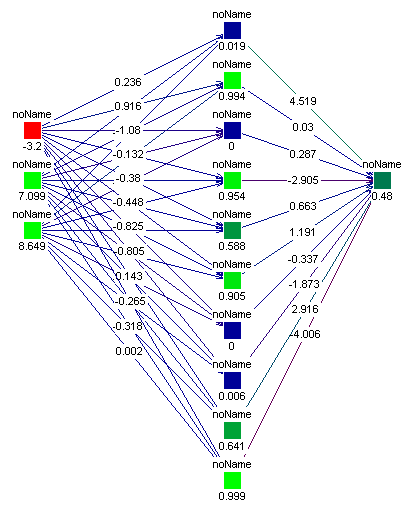
1. **Cel ćwiczenia**

Celem ćwiczenia była aproksymacja funkcji wielu zmiennych za pomocą jednokierunkowej sieci neuronowej. Do realizacji zadania wybrano funkcję trzech zmiennych o wzorze

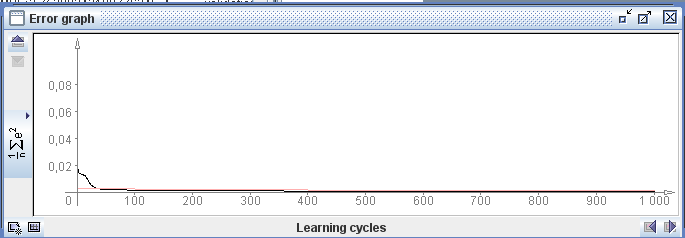
 o wartościach z przedziału . Liczy losowe zmiennoprzecinkowe wygenerowano w programie Microsoft Excel. 1500 przykładów zostało podzielonych na 3 ciągi: ciąg uczący, ciąg walidacyjny oraz ciąg testujący, każdy odpowiednio po 500 przykładów. Funkcja aktywacji sieci jest sigmoidalna, a więc dla wartości 1 dąży do plus nieskończoności, a dla wartości 0 dąży do minus nieskończoności. W celu prawidłowego przebiegu procesu uczenia sieci niezbędne było przeprowadzenie normalizacji danych zgodnie ze wzorem . Wartości sieci mieszczą się w przedziale . Sieć nie osiągnie jednak wartości granicznych, dlatego wartości minimalne i maksymalne y zostały wyznaczone zgodnie ze wzorami . Wartości ymin i ymax zostały użyte do normalizacji wszystkich 3 ciągów zostały pobrane z ciągu uczącego, ponieważ wartości z ciągów walidacyjnego i testowego nie są teoretycznie znane dla sieci podczas jej tworzenia. Otrzymane wartości zostały zapisane w pliku .txt z odpowiednim nagłówkiem obsługiwanym przez środowisko Java.

1. **Tworzenie sieci**

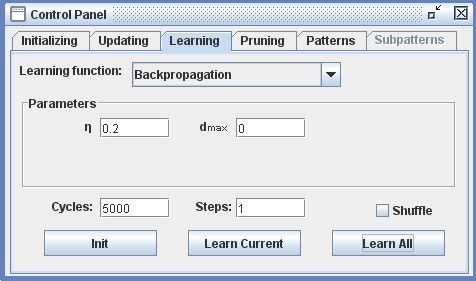
Po znormalizowaniu wartości została utworzona sieć jednokierunkowa w programie JavaNNS. Struktura sieci prezentuje się następująco: 3 neurony w warstwie wejściowej, 10 neuronów w warstwie ukrytej i 1 neuron w warstwie wyjściowej. Neurony zostały połączone ze sobą za pomocą feed-forward, a więc neurony z poprzedniej warstwy łączą się z neuronami z warstwy następnej na zasadzie każdy z każdym i nie występują w tej sieci sprzężenia ani rekurencja. Sieć uczy się za pomocą algorytmu propagacji wstecznej. Podstawowymi parametrami do konfiguracji sieci są krok uczenia, dopuszczalny błąd propagacji, liczba cykli oraz określenie co ile kroków ma nastąpić modyfikacja wag. Istnieje również możliwość podania elementów ciągu uczącego w sposób losowy. Sieć została zbudowana w oparciu o ciąg uczący oraz ciąg walidacyjny. Wartości w ciągu uczącym są znane. Ciąg walidacyjny (estymator obciążony) służy do oceny procesu uczenia sieci i weryfikacji wyników. Za pomocą przycisku Init następuje inicjalizacja doboru wag, a przycisk Learn All uruchamia proces uczenia sieci. Następnie przebieg procesu w postaci wizualizacji błędu SSE (suma błędów kwadratowych) lub MSE (średni błąd kwadratowy) można podejrzeć na wykresie. Tak utworzona sieć została zapisana z rozszerzeniem .net. Wyniki uczenia zapisano do pliku tekstowego z rozszerzeniem .res.



Rys. 1. Wizualizacja sieci 3-10-1 wraz z dobranymi wagami



Rys. 2. Wykres błędu MSE ciągu uczącego i walidacyjnego (czerwony) podczas utworzenia sieci



Rys. 3. Parametry dobrane podczas tworzenia sieci

1. **Zestawienie wyników**

Kolejnym krokiem po utworzeniu sieci było sprawdzenie jak sieć radzi sobie z uczeniem ciągu walidacyjnego oraz ciągu testowego. W celu oceny jakości uczenia dla każdego ciągu obliczono błąd MSE oraz współczynnik korelacji R zgodnie ze wzorami:

gdzie yi – wartość wyjściowa, di – wartość oczekiwana. Wykorzystano do tego program napisany w języku C++ wczytujący dane zapisane w postaci pliku .txt. Wyniki przebiegów zestawiono w tabeli.

Tab. 1. Zestawienie oceny parametrów jakości uczenia sieci

|  |  |  |
| --- | --- | --- |
|  | Błąd MSE | Współczynnik korelacji |
| Ciąg uczący | 2.24113\*10-4 | 0.995359 |
| Ciąg walidacyjny | 1.10443\*10-3 | 0.993997 |
| Ciąg testujący | 8.08588\*10-4 | 0.994425 |

1. **Wnioski**

W celu prawidłowego przebiegu procesu uczenia sieci niezbędna jest normalizacja wartości wyjściowych. Do realizacji zadania wykorzystano sieć jednokierunkową o strukturze 3-10-1 z algorytmem propagacji wstecznej. Najlepszymi parametrami podczas uczenia tej sieci okazały się krok uczenia = 0.2, dopuszczalny błąd propagacji = 0, liczba cykli = 5000 z modyfikacją wag co 1 krok. W celu utworzenia sieci wykorzystano ciąg walidacyjny, który pozwala przewidzieć wynik uczenia się sieci. Do określenia oceny jakości uczenia zastosowano obliczenie błędu MSE oraz współczynnika korelacji R. Wartość błędu MSE jest dzielona przez ilość elementów w ciągu oraz wynika z błędu podniesionego do kwadratu. Zatem im mniejszy będzie błąd, tym mniejsza będzie wartość MSE. W podanym ćwiczeniu najmniejszym błędem charakteryzował się ciąg uczący, a największym ciąg walidacyjny (o rząd większy błąd niż dla ciągu uczącego). Współczynnik korelacji pozwala określić w jaki sposób zmienne są od siebie współzależne. Idealny współczynnik korelacji powinien wynosić 1. Jak widać z analizy tabeli wyników, najlepszą korelacją charakteryzował się ciąg uczący, a najsłabszą walidacyjny. Różnice nie są jednak niewielki i wszystkie ciągi osiągnęły prawie wartość zbliżoną do 1. Analizując wyniki stwierdzono, że proces uczenia sieci przebiegł poprawnie.