

# Przykład optymalizacji struktury sztucznej sieci neuronowej metodą algorytmów genetycznych

**Leszek Grad**

Zakład Automatyki, Instytut Teleinformatyki i Automatyki WAT,  
ul. Kaliskiego 2, 00-908 Warszawa

**STRESZCZENIE:** W artykule przedstawiono przykład optymalizacji struktury jednokierunkowej wielowarstwowej sztucznej sieci neuronowej metodą algorytmów genetycznych. Zaproponowano funkcję przystosowania pozwalającą ocenić jakość proponowanej struktury. Obliczenia wykonano dla sieci neuronowej rozpoznającej cyfry pisane odręcznie.

**SŁOWA KLUCZOWE:** sieć neuronowa, algorytmy genetyczne, optymalizacja struktury

Przedstawiony w artykule problem optymalizacji struktury sztucznej sieci neuronowej jest zagadnieniem istotnym dla projektantów systemów sztucznej inteligencji wykorzystujących jednokierunkowe wielowarstwowe sieci neuronowe. Sieć tego typu realizuje odwzorowanie  $N$ -wymiarowego wektora wejściowego w  $M$ -wymiarowy sygnał wyjściowy. Z teorii Kołmogorowa wynika, że aproksymacja taka jest możliwa przy zastosowaniu jednej warstwy ukrytej o  $2N+1$  neuronach [3]. W praktyce konstruuje się sieci o co najwyżej dwóch warstwach ukrytych, a liczba neuronów w warstwach ukrytych waha się od  $N$  do  $3N$ .

Do optymalizacji struktury sieci neuronowej stosowane są dwa podstawowe podejścia: redukcji sieci oraz rozbudowy sieci. W pierwszym, przyjmuje się nadmiarową architekturę, która w trakcie procesu uczenia bądź po jego zakończeniu jest upraszczana poprzez redukcję najmniej znaczących wag. Do najpopularniejszych metod redukcji sieci należy zaliczyć: metodę OBD (*Optimal Brain Damage*) oraz metodę z funkcją kary. W drugim podejściu, na wstępie przyjmuje się strukturę sieci o niewielkiej liczbie neuronów, która w miarę upływu procesu uczenia jest zwiększana. Spośród wielu metod

rozbudowywania sieci wymienić należy algorytmy: Mezarda-Nadala, Marchanda, Li-Tuftsza, kaskadowej korelacji Fahlmana[3].

W artykule przedstawiony został przykład wykorzystania algorytmów genetycznych do optymalizacji struktury sieci neuronowej rozwiązującej zadanie rozpoznawania znaków (cyfr) pisanych odręcznie.

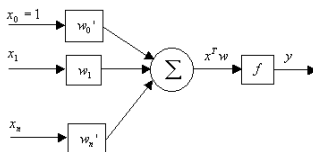
## 1. Sieć neuronowa nieliniowa wielowarstwowa jednokierunkowa

Jednokierunkowa wielowarstwowa nieliniowa sieć neuronowa jest narzędziem pozwalającym na znajdowanie związków (funkcji, relacji) pomiędzy danymi wejściowymi, a wynikami działania systemu na podstawie posiadanych częściowych danych empirycznych. Wykorzystywana jest najczęściej do rozwiązywania zadań klasyfikacji. Składa się z elementarnych elementów przetwarzających zwanych neuronami. Neuron realizuje odwzorowanie postaci:

$$y = f(w^T x)$$

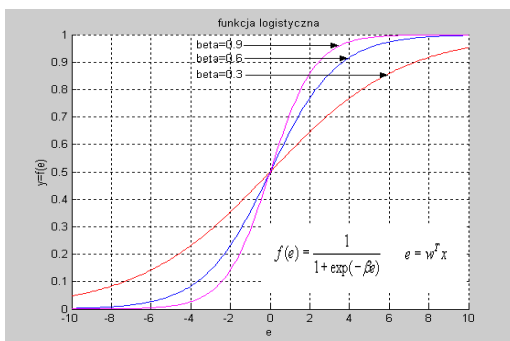
gdzie:  $x$  - sygnał wejściowy,  $w$  - wektor wag,  $y$  - wyjście neuronu,  $f$  - nieliniowa funkcja przejścia (aktywacji).

Na rys.1. przedstawiono schemat neuronu nieliniowego, a na rys.2. logistyczną funkcję aktywacji. Poza tą funkcją częste zastosowanie znajduje funkcja tangensa hiperbolicznego [2][4].

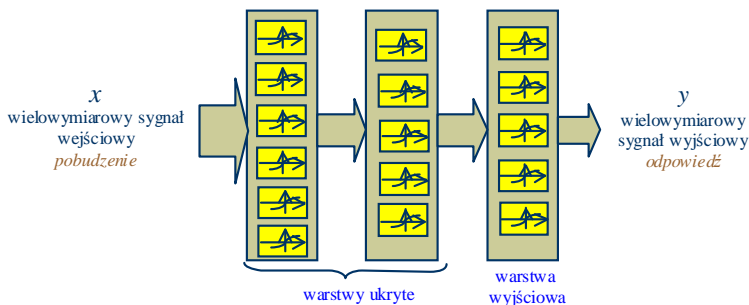


Rys. 1. Neuron nieliniowy

Sieć wielowarstwowa składa się z warstw neuronów. Neurony leżące w tej samej warstwie nie są połączone ze sobą. Wysyłają sygnały do wszystkich neuronów leżących w warstwie kolejnej (poza neuronami warstwy ostatniej). Ostatnia warstwa sieci, której wyjścia są wyjściami całej sieci, nosi nazwę warstwy wyjściowej. Pozostałe warstwy nazywane są ukrytymi. Schemat sieci wielowarstwowej przedstawiony został na rys.3.



Rys. 2. Logistyczna funkcja aktywacji

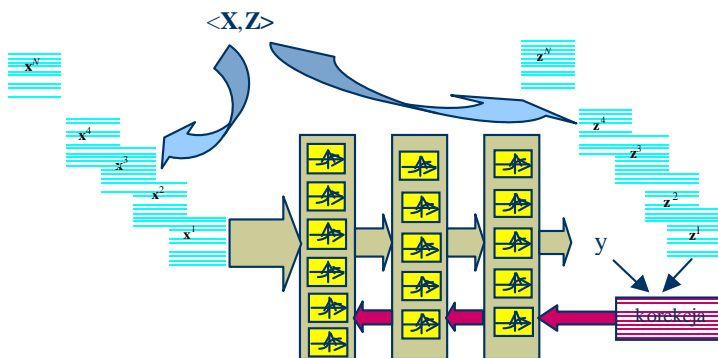


Rys. 3. Jednokierunkowa wielowarstwowa sieć neuronowa

Zanim sieć neuronowa będzie zdolna do realizacji zadania klasyfikacji musi zostać odpowiednio przygotowana. Przygotowanie sieci odbywa się w procesie treningu, w trakcie którego ustalane są wartości wag wszystkich neuronów sieci. Trening oparty jest o zgromadzone wejściowe dane wzorcowe (macierz  $X$ ), dla których tworzy się odpowiadającą (pożądaną) macierz wyjść (macierz  $Z$ ). Skuteczną metodą treningu sieci jest metoda wstecznej propagacji błędów. Jest to metoda gradientowa poszukiwania kierunku najszybszego spadku funkcji błędów. Schematyczne ujęcie procesu treningu wielowarstwowej sieci neuronowej pokazano na rys.4.

W metodzie gradientowej trening polega na wielokrotnym podawaniu wszystkich elementów ciągu uczącego. Po podaniu wejściowego sygnału

wzorcowego następuje korekcja wag sieci. Jednokrotne podanie wszystkich wzorców z ciągu uczącego na sieć nazywane jest epoką. Funkcja określająca błąd sieci jest w metodzie gradientowej funkcją kryterium. Sieć uznaje się za nauczoną w przypadku, kiedy błąd na wyjściu sieci jest mniejszy od zadanego. Najpopularniejszą miarą błędu sieci jest sumaryczny błąd średniokwadratowy (SSE - Sum Square Error) obliczany jako pierwiastek z sumy kwadratów różnic wartości żądanych ( $z$ ) i otrzymywanych ( $y$ ) na wszystkich wyjściach sieci dla całej epoki.



Rys. 4. Trening wielowarstwowej sieci neuronowej

## 2. Sformułowanie zadania klasyfikacji

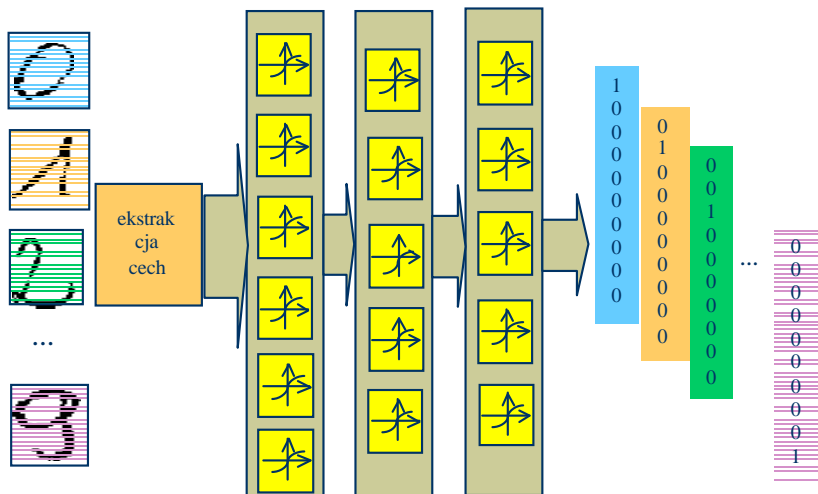
W punkcie tym omówione zostanie zadanie klasyfikacji rozwiązywane przez sieć neuronową. Będzie to zadanie rozpoznawania cyfr arabskich pisanych odręcznie. Zakładamy, że dysponujemy zbiorem wzorców cyfr zapisanych w formie obrazów binarnych.

Rozpoznawanie z wykorzystaniem sieci neuronowej będzie przebiegało zgodnie ze schematem przedstawionym na rys.5.

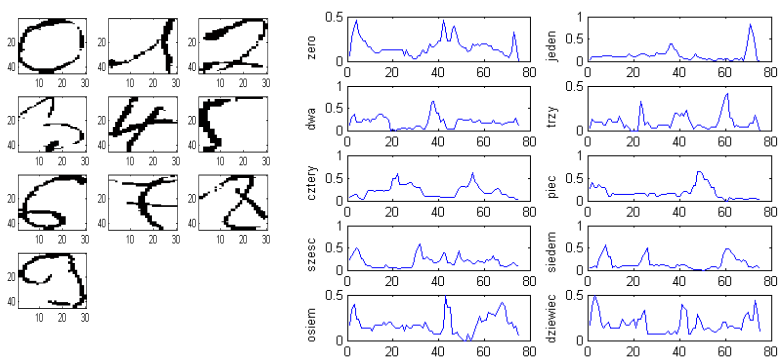
Do rozpoznawania cyfr zastosowano sieć trójwarstwową z możliwością redukcji do warstw dwóch przez mechanizm optymalizacji. Warstwa ostatnia jest warstwą wyjściową, zawierającą na stałe 10 neuronów. Każdy neuron tej warstwy reaguje wartością 1 na jedną, przypisaną mu cyfrę, a wartością 0 na pozostałe.

Obrazy podawane na wejście sieci są poddawane procesowi ekstrakcji cech (redukcji wymiaru wektora wejściowego). W rozpatrywanym rozwiązaniu

przyjęto rozpoznawanie na podstawie wektora zawierającego informację o stopniu zaczerwienia wierszy i kolumn. Wymiar wektora opisującego obraz po procesie ekstrakcji cech wyniósł 75 (rozmiar obrazu 45x30 pikseli) (rys 6).



Rys. 5. Schemat układu rozpoznawania cyfr



Rys. 6. Wynik ekstrakcji cech z obrazu, opis cyfr za pomocą stopnia zaczerwienia w wierszach i kolumnach obrazu

### 3. Metoda algorytmów genetycznych

Algorytmy genetyczne należą do metod optymalizacji. Zasada ich działania oparta jest na doborze naturalnym i dziedziczeniu.

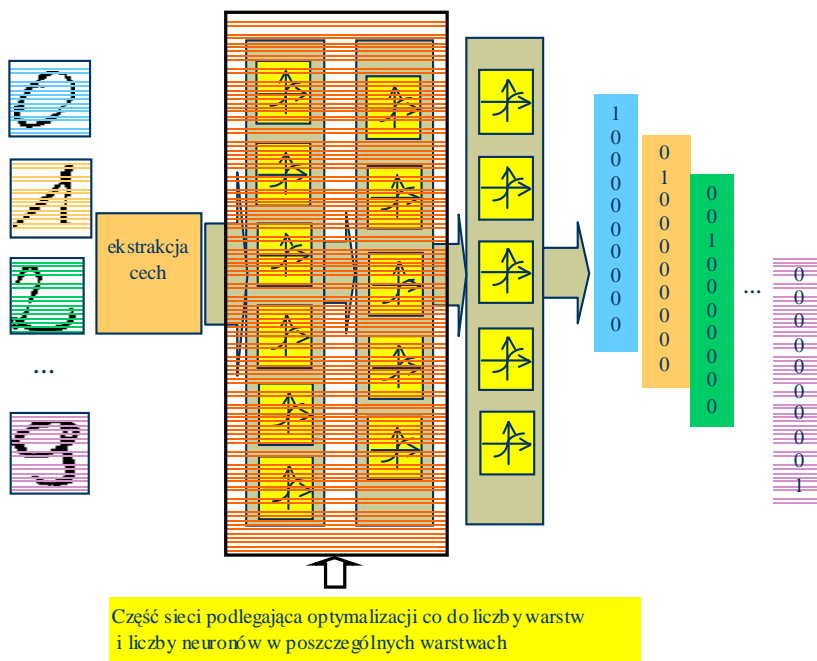
Pierwszym krokiem algorytmu jest wybór podzbioru rozwiązań dopuszczalnych zwanych populacją początkową. Każdy element populacji zwany chromosomem kodowany jest za pomocą liczby binarnej o stałej długości. Jeden bit tej liczby nazywany jest genem. Następnie dla całej populacji obliczana jest wartość tzw. funkcji przystosowania (funkcji kryterium). Wartość tej funkcji stanowi podstawę przeprowadzenia zmian w populacji na drodze selekcji i krzyżowania. Selekcja jest operacją wyboru do następnej populacji (tzw. rodzicielskiej) osobników (chromosomów). W tej operacji do następnej generacji większe szanse przejścia mają osobniki lepiej przystosowane (w rozpatrywanym przypadku o mniejszej wartości funkcji przystosowania). Na etapie krzyżowania następuje wymiana genów pomiędzy dwoma losowo wybranymi osobnikami. Wymiana genów następuje w sposób taki, iż dla losowo wybranego punktu krzyżowania (genu w chromosomie) następuje "rozciecie" osobników i wymiana części chromosomów pomiędzy osobnikami. Dla zapewnienia możliwości wyjścia z podzbioru rozwiązań generowanych przez populację początkową wprowadza się do populacji nowe osobniki na drodze zmiany losowej pewnej liczby genów w chromosomie. Zjawisko to, jak w genetyce, nazywane jest mutacją. Operacje: oceny populacji, selekcji, krzyżowania i mutacji przeprowadza się cyklicznie do osiągnięcia satysfakcjonującego rozwiązania. Rozwiązanie określa najlepiej przystosowany osobnik z populacji końcowej. Warunkiem zakończenia obliczeń może być przykładowo brak postępów w ocenie osobników (wartość funkcji przystosowania nie zmienia się).

Przedstawiony wyżej algorytm można zapisać następująco:

```
wybór populacji początkowej; // rozwiązań ze zbioru rozwiązań dopuszczalnych
ocena populacji; // obliczenie wart. funkcji przystosowania dla wszystkich osobników
while (not stop)
{
selekcja; // określenie populacji rodzicielskiej
krzyżowanie; // wymiana genów
mutacja; // losowa modyfikacja genów (z małym prawdopodobieństwem)
ocena populacji;
}
```

#### 4. Rozwiązanie zadania optymalizacji struktury sieci neuronowej

Formułujemy zadanie optymalizacji struktury sieci neuronowej jako dobór liczby neuronów w warstwach ukrytych sieci neuronowej. Warstwa wyjściowa nie podlega optymalizacji z tej racji, iż jest warstwą sygnalizacyjną (decyzyjną) (rys.7).



Rys. 7. Określenie części sieci neuronowej podlegającej optymalizacji struktury

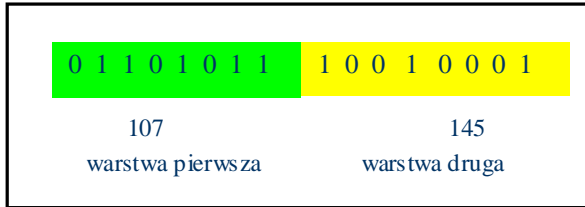
#### Kodowanie struktury sieci neuronowej

Na potrzeby algorytmu genetycznego istnieje potrzeba zakodowania struktury sieci neuronowej (warstw ukrytych sieci) w postaci liczby binarnej. Przyjęto, że struktura sieci zostanie zakodowana przy pomocy liczby binarnej o długości  $k*N$ , gdzie:

- $k$  - liczba warstw ukrytych sieci,
- $N$  - liczba bitów na których zakodowana jest liczba neuronów w jednej warstwie.

Przykładowo, dla  $k=2$ ,  $N=8$  kodujemy strukturę sieci o maksymalnie dwóch warstwach ukrytych i maksymalnej liczbie neuronów w warstwie równej 255 (rys.8). W takim przypadku liczebność zbioru rozwiązań dopuszczalnych wynosi:

$$2^{2*8} - 256 = 65280$$



Rys. 8. Sposób kodowania struktury sieci neuronowej w postaci liczby binarnej

### Określenie funkcji przystosowania

W celu wykorzystania metody algorytmów genetycznych należy określić funkcję przystosowania (kryterium oceny rozwiązania). Ocena jakości działania sieci może być przeprowadzona w zasadzie na podstawie oceny skuteczności np. stopy błędu w zadaniu rozpoznawania. Jednakże złożoność obliczeniowa pełnego procesu treningu jest na tyle duża, że dyskwalifikuje ten sposób oceny.

Przyjmijemy ocenę jakości sieci poprzez ocenę postępu treningu po  $L$  epokach, mierzonego wartością SSE, z dodatkową karą za dużą złożoność sieci. Dla dwóch warstw ukrytych funkcję tę określa wzór:

$$f(n1, n2) = SSE_L(n1, n2) + \alpha \frac{n1 + n2}{2(2^N - 1)} SSE_L(n1, n2), \quad 0 < \alpha < 1$$

gdzie:

- $n1, n2$  - liczby neuronów w warstwach pierwszej i drugiej,
- $SSE_L$  - sumaryczny średniokwadratowy błąd na wyjściu sieci osiągnięty po  $L$  epokach treningu,
- $2(2^N - 1)$  - maksymalna liczba neuronów w sieci,
- $N$  - liczba bitów na których kodowana jest liczba neuronów w warstwie.

### Wyniki obliczeń

W przeprowadzonym eksperymencie przyjęto następujące parametry dla



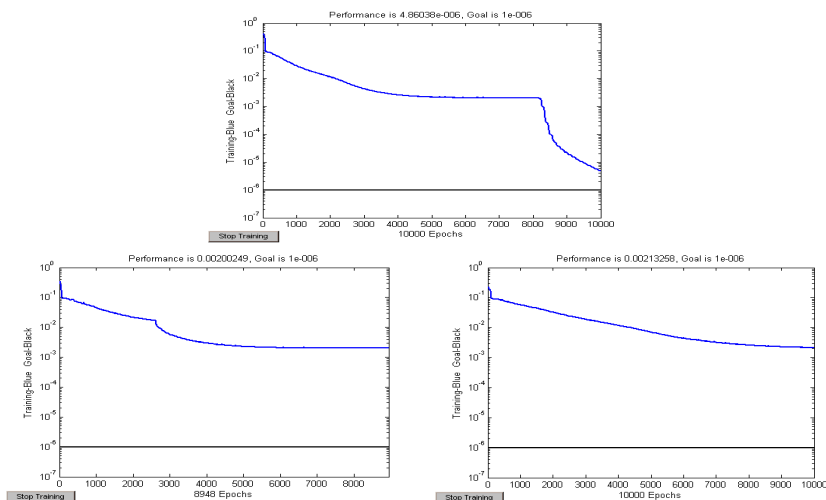
algorytmu genetycznego:

- wielkość populacji – 15,
- liczba iteracji algorytmu – 10,
- metoda selekcji – turniejowa.

W wyniku optymalizacji dla:

- 2 warstw ukrytych sieci neuronowej,
- maksymalnej liczby neuronów w warstwie równej 255,
- liczby epok treningu  $L=1000$  ,
- współczynnika  $\alpha = 0.2$  ,

otrzymano strukturę sieci o  $n1=10$  neuronach w pierwszej warstwie ukrytej oraz  $n2=60$  neuronach w warstwie drugiej.



**Rys.9. Przebieg procesu uczenia sieci zwycięskiej [10 60 10] (przebieg górny) oraz sieci o strukturach [30 100 10] i [100 30 10], wykresy przedstawiają spadek błędów SSE w trakcie ok. 10000 epok**

Na rysunku 9 przedstawiono przebiegi procesu uczenia sieci o strukturze będącej wynikiem działania algorytmu genetycznego na tle innych sieci, o strukturach bardziej złożonych. Sieci, zgodnie z przyjętą funkcją

przystosowania, były oceniane po 1000 epokach. Osiągnięty błąd po tej liczbie epok stanowił jeden składnik oceny (drugi stanowiła kara za złożoność sieci). Przedstawione przebiegi procesu uczenia na odcinku ok. 10000 epok potwierdzają słuszność przyjętego sposobu oceny sieci. Sieć oceniona jako zwycięska na podstawie postępów w treningu w pierwszym tysiącu epok, uzyskała najlepszy wynik po dziesięciokrotnie dłuższym procesie uczenia.

## 5. Podsumowanie

Przedstawiona w artykule metoda optymalizacji struktury sztucznej sieci neuronowej oparta na algorytmie genetycznym przynosi zadowalające rezultaty.

Dla zadania klasyfikacji cyfr pisanych odręcznie realizowanego przez sieć neuronową najlepszą strukturą, będącą wynikiem działania algorytmu genetycznego, okazała się sieć o dwóch warstwach ukrytych i strukturze [10 60 10]. W omówionym przypadku zrezygnowano z oceny sieci neuronowej na podstawie stopy błędnych rozpoznań cyfr, co byłoby miarą najlepszą lecz czasochłonną. Wybrano sposób oceny sieci odzwierciedlający podatność na uczenie, mierzoną błędem SSE, po ustalonej liczbie epok treningu.

## Literatura:

- [1] Goldberg D.E.: *Algorytmy genetyczne i ich zastosowani*, WNT, 1995.
- [2] Kwiatkowski W.: *Metody automatycznego rozpoznawania wzorców*, WAT, 2001.
- [3] Osowski S.: *Sieci neuronowe do przetwarzania informacji*, Oficyna Wydawnicza Politechniki Warszawskiej, Warszawa, 2000.
- [4] Tadeusiewicz R.: *Sieci neuronowe*, Akademicka Oficyna Wydawnicza RM, Warszawa 1993

## **An example of feed forward neural network structure optimisation with genetic algorithm**

ABSTRACT: An example of feed forward neural network structure optimisation with genetic algorithm is presented. In genetic algorithm an original fitness function is applied. All calculations have been realized for a feed forward neural network, which recognizes hand-written signs.

KEYWORDS: Neural network, genetic algorithm, optimisation.

*Recenzent: prof. dr hab. inż. Włodzimierz Kwiatkowski*

*Praca wpłynęła do redakcji: 20.12.2006*