Politechnika Krakowska
im. Tadeusza Kościuszki w Krakowie

Wydział Fizyki, Matematyki i Informatyki
Fizyka Techniczna

Juliusz Chojenka, Paula Świerska, Wojciech Tejchman

Logika rozmyta

Kraków 2017

1. Logika rozmyta

Logika rozmyta: binarne członkostwo (należy (1), nie należy (0)) zostało rozszerzone na przedział liczb rzeczywistych [0,1], gdzie punkty końcowe oznaczają pełną należność i nie należność.

Nieskończona ilość wartości między ptk. końcowymi reprezentuje stopień przynależności elementu x w jakimś zbiorze świata X.

Zbiory rozmyte: zbiory z „przestrzeni” X, które mogą określić stopień członkostwa.

1.1. Przykład

Przykład binarny: możemy rozpoznawać temperaturę powyżej 25°C jako gorącą (1), natomiast poniżej jako zimną (0).

Przykład logiki rozmytej:

$$T \leqslant 0°C \quad \rightarrow \text{bardzo zimno}$$
$$0°C \leqslant T \leqslant 18°C \quad \rightarrow \text{trochę zimno}$$
$$18°C \leqslant T \leqslant 25°C \quad \rightarrow \text{idealnie}$$
$$25°C \leqslant T \leqslant 30°C \quad \rightarrow \text{trochę gorąco}$$
$$T \geqslant 30°C \quad \rightarrow \text{Sahara}$$

1.2. Wprowadzenie oznaczeń

Zbiór rozmyty $\tilde{F}$ na X wyraża się jako:

$$\mu_{\tilde{F}}(x) \;:\; X \rightarrow [0.\alpha]$$

gdzie $\mu_{\tilde{F}}(x)$ jest **funkcją przynależności**.

Przykład: samochód powinien utrzymywać zadaną prędkość d. Stopień bliskości do zadanej prędkości opisuje funkcja przynależności:

$$\mu_{\tilde{F}}(x) = \exp(-\tfrac{(x-d)^2}{2d})$$

Zbiór rozmyty szybkości:

$$\tilde{S} = \{(x, \mu_{\tilde{S}}(x)) \mid x\epsilon[0, 200]\}$$

1.3. Operatory logistyczne

Przecięcie zbiorów rozmytych:

$$\tilde{C} = \tilde{A} \cap \tilde{B}$$
$$\mu_{\tilde{C}}(x) = min\{\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(x)\}$$

Suma zbiorów rozmytych:

$$\tilde{C} = \tilde{A} \cup \tilde{B}$$
$$\mu_{\tilde{C}}(x) = max\{\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(x)\}$$

1.4. Pojęcia z logiki rozmytej

Załóżmy, że zysk na pewnym produkcie należy od 0% do 100%

Duży: wartość 1 otrzymujemy dla 100%, 0 dla 0%.

$$\mu_{\tilde{L}(x)} = \tfrac{x - x_{min}}{x_{max} - x_{min}}$$

Mały: wartość 1 otrzymujemy dla 0%, 0 dla 100%.

$$\mu_{\tilde{S}(x)} = \tfrac{x - x_{max}}{x_{min} - x_{max}}$$

Załóżmy, że zysk na pewnym produkcie wynosi około $p = 40\%$.

$$\mu_{\tilde{N}(x)} = \tfrac{1}{\sqrt{2\pi\sigma}} \exp(\tfrac{-(x-p)^2}{2\sigma^2})$$

gdzie $\sigma = 0,1(x_{max} - x_{min})$ Załóżmy, że zysk na pewnym produkcie będzie większy $p = 40\%$.

$$\mu_{\tilde{G}(x)} = [1 + \exp(\tfrac{-(x-p)}{2\lambda})]^{-1}$$

gdzie $\lambda = 0,1(x_{max} - x_{min})$

1.5. Hedging

Linguistic hedge: zwiększenie lub zmniejszenie wartości przynależności do zbioru.

Przykład:
Stopień przynależności dla temperatur $T\epsilon[0°C, 40°C]$

$T°C$	0	5	10	15	20	25	30	35	40
$\mu_{\tilde{H}(T)}$	0	0.125	0.25	0.375	0.5	0.625	0.75	0.825	1

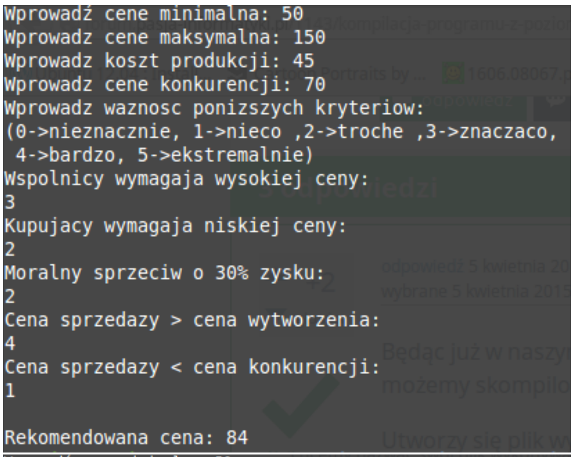
Stopień przynależności dla temperatur $T\epsilon[0°C, 40°C]$ dla bardzo gorących

$T°C$	0	5	10	15	20	25	30	35	40
$\mu_{\tilde{H}(T)}$	0	0.016	0.063	0.141	0.25	0.391	0.563	0.681	1

1.6. Zastosowanie

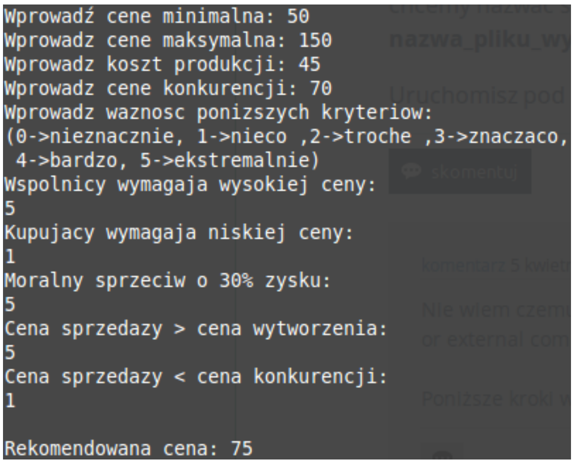
Zastosowanie: przetwarzanie sygnałów, robotyka, inteligentna kontrola procesu technologicznego, systemy ekspertowe, przetwarzanie obrazów, **teoria podejmowania decyzji**, analiza skupień oraz różne metody poznawania takie jak: algorytmy genetyczne i wnioskowanie indukcyjne.

Przykład z teorii podejmowania decyzji: wybór ceny sprzedaży wyrobów ręcznych.
Napisany program znajdował najlepszą cenę produktu w zależności od zadanych mu danych (ceny minimalnej i maksymalnej, ceny konkurencji, ceny produkcji) oraz stopnia przynależności (nieznacznie, nieco, trochę, znacząco, bardzo, ekstremalnie). Wykorzystany został tzw. Hadging. Poniżej znajdują się przykładowe użycia tego programu.



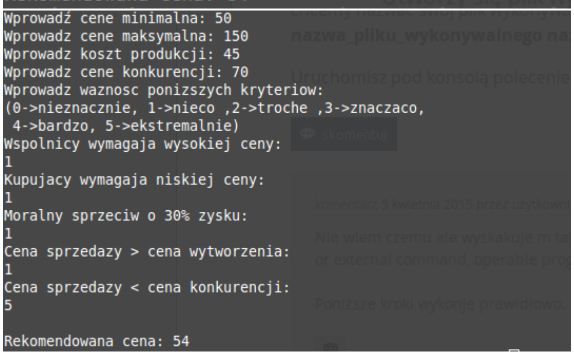
```
Wprowadź cene minimalna: 50
Wprowadz cene maksymalna: 150
Wprowadz koszt produkcji: 45
Wprowadz cene konkurencji: 70
Wprowadz waznosc ponizszych kryteriow:
(0->nieznacznie, 1->nieco ,2->troche ,3->znaczaco,
 4->bardzo, 5->ekstremalnie)
Wspolnicy wymagaja wysokiej ceny:
3
Kupujacy wymagaja niskiej ceny:
2
Moralny sprzeciw o 30% zysku:
2
Cena sprzedazy > cena wytworzenia:
4
Cena sprzedazy < cena konkurencji:
1

Rekomendowana cena: 84
```



```
Wprowadź cene minimalna: 50
Wprowadz cene maksymalna: 150
Wprowadz koszt produkcji: 45
Wprowadz cene konkurencji: 70
Wprowadz waznosc ponizszych kryteriow:
(0->nieznacznie, 1->nieco ,2->troche ,3->znaczaco,
 4->bardzo, 5->ekstremalnie)
Wspolnicy wymagaja wysokiej ceny:
5
Kupujacy wymagaja niskiej ceny:
1
Moralny sprzeciw o 30% zysku:
5
Cena sprzedazy > cena wytworzenia:
5
Cena sprzedazy < cena konkurencji:
1

Rekomendowana cena: 75
```



```
Wprowadź cene minimalna: 50
Wprowadz cene maksymalna: 150
Wprowadz koszt produkcji: 45
Wprowadz cene konkurencji: 70
Wprowadz waznosc ponizszych kryteriow:
(0->nieznacznie, 1->nieco ,2->troche ,3->znaczaco,
 4->bardzo, 5->ekstremalnie)
Wspolnicy wymagaja wysokiej ceny:
1
Kupujacy wymagaja niskiej ceny:
1
Moralny sprzeciw o 30% zysku:
1
Cena sprzedazy > cena wytworzenia:
1
Cena sprzedazy < cena konkurencji:
5

Rekomendowana cena: 54
```

Jak widać na przykładzie użytych cen. Jeżeli przypiszemy odpowiednio stopień ceny sugerując, że cena ma być mała program zasugeruje niską cenę (54), natomiast jeżeli na będziemy zadawać odpowiednie wartości tak jakbyśmy chcieli mieć największy zysk sugerowana cena będzie odpowiednio większa (84).