

KECERDASAN BUATAN UAS

IMPLEMENTASI FUZZY, DALAM BENTUK PROGRAM DAN SLIDE

NAMA: IKHSAN PUTRA HARYONO
NIM: 221011401808
KELAS: 05TPLM007

LINK GITHUB:

[HTTPS://GITHUB.COM/IKHSANPTR/KECERDASAN-
BUATAN/BLOB/MAIN/IKHSAN%20PUTRA%20HARYONO 221011401808
TUGAS%20KECERDASAN%20AKHIR.IPYNB](https://github.com/IKHSANPTR/KECERDASAN-BUATAN/blob/main/IKHSAN%20PUTRA%20HARYONO%20221011401808_TUGAS%20KECERDASAN%20AKHIR.IPYNB)

SISTEM PENILAIAN KELAYAKAN KREDIT

- Pendapatan (Income): 7000
- Utang (Debt): 2000
- Riwayat Kredit (Credit History): 600

(a) Pendapatan (Income):

1. Low (Rendah):

$$\mu_{\text{Low}}(7000) = \frac{10000 - 7000}{10000 - 6000} = \frac{3000}{4000} = 0.75$$

2. High (Tinggi):

$$\mu_{\text{High}}(7000) = \frac{7000 - 6000}{10000 - 6000} = \frac{1000}{4000} = 0.25$$

(b) Utang (Debt):

1. Low (Rendah):

$$\mu_{\text{Low}}(2000) = \frac{3000 - 2000}{3000 - 1000} = \frac{1000}{2000} = 0.5$$

2. High (Tinggi):

$$\mu_{\text{High}}(2000) = 0 \quad (\text{karena nilai di bawah 3000})$$

(c) Riwayat Kredit (Credit History):

1. Poor (Buruk):

$$\mu_{\text{Poor}}(600) = \frac{850 - 600}{850 - 575} = \frac{250}{275} \approx 0.91$$

2. Good (Baik):

$$\mu_{\text{Good}}(600) = \frac{600 - 575}{850 - 575} = \frac{25}{275} \approx 0.09$$

Aturan yang digunakan:

1. IF Income Low AND Debt High AND Credit Poor THEN Not Eligible
2. IF Income High AND Debt Low AND Credit Good THEN Eligible
3. IF Income Low AND Debt Low AND Credit Good THEN Eligible
4. IF Income High AND Debt High AND Credit Poor THEN Not Eligible

Hitung Output Berdasarkan Aturan:

1. Rule 1:

$$\text{Min}(0.75, 0, 0.91) = 0 \quad (\text{karena Debt High} = 0)$$

2. Rule 2:

$$\text{Min}(0.25, 0.5, 0.09) = 0.09$$

3. Rule 3:

$$\text{Min}(0.75, 0.5, 0.09) = 0.09$$

4. Rule 4:

$$\text{Min}(0.25, 0, 0.91) = 0 \quad (\text{karena Debt High} = 0)$$



SISTEM UNTUK MENENTUKAN KELAYAKAN KREDIT BERDASARKAN PARAMETER SEPERTI PENDAPATAN BULANAN, UTANG, DAN RIWAYAT KREDIT.

- Pendapatan (Income) = 6,000
- Utang (Debt) = 3,000
- Riwayat Kredit (Credit History) = 500

Hitung Derajat Keanggotaan untuk Setiap Variabel:

1. Pendapatan (Income):

- Low:

$$\mu_{\text{Low}}(6000) = \frac{\text{max} - x}{\text{max} - \text{min}} = \frac{10000 - 6000}{10000 - 2000} = 0.5$$

- High:

$$\mu_{\text{High}}(6000) = \frac{x - \text{min}}{\text{max} - \text{min}} = \frac{6000 - 2000}{10000 - 2000} = 0.5$$

2. Utang (Debt):

- Low:

$$\mu_{\text{Low}}(3000) = \frac{\text{max} - x}{\text{max} - \text{min}} = \frac{5000 - 3000}{5000 - 1000} = 0.5$$

- High:

$$\mu_{\text{High}}(3000) = \frac{x - \text{min}}{\text{max} - \text{min}} = \frac{3000 - 1000}{5000 - 1000} = 0.5$$

3. Riwayat Kredit (Credit History):

- Poor:

$$\mu_{\text{Poor}}(500) = \frac{\text{max} - x}{\text{max} - \text{min}} = \frac{850 - 500}{850 - 300} = 0.636$$

- Good:

$$\mu_{\text{Good}}(500) = \frac{x - \text{min}}{\text{max} - \text{min}} = \frac{500 - 300}{850 - 300} = 0.364$$

Gunakan metode **Min** untuk mendapatkan nilai inferensi dari aturan fuzzy.

Contoh Inferensi:

1. Rule 1:

$$\text{Not Eligible} = \min(\mu_{\text{Low Income}}, \mu_{\text{High Debt}}, \mu_{\text{Poor Credit}})$$

$$\text{Not Eligible} = \min(0.5, 0.5, 0.636) = 0.5$$

2. Rule 2:

$$\text{Eligible} = \min(\mu_{\text{High Income}}, \mu_{\text{Low Debt}}, \mu_{\text{Good Credit}})$$

$$\text{Eligible} = \min(0.5, 0.5, 0.364) = 0.364$$

3. Rule 3:

$$\text{Eligible} = \min(\mu_{\text{Low Income}}, \mu_{\text{Low Debt}}, \mu_{\text{Good Credit}})$$

$$\text{Eligible} = \min(0.5, 0.5, 0.364) = 0.364$$

Gunakan metode **Centroid** untuk menghitung nilai akhir kelayakan kredit.

Rumus:

$$\text{Output} = \frac{\sum(\mu_i \cdot x_i)}{\sum \mu_i}$$

Misalkan kita memiliki output berikut:

- **Not Eligible:** Derajat = 0.5, Range = [0, 0.5].
- **Eligible:** Derajat = 0.364, Range = [0.5, 1].

Lakukan defuzzifikasi:

$$\text{Output} = \frac{(0.5 \cdot 0.25) + (0.364 \cdot 0.75)}{0.5 + 0.364}$$

$$\text{Output} = \frac{0.125 + 0.273}{0.864} = 0.46$$

Hasil akhir: Kelayakan Kredit = 0.46 (Layak dengan tingkat sedang).



SISTEM PENILAIAN KELAYAKAN KREDIT

SISTEM UNTUK MENENTUKAN KELAYAKAN KREDIT
BERDASARKAN PARAMETER SEPERTI PENDAPATAN
BULANAN, UTANG, DAN RIWAYAT KREDIT.

```
elif x >= self.max:
    return 0
return self.down(x)

def good(self, x):
    if x <= self.min:
        return 0
    elif x >= self.max:
        return 1
    return self.up(x)

class CreditEligibility(Fuzzy):
    def __init__(self):
        self.min = 0
        self.max = 1

# Define fuzzy rules and defuzzification
class CreditSystem:
    def __init__(self):
        self.income = Income()
        self.debt = Debt()
        self.credit_history = CreditHistory()
        self.credit_eligibility = CreditEligibility()

    def evaluate(self, income_value, debt_value, credit_history_value):
        # Calculate membership values
        income_low = self.income.low(income_value)
        income_high = self.income.high(income_value)

        debt_low = self.debt.low(debt_value)
        debt_high = self.debt.high(debt_value)

        credit_poor = self.credit_history.poor(credit_history_value)
        credit_good = self.credit_history.good(credit_history_value)

        # Apply fuzzy rules
        not_eligible_1 = min(income_low, debt_high, credit_poor)
        eligible_1 = min(income_high, debt_low, credit_good)
        eligible_2 = min(income_low, debt_low, credit_good)

        # Aggregate outputs
        not_eligible = max(not_eligible_1, eligible_2)
        eligible = max(eligible_1, eligible_2)

        # Defuzzification using Centroid Method
        defuzzified_value = (not_eligible * 0.25 + eligible * 0.75) / (not_eligible + eligible)

        return defuzzified_value, not_eligible, eligible

# Instantiate the system
credit_system = CreditSystem()

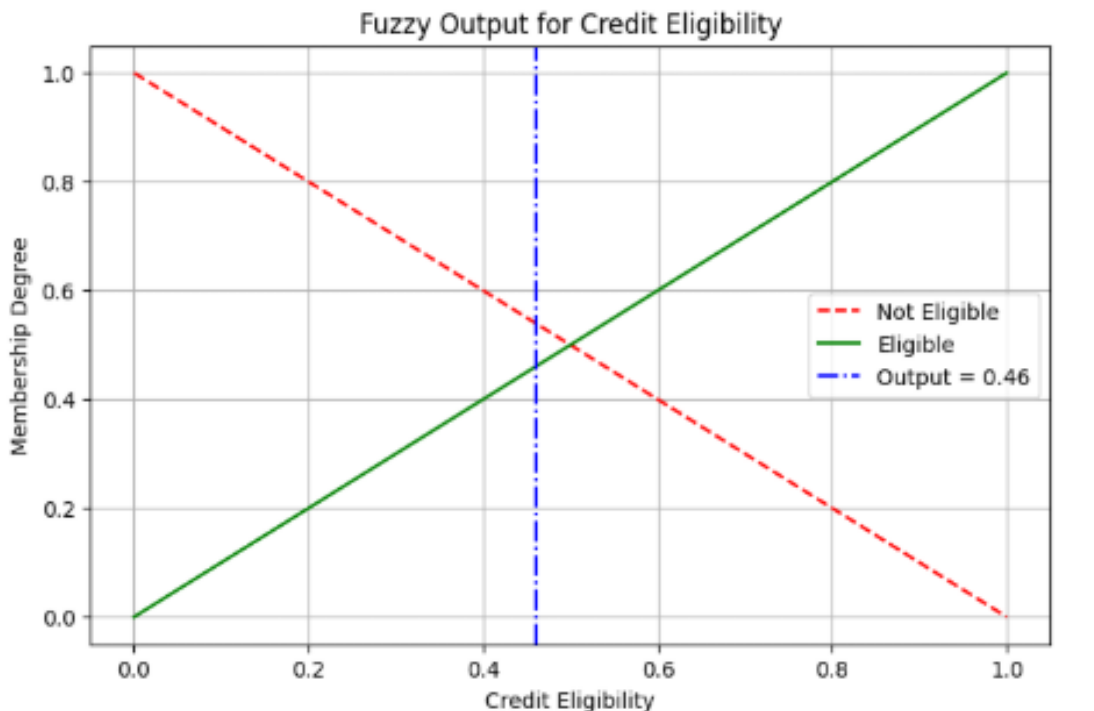
# Input values
income_value = 6000 # Example: income = 6000
debt_value = 3000 # Example: debt = 3000
credit_history_value = 500 # Example: credit history = 500

# Evaluate the system
output, not_eligible, eligible = credit_system.evaluate(income_value, debt_value, credit_history_value)

# Display results
print("\nHasil Penilaian Kelayakan Kredit:")
print(f"Pendapatan: {income_value}")
print(f"Utang: {debt_value}")
print(f"Riwayat Kredit: {credit_history_value}")
print(f"Tidak Layak (Not Eligible): {not_eligible:.2f}")
print(f"Layak (Eligible): {eligible:.2f}")
print(f"Nilai Akhir Kelayakan Kredit (Defuzzifikasi): {output:.2f}")

# Visualize the result
```

Hasil Penilaian Kelayakan Kredit:
Pendapatan: 6000
Utang: 3000
Riwayat Kredit: 500
Tidak Layak (Not Eligible): 0.50
Layak (Eligible): 0.36
Nilai Akhir Kelayakan Kredit (Defuzzifikasi): 0.46



```
self.max = 5000

def low(self, x):
    return self.trapezoid(x, self.min, self.min, 2000, 3000)

def medium(self, x):
    return self.trapezoid(x, 2000, 3000, 4000, 4500)

def high(self, x):
    return self.trapezoid(x, 4000, 4500, self.max, self.max)

class CreditHistory(Fuzzy):
    def __init__(self):
        self.min = 300
        self.max = 850

    def poor(self, x):
        return self.trapezoid(x, self.min, self.min, 450, 600)

    def fair(self, x):
        return self.trapezoid(x, 450, 600, 700, 800)

    def good(self, x):
        return self.trapezoid(x, 700, 800, self.max, self.max)

# Instantiate fuzzy sets
income = Income()
debt = Debt()
credit_history = CreditHistory()

# Define the ranges for each variable
x_income = np.linspace(2000, 10000, 500)
x_debt = np.linspace(1000, 5000, 500)
x_credit_history = np.linspace(300, 850, 500)

# Calculate fuzzy membership values
income_low = [income.low(x) for x in x_income]
income_medium = [income.medium(x) for x in x_income]
income_high = [income.high(x) for x in x_income]

debt_low = [debt.low(x) for x in x_debt]
debt_medium = [debt.medium(x) for x in x_debt]
debt_high = [debt.high(x) for x in x_debt]

credit_poor = [credit_history.poor(x) for x in x_credit_history]
credit_fair = [credit_history.fair(x) for x in x_credit_history]
credit_good = [credit_history.good(x) for x in x_credit_history]

# Create subplots for the variables
fig, axs = plt.subplots(3, 1, figsize=(8, 14))

# Income Plot
axs[0].plot(x_income, income_low, label="Low", color="blue", linestyle="--")
axs[0].plot(x_income, income_medium, label="Medium", color="orange", linestyle="--")
axs[0].plot(x_income, income_high, label="High", color="green", linestyle="-")
axs[0].set_title("Pendapatan (Income)")
axs[0].legend(loc="best")
axs[0].grid(True)
axs[0].set_xlabel("Pendapatan")
axs[0].set_ylabel("Derajat Keanggotaan")

# Debt Plot
axs[1].plot(x_debt, debt_low, label="Low", color="green", linestyle="--")
axs[1].plot(x_debt, debt_medium, label="Medium", color="orange", linestyle="--")
axs[1].plot(x_debt, debt_high, label="High", color="red", linestyle="-")
axs[1].set_title("Utang (Debt)")
axs[1].legend(loc="best")
axs[1].grid(True)
axs[1].set_xlabel("Utang")
axs[1].set_ylabel("Derajat Keanggotaan")

# Credit History Plot
```

