

Perhitungan manual dan Sourcode

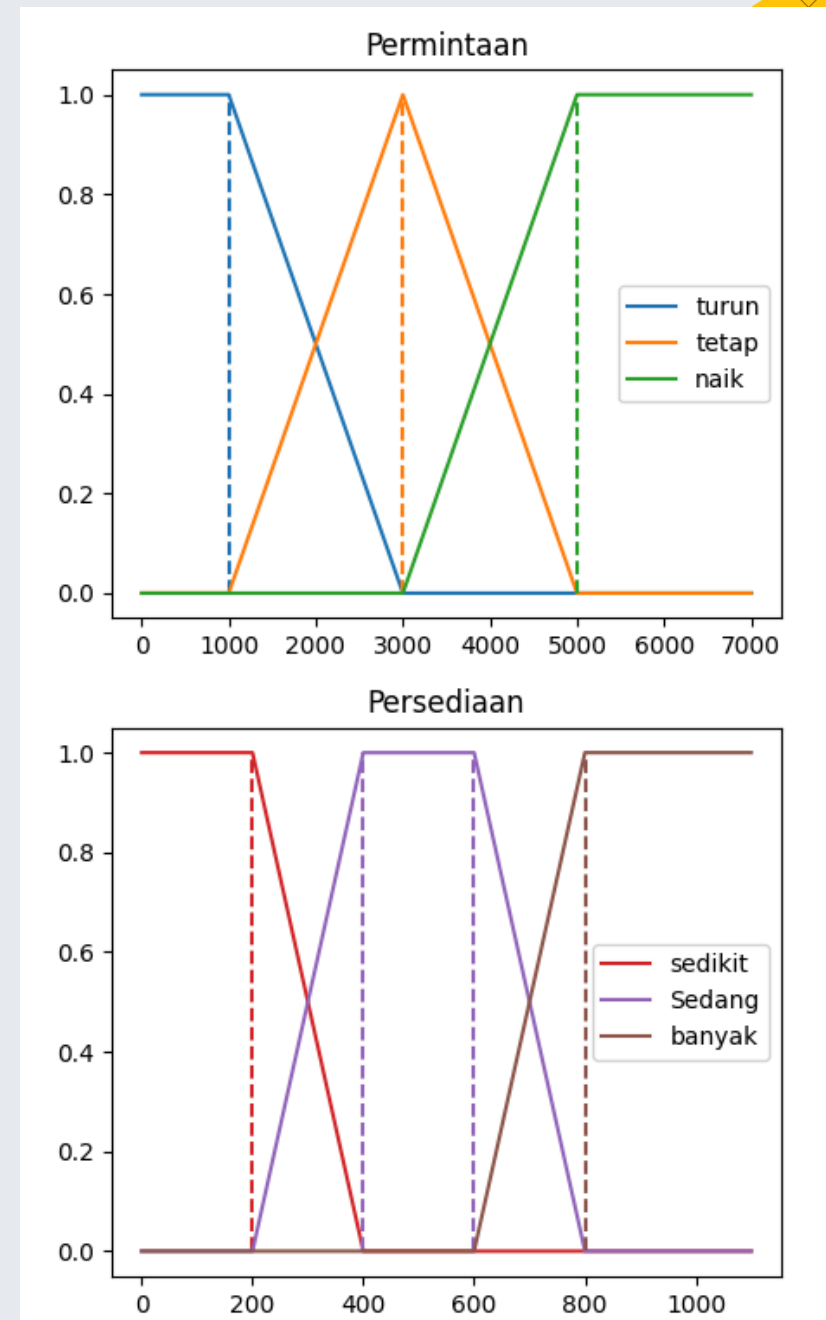
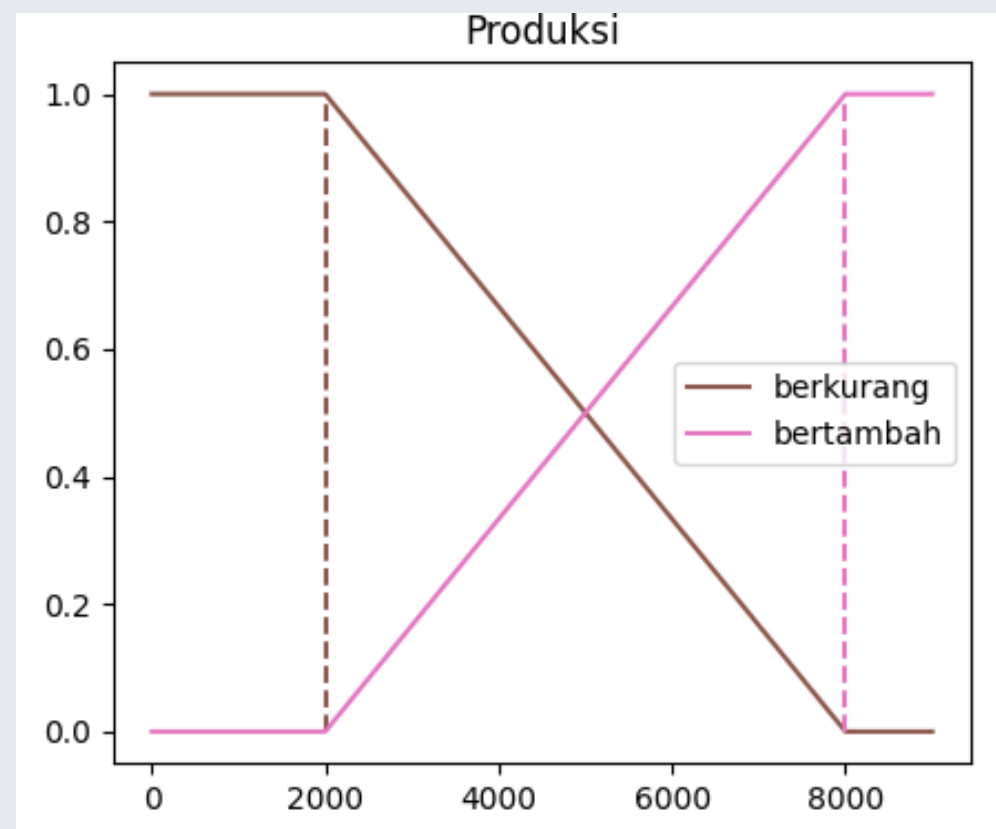
Nama:los Wldria Saragih

NIM : 221011400558

Kelas: 05TPLM007

Soal Latihan

1. Buat Implementasi dengan menggunakan python untuk menghitung Fuzzy Inference System berikut dengan metode tsukamoto



Rule:

1. Jika Permintaan Turun dan Persediaan Sedikit maka produksi Bertambah
2. Jika Permintaan Turun dan Persediaan Sedang maka produksi Berkurang
3. Jika Permintaan Turun dan Persediaan Banyak maka produksi Berkurang
4. Jika Permintaan Tetap dan Persediaan Sedikit maka produksi Bertambah
5. Jika Permintaan Tetap dan Persediaan Sedang maka produksi Berkurang
6. Jika Permintaan Tetap dan Persediaan Banyak maka produksi Berkurang
7. Jika Permintaan Naik dan Persediaan Sedikit maka produksi Bertambah
8. Jika Permintaan Naik dan Persediaan Sedang maka produksi Bertambah
9. Jika Permintaan Naik dan Persediaan Banyak maka produksi Berkurang

1. Langkah Perhitungan Manual

1. Nilai Input

- Permintaan (x_1): 3000 unit
- Persediaan (x_2): 500 unit

2. Fungsi Keanggotaan

a. Permintaan (x_1)

- Turun:

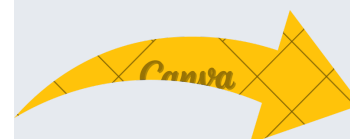
$$\mu_{\text{turun}} = \frac{4000 - x_1}{4000 - 2000} = \frac{4000 - 3000}{2000} = 0.5$$

- Tetap:

$$\mu_{\text{tetap}} = 1 - \left| \frac{x_1 - 3000}{2000} \right| = 1 - \frac{0}{2000} = 1$$

- Naik:

$$\mu_{\text{naik}} = 0 \quad (\text{karena } x_1 < 4000)$$



b. Persediaan (x_2)

- Sedikit:

$$\mu_{\text{sedikit}} = \frac{600 - x_2}{600} = \frac{600 - 500}{600} = 0.1667$$

- Sedang:

$$\mu_{\text{sedang}} = 1 - \left| \frac{x_2 - 600}{400} \right| = 1 - \frac{100}{400} = 0.75$$

- Banyak:

$$\mu_{\text{banyak}} = 0 \quad (\text{karena } x_2 < 800)$$

Langkah Perhitungan Manual

3. Inferensi Aturan

Gunakan $\min(\mu_{\text{permintaan}}, \mu_{\text{persediaan}})$ untuk setiap aturan:

- **Aturan 1:** Permintaan **Turun** dan Persediaan **Sedikit** → Produksi **Bertambah**

$$\alpha_1 = \min(0.5, 0.1667) = 0.1667, z_1 = 8000$$

- **Aturan 2:** Permintaan **Turun** dan Persediaan **Sedang** → Produksi **Berkurang**

$$\alpha_2 = \min(0.5, 0.75) = 0.5, z_2 = 2000$$

- **Aturan 3:** Permintaan **Tetap** dan Persediaan **Sedikit** → Produksi **Bertambah**

$$\alpha_3 = \min(1, 0.1667) = 0.1667, z_3 = 8000$$

- **Aturan 4:** Permintaan **Tetap** dan Persediaan **Sedang** → Produksi **Berkurang**

$$\alpha_4 = \min(1, 0.75) = 0.75, z_4 = 2000$$

4. Defuzzifikasi

Rumus defuzzifikasi:

$$z = \frac{\sum(\alpha_i \cdot z_i)}{\sum \alpha_i}$$

Substitusi:

$$z = \frac{(0.1667 \cdot 8000) + (0.5 \cdot 2000) + (0.1667 \cdot 8000) + (0.75 \cdot 2000)}{0.1667 + 0.5 + 0.1667 + 0.75}$$

Hitung pembilang:

$$z = \frac{1333.6 + 1000 + 1333.6 + 1500}{1.5834} = \frac{5167.2}{1.5834} = 4677.083$$

Hasil Akhir

Produksi: **4677.083 unit.**


```
# Derajat keanggotaan input
permintaan_turun = fuzz.interp_membership(x_permintaan, turun, permintaan_input)
permintaan_tetap = fuzz.interp_membership(x_permintaan, tetap, permintaan_input)
permintaan_naik = fuzz.interp_membership(x_permintaan, naik, permintaan_input)

persediaan_sedikit = fuzz.interp_membership(x_persediaan, sedikit, persediaan_input)
persediaan_sedang = fuzz.interp_membership(x_persediaan, sedang, persediaan_input)
persediaan_banyak = fuzz.interp_membership(x_persediaan, banyak, persediaan_input)

# Inferensi aturan
rules = []
rules.append((min(permintaan_turun, persediaan_sedikit), bertambah))
rules.append((min(permintaan_turun, persediaan_sedang), berkurang))
rules.append((min(permintaan_turun, persediaan_banyak), berkurang))
rules.append((min(permintaan_tetap, persediaan_sedikit), bertambah))
rules.append((min(permintaan_tetap, persediaan_sedang), berkurang))
rules.append((min(permintaan_tetap, persediaan_banyak), berkurang))
rules.append((min(permintaan_naik, persediaan_sedikit), bertambah))
rules.append((min(permintaan_naik, persediaan_sedang), bertambah))
rules.append((min(permintaan_naik, persediaan_banyak), berkurang))

# Perhitungan output
produk = []
for rule_strength, output in rules:
    z = fuzz.defuzz(x_produksi, output, 'centroid')
    produk.append(rule_strength * z)

hasil = sum(produk) / sum(rule[0] for rule in rules)
return hasil

# Contoh Penghitungan
permintaan_input = 2500 # Contoh input permintaan
persediaan_input = 300 # Contoh input persediaan
produksi_output = inferensi_tsukamoto(permintaan_input, persediaan_input)

print("Output produksi (unit):", produksi_output)
```

91 ✓ 0.2s

Output produksi (unit): 4677.083333333331

```
# Derajat keanggotaan input
permintaan_turun = fuzz.interp_membership(x_permintaan, turun, permintaan_input)
permintaan_tetap = fuzz.interp_membership(x_permintaan, tetap, permintaan_input)
permintaan_naik = fuzz.interp_membership(x_permintaan, naik, permintaan_input)

persediaan_sedikit = fuzz.interp_membership(x_persediaan, sedikit, persediaan_input)
persediaan_sedang = fuzz.interp_membership(x_persediaan, sedang, persediaan_input)
persediaan_banyak = fuzz.interp_membership(x_persediaan, banyak, persediaan_input)

# Inferensi aturan
rules = []
rules.append((min(permintaan_turun, persediaan_sedikit), bertambah))
rules.append((min(permintaan_turun, persediaan_sedang), berkurang))
rules.append((min(permintaan_turun, persediaan_banyak), berkurang))
rules.append((min(permintaan_tetap, persediaan_sedikit), bertambah))
rules.append((min(permintaan_tetap, persediaan_sedang), berkurang))
rules.append((min(permintaan_tetap, persediaan_banyak), berkurang))
rules.append((min(permintaan_naik, persediaan_sedikit), bertambah))
rules.append((min(permintaan_naik, persediaan_sedang), bertambah))
rules.append((min(permintaan_naik, persediaan_banyak), berkurang))

# Perhitungan output
produk = []
for rule_strength, output in rules:
    z = fuzz.defuzz(x_produksi, output, 'centroid')
    produk.append(rule_strength * z)

hasil = sum(produk) / sum(rule[0] for rule in rules)
return hasil

# Contoh Penghitungan
permintaan_input = 2500 # Contoh input permintaan
persediaan_input = 300 # Contoh input persediaan
produksi_output = inferensi_tsukamoto(permintaan_input, persediaan_input)

print("Output produksi (unit):", produksi_output)
```

91 ✓ 0.2s

Output produksi (unit): 4677.083333333331