

LAPORAN TUGAS PEMROGRAMAN 2

Dibuat untuk memenuhi salah satu tugas mata kuliah Pengantar Kecerdasan Buatan

Oleh:

Priyan Fadhil Supriyadi / 1301190442

Claudia Mei Serin Sitio / 1301190424

Vena Erla Candrika / 1301194040



**PROGRAM STUDI INFORMATIKA
FAKULTAS INFORMATIKA
2021**

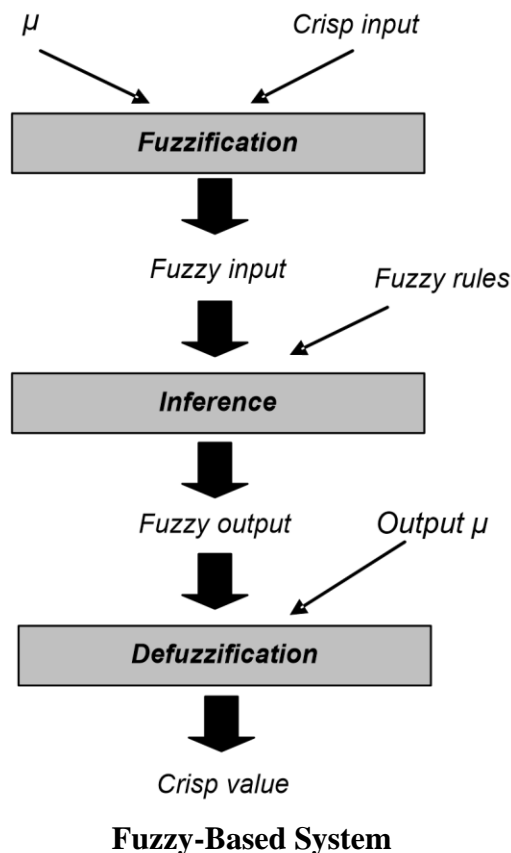
DAFTAR ISI

DAFTAR ISI	2
I. Pendahuluan	3
II. Hal yang diobservasi:	3
1. Read dan Load Data	4
2. Jumlah dan Nama Linguistik setiap input	4
3. Bentuk dan Batas Fungsi Keanggotaan Input	4
4. Fuzzifikasi	8
5. Inferensi	9
6. Metode Defuzzifikasi	11
7. Bentuk dan Batas Fungsi Keanggotaan Output (sesuai metode defuzzifikasi)	12
III. Nilai-nilai parameter Fuzzy yang paling optimum	14
- Fungsi keanggotaan makanan terdapat parameter:	14
IV. Kesimpulan	15
V. Link Presentasi	15
References	16

I. Pendahuluan

Dalam menyelesaikan suatu masalah dapat menggunakan beberapa teknik tertentu. Salah satunya ialah dengan merepresentasikan masalah tersebut ke dalam pengetahuan dan melakukan proses logika (penalaran) untuk mendapatkan solusi. Teknik inilah yang disebut dengan reasoning. Adapun teknik reasoning memiliki beberapa jenis logic seperti *propositional logic* dan *first order logic* namun hanya untuk masalah yang pasti. Dalam menyelesaikan masalah seperti pembuatan sistem fuzzy logic untuk memilih 10 restoran terbaik di kota A maka *fuzzy logic* merupakan pilihan yang tepat. *Fuzzy logic* merupakan salah satu jenis logic yang terdapat pada teknik reasoning yang dapat digunakan untuk menyelesaikan masalah yang mengandung ketidakpastian. Fuzzy logic juga banyak digunakan untuk membangun sistem kontrol. Dipilihnya logika fuzzy untuk memilih 10 restoran terbaik di kota A antara lain karena logika fuzzy mudah dimengerti, fleksibel, memiliki toleransi terhadap data – data yang tidak pasti, dan logika fuzzy menggunakan bahasa alami. (wibowo, 2015)

II. Hal yang diobservasi:



1. Read dan Load Data

File restoran.xls dimasukkan ke dalam github agar dapat diakses melalui url. Data dibaca dengan menggunakan library pandas dengan memanggil url github tadi dan kemudian menuliskannya dalam bentuk kolom sesuai dengan file excel tersebut.

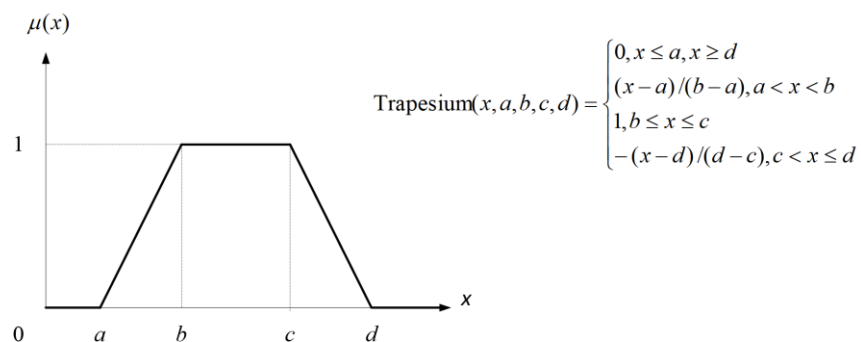
```
url = 'https://github.com/priyanfadhil/simplefuzzy/blob/main/restoran.xlsx?raw=true'
data_resto = pd.read_excel(url)
data_resto.columns = ["id", "pelayanan", "makanan"]
data_resto.head(10)
```

2. Jumlah dan Nama Linguistik setiap input

Pada data file restoran.xls yang berupa himpunan data 100 restoran yang ada di kota A terdapat dua atribut yaitu kualitas pelayanan dan kualitas makanan. Kedua atribut itulah yang akan dijadikan input pada sistem. Input kualitas pelayanan memiliki range bilangan real 1-100. Pada input ini kami menggunakan empat linguistik yang terdiri dari buruk, cukup, bagus, dan sangat bagus. Sedangkan input kualitas makanan memiliki range bilangan real 1-10. Pada input ini kami juga menggunakan empat linguistik yang terdiri dari tidak enak, cukup, enak, dan sangat enak.

3. Bentuk dan Batas Fungsi Keanggotaan Input

Fungsi keanggotaan merupakan fungsi yang memetakan elemen suatu himpunan ke nilai keanggotaan pada interval $[0,1]$. Terdapat beberapa bentuk fungsi keanggotaan pada sistem *fuzzy*. Salah satunya fungsi yang kami gunakan pada sistem ini yaitu fungsi trapesium seperti di bawah ini.

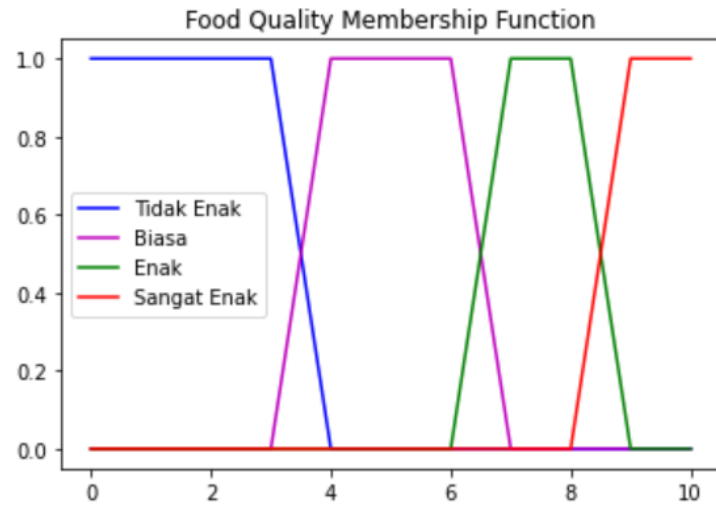


Representasi Kurva Trapesium

Bentuk dan batas fungsi keanggotaan input :

a. Fungsi Keanggotaan Makanan

Bentuk fungsi keanggotaan untuk input makanan yang kami gunakan ialah fungsi trapesium sebagai berikut :



Batas-batas fungsi keanggotaan makanan :

$a = 3$
 $b = 4$
 $c = 6$
 $d = 7$
 $e = 8$
 $f = 9$

1) Tidak Enak berada pada interval $[0,4]$

```
def tidak_enak(makanan):
    if makanan <= a:
        return 1
    if makanan > a and makanan < b :
        return -(makanan - b) / (b - a)
    elif makanan >= b:
        return 0
```

2) Biasa berada pada interval [3,7]

```
def biasa(makanan):  
    if makanan > a and makanan < b:  
        return (makanan - a) / (b - a)  
    if makanan >= b and makanan <= c:  
        return 1  
    if makanan > c and makanan < d:  
        return -(makanan - d) / (d - c)  
    elif makanan <= a or makanan >= d:  
        return 0
```

3) Enak berada pada interval [6,9]

```
def enak(makanan):  
    if makanan > c and makanan < d:  
        return (makanan - c) / (d - c)  
    if makanan >= d and makanan <= e:  
        return 1  
    if makanan > e and makanan < f:  
        return -(makanan - f) / (f - e)  
    elif makanan <= c or makanan >= f:  
        return 0
```

4) Sangat Enak berada pada interval [8,10]

```
def sangat_enak(makanan):  
    if makanan >= f:  
        return 1  
    if makanan > e and makanan < f :  
        return (makanan - e) / (f - e)  
    elif makanan <= e:  
        return 0
```

b. Fungsi Keanggotaan Pelayanan

Bentuk fungsi keanggotaan untuk input pelayanan yang kami gunakan ialah fungsi trapesium sebagai berikut :



Batas-batas fungsi keanggotaan pelayanan :

- 1) Buruk berada pada interval

```
def buruk(pelayanan):
    if pelayanan <= g:
        return 1
    if pelayanan > g and pelayanan < h :
        return -(pelayanan - h) / (h - g)
    elif pelayanan >= h:
        return 0
```

- 2) Cukup berada pada interval

```
def cukup(pelayanan):
    if pelayanan > g and pelayanan < h:
        return (pelayanan - g) / (h - g)
    if pelayanan >= h and pelayanan <= i:
        return 1
    if pelayanan > i and pelayanan < j:
        return -(pelayanan - j) / (j - i)
    elif pelayanan <= g or pelayanan >= j:
        return 0
```

3) Bagus berada pada interval

```
def bagus(pelayanan):  
    if pelayanan > i and pelayanan < j:  
        return (pelayanan - i) / (j - i)  
    if pelayanan >= j and pelayanan <= k:  
        return 1  
    if pelayanan > k and pelayanan < l:  
        return -(pelayanan - l) / (l - k)  
    elif pelayanan <= i or pelayanan >= l:  
        return 0
```

4) Sangat Bagus berada pada interval

```
def sangat_bagus(pelayanan):  
    if pelayanan >= l:  
        return 1  
    if pelayanan > k and pelayanan < l:  
        return (pelayanan - k) / (l - k)  
    elif pelayanan <= k:  
        return 0
```

4. Fuzzifikasi

Menurut Wang (1997:105), fuzzifikasi didefinisikan sebagai pemetaan dari himpunan tegas ke himpunan *fuzzy*. Dengan kata lain pada fuzzifikasi akan terbentuk *fuzzy* set untuk setiap input yang ada.

Untuk input makanan proses fuzzifikasi akan merubah variabel nilai_makanan menjadi food_set yang merupakan bentuk linguistik yang sudah ditentukan dengan cara memanggil fungsi keanggotaan untuk kualitas makanan.

```
def fuzzifikasi_makanan(nilai_makanan):  
    food_set = {}  
    food_set['tidak enak'] = tidak_enak(nilai_makanan)  
    food_set['biasa'] = biasa(nilai_makanan)  
    food_set['enak'] = enak(nilai_makanan)  
    food_set['sangat enak'] = sangat_enak(nilai_makanan)  
    return food_set
```


Sedangkan untuk input pelayanan, proses fuzzifikasi akan merubah nilai_pelayanan menjadi service_set sesuai dengan bentuk linguistik yang ada dengan cara memanggil fungsi keanggotaan untuk kualitas pelayanan.

```
def fuzzifikasi_pelayanan(nilai_pelayanan):
    service_set = {}
    service_set['buruk'] = buruk(nilai_pelayanan)
    service_set['cukup'] = cukup(nilai_pelayanan)
    service_set['bagus'] = bagus(nilai_pelayanan)
    service_set['sangat bagus'] = sangat_bagus(nilai_pelayanan)
    return service_set
```

Berikut beberapa hasil dari fuzzifikasi pada input makanan dan input pelayanan(service) :

```
{'tidak enak': 0, 'biasa': 0, 'enak': 1, 'sangat enak': 0} {'buruk': 0, 'cukup': 1, 'bagus': 0, 'sangat bagus': 0}
{'tidak enak': 1, 'biasa': 0, 'enak': 0, 'sangat enak': 0} {'buruk': 0, 'cukup': 1, 'bagus': 0, 'sangat bagus': 0}
{'tidak enak': 1, 'biasa': 0, 'enak': 0, 'sangat enak': 0} {'buruk': 0, 'cukup': 0, 'bagus': 0, 'sangat bagus': 1}
{'tidak enak': 0, 'biasa': 1, 'enak': 0, 'sangat enak': 0} {'buruk': 0, 'cukup': 1, 'bagus': 0, 'sangat bagus': 0}
{'tidak enak': 0, 'biasa': 1, 'enak': 0, 'sangat enak': 0} {'buruk': 1, 'cukup': 0, 'bagus': 0, 'sangat bagus': 0}
{'tidak enak': 0, 'biasa': 0, 'enak': 0, 'sangat enak': 1} {'buruk': 0, 'cukup': 1, 'bagus': 0, 'sangat bagus': 0}
{'tidak enak': 0, 'biasa': 0, 'enak': 1, 'sangat enak': 0} {'buruk': 0, 'cukup': 0.9, 'bagus': 0.1, 'sangat bagus': 0}
{'tidak enak': 0, 'biasa': 0, 'enak': 0, 'sangat enak': 1} {'buruk': 1, 'cukup': 0, 'bagus': 0, 'sangat bagus': 0}
{'tidak enak': 1, 'biasa': 0, 'enak': 0, 'sangat enak': 0} {'buruk': 0, 'cukup': 1, 'bagus': 0, 'sangat bagus': 0}
{'tidak enak': 0, 'biasa': 0, 'enak': 0, 'sangat enak': 1} {'buruk': 0.4, 'cukup': 0.6, 'bagus': 0, 'sangat bagus': 0}
```

5. Inferensi

Inferensi akan melakukan penalaran menggunakan *fuzzy input* dan *fuzzy rules* yang telah ditentukan, sehingga dapat menghasilkan *fuzzy output*. Berikut *fuzzy rules* yang terbentuk :

IF food_set = sangat enak AND service_set = sangat bagus THEN inference_set = direkomendasikan

IF food_set = sangat enak AND service_set = bagus THEN inference_set = direkomendasikan

IF food_set = sangat enak AND service_set = cukup THEN inference_set = sedang

IF food_set = sangat enak AND service_set = buruk THEN inference_set = tidak direkomendasikan

IF food_set = enak AND service_set = sangat bagus THEN inference_set = direkomendasikan

IF food_set = enak AND service_set = bagus THEN inference_set = sedang

IF food_set = enak AND service_set = cukup THEN inference_set = sedang

IF food_set = enak AND service_set = buruk THEN inference_set = tidak direkomendasikan

IF food_set = biasa AND service_set = sangat bagus THEN inference_set = sedang

IF food_set = biasa AND service_set = bagus THEN inference_set = sedang
IF food_set = biasa AND service_set = cukup THEN inference_set = tidak direkomendasikan
IF food_set = biasa AND service_set = buruk THEN inference_set = tidak direkomendasikan
IF food_set = tidak enak AND service_set = sangat bagus THEN inference_set = tidak direkomendasikan
IF food_set = tidak enak AND service_set = bagus THEN inference_set = tidak direkomendasikan
IF food_set = tidak enak AND service_set = cukup THEN inference_set = tidak direkomendasikan
IF food_set = tidak enak AND service_set = buruk THEN inference_set = tidak direkomendasikan

Adapun tabel inferensi yang terbentuk adalah :

Makanan	Pelayanan	Nilai
sangat enak	sangat bagus	direkomendasikan
sangat enak	bagus	direkomendasikan
sangat enak	cukup	sedang
sangat enak	buruk	tidak direkomendasikan
enak	sangat bagus	direkomendasikan
enak	bagus	sedang
enak	cukup	sedang
enak	buruk	tidak direkomendasikan
biasa	sangat bagus	sedang
biasa	bagus	sedang
biasa	cukup	tidak direkomendasikan
biasa	buruk	tidak direkomendasikan
tidak enak	sangat bagus	tidak direkomendasikan
tidak enak	bagus	tidak direkomendasikan
tidak enak	cukup	tidak direkomendasikan
tidak enak	buruk	tidak direkomendasikan

Berikut beberapa hasil inferensi food_set dengan service_set :

```
{'direkomendasikan': 0, 'sedang': 1, 'tidak direkomendasikan': 0}
{'direkomendasikan': 0, 'sedang': 0, 'tidak direkomendasikan': 1}
{'direkomendasikan': 0, 'sedang': 0, 'tidak direkomendasikan': 1}
{'direkomendasikan': 0, 'sedang': 0, 'tidak direkomendasikan': 1}
{'direkomendasikan': 0, 'sedang': 0, 'tidak direkomendasikan': 1}
{'direkomendasikan': 0, 'sedang': 1, 'tidak direkomendasikan': 0}
{'direkomendasikan': 0, 'sedang': 0.9, 'tidak direkomendasikan': 0}
{'direkomendasikan': 0, 'sedang': 0, 'tidak direkomendasikan': 1}
{'direkomendasikan': 0, 'sedang': 0, 'tidak direkomendasikan': 1}
{'direkomendasikan': 0, 'sedang': 0.6, 'tidak direkomendasikan': 0.4}
```

6. Metode Defuzzifikasi

Defuzzication atau metode defuzzikasi merupakan langkah terakhir dalam suatu sistem logika fuzzy yang bertujuan mengkonversi setiap hasil dari inference engine yang di ekspresikan dalam bentuk fuzzy set ke suatu bilangan real. Hasil konversi merupakan aksi yang diambil oleh sistem kendali logika fuzzy.

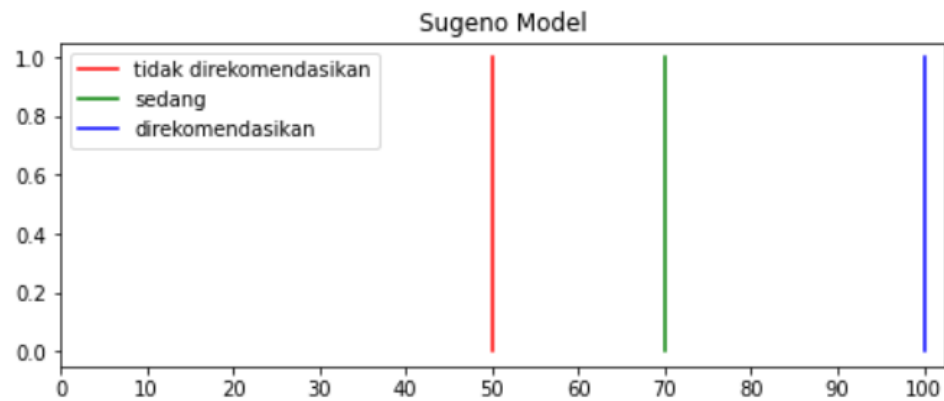
Berdasarkan latar belakang yang telah diuraikan diatas, maka permasalahan yang dibahas dalam penelitian ini adalah pembuatan sistem fuzzy logic untuk memilih 10 restoran terbaik di kota A. Metode yang digunakan dalam penelitian ini adalah metode Sugeno dalam sistem logika fuzzy. Metode Sugeno adalah metode inferensi fuzzy untuk aturan yang direpresentasikan dalam bentuk IF – THEN dimana output sistem tidak dalam bentuk himpunan fuzzy tetapi dalam bentuk konstanta atau persamaan linier. (rohimah, rukiastiandari, & siregar, 2018)

```
def defuzzifikasi(inference_set):
    numerator, denominator = 0, 0
    defuz_set = {'direkomendasikan': 100, 'sedang': 75, 'tidak direkomendasikan': 50}

    for output in defuz_set.keys():
        numerator += inference_set[output] * defuz_set[output]
        denominator += inference_set[output]
    return numerator/denominator
```

7. Bentuk dan Batas Fungsi Keanggotaan Output (sesuai metode defuzzifikasi)

- Fungsi keanggotaan sugeno model



Kami memilih batas fungsi keanggotaan dengan metode sugeno dengan parameter, tidak direkomendasikan : 50, sedang : 70, dan direkomendasikan :100. Angka-angka tersebut kami pilih berdasarkan hasil observasi kami yang dilakukan dengan membaca berbagai jurnal dan power point dari analisis fuzzy logic yang sudah lebih dahulu dilakukan.

Program Utama :

Di dalam program utama ini, kami melakukan pemanggilan dengan melibatkan metode-metode yang telah digunakan sebelumnya.

```
hasil_defuz = []

def fuzzy_system(data_resto):
    for i in range (len(data_resto)):
        food = fuzzifikasi_makanan(data_resto['makanan'][i])
        service = fuzzifikasi_pelayanan(data_resto['pelayanan'][i])
        inference_set = inferensi(food, service)
        result = defuzzifikasi(inference_set)
        hasil_defuz.append(result)

    return hasil_defuz

data = fuzzy_system(data_resto)
data_resto['hasil'] = data
```

Kami melakukan output data dengan melakukan sorting terlebih dahulu secara descending kemudian melakukan output berupa file berjenis excel yang mengeluarkan 10 id terbaiknya :

```
data_resto = data_resto.sort_values(by='hasil', ascending=False)
data_resto = data_resto.head(10)

data_resto['id'].to_excel('peringkat.xls', index=False, header=False)
data_resto
```

Sehingga menghasilkan output dari main program yaitu 10 restoran terbaik di kota A :

id		pelayanan	makanan	hasil
23	24	100	9	100.0
41	42	94	10	100.0
24	25	61	10	100.0
30	31	74	9	100.0
78	79	87	9	100.0
69	70	78	8	100.0
68	69	86	10	100.0
62	63	78	7	100.0
21	22	79	9	100.0
53	54	64	10	100.0

III. Nilai-nilai parameter Fuzzy yang paling optimum

- Fungsi keanggotaan makanan terdapat parameter:
a = 3
b = 4
c = 6
d = 7
e = 8
f = 9
- Fungsi keanggotaan pelayanan terdapat parameter:
g = 30
h = 40
i = 60
j = 70
k = 80
l = 90
- Fuzifikasi kami set dengan: `food_set['sangat enak'] = sangat_enak(nilai_makanan)`
- Inferensi dan disjunction/conjunction menggunakan:
`inference_set['direkomendasikan'].append(min(food_set['sangat enak'],service_set['sangat bagus']))`
- Defuzifikasi dengan ‘direkomendasikan: 100’ adalah nilai optimum

Dari nilai parameter optimum diatas akan di dapatkan output yang optimum yaitu dengan id = 24, pelayanan = 100, makanan = 9, dan hasil = 100,00.

IV. Kesimpulan

Berdasarkan hasil analisis dan perancangan sistem berbasis fuzzy logic untuk memilih 10 restoran terbaik di kota A, dengan menggunakan metode sugeno, maka dapat diambil kesimpulan bahwa dengan dibangunnya sistem ini dapat membantu customer untuk menentukan restoran yang sesuai dengan ekspektasi mereka masing masing, baik itu dari segi makanan maupun pelayanannya. Sistem yang kami buat menggunakan metode sugeno terbukti efektif dengan nilai-nilai parameter Fuzzy yang kelompok 6 anggap paling optimum, sehingga menghasilkan 10 restoran terbaik di kota A

V. Link Presentasi

Berikut link presentasi Tugas Pemrograman 2 Pengantar Kecerdasan Buatan Kelompok 6 : <https://youtu.be/WB0P7F1QSqg>

References

- Power point Pengantar Kecerdasan Buatan
- nisa, auliah khoirun, dkk (2020, 04 01). Journal of Mathematics, Computations, and Statistics. From <http://www.ojs.unm.ac.id/jmatchos>
- rohimah, l., rukiastiandari, s., & siregar, j. (2018). Penerapan Logika Fuzzy Metode Sugeno Untuk Optimalisasi Nilai Ekspor Ikan Tuna Hs 160414 Ke Italia . *Jurnal Teknik Komputer AMIK BSI*.
- wibowo, s. (2015). Penerapan Logika Fuzzy Dalam Penjadwalan Waktu Kuliah. *Jurnal Informatika UPGRIS*.