

# Sprawozdanie

## Uczenie nadzorowane SNN

02.10.2009

Karol Bonenberg

### Zad. 5.2

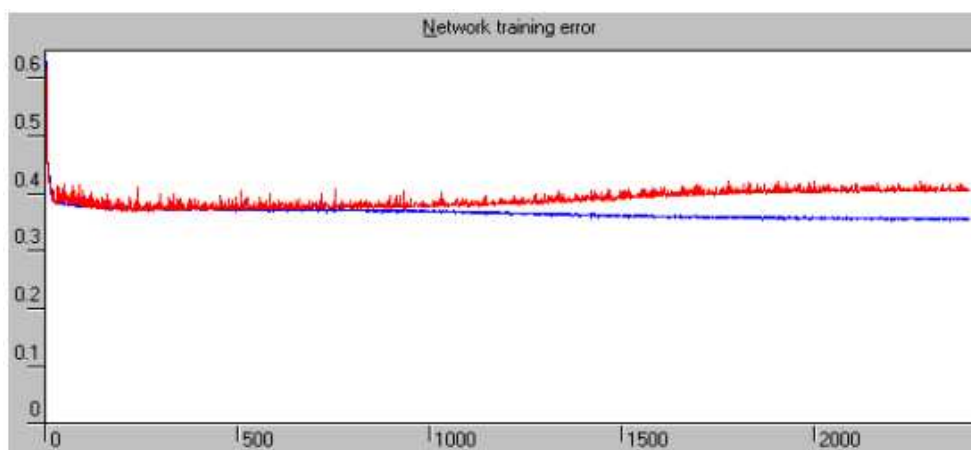
Wczytanie zbioru PIMA i skonstruowanie dla niego dwuwarstwowej sieci nieliniowej o architekturze 8-4-1. Uczenie algorytmem wstecznej propagacji błędu przeprowadzono robiąc pomiary co mniej więcej 300 epok uzyskując poniższe wyniki:

Liczba epok	Zbiór uczący			Zbiór weryfikujący		
	Trafność klasyfikowania	Błąd klasyfikowania	% niezaklasyfikowanych	Trafność klasyfikowania	Błąd klasyfikowania	% niezaklasyfikowanych
300	0,797	0,202572347	0	0,797101449	0,202898551	0
600	0,788	0,21221865	0	0,753623188	0,246376812	0
900	0,804	0,196141479	0	0,768115942	0,231884058	0
1200	0,801	0,199356913	0	0,768115942	0,231884058	0
1500	0,809	0,191318328	0	0,753623188	0,246376812	0
1800	0,828	0,172025723	0	0,753623188	0,246376812	0
2100	0,826	0,173633441	0	0,753623188	0,246376812	0
2400	0,822	0,178456592	0	0,753623188	0,246376812	0





Z obserwacji wynika, że trafność klasyfikowania na zbiorze uczącym w nieznaczny sposób wzrasta wraz ze zmniejszaniem się błędu klasyfikowania. Z obserwacji na zbiorze weryfikującym widać, że trafność klasyfikowania nieznacznie maleje w czasie, gdy błąd klasyfikowania rośnie. Sytuację można wytłumaczyć pojawieniem się błędu przeuczenia, co potwierdza wykres Training Error Graph z programu StatisticaNN:



### Zad 5.3

Dla zbioru IRIS przeprowadzono po 5 eksperymentów uczenia dla każdej z sieci neuronowych o architekturze 4-n-3 (n=2..10) wg polecenia. Uśrednione dane zamieszczono poniżej.

n	Zbiór uczący			Zbiór weryfikujący		
	Trafność klasyfikowania	Błąd klasyfikowania	% niezaklasyfikowanych	Trafność klasyfikowania	Błąd klasyfikowania	% niezaklasyfikowanych
2	0,908	0,000	0,092	0,900	0,000	0,100
3	0,915	0,002	0,083	0,893	0,000	0,107
4	0,918	0,000	0,082	0,893	0,000	0,107
5	0,922	0,000	0,078	0,913	0,013	0,073
6	0,930	0,000	0,070	0,920	0,027	0,053
7	0,917	0,003	0,080	0,907	0,013	0,080
8	0,928	0,000	0,072	0,920	0,013	0,067
9	0,922	0,000	0,078	0,907	0,013	0,080
10	0,937	0,000	0,063	0,920	0,020	0,060

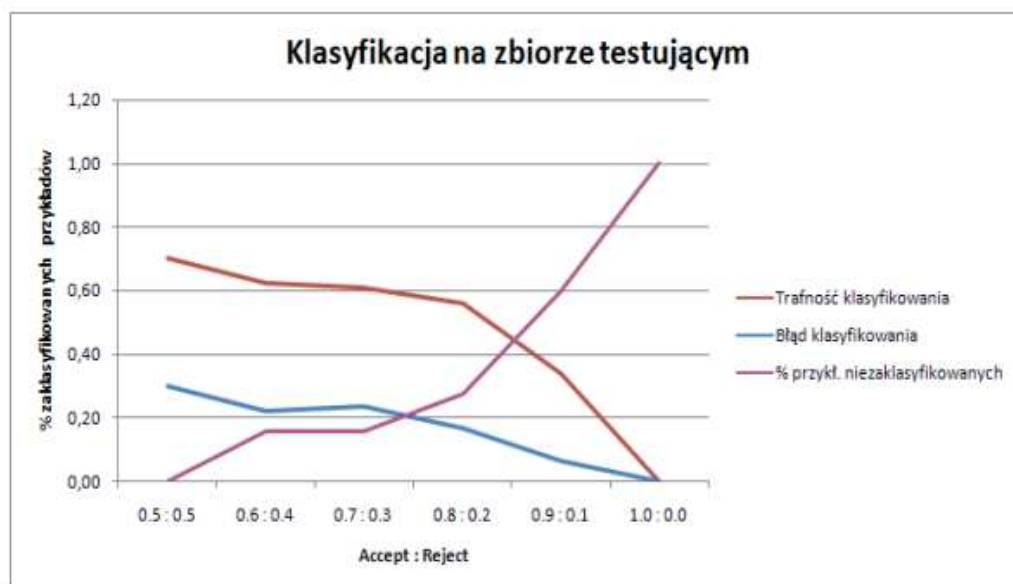


Błąd klasyfikowania dla zbioru klasyfikującego był przez większy czas równy zeru. Dlatego też biorąc pod uwagę jedynie przebieg błędu klasyfikowania dla zbioru uczącego nie można ustalić tutaj optymalnej liczby neuronów w warstwie ukrytej. Dla zbioru weryfikującego błąd klasyfikowania był równy zeru jedynie dla  $n < 5$ . A więc w tym przypadku duża liczba neuronów w drugiej warstwie sieci nie jest korzystna. Trafności klasyfikowania wzrasta wraz ze wzrostem liczby  $n$ .

#### Zad 6.1

Stworzono i nauczono sieć na zbiorze PIMA w architekturze 8-4-1. Następnie przeprowadzono klasyfikację przykładów ze zbioru testującego. Ustawiono wartości Accept i Reject i dla zbioru testującego sporządzono wykres przebiegów przypadków poprawnie zaklasyfikowanych, niepoprawnie zaklasyfikowanych oraz niezaklasyfikowanych.

Accept	Reject	Trafność klasyfikowania	Błąd klasyfikowania	% niezaklasyfikowanych
0,50	0,50	0,70	0,30	0,00
0,60	0,40	0,62	0,22	0,16
0,70	0,30	0,61	0,23	0,16
0,80	0,20	0,56	0,17	0,27
0,90	0,10	0,34	0,06	0,60
1,00	0,00	0,00	0,00	1,00

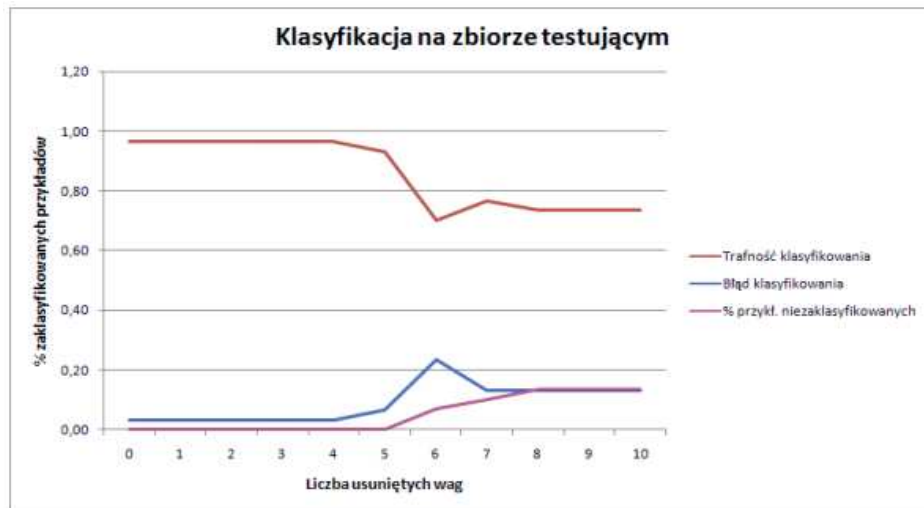


Na podstawie otrzymanego wykresu zasugerowałbym, jako optymalną wartość obu progów 0,5. Z danych wynika, że najwyższą trafność klasyfikacji zanotowano w przypadku, gdy wartości parametrów *Accept* i *Reject* wynosiły 0.5. Wraz z dalszym ich modyfikowaniem trafność klasyfikacji malała monotonicznie.

## Zad 6.2

Aby zbadać odporność sieci na uszkodzenia skonstruowano dwuwarstwową sieć o architekturze 4-3-3, a następnie nauczono ją metodą back propagation. Kolejno usuwano kolejne wagi notując trafność klasyfikowania. Wyniki zamieszczono w poniżej:

Liczba usuniętych wag	Trafność na zbiorze testującym	błąd klasyfikowania	% niezaklasyfikowanych
0	0,97	0,03	0,00
1	0,97	0,03	0,00
2	0,97	0,03	0,00
3	0,97	0,03	0,00
4	0,97	0,03	0,00
5	0,93	0,07	0,00
6	0,70	0,23	0,07
7	0,77	0,13	0,10
8	0,73	0,13	0,13
9	0,73	0,13	0,13
10	0,73	0,13	0,13



Odporność można do pewnego momentu uznać za wysoką. Gdy usunięto szóstą wagę trafność klasyfikacji znacząco się pogorszyła. Na podstawie tak zredukowanej sieci można wysnuć pewne wnioski na temat ważności poszczególnych atrybutów w dostarczaniu informacji. Wejścia neuronów w drugiej warstwie pochodzące od kryterium SLENGTH miały niskie wagi. Po redukcji sześciu wag, atrybut SLENGTH nie miał już wpływu na wynik (wagi wyzerowane). Spadek jakości klasyfikacji wskazuje, że atrybut ten dostarczał dość istotną informację do sieci. Po dokonaniu 9. redukcji wszystkie wagi atrybutu wejściowego SWIDTH były wyzerowane. Jednak trafność klasyfikacji zdołała się utrzymać. Stąd można by wnioskować, że atrybut ten prawdopodobnie nie wnosił istotnej informacji. W konsekwencji „przerzedzenia” sieci można usunąć pewne neurony, w tym wypadku prawdopodobnie można by usunąć np. neuron w warstwie pierwszej odpowiedzialny za wprowadzanie do sieci informacji o atrybucie SWIDTH.

### Zad 6.3

Eksperymentalny dobór rozmiaru zbioru weryfikującego wykonano zmieniając proporcje między zbiorami uczącymi i weryfikującymi wg polecenia. Zgodnie z poleceniem liczba epok jest większa od 300.

x	Trafność klasyfikowania	Błąd klasyfikowania	% niezaklasyfikowanych
0,11	0,75	0,25	0,00
0,22	0,71	0,29	0,00
0,33	0,75	0,25	0,00
0,44	0,68	0,32	0,00
0,56	0,66	0,34	0,00
0,67	0,73	0,27	0,00
0,78	0,65	0,35	0,00
0,89	0,66	0,34	0,00



Z powyższych danych wynika, że najlepszymi z przebadanych konfiguracji pod względem trafności klasyfikacji są konfiguracje 8:1:1 oraz 6:3:1 (zbiór uczący : weryfikujący : testujący). Niestety przedstawione zależności nie są monotoniczne na dłuższych odcinkach, dlatego moim zdaniem nie można wyciągnąć jednoznacznych wniosków.

### Zad. 6.5

Dla problemu IRIS zbudowano sieć RBF. Czas uczenia był zdecydowanie krótszy niż w przypadku sieci typu *Multilayer Perceptron*. W sieciach typu RBF ważnym parametrem są promienie, które decydują o stopniu „spłaszczenia” funkcji Gaussowskich. Jeżeli funkcja Gaussowska jest zbyt „spłaszczona”, neuron RBF może odpowiadać dla prawie każdego wejścia, nawet gdy wektor wymuszeń nie pokrywa się zbyt z wektorem wag, jednakże jego odpowiedzi nie będą silne. Maksymalna wartość odpowiedzi (gdy wektor wymuszeń pokrywa się z wektorem wag) również nie będzie duża. Gdy funkcja Gaussowska będzie zbyt „wysoka”, neuron RBF będzie dawał odpowiedź jedynie dla wektora wymuszeń bardzo zbliżonego do wektora wag. Ponadto wymuszenie to będzie silne.