

IMPLEMENTASI FUZZY, VARIABEL, HIMPUNAN FUZZY DAN RULE

NAMA: IOS WIDRIA SARAGIH

NIM : 221011400558

KELAS : 05TPLMOO5

https://github.com/iossaragih/ioswidriasaragih_uas/blob/main/kecerdasan_buatan_UAS.ipynb



PERHITUNGAN MANUAL

Berikut adalah langkah-langkah perhitungan manual berdasarkan kode dengan input:

- Pendapatan = 15000
- Utang = 3000
- Riwayat Kredit = 8

1. Fuzzifikasi

Langkah ini menghitung derajat keanggotaan untuk masing-masing parameter berdasarkan fungsi keanggotaan yang telah didefinisikan.

Pendapatan = 15000

- Rendah:
Fungsi trapezoid:

$$\mu_{\text{rendah}} = 0, \quad \text{karena } 15000 > 8000$$

- Sedang:
Fungsi triangle:

$$\mu_{\text{sedang}} = \frac{16000 - 15000}{16000 - 10000} = \frac{1000}{6000} = 0.167$$

- Tinggi:
Fungsi trapezoid:

$$\mu_{\text{tinggi}} = 1, \quad \text{karena } 15000 \geq 12000$$

Utang = 3000

- Rendah:
Fungsi trapezoid:

$$\mu_{\text{rendah}} = 1, \quad \text{karena } 3000 < 4000$$

- Sedang:
Fungsi triangle:

$$\mu_{\text{sedang}} = \frac{3000 - 2000}{5000 - 2000} = \frac{1000}{3000} = 0.333$$

- Tinggi:
Fungsi trapezoid:

$$\mu_{\text{tinggi}} = 0, \quad \text{karena } 3000 < 6000$$

Riwayat Kredit = 8

- Buruk:
Fungsi trapezoid:

$$\mu_{\text{buruk}} = 0, \quad \text{karena } 8 > 5$$

- Sedang:
Fungsi triangle:

$$\mu_{\text{sedang}} = \frac{10 - 8}{10 - 7} = \frac{2}{3} = 0.667$$

- Baik:
Fungsi trapezoid:

$$\mu_{\text{baik}} = 1, \quad \text{karena } 8 \geq 7$$

PERHITUNGAN MANUAL

2. Inferensi

Berikut adalah aturan fuzzy yang diterapkan:

1. Rule 1: Pendapatan = Rendah, Utang = Tinggi, Riwayat Kredit = Buruk

$$\text{Min}(\mu_{\text{rendah}}[\text{pendapatan}], \mu_{\text{tinggi}}[\text{utang}], \mu_{\text{buruk}}[\text{riwayat}]) = \text{Min}(0, 0, 0) = 0$$

Keluar: Tidak Layak

2. Rule 2: Pendapatan = Rendah, Utang = Rendah, Riwayat Kredit = Baik

$$\text{Min}(\mu_{\text{rendah}}[\text{pendapatan}], \mu_{\text{rendah}}[\text{utang}], \mu_{\text{baik}}[\text{riwayat}]) = \text{Min}(0, 1, 1) = 0$$

Keluar: Layak

3. Rule 3: Pendapatan = Tinggi, Utang = Rendah, Riwayat Kredit = Baik

$$\text{Min}(\mu_{\text{tinggi}}[\text{pendapatan}], \mu_{\text{rendah}}[\text{utang}], \mu_{\text{baik}}[\text{riwayat}]) = \text{Min}(1, 1, 1) = 1$$

Keluar: Sangat Layak

4. Rule 4: Pendapatan = Sedang, Utang = Sedang, Riwayat Kredit = Sedang

$$\text{Min}(\mu_{\text{sedang}}[\text{pendapatan}], \mu_{\text{sedang}}[\text{utang}], \mu_{\text{sedang}}[\text{riwayat}]) = \text{Min}(0.167, 0.333, 0.667) = 0.167$$

Keluar: Layak

5. Rule 5: Pendapatan = Tinggi, Utang = Tinggi, Riwayat Kredit = Buruk

$$\text{Min}(\mu_{\text{tinggi}}[\text{pendapatan}], \mu_{\text{tinggi}}[\text{utang}], \mu_{\text{buruk}}[\text{riwayat}]) = \text{Min}(1, 0, 0) = 0$$

Keluar: Tidak Layak

6. Rule 6: Pendapatan = Tinggi, Utang = Rendah, Riwayat Kredit = Sedang

$$\text{Min}(\mu_{\text{tinggi}}[\text{pendapatan}], \mu_{\text{rendah}}[\text{utang}], \mu_{\text{sedang}}[\text{riwayat}]) = \text{Min}(1, 1, 0.667) = 0.667$$

Keluar: Sangat Layak

3. Agregasi

Hasil dari inferensi digabungkan dengan cara mengambil nilai maksimum dari masing-masing kategori keluaran:

- Tidak Layak: $\max(0, 0) = 0$
- Layak: $\max(0, 0.167) = 0.167$
- Sangat Layak: $\max(1, 0.667) = 1$

4. Defuzzifikasi

1. Dari Agregasi
Hasil maksimum dari
setiap kategori:

- Tidak Layak:
 $\mu=0 \setminus \mu = 0$
(tidak dihitung
karena kontribusi
nol).
- Layak: $\mu=0.167 \setminus \mu = 0.167$,
centroid sekitar
505050.
- Sangat Layak:
 $\mu=1 \setminus \mu = 1$,
centroid sekitar
85858

2. Perhitungan Centroid

Formula:

$$\text{Kelayakan Kredit} = \frac{\sum(\mu \cdot \text{centroid})}{\sum \mu}$$

Substitusi nilai:

$$\text{Kelayakan Kredit} = \frac{(0.167 \cdot 50) + (1 \cdot 85)}{0.167 + 1}$$

Hitung:

$$\text{Kelayakan Kredit} = \frac{8.35 + 85}{1.167} = \frac{93.35}{1.167} \approx 80$$

Kelayakan Kredit = 80.

Artinya, dengan input yang diberikan, sistem
menilai kredit sangat layak.

SOURCECODE

```
[1]: import numpy as np
import skfuzzy as fuzz
from skfuzzy import control as ctrl

# Step 1: Define Fuzzy Variables
# Input Variables
pendapatan = ctrl.Antecedent(np.arange(0, 20001, 1000), 'pendapatan') # Pendapatan bulanan (0 - 20,000)
utang = ctrl.Antecedent(np.arange(0, 10001, 500), 'utang') # Utang (0 - 10,000)
riwayat_kredit = ctrl.Antecedent(np.arange(0, 11, 1), 'riwayat_kredit') # Riwayat Kredit (0 - 10)

# Output Variable
kelayakan = ctrl.Consequent(np.arange(0, 101, 1), 'kelayakan') # Kelayakan Kredit (0 - 100)

# Step 2: Define Membership Functions
# Membership for Pendapatan
pendapatan['rendah'] = fuzz.trapmf(pendapatan.universe, [0, 0, 4000, 8000])
pendapatan['sedang'] = fuzz.trimf(pendapatan.universe, [4000, 10000, 16000])
pendapatan['tinggi'] = fuzz.trapmf(pendapatan.universe, [12000, 16000, 20000, 20000])

# Membership for Utang
utang['rendah'] = fuzz.trapmf(utang.universe, [0, 0, 2000, 4000])
utang['sedang'] = fuzz.trimf(utang.universe, [2000, 5000, 8000])
utang['tinggi'] = fuzz.trapmf(utang.universe, [6000, 8000, 10000, 10000])

# Membership for Riwayat Kredit
riwayat_kredit['buruk'] = fuzz.trapmf(riwayat_kredit.universe, [0, 0, 3, 5])
riwayat_kredit['sedang'] = fuzz.trimf(riwayat_kredit.universe, [3, 5, 7])
riwayat_kredit['baik'] = fuzz.trapmf(riwayat_kredit.universe, [5, 7, 10, 10])

# Membership for Kelayakan Kredit
kelayakan['tidak_layak'] = fuzz.trapmf(kelayakan.universe, [0, 0, 30, 50])
kelayakan['layak'] = fuzz.trimf(kelayakan.universe, [30, 50, 70])
kelayakan['sangat_layak'] = fuzz.trapmf(kelayakan.universe, [60, 80, 100, 100])
```

```
# Step 3: Define Fuzzy Rules
rule1 = ctrl.Rule(pendapatan['rendah'] & utang['tinggi'] & riwayat_kredit['buruk'], kelayakan['tidak_layak'])
rule2 = ctrl.Rule(pendapatan['rendah'] & utang['rendah'] & riwayat_kredit['baik'], kelayakan['layak'])
rule3 = ctrl.Rule(pendapatan['tinggi'] & utang['rendah'] & riwayat_kredit['baik'], kelayakan['sangat_layak'])
rule4 = ctrl.Rule(pendapatan['sedang'] & utang['sedang'] & riwayat_kredit['sedang'], kelayakan['layak'])
rule5 = ctrl.Rule(pendapatan['tinggi'] & utang['tinggi'] & riwayat_kredit['buruk'], kelayakan['tidak_layak'])
rule6 = ctrl.Rule(pendapatan['tinggi'] & utang['rendah'] & riwayat_kredit['sedang'], kelayakan['sangat_layak'])

# Step 4: Build Control System
kelayakan_ctrl = ctrl.ControlSystem([rule1, rule2, rule3, rule4, rule5, rule6])
kelayakan_simulasi = ctrl.ControlSystemSimulation(kelayakan_ctrl)

# Step 5: Input Example
kelayakan_simulasi.input['pendapatan'] = 15000
kelayakan_simulasi.input['utang'] = 3000
kelayakan_simulasi.input['riwayat_kredit'] = 8

# Step 6: Compute
kelayakan_simulasi.compute()
print(f"Kelayakan Kredit: {kelayakan_simulasi.output['kelayakan']:.2f}")

# Step 7: Visualize
pendapatan.view()
utang.view()
riwayat_kredit.view()
kelayakan.view(sim=kelayakan_simulasi)
```

Kelayakan Kredit: 82.38

HASIL RUNNING

