

Sprawozdanie

Scenariusz 1

Zadanie: Implementacja sztucznego neuronu, wygenerowanie danych uczących i testujących wybranej funkcji logicznej dwóch zmiennych oraz uczenie perceptronu dla różnej liczby danych uczących i testowanie perceptronu.

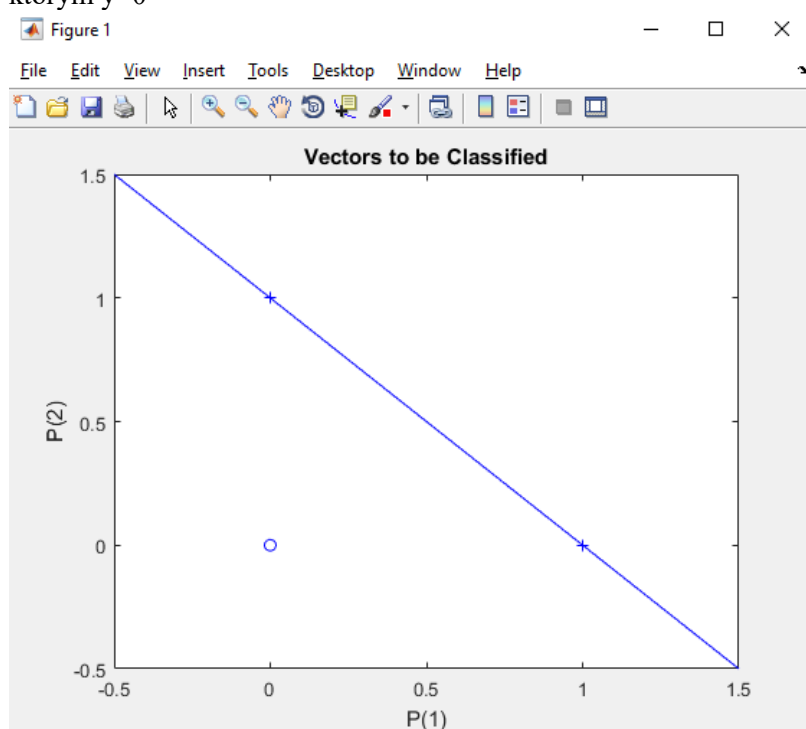
Do wykonania zadania użyłem pakietu MATLAB oraz biblioteki narzędziowej „Neural Network Toolbox”, która posiadała funkcje pozwalające na implementację perceptronu. Jako funkcję logiczną wybrałem bramkę logiczną OR.

Do nauki perceptronu użyłem funkcji train, która domyślnie wykorzystuje algorytm Levenberga-Marquardta. Cechuje się on szybkością działania i dużym zapotrzebowaniem na pamięć. Algorytm zmienia przy każdej iteracji wagi synaptyczne, które domyślnie mają wartość 0.

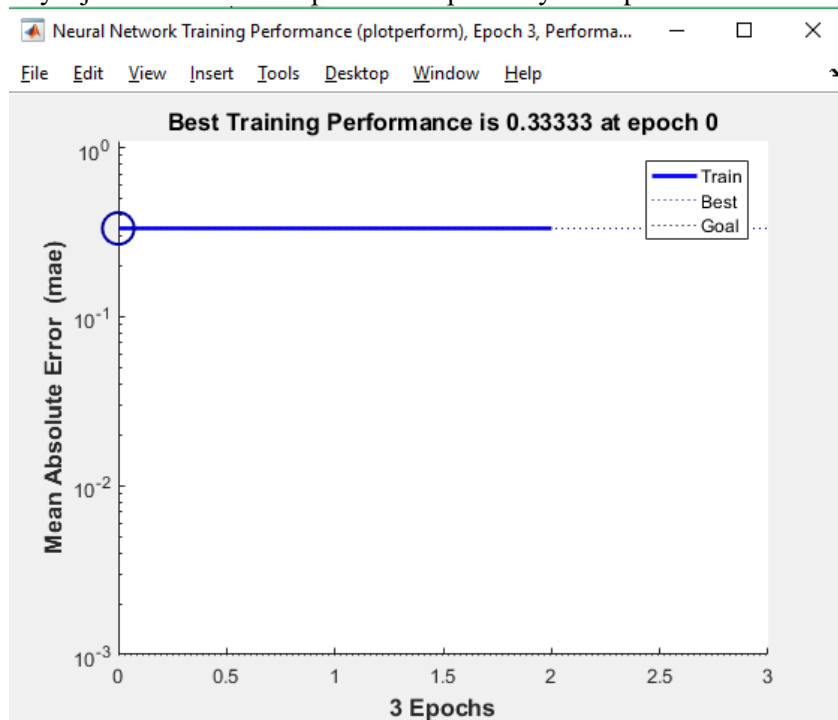
Wyniki:

-Do poprawnego działania perceptron potrzebował w sumie 9 danych , mniejsza ilość powodowała, że otrzymane wyniki nie zawsze były poprawne.

-do osiągnięcia poprawnych wyników potrzeba było nie mniej niż 3 epok treningowych, przy mniejszych ilościach zdarzały się błędy w wynikach. 3 epoki to również minimalna ilość dla której poprawnie została wyznaczona granica decyzyjna. Poniżej wykresy z wartościami dla poszczególnych x_1 i x_2 oraz granica decyzyjna: krzyżyki oznaczają miejsca, dla których wartość y wyniesie 1 natomiast koło to miejsce w którym $y=0$



-wydajność nauczania na przestrzeni pierwszych 3 epok:



Wnioski:

- aby działać poprawnie perceptron nie potrzebował wprowadzenia wszystkich wariantów.
- najlepsze efekty podczas uczenia miały miejsce przy domyślnych wagach. Zapewne wynika to z ich niskiej wartości (domyślnie 0) ponieważ zauważyłem że przy zwiększaniu wartości wag spada efektywność nauki.
- zastosowanie niskiego współczynnika nauki poprawia jakość uczenia natomiast wysoki współczynnik może doprowadzić do niepoprawnego działania perceptronu

Kod programu znajduje się w pliku OR.m