### Nama : Ariq Abdurrahman NPM : G1A021036

**Mata Kuliah : Logika Fuzzy**

# A . Fungsi Keanggotaan

Fungsi keanggotaan adalah komponen penting dalam sistem fuzzy yang menentukan sejauh mana nilai input masuk ke dalam kategori fuzzy tertentu. Dalam sistem ini, kami mendefinisikan fungsi keanggotaan untuk masing-masing variabel input dan output.

1. **Variabel Input dan Output**

### Suhu (°C):

1. dingin: [22, 22, 23, 25]

2. normal: [23, 26, 29, 30]

3. panas: [29, 32, 32, 32]

### Kebisingan (dB):

1. rendah: [55, 55, 60, 65]

2. sedang: [60, 70, 80, 90]

3. tinggi: [85, 90, 90, 90]

### Pencahayaan (lux):

1. redup: [150, 150, 200, 250]

2. normal: [200, 300, 400, 500]

3. terang: [450, 500, 500, 500]

### Produksi (unit):

1. rendah: [120, 130, 135, 140]

2. sedang: [135, 145, 150, 155]

3. tinggi: [150, 155, 160, 160]

## Kode untuk Definisi Fungsi Keanggotaan

# 1. Definisikan variabel input dan output

suhu = ctrl.Antecedent(np.arange(22, 33, 1), 'suhu') # Suhu dalam derajat Celsius

kebisingan = ctrl.Antecedent(np.arange(55, 91, 1), 'kebisingan') #

Kebisingan dalam dB

pencahayaan = ctrl.Antecedent(np.arange(150, 501, 1), 'pencahayaan') # Pencahayaan dalam lux

produksi = ctrl.Consequent(np.arange(120, 161, 1), 'produksi') #

Produksi dalam unit

# 2. Definisikan fungsi keanggotaan trapezoid untuk suhu suhu['dingin'] = fuzz.trapmf(suhu.universe, [22, 22, 23, 25])

suhu['normal'] = fuzz.trapmf(suhu.universe, [23, 26, 29, 30])

suhu['panas'] = fuzz.trapmf(suhu.universe, [29, 32, 32, 32])

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| # 3. Definisikan fungsi keanggotaan trapezoid untuk kebisingan | | | | | | | |
| kebisingan['rendah'] 65])  kebisingan['sedang'] | | =  = | fuzz.trapmf(kebisingan.universe,  fuzz.trapmf(kebisingan.universe, | [55,  [60, | | 55,  70, | 60,  80, |
| 90])  kebisingan['tinggi'] | | = | fuzz.trapmf(kebisingan.universe, | [85, | | 90, | 90, |
| 90]) | |  |  |  | |  |  |
| # 4. Definisikan fungsi keanggotaan trapezoid untuk pencahayaan pencahayaan['redup'] = fuzz.trapmf(pencahayaan.universe, [150, 200, 250])  pencahayaan['normal'] = fuzz.trapmf(pencahayaan.universe, [200, | | | | | | | 150,  300, |
| 400, 500])  pencahayaan['terang'] = fuzz.trapmf(pencahayaan.universe, | | | | | [450, | | 500, |
| 500, 500]) | | | | |  | |  |
| # 5. Definisikan fungsi keanggotaan trapezoid untuk produksi | | | | | | | |
| produksi['rendah'] 140])  produksi['sedang'] | =  = | fuzz.trapmf(produksi.universe, [120,  fuzz.trapmf(produksi.universe, [135, | | | 130,  145, | | 135,  150, |
| 155])  produksi['tinggi'] | = | fuzz.trapmf(produksi.universe, [150, | | | 155, | | 160, |
| 160]) |  |  | | |  | |  |

# 6. Gambar fungsi keanggotaan suhu.view(title="Fungsi Keanggotaan Suhu")

kebisingan.view(title="Fungsi Keanggotaan Kebisingan") pencahayaan.view(title="Fungsi Keanggotaan Pencahayaan") produksi.view(title="Fungsi Keanggotaan Produksi") plt.show()

## Visualisasi Fungsi Keanggotaan

Fungsi keanggotaan untuk masing-masing variabel akan digambarkan menggunakan fungsi .view(), memberikan gambaran visual yang jelas tentang bagaimana setiap nilai terdistribusi dalam himpunan fuzzy.

|  |  |  |  |
| --- | --- | --- | --- |
| **B.** **27 Aturan Fuzzy** |  | | |
| **A. Kode untuk Aturan Fuzzy** |  |  |  |
| # 7. Definisikan aturan fuzzy rules = []  # Kombinasi untuk suhu 'dingin'  rules.append(ctrl.Rule(suhu['dingin'] | & | kebisingan['rendah'] | & |

pencahayaan['redup'], produksi['rendah'])) rules.append(ctrl.Rule(suhu['dingin'] & kebisingan['rendah'] & pencahayaan['normal'], produksi['rendah'])) rules.append(ctrl.Rule(suhu['dingin'] & kebisingan['rendah'] & pencahayaan['terang'], produksi['sedang'])) rules.append(ctrl.Rule(suhu['dingin'] & kebisingan['sedang'] & pencahayaan['redup'], produksi['rendah'])) rules.append(ctrl.Rule(suhu['dingin'] & kebisingan['sedang'] & pencahayaan['normal'], produksi['rendah']))

rules.append(ctrl.Rule(suhu['dingin'] & kebisingan['sedang'] & pencahayaan['terang'], produksi['sedang'])) rules.append(ctrl.Rule(suhu['dingin'] & kebisingan['tinggi'] & pencahayaan['redup'], produksi['rendah'])) rules.append(ctrl.Rule(suhu['dingin'] & kebisingan['tinggi'] & pencahayaan['normal'], produksi['sedang'])) rules.append(ctrl.Rule(suhu['dingin'] & kebisingan['tinggi'] & pencahayaan['terang'], produksi['sedang']))

# Kombinasi untuk suhu 'normal'

rules.append(ctrl.Rule(suhu['normal'] & kebisingan['rendah'] & pencahayaan['redup'], produksi['rendah'])) rules.append(ctrl.Rule(suhu['normal'] & kebisingan['rendah'] & pencahayaan['normal'], produksi['sedang'])) rules.append(ctrl.Rule(suhu['normal'] & kebisingan['rendah'] & pencahayaan['terang'], produksi['tinggi'])) rules.append(ctrl.Rule(suhu['normal'] & kebisingan['sedang'] & pencahayaan['redup'], produksi['sedang'])) rules.append(ctrl.Rule(suhu['normal'] & kebisingan['sedang'] & pencahayaan['normal'], produksi['sedang'])) rules.append(ctrl.Rule(suhu['normal'] & kebisingan['sedang'] & pencahayaan['terang'], produksi['tinggi'])) rules.append(ctrl.Rule(suhu['normal'] & kebisingan['tinggi'] & pencahayaan['redup'], produksi['sedang'])) rules.append(ctrl.Rule(suhu['normal'] & kebisingan['tinggi'] & pencahayaan['normal'], produksi['sedang'])) rules.append(ctrl.Rule(suhu['normal'] & kebisingan['tinggi'] & pencahayaan['terang'], produksi['tinggi']))

# Kombinasi untuk suhu 'panas'

rules.append(ctrl.Rule(suhu['panas'] & kebisingan['rendah'] & pencahayaan['redup'], produksi['sedang'])) rules.append(ctrl.Rule(suhu['panas'] & kebisingan['rendah'] & pencahayaan['normal'], produksi['tinggi'])) rules.append(ctrl.Rule(suhu['panas'] & kebisingan['rendah'] & pencahayaan['terang'], produksi['tinggi'])) rules.append(ctrl.Rule(suhu['panas'] & kebisingan['sedang'] & pencahayaan['redup'], produksi['sedang'])) rules.append(ctrl.Rule(suhu['panas'] & kebisingan['sedang'] & pencahayaan['normal'], produksi['tinggi'])) rules.append(ctrl.Rule(suhu['panas'] & kebisingan['sedang'] & pencahayaan['terang'], produksi['tinggi'])) rules.append(ctrl.Rule(suhu['panas'] & kebisingan['tinggi'] & pencahayaan['redup'], produksi['sedang'])) rules.append(ctrl.Rule(suhu['panas'] & kebisingan['tinggi'] & pencahayaan['normal'], produksi['tinggi'])) rules.append(ctrl.Rule(suhu['panas'] & kebisingan['tinggi'] & pencahayaan['terang'], produksi['tinggi']))

# C. Derajat Keanggotaan

Derajat keanggotaan menghitung seberapa baik input yang diberikan sesuai dengan fungsi keanggotaan yang telah didefinisikan.

**A. Kode untuk Menghitung Derajat Keanggotaan**

# 8. Setup sistem kontrol fuzzy produksi\_ctrl = ctrl.ControlSystem(rules)

produksi\_simulasi = ctrl.ControlSystemSimulation(produksi\_ctrl) # 9. Masukkan nilai input (contoh)

suhu\_value = 28 # Suhu dalam derajat Celsius

kebisingan\_value = 90 # Kebisingan dalam dB pencahayaan\_value = 300 # Pencahayaan dalam lux # Masukkan nilai input ke dalam sistem produksi\_simulasi.input['suhu'] = suhu\_value

produksi\_simulasi.input['kebisingan'] = kebisingan\_value produksi\_simulasi.input['pencahayaan'] = pencahayaan\_value # Hitung derajat keanggotaan suhu.view(sim=produksi\_simulasi) kebisingan.view(sim=produksi\_simulasi) pencahayaan.view(sim=produksi\_simulasi) produksi.view(sim=produksi\_simulasi)

# D. α-Predikat untuk Setiap Aturan

α-predikat digunakan untuk menghitung output berdasarkan aturan fuzzy yang telah ditetapkan.

**Kode untuk Menghitung Output**

# 10. Hitung hasil output produksi\_simulasi.compute()

# 11. Lihat hasil output produksi

output\_produksi = produksi\_simulasi.output['produksi']print(f"Output produksi: {output\_produksi:.2f}")

# E. Rata-rata Jumlah Produk

Metode defuzzy yang digunakan adalah Weighted Average, yang memberikan hasil akhir berdasarkan kontribusi dari setiap aturan.

**1. Kode untuk Visualisasi Hasil Output**

# 12. Gambar hasil output produksi.view(sim=produksi\_simulasi) plt.show()

**NOTE : link ke NoteBook (https://colab.research.google.com/drive/1v9v6- JTQkmi5uv\_b\_QxEBYLr\_ycLpCdt?usp=sharing)**