczenia `lr` (`net.inputWeights{1}.learnParam`). Czy jest to wartość optymalna dla analizowanego problemu?

(d) Przeanalizować wpływ współczynników `bias` na jakość uczenia sieci.

(e) Sprawdzić wpływ liczby iteracji na jakość uczenia sieci.

(f) Przeanalizować algorytm strojenia sieci Kohonena.

## Literatura

- [1] Osowski Stanisław, Sieci neuronowe w ujęciu algorytmicznym, Wydawnictwa Naukowo-Techniczne, Warszawa 1996.
- [2] Sieci Neuronowe, W. Duch, J. Korbicz, L. Rutkowski, R. Tadeusiewicz, Akademicka Oficyna Wydawnicza EXIT, Warszawa 2000.