Podstaw sztucznej inteligencji-projekt 1.

Program został wykonany w języku C++. Jego celem jest uczenie, a następnie próba odgadnięcia wartości funkcji Boolowskiej AND składającej się z dwóch zmiennych.

Próg aktywacji ustawiony jest na wartość 1. Jeśli wynik sumowania iloczynów sygnałów wejściowych i wag przekroczy wartość 1, wtedy następuje aktywacja perceptronu.

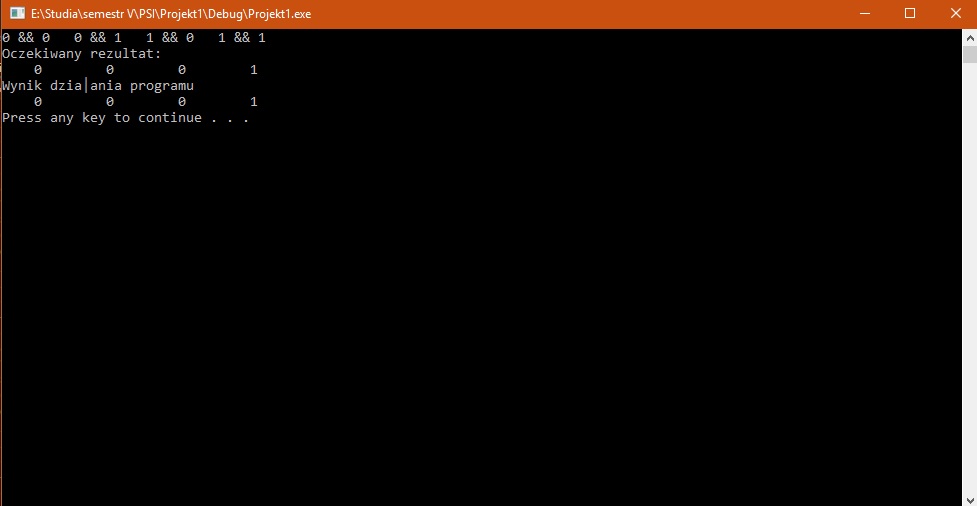
Regulacji ulegają: współczynnik uczenia, ilość danych uczących, a co za tym idzie-ilość danych testujących oraz ilość cykli uczenia.

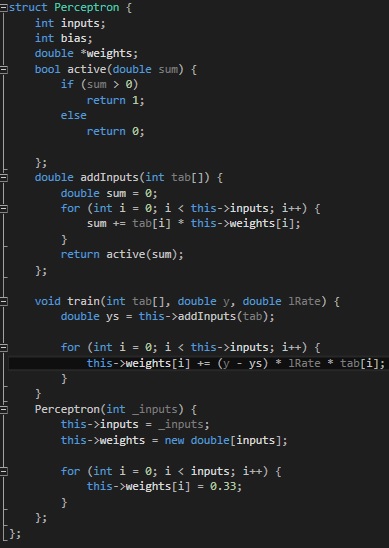
Dla funkcji logicznej dwóch zmiennych możliwe są cztery scenariusze. Dla iloczynu logicznego przedstawia się to następująco:

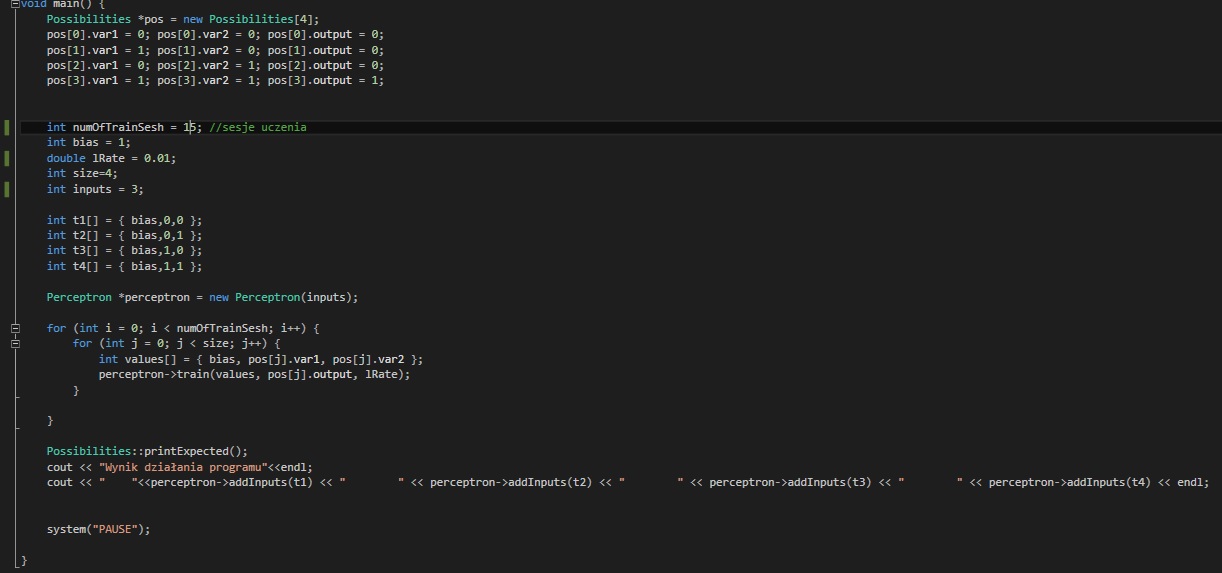
|  |  |  |
| --- | --- | --- |
| Wartość 1 | Wartość 2 | Wynik |
| 1 | 1 | 1 |
| 1 | 0 | 0 |
| 0 | 1 | 0 |
| 0 | 0 | 0 |

Początkową wagę ustaliłem losowo na wartość 0.33. W dalszej części programu zmienia się ona zgodnie z zasadą Widrow-Hoffa.

Najbliższym ideału współczynnikiem uczenia było 0.1. Zmiejszenie, lub zwiększenie go skutkowały gorszymi wynikami, dość proporcjonalnie do różnicy tej wartości. Przy ustawieniu 2 danych jako uczące oraz 3 sesjach nauki trafność programu zawsze wynosiła 100%. Przy ustawieniu 2 danych jako uczące, program miał 50% skuteczności, przy 1 i 4-25%.

Gdy współczynnik uczenia wynosił 1 wyniki pogorszyły się. Program był w stanie podać prawidłową konfigurację tylko, gdy daliśmy mu 3 wiersze do nauki. Przy współczynniku 0.01 i 3 wierszach uczących program nadal potrafił nauczyć się funkcji.Screen z działania programu.





Wnioski:

Wbrew możliwemu początkowemu przykonaniu, wyższy współczynnik uczenia nie jest proporcjonalny do ilości prawidłowych odpowiedzi. Jednoznacznie zaprzecza temu test, który wykonałem dla wsp 0.1, kiedy to procent prawidłowych odpowiedzi wynosił 100%, oraz dla wsp 1, kiedy procent prawidłowych odpowiedzi wynosił 75%.