



**Kolegium Nauk Przyrodniczych
Uniwersytet Rzeszowski**

Przedmiot:

Zastosowanie agregacji w uczeniu maszynowym

**Zastosowanie T-norm i T-konorm
do łączenia Rozmytych Reguł Decyzyjnych.**

Wykonał:

Michał Wajdowicz 117841

Jakub Słota 117833

Informatyka Rok 2 magisterskie

Prowadzący: mgr inż. Aleksander Wojtowicz

Rzeszów 2025

Spis treści

1.	Problem decyzyjny	3
2.	Funkcje przynależności	3
2.1.	Dochód.....	4
2.2.	Zadłużenie.....	5
3.	Definicje T-norm.....	5
4.	Definicje T-konorm	6
5.	Definicja reguł	7
6.	Porównanie wyników dla różnych operatorów	8
7.	Wnioski.....	9

1. Problem decyzyjny

Celem projektu jest zbudowanie prostego modelu decyzyjnego opartego na logice rozmytej, który pomoże w ocenie, czy dana osoba powinna otrzymać kredyt. W przeciwieństwie do klasycznego podejścia zero-jedynkowego, wykorzystujemy logikę rozmytą, która pozwala na wyrażenie niepewności i stopniowości w ocenie.

Założenia problemu:

- Decyzja dotyczy udzielenia lub odmowy kredytu.
- W ocenie ryzyka kredytowego uwzględniane są dwie cechy:
 - Dochód klienta
 - Wysokość zadłużenia
- Obie cechy są modelowane za pomocą funkcji przynależności rozmytej, dzięki czemu wartości nie są oceniane binarnie (np. "niski" albo "wysoki"), ale z pewnym stopniem przynależności do każdego z pojęć.

Cel:

Zaimplementować system reguł decyzyjnych z wykorzystaniem różnych kombinacji operatorów:

- **T-norm** – do łączenia warunków w obrębie jednej reguły.
- **T-konorm** – do agregacji wyników wielu reguł.

Następnie porównać wpływ wyboru tych operatorów na końcową decyzję systemu: czy kredyt powinien zostać przyznany (TAK), czy nie (NIE).

2. Funkcje przynależności

W celu odwzorowania nieprecyzyjnych pojęć takich jak „niski dochód” czy „wysokie zadłużenie”, wykorzystano funkcje przynależności rozmytej. Zamiast przypisywać klienta do jednej kategorii, funkcje te pozwalają określić stopień przynależności do każdego z rozmytych zbiorów (wartość od 0 do 1).

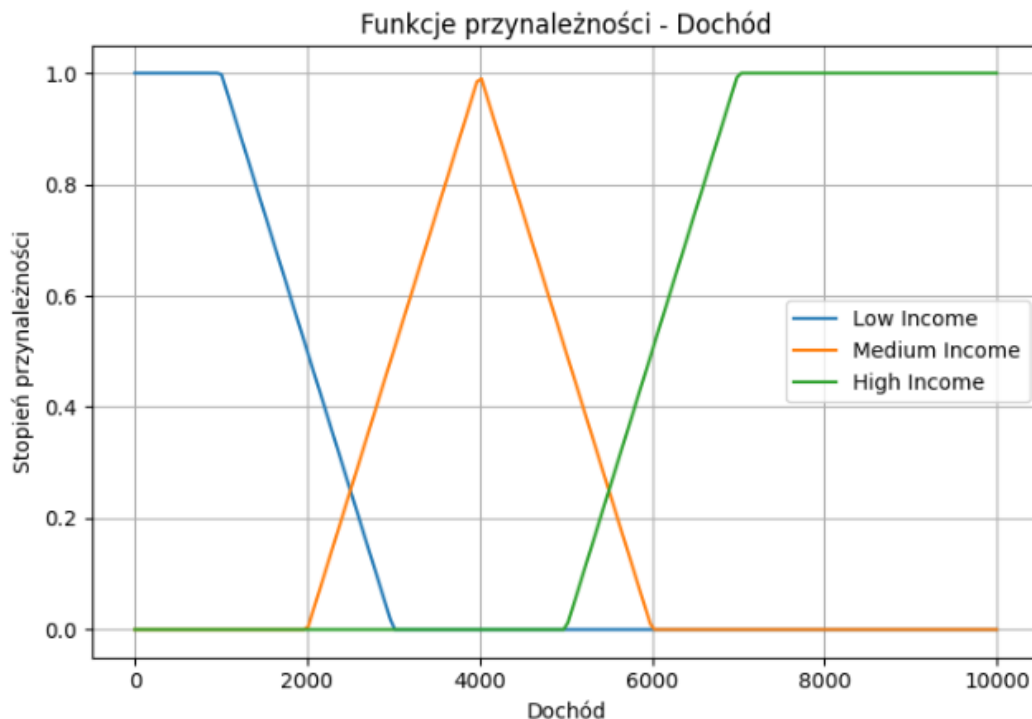
2.1. Dochód

Dla cechy dochód wyróżniono trzy zbiory rozmyte:

- Niski dochód: jest w pełni spełniony (wartość 1), gdy dochód wynosi 1000 zł lub mniej. Powyżej tej wartości przynależność maleje liniowo, aż do 0 przy dochodzie równym 3000 zł.
- Średni dochód: opisany funkcją trójkątną. Maksymalny stopień przynależności (1) osiąga przy dochodzie 4000 zł. Przynależność rośnie liniowo od 0 do 1 w zakresie 2000–4000 zł i następnie maleje od 1 do 0 w zakresie 4000–6000 zł.
- Wysoki dochód: nie występuje dla dochodów poniżej 5000 zł. Od tej wartości przynależność rośnie liniowo i osiąga maksimum (1) przy dochodzie 7000 zł lub więcej.

```
# Funkcje przynależności dochodu
def low_income(x): return np.maximum(0, np.minimum(1, (3000 - x) / 2000)) 2 usages
def medium_income(x): return np.maximum(0, np.minimum((x - 2000) / 2000, (6000 - x) / 2000)) 2 usages
def high_income(x): return np.maximum(0, np.minimum(1, (x - 5000) / 2000)) 2 usages
```

Rys. 1 Definicja funkcji przynależności w pythonie dla cechy „Dochód”



Rys. 2 Wykres funkcji przynależności dla cechy "Dochód"

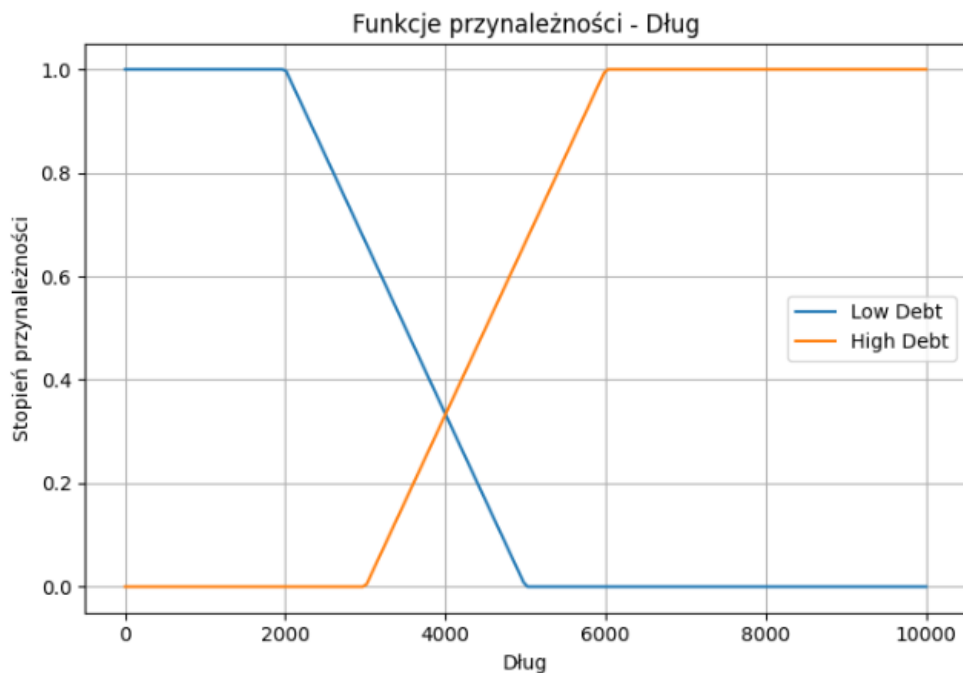
2.2. Zadłużenie

Dla cechy zadłużenie zdefiniowano dwa zbiory rozmyte:

- Niskie zadłużenie: dla wartości zadłużenia do 2000 zł przynależność wynosi 1. Powyżej tej wartości maleje liniowo, osiągając 0 przy 5000 zł.
- Wysokie zadłużenie: nie występuje dla wartości poniżej 3000 zł. Od tego poziomu przynależność rośnie liniowo i osiąga wartość 1 przy 6000 zł.

```
# Funkcje przynależności zadłużenia
def low_debt(x): return np.maximum(0, np.minimum(1, (5000 - x) / 3000)) 2 usages
def high_debt(x): return np.maximum(0, np.minimum(1, (x - 3000) / 3000)) 2 usages
```

Rys. 3 Definicja funkcji przynależności w pythonie dla cechy „Zadłużenie”



Rys. 4 Wykres funkcji przynależności dla cechy "Zadłużenie"

3. Definicje T-norm

T-normy (triangular norms) to operatory logiczne wykorzystywane w logice rozmytej do łączenia warunków w jednej regule decyzyjnej. Odpowiadają one rozmytemu odpowiednikowi logicznej koniunkcji – czyli spójnika „i”.

W tym projekcie, t-norma służy do określenia stopnia spełnienia reguły warunkowej, w której występuje więcej niż jedna przesłanka. Przykładowo: „Jeśli dochód jest wysoki i zadłużenie jest niskie, to decyzja = TAK.” W logice rozmytej stopień spełnienia takiej reguły to t-norma zastosowana do stopni przynależności każdej przesłanki. W projekcie zaimplementowano trzy najczęściej wykorzystywane t-normy:

Tabela 1 T-normy

Nazwa	Wzór matematyczny	Opis
Minimum	$\min(a, b)$	Wybiera mniejszą wartość – konserwatywna koniunkcja
Iloczyn (produkt)	$a * b$	Mnoży wartości – zachowanie probabilistyczne
Łukasiewicz	$\max(0, a + b - 1)$	Suma pomniejszona o 1, minimum to 0

```
# T-norms
def t_norm_min(a, b): return np.minimum(a, b) 1 usage
def t_norm_product(a, b): return a * b 1 usage
def t_norm_lukasiewicz(a, b): return np.maximum(0, a + b - 1) 1 usage
```

Rys. 5 Definicja T-norm w pythonie

4. Definicje T-konorm

T-konormy (s-normy) to operatory logiczne stosowane w logice rozmytej do łączenia wyników z różnych reguł. Są one odpowiednikiem rozmytej alternatywy logicznej – czyli spójnika „lub”. T-konorma określa, w jakim stopniu co najmniej jedna z reguł sugeruje daną decyzję (np. „TAK”). W projekcie wykorzystano trzy popularne t-konormy:

Tabela 2 T-konormy

Nazwa	Wzór matematyczny	Opis
Maksimum	$\max(a, b)$	Wybiera większą wartość – najbardziej optymistyczna
Suma probabilistyczna	$a + b - a * b$	Unia probabilistyczna – uwzględnia nakładanie się wartości
Łukasiewicz	$\min(1, a + b)$	Suma z ograniczeniem do 1 – zrównoważona agregacja

```
# T-conorms
def t_conorm_max(a, b): return np.maximum(a, b) 1 usage
def t_conorm_probabilistic_sum(a, b): return a + b - a * b 1 usage
def t_conorm_lukasiewicz(a, b): return np.minimum(1, a + b) 1 usage
```

Rys. 6 Definicja T-konorm w pythonie

5. Definicja reguł

W logice rozmytej reguły decyzyjne opisują zależność między cechami wejściowymi a decyzją wyjściową. Są one oparte na wiedzy eksperckiej i mają postać:

Jeśli (warunek1) i/lub (warunek2) to (decyzja)

W tym projekcie zdefiniowano trzy proste reguły eksperckie, które uwzględniają dochód i zadłużenie klienta w kontekście decyzji kredytowej.

Zdefiniowane reguły:

- Jeśli dochód jest wysoki i dług jest niski, to decyzja = TAK
- Jeśli dochód jest średni i dług jest niski, to decyzja = TAK
- Jeśli dochód jest niski lub dług jest wysoki, to decyzja = NIE

Sposób działania:

- Warunki w regule są łączone za pomocą T-normy (np. min, iloczyn, Łukasiewicz).
- Wyniki różnych reguł są agregowane za pomocą T-konormy (np. max, suma probabilistyczna, Łukasiewicz).

```

r1 = tnorm(high_income, low_debt) # Reguła 1
r2 = tnorm(medium_income, low_debt) # Reguła 2
r3 = tconorm(low_income, high_debt) # Reguła 3

credit_yes = tconorm(r1, r2) # Agregacja reguł prowadzących do TAK
credit_no = r3 # Reguła prowadząca do NIE

```

Rys. 7 Implementacja reguł w pythonie

Na końcu system porównuje, która z wartości (TAK czy NIE) jest większa i na tej podstawie podejmuje decyzję.

Decyzja końcowa:

Jeśli $\text{credit_yes} > \text{credit_no} \rightarrow \text{decyzja} = \text{TAK}$

W przeciwnym razie $\rightarrow \text{decyzja} = \text{NIE}$

6. Porównanie wyników dla różnych operatorów

Celem tego etapu było sprawdzenie, jak wybór T-normy i T-konormy wpływa na końcową decyzję systemu rozmytego. W tym celu przygotowano cztery przykładowe przypadki (kombinacje dochodu i zadłużenia), a dla każdego przetestowano 9 kombinacji operatorów (3 t-normy \times 3 t-konormy).

Przykład	Dochód (zł)	Zadłużenie (zł)
1	4500	2500
2	2000	5500
3	2632	3938
4	5571	3775

Dla każdego przypadku obliczano:

- Stopień przynależności do każdego zbioru (np. „średni dochód”)
- Stopień spełnienia każdej reguły (z użyciem t-norm)
- Wynik agregacji reguł (z użyciem t-konorm)
- Decyzję końcową: TAK lub NIE

Przykładowy fragment wyników:

Przykład	T-norma	T-konorma	TAK	NIE	Decyzja
1	min	max	0.75	0.0	TAK
2	product	prob_sum	0.0	0.92	NIE
3	min	max	0.316	0.313	TAK
4	product	max	0.117	0.258	NIE

7. Wnioski

W projekcie przedstawiono zastosowanie logiki rozmytej do prostego systemu wspomagania decyzji – oceny ryzyka kredytowego na podstawie dochodu i zadłużenia klienta. Kluczowym celem było zbadanie wpływu różnych operatorów T-norm i T-konorm na wynik systemu decyzyjnego.

Najważniejsze obserwacje:

- Logika rozmyta pozwala modelować nieprecyzyjne pojęcia i warunki, które trudno opisać w sposób binarny. Przykłady to: „średni dochód” czy „wysokie zadłużenie”.
- Reguły można zapisać w formie rozmytej i przetwarzać z użyciem t-norm oraz t-konorm, co pozwala elastycznie łączyć przesłanki i decyzje.
- Wybór operatorów T-normy (do łączenia warunków) i T-konormy (do agregacji wyników reguł) ma znaczący wpływ na decyzję końcową.
- Niektóre operatory, jak min i max, prowadzą do bardziej „zero-jedynkowego” zachowania systemu. Inne, jak product i Łukasiewicz, umożliwiają stopniowanie decyzji.