

Barbara Partyka

Scenariusz 6

Temat ćwiczenia: Budowa i działanie sieci Kohonena dla WTM.

Celem ćwiczenia jest poznanie budowy i działania sieci Kohonena przy wykorzystaniu reguły WTM do odwzorowywania istotnych cech liter alfabetu.

Algorytm WTA

Pierwszym algorytmem użytym w badaniach był zwykły algorytm WTA (Winner Takes All). Dąży się w nim do minimalizacji odległości wektora x od wektora wagowego neuronu zwycięskiego lub do maksymalizacji iloczynu skalarnego wektora uczącego x oraz wektora wag w : $x^T w \rightarrow \max$. W ten sposób wyłania się neuron zwycięzca. Zwycięski neuron adaptuje swoje wagi w kierunku wektora x według wzoru (3), w którym η oznacza współczynnik uczenia o wartości malejącej liniowo w kolejnych iteracjach. Pozostałe neurony nie adaptują swoich wag.

$$(3) w[w] \leftarrow w[m] + \eta(x - w[m])$$

(źródło: <http://pe.org.pl/articles/2016/10/48.pdf>)

Algorytmy WTA, w których tylko jeden neuron może podlegać adaptacji w każdej iteracji, są algorytmami słabo zbieżnymi, szczególnie przy dużej liczbie neuronów. W praktyce zostały one zastąpione algorytmami WTM (Winner Takes Most), w których oprócz zwycięzcy uaktualniają swoje wagi również neurony z jego sąsiedztwa, przy czym im dalsza jest odległość od zwycięzcy, tym mniejsza jest zmiana wartości wag neuronu.

Nauka sieci

Podczas uczenia się danego rekordu, będzie wyłaniany jeden zwycięski neuron i następnie dostosowywane wagi w nim oraz jego sąsiedztwie. Po skończonym procesie nauki taki neuron będzie rozpoznawał daną klasę obiektów, ponieważ tylko on będzie się aktywował.

Zwycięski neuron typowany jest za pomocą miary Euklidesowej. Zwycięża neuron zwracający najmniejszą wartość d dla danej iteracji algorytmu.

X – wektor rekordu podawanego w warstwie wejściowej

W – wektor wag i -tego neuronu

Współczynnik uczenia

Współczynnik uczenia jest obliczany przy każdej iteracji programu. Jest on malejący.

Został wykorzystany poniższy wzór:

L_0 - Jest początkową wartością współczynnika uczenia

t – Jest numerem epoki uczenia

- Stanowi końcową ilość epok uczenia(limit)

0 – Jest początkową średnicą sąsiedztwa (często ustalane jako szerokość całej mapy)

t – Jest numerem epoki uczenia

– Stanowi końcową ilość epok uczenia(limit)

Funkcja theta

Wpływ neuronu zwycięskiego na neurony sąsiadujące

Dist - Jest odległością neuronu dla którego obliczamy zmianę wag, od neuronu zwycięskiego

X – Wektor uczący

W_i , winner – Wektor wag zwycięskiego neuronu

Znając wartości błędów dla wszystkich wektorów uczących, można obliczyć przeciętny błąd kwantyzacji:

Stworzona sieć posiada 35 wejść oraz 30 neuronów w warstwie wyjściowej.

Dane uczące oraz testujące

Dane uczące oraz testujące są takie same i są zawarte w pliku „learn.txt”

Siatki liter zostały pobrane ze strony:

<http://www.ai.c-labtech.net/sn/litery.html>

Pierwsza linia oznacza liczbę rekordów

Druga linia oznacza liczbę danych

Każdy rekord ma na końcu literę, którą reprezentuje.

Plik „learn.txt”:

35

-1 1 1 1 -1 1 -1 -1 -1 1 1 -1 -1 -1 1 1 1 1 1 1 -1 -1 -1 1 1 -1 -1 -1 1 1 -1 -1 -1 1

A

1 1 1 1 -1 1 -1 -1 -1 1 1 -1 -1 -1 1 1 1 1 1 -1 1 -1 -1 -1 1 1 -1 -1 -1 1 1 1 1 1 -1 B

-1 1 1 1 -1 1 -1 -1 -1 1 1 -1 -1 -1 -1 1 -1 -1 -1 1 -1 -1 -1 1 -1 -1 -1 1 -1 1 1

1 -1 C

1 1 1 1 -1 1 -1 -1 -1 1 1 -1 -1 -1 1 1 -1 -1 -1 1 1 -1 -1 -1 1 1 1 1 1 -1

D

1 1 1 1 1 1 -1 -1 -1 -1 1 -1 -1 -1 -1 1 1 1 1 -1 1 -1 -1 -1 -1 1 -1 -1 -1 -1 1 1 1 1

E

1 1 1 1 1 1 -1 -1 -1 -1 1 -1 -1 -1 -1 1 1 1 1 -1 1 -1 -1 -1 -1 1 -1 -1 -1 1 -1 -1 -1

-1 F

-1 1 1 1 -1 1 -1 -1 -1 1 1 -1 -1 -1 -1 1 -1 1 1 1 1 -1 -1 -1 1 1 -1 -1 -1 1 1 1 1 -1

G

1 -1 -1 -1 1 1 -1 -1 -1 1 1 -1 -1 -1 1 1 1 1 1 1 -1 -1 -1 1 1 -1 -1 -1 1 1 -1 -1 -1 1

H

-1 1 1 1 -1 -1 -1 1 -1 -1 -1 -1 1 -1 -1 -1 -1 1 -1 -1 -1 -1 1 -1 -1 -1 -1 1 -1 -1 1

1 1 -1 I

1 1 1 1 1 -1 -1 -1 -1 1 -1 -1 -1 -1 1 -1 -1 -1 -1 1 1 -1 -1 -1 1 -1 1 1 1

-1 J

1 -1 -1 -1 1 1 -1 -1 1 -1 1 -1 -1 1 1 -1 -1 -1 1 -1 -1 1 -1 -1 1 -1 1 -1 -1 0

1 K

1 -1 -1 -1 -1 1 -1 -1 -1 1 -1 -1 -1 1 -1 -1 -1 1 -1 -1 -1 1 -1 -1 -1 1 1

1 1 1 L

1 -1 -1 -1 1 1 1 -1 1 1 1 -1 1 -1 1 1 -1 -1 -1 1 1 -1 -1 -1 1 1 -1 -1 -1 1

M

1 -1 -1 -1 1 1 -1 -1 -1 1 1 1 -1 -1 1 1 -1 1 -1 1 1 -1 -1 -1 1 1 -1 -1 -1 1

N

-1 1 1 1 -1 1 -1 -1 -1 1 1 -1 -1 -1 1 1 -1 -1 -1 1 1 -1 -1 -1 1 -1 1 1 1

-1 O

1 1 1 1 -1 1 -1 -1 -1 1 1 -1 -1 -1 1 1 1 1 -1 1 -1 -1 -1 -1 1 -1 -1 -1 1 -1 -1 -1

-1 P

-1 1 1 1 -1 1 -1 -1 -1 1 1 -1 -1 -1 1 1 -1 1 -1 1 1 -1 -1 1 -1 -1 1 1 -1 1

Q

1 1 1 1 -1 1 -1 -1 -1 1 1 -1 -1 -1 1 1 1 1 -1 1 -1 1 -1 -1 1 -1 1 -1 -1 -1 1

R

-1 1 1 1 -1 1 -1 -1 -1 1 1 -1 -1 -1 -1 1 1 1 -1 -1 -1 -1 1 1 -1 -1 -1 1 1 1

-1 S

1 1 1 1 1 -1 -1 1 -1 -1 -1 -1 1 -1 -1 -1 -1 1 -1 -1 -1 1 -1 -1 -1 1 -1 -1 -1 1

-1 -1 T

1 -1 -1 -1 1 1 -1 -1 -1 1 1 -1 -1 -1 1 1 -1 -1 -1 1 1 -1 -1 -1 1 -1 1 1 1

-1 U

```

1 -1 -1 -1 1 1 -1 -1 -1 1 1 -1 -1 -1 1 1 -1 -1 -1 1 -1 1 -1 1 -1 -1 -1 1
-1 -1 V
1 -1 -1 -1 1 1 -1 -1 -1 1 1 -1 -1 -1 1 1 -1 -1 -1 1 1 -1 1 -1 1 -1 1 -1 1
-1 W
1 -1 -1 -1 1 1 -1 -1 -1 1 1 -1 1 -1 -1 -1 1 -1 -1 1 -1 1 -1 -1 -1 1 1 -1 -1
-1 1 X
1 -1 -1 -1 1 1 -1 -1 -1 1 1 -1 1 -1 -1 -1 1 -1 -1 -1 1 -1 -1 -1 1 -1 -1 -1
1 -1 -1 Y
1 1 1 1 1 -1 -1 -1 -1 1 -1 -1 -1 1 -1 -1 -1 1 -1 -1 -1 1 -1 -1 -1 1 1 1 1
1 Z

```

Wnioski

- Dzięki Sieci Kohonena z użyciem algorytmu WTM możemy rozpoznawać charakterystyczne cechy danych wejściowych i grupować je względem ich podobieństwa
- Algorytm WTM dobrze sprawdza się przy dużych wymiarach danych wejściowych, które później można reprezentować w małych wymiarach
- Aby sieć poprawnie działała należy dobrać odpowiednie współczynniki początkowe
- Każdy ze współczynników miał duży wpływ na działanie programu

Listing kodu programu

https://github.com/barbarapar/PSI_GCP03_zima_2017-2018_Barbara_Partyka