

LAPORAN FUZZY LOGIC

Dibuat untuk memenuhi tugas mata kuliah “Kecerdasan Buatan”



Nama Anggota (Kelompok 18 - IF4408) :

- 1. Asfa Amalia Dinata - 1301204498**
- 2. Mohamad Zulistiyan - 1301204037**
- 3. Kelvyn lukito - 1301200104**

**Universitas Telkom
Bandung**

DAFTAR ISI

- A. Definisi Fuzzy Logic
- B. Rumusan Masalah
- C. Proses Pengkodean Fuzzy Logic
 - Fuzzifikasi
 - Inferensi
 - Defuzzifikasi
- D. Kesimpulan

A. Definisi Fuzzy Logic

Reasoning merupakan salah satu cabang dalam keilmuan Kecerdasan Buatan yang merepresentasikan masalah ke dalam basis pengetahuan dan melakukan proses penalaran untuk menemukan solusi. Dalam Kecerdasan Buatan, teknik *reasoning* menggunakan *logic* atau bahasa yang dipahami komputer. Terdapat lima jenis *logic* dalam *reasoning*, yaitu *Propositional logic*, *First-order logic*, *Temporal logic*, *Probability theory*, dan *Fuzzy logic*.

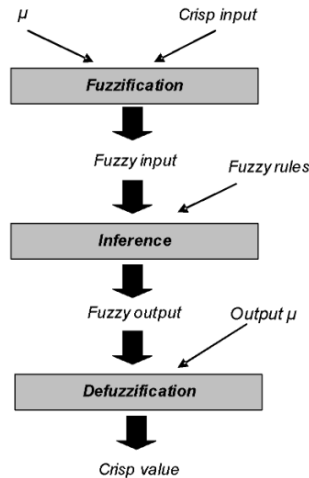
Tiga *logic* yang pertama, yaitu *propositional logic*, *first order logic*, dan *temporal logic* digunakan untuk masalah yang pasti: benar, salah, atau tidak diketahui. Sedangkan dua *logic* lainnya, yaitu *probability theory* dan *fuzzy logic*, digunakan untuk menyelesaikan masalah yang tidak pasti.

Fuzzy secara bahasa diartikan sebagai kabur atau samar yang artinya suatu nilai dapat bernilai benar atau salah secara bersamaan. Logika *fuzzy* memiliki derajat keanggotaan dalam rentang 0 hingga 1 dan logika *fuzzy* menunjukkan sejauh mana suatu nilai benar dan sejauh mana suatu nilai itu salah. Logika *fuzzy* adalah suatu cara yang tepat untuk memetakan suatu ruang input ke dalam suatu ruang output dan mempunyai nilai kontinu. *Fuzzy* dinyatakan dalam derajat keanggotaan dan derajat kebenaran. Oleh sebab itu sesuatu dapat dikatakan sebagian benar dan sebagian salah pada waktu yang sama (Kusumadewi, 2004).

Sistem fuzzy mempunyai beberapa keuntungan bila dibandingkan dengan sistem tradisional. Contohnya yaitu pada jumlah aturan yang dipergunakan. Pemrosesan awal sejumlah besar nilai menjadi sebuah nilai derajat keanggotaan pada sistem fuzzy mengurangi jumlah nilai yang harus dipergunakan pengontrol untuk membuat suatu keputusan. Keuntungan lainnya adalah sistem fuzzy mempunyai kemampuan penalaran yang mirip dengan kemampuan penalaran manusia. Hal ini disebabkan karena sistem fuzzy mempunyai kemampuan untuk memberikan respon berdasarkan informasi yang bersifat kualitatif, tidak akurat, dan ambigu.

Ada beberapa alasan penggunaan Logika Fuzzy :

- Logika fuzzy sangat fleksibel.
- Logika fuzzy memiliki toleransi
- Konsep logika fuzzy mudah dimengerti. Konsep matematis yang mendasari penalaran fuzzy sangat sederhana dan mudah dipahami.
- Logika fuzzy mampu memodelkan fungsi-fungsi nonlinear yang sangat kompleks.
- Logika fuzzy dapat bekerja sama dengan teknik-teknik kendali secara konvensional.



Fuzzy-Based System terdiri dari tiga komponen, yaitu: *Fuzzification*, *Inference*, dan *Defuzzification*. Seperti pada gambar ini. *Fuzzification* mengubah masukan yang nilai kebenarannya bersifat pasti (*crisp input*) menjadi *fuzzy input*, yang berupa nilai linguistik, di mana semantiknya ditentukan berdasarkan fungsi keanggotaan tertentu. *Inference* melakukan penalaran menggunakan *fuzzy input* dan *fuzzy rules* yang telah ditentukan sehingga menghasilkan *fuzzy output*. Terdapat dua model inferensi yang bisa digunakan, yaitu: Model Mamdani dan Model Sugeno. Model Mamdani lebih intuitif mengikuti perasaan manusia sedangkan Model Sugeno lebih sesuai untuk sistem kontrol. *Defuzzification* mengubah *fuzzy output* menjadi *crisp value* berdasarkan fungsi keanggotaan yang telah ditentukan.

B. Rumusan Masalah

Diberikan sebuah file bernama “bengkel.xlsx” berupa himpunan data 100 bengkel mobil yang ada di kota Bandung dengan dua atribut: Kualitas Servis (bilangan real 1-100; semakin tinggi semakin baik) dan Harga (bilangan real 1-10, semakin tinggi semakin mahal). Sebuah sistem berbasis Fuzzy Logic digunakan untuk memilih 10 bengkel terbaik di kota Bandung. Sistem membaca masukan file bengkel.xlsx dan mengeluarkan output berupa sebuah file peringkat.xlsx yang berisi 10 nomor atau ID bengkel terbaik beserta skor-nya.

1. Fungsi Keanggotaan

Terdapat dua atribut yang diperhitungkan dalam studi kasus ini, yaitu kualitas servis dan harga servis yang masing-masing memiliki batasan tersendiri. Batasan untuk fungsi keanggotaan dari atribut kualitas servis sendiri dirumuskan seperti di bawah ini :

Sangat buruk : $0 \leq \text{nilai} < 30$

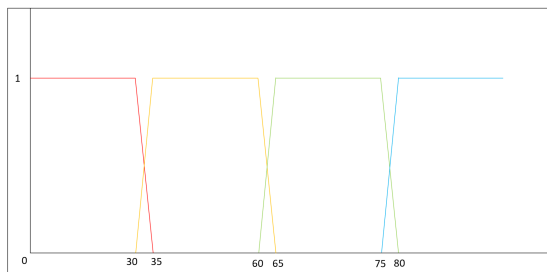
Buruk : $25 < \text{nilai} < 55$

Baik : $50 < \text{nilai} < 80$

Sangat baik : $75 < \text{nilai} \leq 100$

Batasan fungsi keanggotaan dari atribut kualitas servis di atas digambarkan dengan grafik fungsi keanggotaan yang berbentuk trapesium seperti pada gambar di bawah ini. Warna

merah untuk sangat buruk, kuning untuk buruk, hijau untuk baik, dan biru untuk sangat baik.



Sementara untuk batasan fungsi keanggotaan dari atribut harga servis dirumuskan seperti di bawah ini :

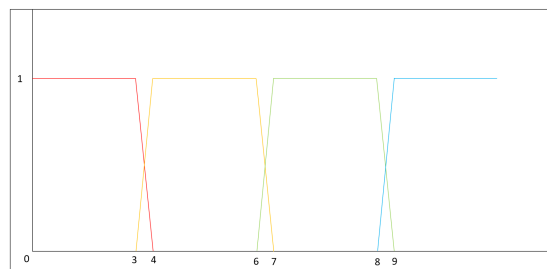
Sangat murah : $0 \leq \text{harga} < 4$

Murah : $3 < \text{harga} < 7$

Mahal : $6 < \text{harga} < 9$

Sangat mahal : $8 < \text{harga} \leq 10$

Batasan fungsi keanggotaan dari atribut harga servis di atas digambarkan dengan grafik fungsi keanggotaan yang berbentuk trapesium seperti pada gambar di bawah ini. Warna merah untuk sangat murah, kuning untuk murah, hijau untuk mahal, dan biru untuk sangat mahal.



2. Fungsi Inferensi

Terdapat 4 variable linguistik untuk masing masing variable pada kualitas servis dan harga, sehingga akan menghasilkan aturan inferensi, Aturan tersebut diatur berdasarkan penilaian kualitas servis yang yaitu sangat buruk, buruk, baik, sangat baik dan juga penilaian harga yaitu sangat mahal, mahal murah, sangat murah. Aturan tersebut akan dikelompokkan menjadi suatu bentuk fungsi keanggotaan seperti tidak rekom, rekomendasi, sangat direkomendasi.

ATURAN FUZZY NILAI KELAYAKAN (NK)				
Harga\Pelayanan	sangat buruk	buruk	baik	sangat baik
sangat mahal	tidak rekom	tidak rekom	rekomendasi	sangat direkomendasi
mahal	tidak rekom	tidak rekom	rekomendasi	sangat direkomendasi
murah	tidak rekom	tidak rekom	sangat direkomendasi	sangat direkomendasi
sangat murah	tidak rekom	rekomendasi	sangat direkomendasi	sangat direkomendasi

3. Nilai Sugeno

Pada metode Sugeno, terdapat variabel random. Pada studi kasus bengkel ini digunakan nilai random Sugeno 30 untuk tidak rekomendasi, 75 untuk rekomendasi, dan 100 untuk sangat rekomendasi.

C. IMPLEMENTASI

1. Fuzzifikasi

Saat melakukan proses fuzzy, kita menentukan besaran dan nama bahasa dari setiap input serta bentuk dan limit dari fungsi keanggotaan input untuk mendapatkan crisp Value dari setiap data yang tersedia.

1.1. Nama dan jumlah linguistik

Dalam sistem rekomendasi bengkel terdapat dua variabel linguistik, yaitu kualitas servis dan harga. Kami menggunakan dua input dan lima variabel linguistik untuk kualitas servis dan dua input dan tiga variabel linguistik untuk harga sehingga rentang nilai yang dihasilkan dapat mencapai nilai maksimum untuk setiap variabel linguistik.

1.1.1 Servis

Terdapat variabel linguistik pada function tersebut adalah sangat buruk, buruk, baik, sangat baik

```
nilaiServis = {'sangatburuk' : 0, 'buruk' : 0, 'baik' : 0, 'sangatbaik' : 0}
```

1.1.2 Harga

Terdapat variabel linguistik pada function tersebut adalah sangat buruk, buruk, baik, sangat baik

```
nilaiHarga = {'sangatmurah' : 0, 'murah' : 0, 'mahal' : 0, 'sangatmahal' : 0 }
```

1.2. Batasan Fungsi Keanggotaan

1.2.1. Kualitas Servis

Pada bagian sebelumnya, telah dijelaskan apabila pada tiap atribut yang menjadi batasan-batasannya. Di bawah ini merupakan implementasi batasan pada atribut kualitas servis yang stagnan di nilai 1 sebelum derajat kebenarannya turun.

```
crispValue = []

for x in range(100):
    nilaiServis = {'sangatburuk' : 0, 'buruk' : 0, 'baik' : 0, 'sangatbaik' : 0}
    a,b,c,d,e,f = 30, 35, 60, 65, 75, 80
    #Fuzzification
    #Servis
    if servis[x] <= a:
        nilaiServis['sangatburuk'] = 1
    elif b <= servis[x] <= c:
        nilaiServis['buruk'] = 1
    elif d <= servis[x] <= e:
        nilaiServis['baik'] = 1
    elif servis[x] >= f:
        nilaiServis['sangatbaik'] = 1
```

Kemudian, terdapat implementasi untuk bagian diagram yang membentuk garis miring naik atau turun. Pada garis miring yang naik, menggambarkan derajat kepercayaan yang perlahan naik dari 0 hingga 1. Sebaliknya, pada garis miring turun menggambarkan derajat kepercayaan yang perlahan turun dari 1 hingga 0.

```
if a < servis[x] < b:
    nilaiServis['sangatburuk'] = -(servis[x] - b) / (b - a)
    nilaiServis['buruk'] = (servis[x] - a) / (b - a)
elif c < servis[x] < d:
    nilaiServis['buruk'] = -(servis[x] - d) / (d - c)
    nilaiServis['baik'] = (servis[x] - c) / (d - c)
elif e < servis[x] < f:
    nilaiServis['baik'] = -(servis[x] - f) / (f - e)
    nilaiServis['sangatbaik'] = (servis[x] - e) / (f - e)
```


1.2.3 Harga Servis

```
nilaiHarga = {'sangatmurah' : 0, 'murah' : 0, 'mahal' : 0, 'sangatmahal' : 0 }
a,b,c,d,e,f = 3, 4, 6, 7, 8, 9
#Fuzzification
#Harga
if harga[x] <= a:
    nilaiHarga['sangatmurah'] = 1
elif harga[x] >= b and harga[x] <= c:
    nilaiHarga['murah'] = 1
elif harga[x] >= d and harga[x] <= e:
    nilaiHarga['mahal'] = 1
elif harga[x] >= f:
    nilaiHarga['sangatmahal'] = 1
```

2. Inferensi

Pada tahap inferensi, telah dijelaskan pada bagian-bagian sebelumnya kombinasi dari hasil fuzzifikasi untuk dua atribut kualitas servis dan harga servis membentuk nilai-nilai inferensi yaitu tidak rekomendasi, rekomendasi, dan sangat rekomendasi.

2.1. Tidak rekomendasi

```
#inference
TR = [] #Tidak Rekomendasi
#if nilaiHarga['sangatmurah'] and nilaiServis['sangatburuk']:
TR.append(min(nilaiHarga['sangatmurah'], nilaiServis['sangatburuk']))
#if nilaiHarga['murah'] and nilaiServis['sangatburuk']:
TR.append(min(nilaiHarga['murah'], nilaiServis['sangatburuk']))
#if nilaiHarga['mahal'] and nilaiServis['sangatburuk']:
TR.append(min(nilaiHarga['mahal'], nilaiServis['sangatburuk']))
#if nilaiHarga['sangatmahal'] and nilaiServis['sangatburuk']:
TR.append(min(nilaiHarga['sangatmahal'], nilaiServis['sangatburuk']))
#if nilaiHarga['sangatmahal'] and nilaiServis['buruk']:
TR.append(min(nilaiHarga['sangatmahal'], nilaiServis['buruk']))
#if nilaiHarga['mahal'] and nilaiServis['buruk']:
TR.append(min(nilaiHarga['mahal'], nilaiServis['buruk']))

if TR != []:
    maxTR = max(TR)
    print(maxTR)
```

2.2. Rekomendasi

```
R = [] #Rekomendasi
#if nilaiHarga['sangatmurah'] and nilaiServis['buruk']:
R.append(min(nilaiHarga['sangatmurah'],nilaiServis['buruk']))
#if nilaiHarga['sangatmahal'] and nilaiServis['baik']:
R.append(min(nilaiHarga['sangatmahal'],nilaiServis['baik']))
#if nilaiHarga['mahal'] and nilaiServis['baik']:
R.append(min(nilaiHarga['mahal'],nilaiServis['baik']))
#if nilaiHarga['murah'] and nilaiServis['buruk']:
R.append(min(nilaiHarga['murah'],nilaiServis['buruk']))

if R != []:
    maxR = max(R)
    print(maxR)
```

2.3 Sangat rekomendasi

```
SR = [] #Sangat Rekomendasi
#if nilaiHarga['sangatmurah'] and nilaiServis['baik']:
SR.append(min(nilaiHarga['sangatmurah'],nilaiServis['baik']))
#if nilaiHarga['murah'] and nilaiServis['baik']:
SR.append(min(nilaiHarga['murah'],nilaiServis['baik']))
#if nilaiHarga['sangatmurah'] and nilaiServis['sangatbaik']:
SR.append(min(nilaiHarga['sangatmurah'],nilaiServis['sangatbaik']))
#if nilaiHarga['murah'] and nilaiServis['sangatbaik']:
SR.append(min(nilaiHarga['murah'],nilaiServis['sangatbaik']))
#if nilaiHarga['mahal'] and nilaiServis['sangatbaik']:
SR.append(min(nilaiHarga['mahal'],nilaiServis['sangatbaik']))
#if nilaiHarga['sangatmahal'] and nilaiServis['sangatbaik']:
SR.append(min(nilaiHarga['sangatmahal'],nilaiServis['sangatbaik']))

if SR != []:
    maxSR = max(SR)
    print(maxSR)
```

3. Defuzzifikasi

Pada tahap defuzzifikasi, digunakan model Sugeno. Implementasi Sugeno model untuk defuzzifikasi dapat dilihat pada gambar di bawah ini.

```
pembagi = maxR + maxSR + maxTR
result = ((30*maxTR)+(75*maxR)+(100*maxSR)) / pembagi
crispValue.append(result)
print('id    Servis    Harga    Hasil')
print(id[x], ' ', servis[x], ' ', harga[x], ' ', result)
print('='*40)
```

4. Kesimpulan

Beberapa hal terkait penjelasan mengenai program *fuzzy logic* yang telah dibuat, maka dapat disimpulkan bahwa dengan menggunakan metode Sugeno untuk melakukan uji kelayakan bengkel dengan data yang diberikan, berhasil didapatkan keluaran hasil berupa 10 data bengkel dengan nilai kelayakan terbaik.

	id	servis	harga	Kelayakan
59	60	79	6	100.0
91	92	83	3	100.0
68	69	86	10	100.0
23	24	100	9	100.0
41	42	94	10	100.0
78	79	87	9	100.0
16	17	70	3	100.0
15	16	82	6	100.0
90	91	98	3	100.0
12	13	80	3	100.0