Anna Budzoń

Sprawozdanie do scenariusza 6.

05.01.18r.

1. Cel ćwiczenia:

Poznanie budowy i działania sieci Kohonena przy wykorzystaniu reguły WTM do odwzorowywania istotnych cech liter alfabetu.

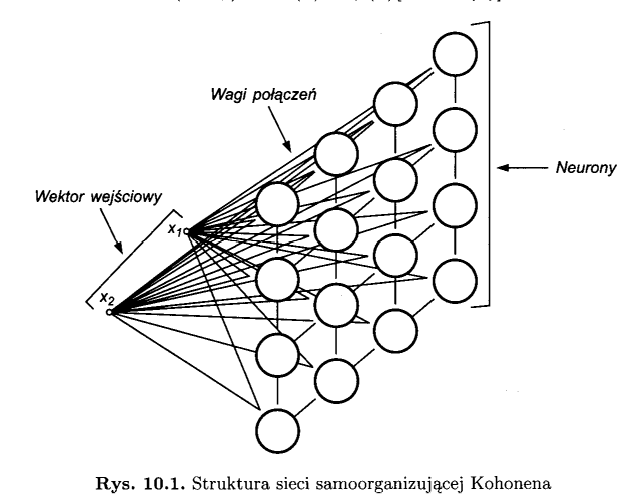
1. Opis syntetyczny:
2. Sieć Kohonena:

* Sieć samoorganizująca się przez współzawodnictwo – podstawą uczenia się sieci jest konkurencja między neuronami. Polega na wyłanianiu na każdym etapie zwycięzcy, czyli neuronu, którego wektor wagowy różni się najmniej od przyłożonego na wejściu sieci wektora x. Uczenie przebiega bez nauczyciela.
* Sieci samoorganizujące są zwykłe jednowarstwowe, a każdy neuron połączony jest ze wszystkimi składowymi N-wymiarowego wektora wejściowego x (na rysunku N = 2). Wagi połączeń synaptycznych neuronów tworzą wektor wi. Zwycięża neuron, którego wagi najmniej różnią się od odpowiednich składowych tego wektora. Spełnia on relację:

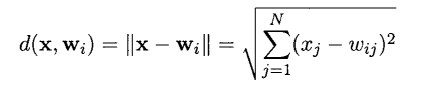


d(x,w) to odległość między wektorem x, a wektorem w neuronu zwycięzcy,

n – liczba neuronów.



* Wejścia połączone są ze wszystkimi węzłami sieci. Każdy węzeł sieci przechowuje wektor wag o wymiarze identycznym z wektorami wejściowymi.
* Miara odległości euklidesowa (odległość między wektorem x i wektorem wi):



1. Reguła WTM (Winner Takes Most) – strategia, w której wokół neuronu zwycięzcy przyjmuje się sąsiedztwo Sw(k) o określonym promieniu malejącym w czasie. Jednym słowem neuron zwycięzca i wszystkie neurony w sąsiedztwie podlegają uaktualnianiu swoich wag w kierunku wektora x, zgodnie z regułą Kohonena:

, gdzie

* jest współczynnikiem uczenia i-tego neuronu z sąsiedztwa Sw(k) w k-tej chwili. Wartość ta malenie wraz z odległością neuronu od zwycięzcy.

Wagi neuronów spoza sąsiedztwa nie podlegają zmianom. Rozmiar sąsiedztwa oraz wartości współczynników uczenia poszczególnych neuronów są funkcjami malejącymi w czasie wraz z uczeniem się sieci.

Natomiast im dalsza odległość neuronu od zwycięzcy, tym mniejsza zmiana wartości wag tego neuronu.

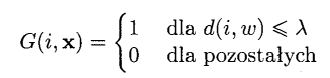
Uogólniona zależność uaktualniania wartości wag neuronu:



Zależność współczynnika uczenia ηi uniezależniono od odległości względem prezentowanego wektora x, poprzez uwzględnienie funkcji sąsiedztwa G(i,x).

1. Funkcja sąsiedztwa dla sieci Kohonena:

* Sąsiedztwo prostokątne:

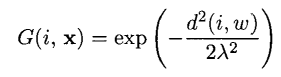


d(i,w) może oznaczać odległość euklidesową między wektorami wag neuronu zwycięzcy w i neuronu i-tego, bądź odległością mierzoną w liczbie neuronów.

– promień sąsiedztwa o wartości malejącej z czasem uczenia do zera

Każdy neuron należący do sąsiedztwa zwycięzcy podlega adaptacji wag w jednakowym stopniu.

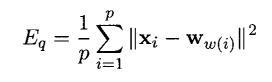
* Sąsiedztwo typu gaussowskiego – użyte w projekcie



O stopniu adaptacji neuronów sąsiedztwa zwycięzcy decyduje odległość euklidesowa d(i,w) neuronu i-tego od zwycięzcy, a także promień sąsiedztwa . W odróżnieniu od sąsiedztwa typu prostokątnego, stopień adaptacji jest zróżnicowany dla neuronów sąsiedztwa i zależy od wartości funkcji Gaussa. Dla zwycięzcy G(w,w) = 1, dla pozostałych neuronów 0 < G(i,w) < 1.

1. Cel uczenia sieci samoorganizujący się przez konkurencje:

* Takie uporządkowanie neuronów (dobór wartości ich wag), które zminimalizuje wartość oczekiwaną zniekształcenia, określaną jako błąd popełniany przy aproksymacji wektora wejściowego x, wartościami wag neuronu zwyciężającego w konkurencji. Przy p wektorach wejściowych x i zastosowaniu normy euklidesowej, błąd kwantyzacji może być wyrażony:

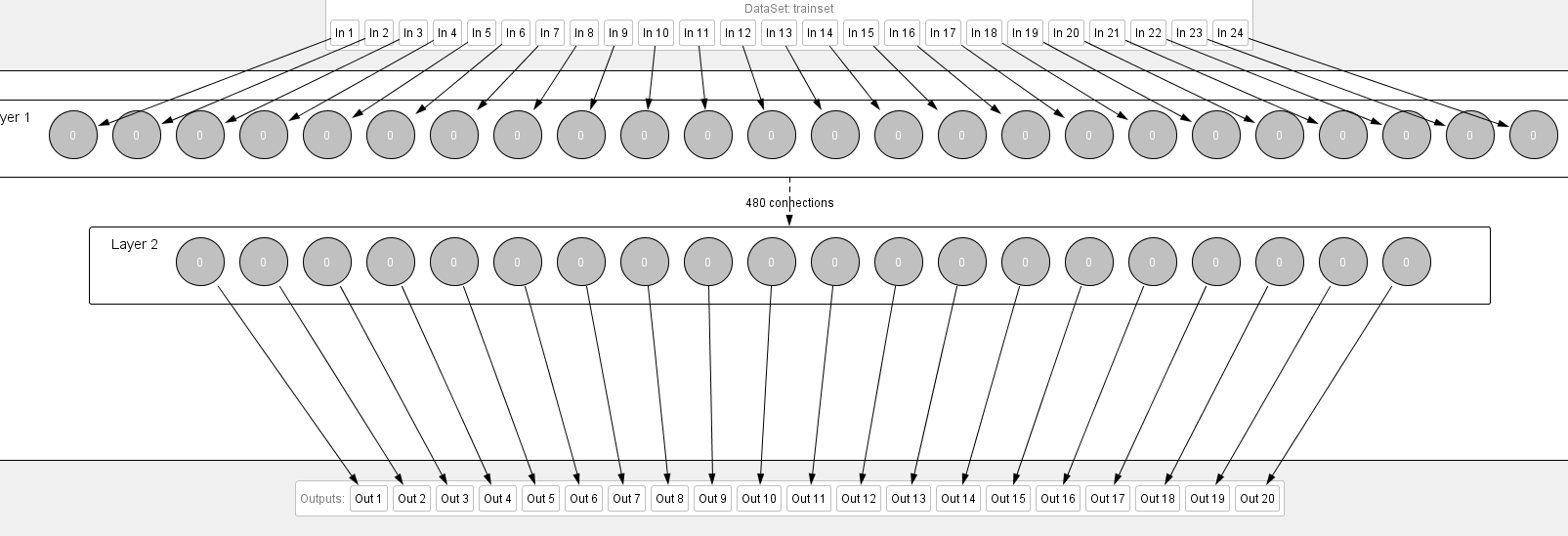


Ww(i) – waga neuronu zwyciężającego przy prezentacji wektora xi

1. Wykorzystana sieć:

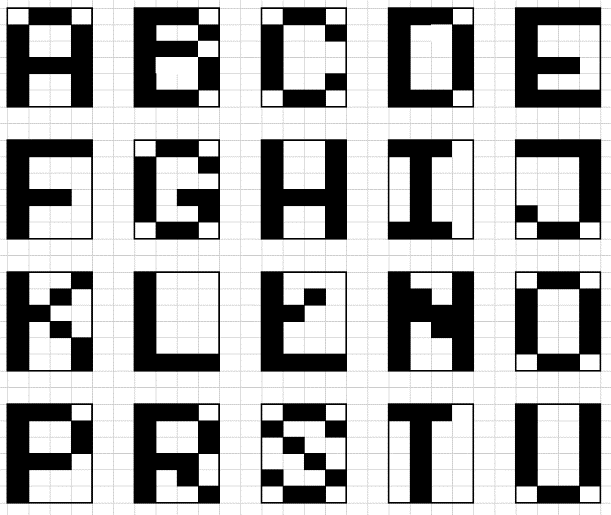
- 24 neuronów wejściowych

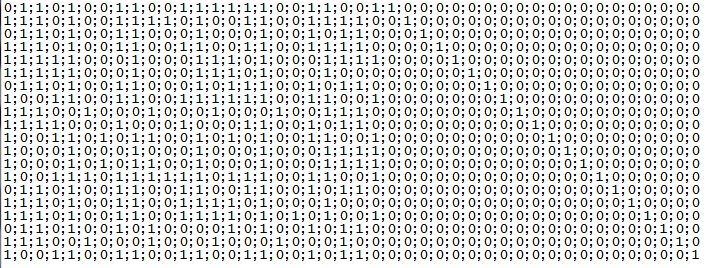
- 20 danych wyjściowych



1. Zestaw uczący:

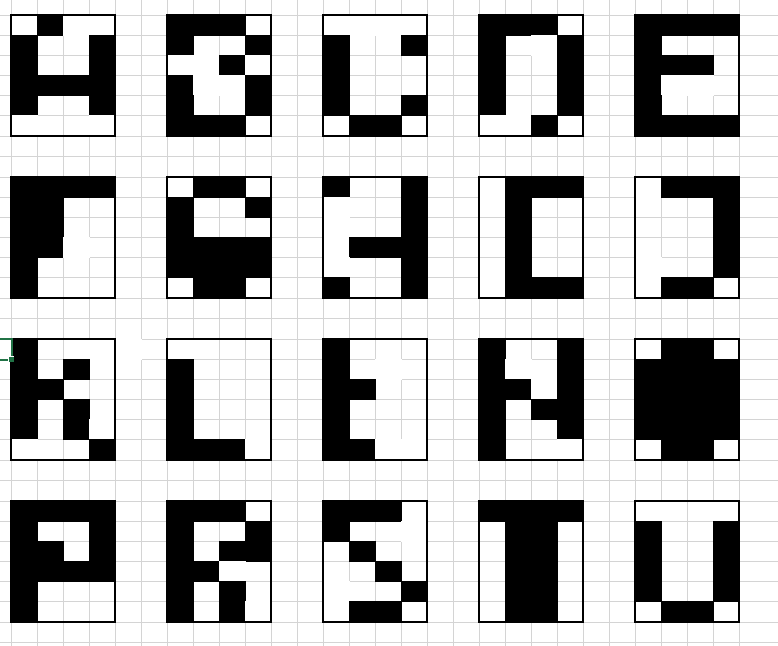
* 20 dużych liter o wymiarach 4x6.

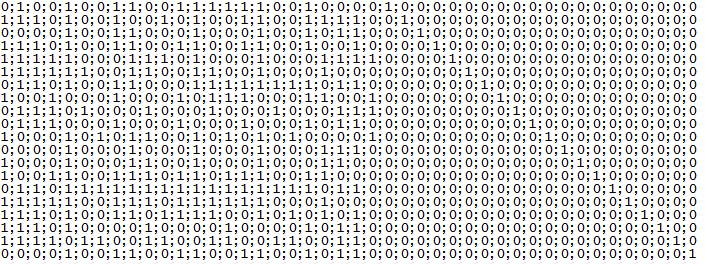




1. Zestaw testujący:

* Zmodyfikowane litery z zestawu uczącego.





1. Wyniki:

**EXCEL**

1. Analiza wykresów:

Jak możemy wywnioskować z wykresów, ilość epok maleje wraz ze wzrostem współczynnika uczenia i promienia. Sieć uczy się zdecydowanie lepiej przy stałej uczenia zbliżającej się do 1, jednakże otrzymujemy wtedy mniej dokładne rezultaty i sieć popełnia zdecydowanie więcej pomyłek. Ponadto najbardziej optymalny promień ma mniejsze wartości, najlepiej 1 lub 2, ponieważ przy wartości 3 i 4 sieć bardzo często nie rozpoznawała żadnych liter przy podanych danych testowych.

1. Analiza otrzymanych wyników:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Litera | Testowanie 1 | Testowanie 2 | Testowanie 3 | Testowanie 4 |
|  |  |  |  | (0.41) |
|  |  |  |  |  |
|  |  | (0.88) (0.36) |  |  |
|  | (0.62) (0.4) |  |  | Nie rozpoznano żadnej litery |
|  | (0.74) (0.46) (0.36) |  | (0.75) (0.27) | Nie rozpoznano żadnej litery |
|  | (0.88) (0.23) |  |  |  |
|  |  | (0.77) (0.32) |  | (0.42) |
|  | (0.85) (0.25) | Nie rozpoznano żadnej litery |  |  |
|  |  | (0.32) (0.77) |  | Nie rozpoznano żadnej litery |
|  |  |  |  |  |
|  | (0.88) (0.22) | (0.65) (0.35) |  | (0,53) |
|  |  | (0.42) | (0.72) (0.25) | (0.92) (0.32) |
|  | (0.71) (0.32) | (0.52) (0.42) |  |  |
|  |  |  |  |  |
|  | (0.81) (0.35) (0.28) |  |  | (0.26) |
|  | (0.46) (0.28) |  |  | (0.63) |
|  | (0.27) | (0.67) (0.47) (0.39) |  | (0.27) |
|  |  | (0.75) (0.34) |  | Nie rozpoznano żadnej litery |
|  | (0.88) (0.23) |  |  | Nie rozpoznano żadnej litery |
|  |  | (0.84) (0.38) |  | (0.85) (0.5) |

1. Wnioski:
2. Proces uczenia przebiegł pomyślnie dla każdego współczynnika uczenia i przyjętego promienia sąsiedztwa, mimo niewielkich błędów otrzymanych wyników dla testowania na danych uczących. Sieć otrzymywała wartości <0.8-1) dla wartości oczekiwanej 1. Najgorsze wyniki otrzymano przy uczeniu numer 4, gdyż wartości maksymalnie osiągnęły wartość 0.9. Mogło to wskazywać na to, że sieć nie osiągnęła najlepszej dokładności, na co również wskazują otrzymane wyniki testowania sieci.
3. Uczenie pierwsze przebiegło aż w 21 000 iteracjach, ze względu na przyjęty mały współczynnik uczenia się – 0,01. Wartość błędu wraz z kolejnymi epokami malała liniowo, płynnie. Sieć bardzo często pokazywała wiele odpowiedzi dla jednej danej testowej i nie klasyfikowała litery jednoznacznie do jednej grupy. Rozpoznała prawidłowo I, J, K, S, T, U. Jest to mały procent prawidłowych odpowiedzi (6/20), jednak trzeba wziąć pod uwagę, że podany na wejściu zestaw testujący znacznie różnił się od uczącego i litery nieraz nie przypominały tych przez modyfikacją. Dlatego oczywistym jest, że sieć rozpoznawała cechy innych znaków niż oczekiwanych.
4. Uczenie drugie przebiegło podobnie jak uczenie pierwsze. Obejmowało 4150 epok przy współczynniku uczenia 0.8 i promieniu sąsiedztwa 2. Wartość błędu malała przez większość czasu liniowo, jednak pod koniec nauki możemy zaobserwować wyraźne zaburzenia i skoki błędu. Może to wskazywać na to, że sieć nie do końca została nauczona prawidłowo. Sieć raz nie rozpoznała żadnej litery – przy zmodyfikowanej literze H. Prawidłowe wyniki zostały podane dla: B, D, E, J, K, N, T, U (8/20), więc więcej niż dla pierwszego uczenia. Jednoznaczna klasyfikacja litery nastąpiła 10 razy, a przy drugiej połowie litery zostały przypasowane do wielu kategorii.
5. Trzecie uczenie przebiegło w 950 iteracjach przy współczynniku uczenia się równym 0.2 i promieniu 3. Analizując wykres, możemy stwierdzić, że błąd często, impulsywnie wzrastał i malał, przez co proces uczenia nie przebiegł płynnie. Może to przyczynić się do popełnianych przez sieć błędów i jej niedokładności wyników. Jednakże, analizując wyniki, możemy stwierdzić, że sieć w większości przypadków, bo aż 18/20 razy, zaklasyfikowała litery do konkretnych grup, a wyniki dla każdego przypadku były większe od 0.9. Oczekiwane wyniki osiągnęliśmy dla: A, B, F, I, R, S (6/20), podobnie jak w pierwszym uczeniu, nie dość zadawalające.
6. Ostatnie uczenie zajęło jedynie 215 epok ze względu na wysoką stałą uczenia równą 0.5 i duży promień sąsiedztwa równy 4. Proces ten przebiegł najmniej pomyślnie, co możemy wywnioskować po otrzymanych wynikach testowania. Żadnego wpływu na rezultaty nie miał fakt, że wykres błędu od czasu uczenia się zmniejszał się liniowo. 5/20 razy sieć nie rozpoznała w danych testowych żadnej z podanych liter uczących. Jednoznacznie i wyższe od 0.7 wyniki podała jedynie dla 6/20 przypadków. Dla pozostałych klasyfikowała znaki do wielu kategorii lub podawała wartość nie zbliżoną do oczekiwanej (1). Prawidłowo rozpoznała F, H, L, U (4/20). Możemy podejrzewać, że błędny proces uczenia mógł wynikać z zbyt dużego współczynnika uczenia, gdyż sieć nie zdążyła dopasować dokładnie odpowiednich wag. Wpływ mogła mieć również zbyt wysoka wartość promienia sąsiedztwa, co może wskazywać na to, że optymalne wartości promienia należało zawrzeć w zakresie 1-3.