

Metody identyfikacji i dostrajania modeli rozmytych

Autor: Stanisław Swianiewicz
Opiekun naukowy: dr inż. Piotr Marusak

2. grudnia 2010 r.

- 1 Modelowanie rozmyte
- 2 Rozmyte sieci neuronowe
- 3 Algorytmy ewolucyjne w strojeniu modeli rozmytych
 - Podejście oparte na dyskretyzacji
 - Podejście z genotypem ciągłym
- 4 Implementacja

Modele rozmyte Takagi-Sugeno-Kanga

- Obszar zmienności wejść modelu dzielony na zbiory rozmyte
- Funkcje przynależności
- Reguły wnioskowania
- Liniowe modele lokalne

Modele rozmyte Takagi-Sugeno-Kanga

Zbiór reguł — baza wiedzy

R^1 : JEŚLI x_1 jest X_{11} i x_2 jest X_{21} to $y^{11} = f_{11}(x)$

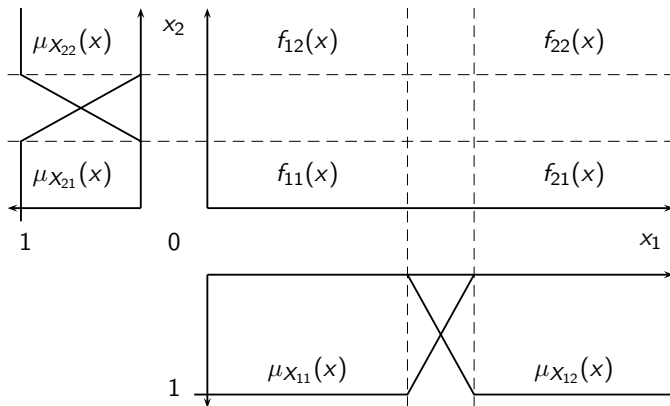
R^2 : JEŚLI x_1 jest X_{11} i x_2 jest X_{22} to $y^{12} = f_{12}(x)$

R^3 : JEŚLI x_1 jest X_{12} i x_2 jest X_{21} to $y^{21} = f_{21}(x)$

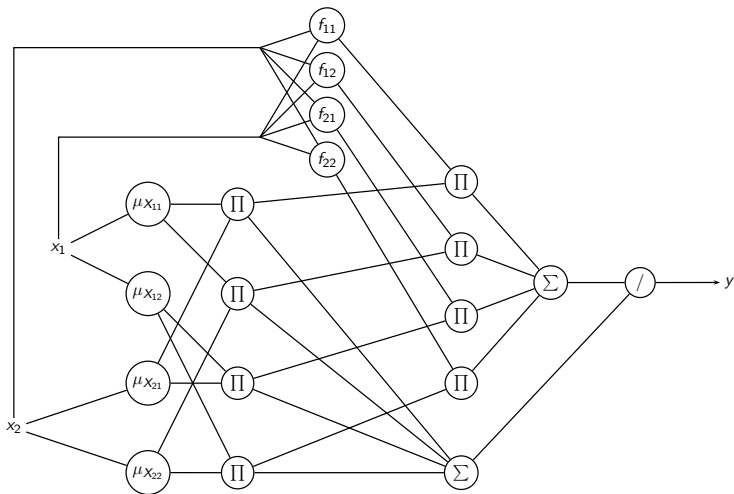
R^4 : JEŚLI x_1 jest X_{12} i x_2 jest X_{22} to $y^{22} = f_{22}(x)$

Modele rozmyte Takagi-Sugeno-Kanga

Funkcje przynależności, wnioskowanie rozmyte



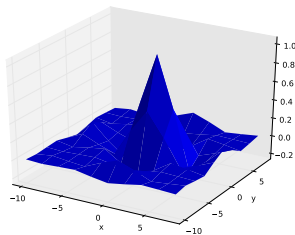
Struktura rozmytej sieci neuronowej



Algorytm uczenia rozmytej sieci neuronowej

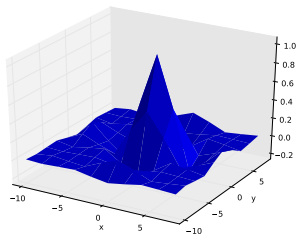
- Algorytm hybrydowy
- Dostrajanie parametrów liniowych następników - metoda najmniejszych kwadratów
- Dostrajanie parametrów poprzedników - uczenie SN
 - Optymalizacja gradientowa - algorytm propagacji wstecznej
 - Optymalizacja bezgradientowa

Algorytm uczenia rozmytej sieci neuronowej

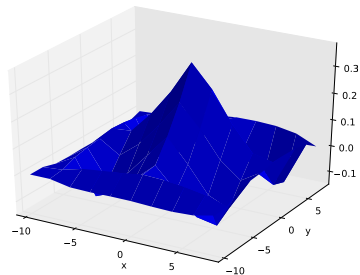


dane testowe

Algorytm uczenia rozmytej sieci neuronowej



dane testowe

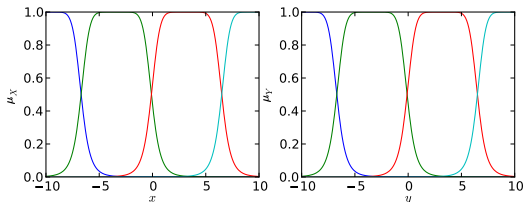


następniki dostrojone mnk

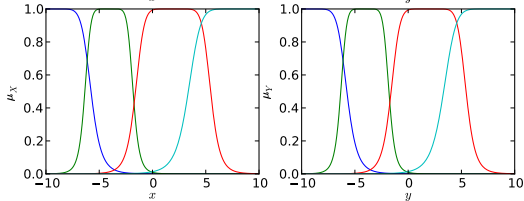
Algorytm uczenia rozmytej sieci neuronowej

funkcje przynależności

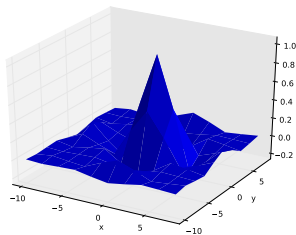
sieć nie uczona



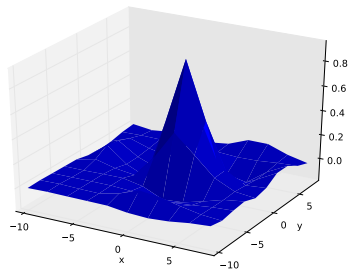
50 epok uczenia



Algorytm uczenia rozmytej sieci neuronowej



dane testowe



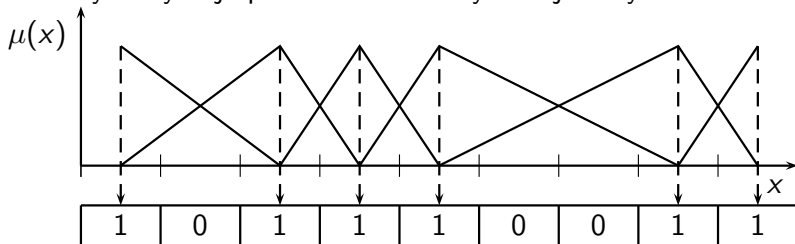
50 epok uczenia

Algorytmy ewolucyjne

- Ewolucja sztucznych osobników należących do populacji
- Rozmnażanie - Mutacja i krzyżowanie
- Dobór naturalny

Podejście oparte na dyskretyzacji

- Dyskretyzacja przestrzeni zmiennych wejściowych



- Mutacja – zamiana dwóch różnych bitów
- Krzyżowanie – wymiana parzystej liczby różnych bitów
- Funkcja dopasowania – suma kwadratów uchybów dla danych ze zbioru uczącego

Podejście z genotypem ciągłym

- Genotyp – wektor rzeczywistych parametrów modelu
- Mutacja – mutacja nierównomierna

$$c^{l+1} = \begin{cases} c^l + \Delta(l, \delta_{\max}), & b = 0 \\ c^l - \Delta(l, \delta_{\max}), & b = 1 \end{cases}$$

$$\Delta(l, y) = y(1 - r^{(1 - \frac{l}{l_{\max}})^b})$$

- Krzyżowanie

$$\begin{aligned} C_1^{l+1} &= aC_r^l + (1 - a)C_s^l, & C_2^{l+1} &= (1 - a)C_r^l + aC_s^l, \\ C_3^{l+1} &= \min(C_r^l, C_s^l), & C_4^{l+1} &= \max(C_r^l, C_s^l) \end{aligned}$$

Inne zastosowania algorytmów ewolucyjnych

- Dostrajanie parametrów liniowych następników – reprezentacja kątowa
- Ewolucyjny dobór struktury sieci
- Wybór istotnych wejść modelu

Implementacja

- Wygodne i uniwersalne API do tworzenia i strojenia modeli rozmytych
- Elastyczność - swoboda wyboru struktury i parametrów dostrajalnych modelu
- Interfejs graficzny oparty o przeglądarkę
- Możliwość importu i eksportu danych wykorzystywanych przez MATLAB Fuzzy Toolbox
- Wykorzystywane technologie: Python, NumPy, SciPy, Django

Dziękuję za uwagę