Daniel Wysowski  
Nr. albumu: 286136  
Inżynieria Obliczeniowa   
Wydział Inżynierii Metali i Informatyki Przemysłowej   
Akademia Górniczo-Hutnicza w Krakowie

**Sprawozdanie z przedmiotu: Podstawy sztucznej inteligencji**

**Scenariusz 1 – Budowa i działanie perceptronu**

# **Spis treści:**

1. Wstęp teoretyczny
2. Cel ćwiczenia
3. Wykonanie zadania
4. Podsumowanie – wyniki i wnioski

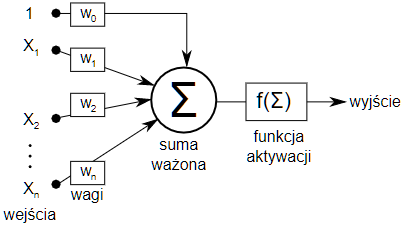
I.WSTĘP TEORETYCCZNY

**Sieć neuronowa** – model matematyczny inspirowany budową naturalnych neuronów znajdujących się w mózgu człowieka. Modele takie składają się z neuronów które to realizują podstawowe obliczenia na swoim wejściu i nazywamy je sztucznymi neuronami.

Systemy oparte o sieci neuronowe wykazują zdolności uczenia się, na podstawie przykładów uczymy taką sieć aby następnie ona już samodzielnie analizowała nowe dane przedstawione w innym lecz analogicznym kontekście.

Sieć neuronowa spisuje się świetnie w zastosowaniach przy których klasyczny komputer nie mógłby sobie poradzić tzn. Uogólnienie posiadanej wiedzy na inne przypadki włącznie z wymyśleniem nowego sposobu rozwiązania nieznanego jej dotychczas problemu, odporność na uszkodzenia – niepełne, błędne dane, sprawne działanie nawet gdy część jej elementów nie działa poprawnie lub nie działa w ogóle. Sieci neuronowe nadają się idealnie do rozwiązywania problemów przy których klasyczny język programowania nie daje sobie rady tzn. brakuje odpowiedniego algorytmu.

**Neuron McCullocha-Pittsa** – jest to jeden z matematycznych modeli neuronu oraz podstawowy budulec wspomnianej wyżej sieci neuronowej (perceptron). Posiada wiele wejść gdzie każde ma swoją wagę (tzw. Wagę wejścia), oraz jedno wyjście którego waga jest obliczona dodatkowo za pomocą wzoru (funkcja aktywacji).



**Perceptron** – jest to podstawowa sieć neuronowa składająca się z jednego lub wielu neuronów McCullocha-Pittsa. Perceptron jest w stanie określić przynależność parametrów wejściowy do jednej z dwóch klas lecz jego ogranicznikiem jest fakt ze może zrobić to wyłącznie dla klas liniowo separowalnych.

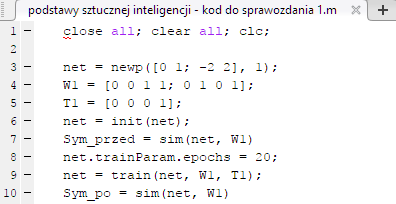
II. CEL ĆWICZENIA

Celem ćwiczenia było poznanie budowy i działania perceptronu poprzez implementację oraz uczenie perceptronu realizującego wybraną funkcję logiczną dwóch zmiennych. Zadanie jakie mieliśmy wykonać było oparte o implementację sztucznego neuronu w środowisku MATLAB, następnie mieliśmy przygotować dla niego dane pozwalające mu się uczyć (odbywało się to ze zmienną liczbą danych uczących). Ostatnim etapem ćwiczeń który mieliśmy przygotować był test pracy perceptronu.

III. WYKONANIE ZADANIA

Poniżej zamieściłem listing całego kodu który przedstawia wywołanie funkcji odpowiedzialnej za tworzenie sieci składającej się z pojedynczego neuronu o dwóch wejściach.

Zakres wartości podawanych na pierwszym wejściu to [0,1], na drugim [-2,2].



ANALIZA KODU:

Kod:   
net=newp([0 1; -2 2], 1);

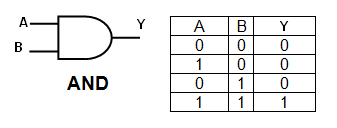
Ta linia kod odpowiedzialna jest za tworzenie wspomnianego wcześniej pojedynczego neuronu o podanych przedziałach wartości tzn. wejście pierwsze [0,1] oraz drugie [-2,2].

Kod:

W1 = [0 0 1 1; 0 1 0 1];

T1 = [0 0 0 1];

Ta część kodu odpowiedzialna jest za tworzenie wektorów W1 które opisują wartości na wejściach oraz T1 który opisuje wartości wynikowe w ten sposób otrzymujemy zestaw trzech wektorów które opisują działanie bramki logicznej **AND** jej schemat prezentuje się następująco:



Kod:  
net = init(net);

Ta linia kodu odpowiedzialna jest za inicjalizację sieci perceptronowej w której wartości wag wejść i progów są losowe. Wykorzystywana jest do tego funkcja „init”.

Kod:

Sym\_przed = sim(net, W1)

Ta linia kodu odpowiedzialna jest za wywołanie symulacji przed treningiem. Używana jest do tego funkcja „sim”.

Kod:

net.trainParam.epochs = 20;

Ta linia kodu określa liczbę epok, które należy wykonać w treningu sieci.

Kod:

net = train(net, W1, T1);

Za pomocą funkcji train został uruchomiony trening naszego perceptronu.

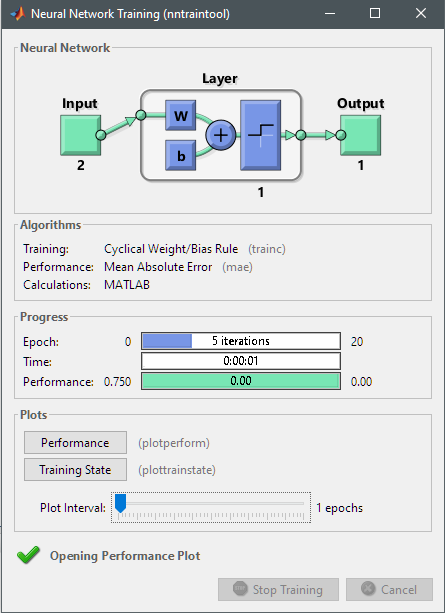
Kod:

Sym\_po = sim(net, W1)

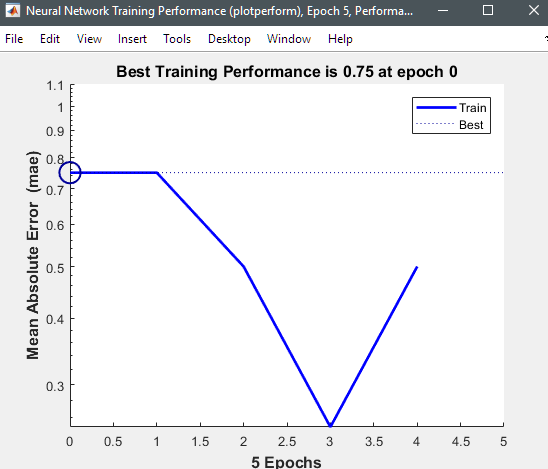
Końcowa symulacja sieci neuronowej dla wartości które były wyznaczone podczas treningu.

IV.PODSUMOWANIE

Proces uczenia się trwał tylko 5 epok, osiągając wydajność na poziomie 0.750. Możemy to zaobserwować na zdjęciu interfejsu znajdującym się poniżej.



Po uzyskaniu wykresu uczenia się można dostrzec że w epoce zerowej MAE czyli ilość błędów pomiarowych, wynosi 0.75 jest to wartość stabilna aż do iteracji pierwszej. Na wykresie możemy dostrzec że w epoce trzeciej sieć osiąga wartość minimalną błędu czyli zostaje nauczona. W późniejszych epokach błąd rośnie ponieważ zachodzi tzw. Zjawisko przeuczenia się jest to moment w którym nasza sieć neuronowa nazbyt dopasowuje się do konkretnych danych uczących. Wszystko to możemy zaobserwować na wykresie poniżej:



Celem tego projektu było zapoznanie się z siecią neuronową (perceptronem). Naszym zadaniem było przeprowadzić proces nauki perceptron, osobiście wybrałem bramkę logiczną AND. Z moich obserwacji wynika iż sieć została nauczona do trzeciej epoki później wystąpiło przetrenowanie. Dzięki tym ćwiczeniom zrozumiałem budowę perceptronu, zasadę działania prostej sieci neuronowej i jej budowę, zamierzony cel został osiągnięty ponieważ sieć osiągnęła maksymalny stan ‘wytrenowania’.