

Laporan Observasi
Tugas Pemrograman 01 CII-2M3 Pengantar Kecerdasan Buatan
Genap 2020/2021



Disusun Oleh :

1. Irwan Reza Firmansyah (1301194018)
2. Ramadito Ferdian Assa (1301194005)
3. Artisa Bunga Syahputri (1301194007)

FAKULTAS INFORMATIKA
PROGRAM STUDI S1 INFORMATIKA
UNIVERSITAS TELKOM
2020/2021

Pada tugas pemrograman 2 untuk mata kuliah pengantar kecerdasan buatan kali ini kami membangun sistem berbasis fuzzy untuk memilih 10 restoran terbaik di kota A berdasarkan data pada file restoran.xls yang menghimpun data 100 restoran yang ada di kota A dengan atribut Kualitas Pelayanan (bilangan real 1-100) dan kualitas makanan (bilangan real 1-10). Berikut adalah hal-hal yang kami observasi dari sistem fuzzy yang telah kami bangun

1. Jumlah dan Nama Linguistik setiap input

Linguistik atau pada sistem fuzzy disebut linguistik variabel adalah nilai variabel yang merepresentasikan sebuah nilai dengan menggunakan kata-kata dibandingkan menggunakan angka untuk setiap anggota atau membership dari setiap atributnya.

Pada tugas pemrograman kali ini, kami menggunakan 8 variabel linguistik, dengan setiap atribut memiliki 4 variabel linguistik untuk merepresentasikan nilai atributnya

Pada atribut Pelayanan kami menggunakan variabel linguistik berupa kategori 'bad', 'ordinary', 'good', dan 'very good'. Kami menggunakan variabel linguistik tersebut untuk mendeskripsikan atribut pelayanan adalah karena dengan semakin banyak linguistik maka makin detil aturan dan Semakin akurat dalam menyeleksi data terbaik pada atribut pelayanan

Sedangkan untuk atribut Makanan, kami menggunakan variabel linguistik nya berupa kategori 'not good', 'not bad', 'delicious', dan 'very delicious'. Kami menggunakan variabel linguistik tersebut untuk mendeskripsikan atribut pelayanan adalah karena dengan semakin banyak linguistik maka makin detil aturan dan Semakin akurat dalam menyeleksi data terbaik pada atribut Makanan karena dari jumlah linguistik yang kami gunakan berjumlah 8, maka jumlah rule yang terbentuk ada 16.

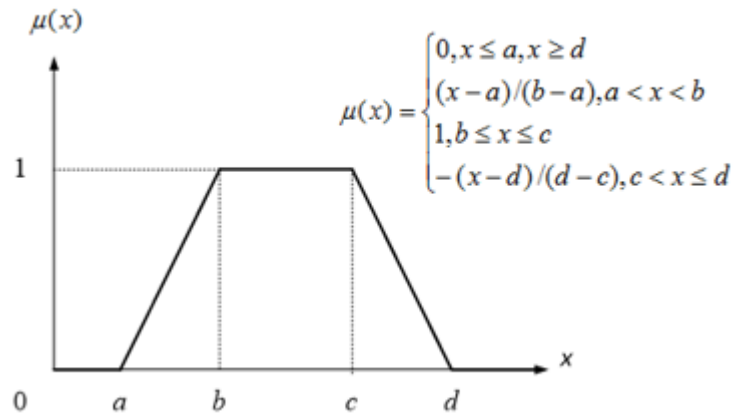
Berikut adalah code program yang dibuat untuk mendeklarasikan variabel linguistik pada sistem fuzzy yang kamu buat dengan menggunakan bahasa pemrograman python.

```
pelayanan = {
    'bad' : [ 0, 0, 10, 30],
    'ordinary' : [ 25, 30, 40, 55],
    'good' : [ 45, 50, 60, 75],
    'very good' : [70, 90, 100, 100]
}

# Membership Makanan
makanan = {
    'not good' : [0, 0, 2, 4],
    'not bad' : [3, 5, 6, 8],
    'delicious' : [6, 7, 8, 9],
    'very delicious' : [8, 9, 10, 10]
}
```

2. Bentuk dan Batas Fungsi Keanggotaan Input

Untuk fungsi keanggotaan pada pembangunan sistem fuzzy ini kami menggunakan array dengan 4 index, dimana setiap index membentuk kurva linear dimana perpaduan dari 2 atau lebih kurva tersebut akan membentuk sebuah kurva berbentuk trapesium, bentuk kurva kamu menyerupai gambar dibawah ini:

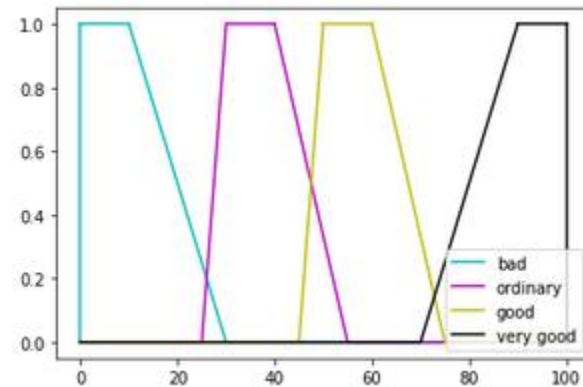


Pada pembangunan sistem fuzzy ini, batas keanggotaan yang kami gunakan pada inputan pertama yaitu pada atribut pelayanan diantaranya

‘Bad’ : [0,0,10,30]
‘Ordinary’ : [25,30,40,55]
‘Good’ : [45,50,60,75]
‘Very good’ : [70,90,100,100]

```
pelayanan = {  
    'bad' : [ 0, 0, 10, 30],  
    'ordinary' : [ 25, 30, 40, 55],  
    'good' : [ 45, 50, 60, 75],  
    'very good' : [70, 90, 100, 100]  
}
```

Dan hasil kurva yang terbentuk:

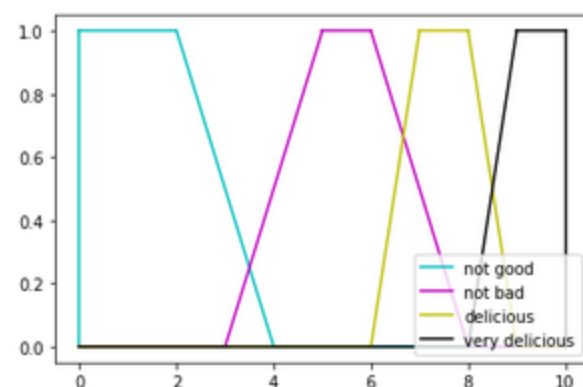


Sedangkan batas keanggotaan untuk atribut makanan diantaranya

'Not good' : [0,0,2,4]
'Not bad' : [3,5,6,8]
'Delicious' : [6,7,8,9]
'Very delicious' : [8,9,10,10]

```
makanan = {  
    'not good' : [0, 0, 2, 4],  
    'not bad' : [3, 5, 6, 8],  
    'delicious' : [6, 7, 8, 9],  
    'very delicious' : [8, 9, 10, 10]  
}
```

Dan hasil kurva yang terbentuk:



3. Aturan Inferensi

Aturan inferensi yang kami gunakan pada pembangunan sistem fuzzy ini adalah dengan menentukan hasil yang akan dikeluarkan terlebih dahulu, dimana hasil yang kami inginkan ada 3 diantaranya adalah reject, consider, dan accept. Kami menggunakan kombinasi 3 nilai yang mana tiap nilai memiliki 4 jenis yang berbeda sehingga terbentuk 16 kombinasi hasil yang menjadi aturan inferensi pada sistem fuzzy ini. Hal ini bertujuan untuk memperketat seleksi penentuan hasil terbaik pada setiap data yang diberikan.

```
# Rule Inference
rule = {
    ('bad' , 'not good')    : 'reject',
    ('bad' , 'not bad')    : 'reject',
    ('bad' , 'delicious')   : 'consider',
    ('bad' , 'very delicious') : 'consider',
    ('ordinary' , 'not good') : 'reject',
    ('ordinary' , 'not bad') : 'reject',
    ('ordinary' , 'delicious') : 'consider',
    ('ordinary' , 'very delicious') : 'consider',
    ('good' , 'not good')   : 'reject',
    ('good' , 'not bad')   : 'reject',
    ('good' , 'delicious')  : 'consider',
    ('good' , 'very delicious') : 'accept',
    ('very good' , 'not good') : 'reject',
    ('very good' , 'not bad') : 'consider',
    ('very good' , 'delicious') : 'consider',
    ('very good' , 'very delicious') : 'accept',
}
```

4. Metode Defuzzifikasi

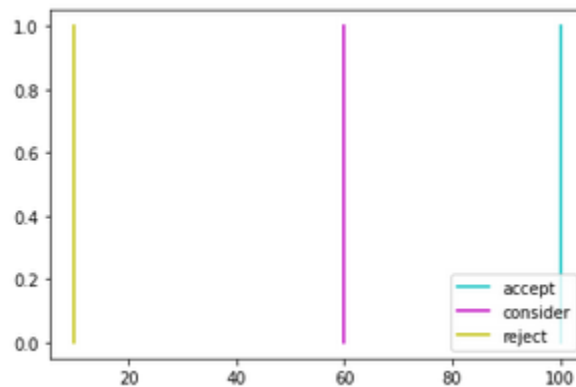
Metode defuzzifikasi yang kami gunakan pada sistem fuzzy ini adalah metode constant defuzzification (Takagi-sugeno-style), dimana metode ini memilih nilai konstan sebagai acuan untuk menentukan output setiap linguistik, lalu menghitung crips nya dengan rumus:

$$z^* = \frac{\sum_{i=1}^l \mu B_i \cdot c_i}{\sum_{i=1}^l \mu B_i}$$

$c_i = \text{constant for } i^{\text{th}} \text{ linguistic}$
 $\mu B_i = \text{membership for } i^{\text{th}} \text{ linguistic}$

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

Batas fungsi keanggotaan yang kami terapkan pada pembangunan sistem fuzzy kali ini adalah 'accept':100, 'consider':60, dan 'reject':10, sehingga didapatkan bentuk nya dalam plot adalah sebagai berikut:



Proses yang dibangun untuk membuat sistem fuzzy ini diantaranya:

1. Membaca file

Kami melakukan download file restoran.xls yang sudah kamu masukan ke dalam drive yang nantinya akan dijadikan sebagai data untuk membangun sistem fuzzy yang akan dibuat.

```
!gdown --id 1LLIqdkukJlMhIH5ezUSj3DcKHysyIT__
#melakukan download dataset pada google drive
#https://drive.google.com/file/d/1LLIqdkukJlMhIH5ezUSj3DcKHysyIT__/view?usp=sharing

Downloading...
From: https://drive.google.com/uc?id=1LLIqdkukJlMhIH5ezUSj3DcKHysyIT__
To: /content/restoran.xlsx
100% 10.5k/10.5k [00:00<00:00, 16.1MB/s]
```

```
#import library pandas
import pandas as pd

# import dataset dan lihat 10 data pertama
data_frame = pd.read_excel("restoran.xlsx")
data_frame.head(10)
```

	id	pelayanan	makanan
0	1	58	7
1	2	54	1
2	3	98	2
3	4	52	4
4	5	11	4
5	6	59	10
6	7	61	8
7	8	30	10
8	9	45	1
9	10	36	9

2. Fuzzifikasi

Fuzzifikasi adalah proses yang dilakukan untuk mengubah suatu makna dari bentuk crips menjadi variabel-variabel linguistik yang membentuk himpunan fuzzy dengan fungsi keanggotaan nya masing-masing.

Hal yang kami lakukan pertama kali dalam membangun proses fuzzifikasi ini adalah dengan membuat fungsi fuzzy linguistik yang akan menerima inputan berupa nilai input yang akan di-fuzzifikasi (x) dan juga list berisi 4 nilai dari representasi linguistik yang akan mengoutputkan nilai fuzzifikasi dalam bilangan real.

Lalu kami membuat fungsi fuzzy keanggotaan yang akan menerima inputan berupa Nilai input yang akan di-fuzzifikasi (x) dan dictionary berisi kumpulan linguistik dan nilai linguistiknya yang lalu akan mengoutputkan dictionary berupa semua linguistik anggota dan nilainya

Lalu kami membuat fungsi fuzzifikasi yang akan menghitung seluruh nilai fuzzifikasi dari x terhadap seluruh linguistik yang ada dan akan mengoutputkan dictionary dan hasil fuzzifikasi seluruh anggota

```
def fuzzy_linguistik(x, point):

    # jika point berada didepan dan dibelakang grafik
    if x < point[0] or x > point[3]:
        return 0

    # jika point berada ditengah grafik
    elif x >= point[1] and x <= point[2]:
        return 1

    # jika point berada di 1/4 grafik kiri
    elif x >= point[0] and x < point[1]:
        return (x - point[0]) / (point[1] - point[0])

    # jika point berada di 1/4 grafik kanan
    elif x >= point[2] and x < point[3]:
        return (point[3] - x) / (point[3] - point[2])

    return 0
```

```
def fuzzy_keanggotaan(x, keanggotaan):

    result = {}

    # Mencari nilai fuzzy untuk masing-masing linguistik dalam Membership
    for ling in keanggotaan:
        result[ling] = fuzzy_linguistik(x, keanggotaan[ling])

    return result

def fuzzyfikasi(x, list_keanggotaan):

    result = []

    # hitung seluruh hasil fuzzyfication dari seluruh membership
    for i in range(len(list_keanggotaan)):
        result.append(fuzzy_keanggotaan(x[i], list_keanggotaan[i]))

    return result
```


3. Inferensi

Selanjutnya kami membangun proses inferensi yang akan melakukan inferensi dari hasil fuzzifikasi sesuai dengan rule yang sudah ditetapkan sebelumnya dan akan mengeluarkan hasil berupa dictionary hasil inferensi.

```
def inferensi(nilai_fuzzy, rule):  
  
    result = {}  
  
    # loop seluruh linguistik 1 dan value v dari hasil fuzzyfication  
    for l1, v1 in nilai_fuzzy[0].items():  
        for l2, v2 in nilai_fuzzy[1].items():  
  
            # r dan min_value merupakan kombinasi dua linguistik  
            # dan value minimal dari kedua linguistik  
            r, min_value = (l1, l2), min(v1, v2)  
  
            # lihat nilai yang sudah didapat pada hasil  
            current_value = result.get(rule[r], 0)  
  
            # replace nilai hasil apabila current value lebih kecil  
            result[rule[r]] = max(min_value, current_value)  
  
    return result
```

4. Defuzzifikasi

Langkah berikutnya kami melakukan proses defuzzifikasi dengan menggunakan metode constant defuzzification (Takagi-sugeno-style), setelah melakukan proses defuzzifikasi maka dapat ditentukan 10 restoran terbaik di kota A.

```
def defuzz_sugeno(x_infer, membership):  
  
    num, den = 0, 0  
  
    # Penggunaan Rumus Defuzz Sugeno  
    for ling in membership:  
        num += x_infer[ling] * membership[ling]  
        den += x_infer[ling]  
  
    return num / den
```

Hasil dan kesimpulan

Untuk menentukan 10 restoran terbaik di kota A dari data restoran.xls kami menggunakan fuzzy logic dengan membangun proses-proses logika nya, dari proses-proses yang dibangun itu akan menghasilkan keluaran seperti dibawah ini:

- Data yang kami miliki sebelum melakukan proses fuzzy adalah

	id	pelayanan	makanan
0	1	58	7
1	2	54	1
2	3	98	2
3	4	52	4
4	5	11	4
5	6	59	10
6	7	61	8
7	8	30	10
8	9	45	1
9	10	36	9

- Dan data yang kami dapatkan setelah membangun proses fuzzy adalah sebagai berikut

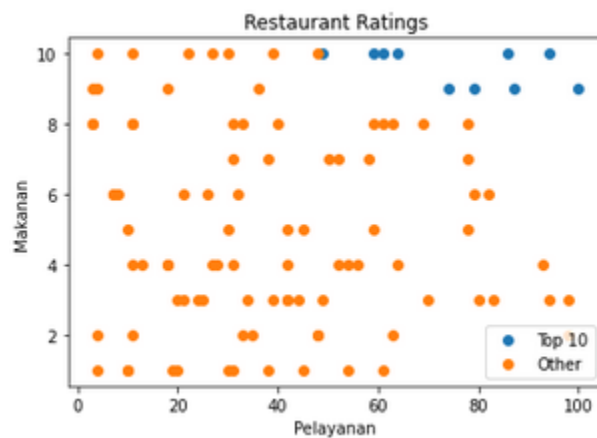
	id	pelayanan	makanan
68	69	86	10
78	79	87	9
41	42	94	10
24	25	61	10
23	24	100	9
5	6	59	10
21	22	79	9
53	54	64	10
30	31	74	9
19	20	49	10

Hasil keluaran setelah melakukan proses fuzzy untuk menentukan peringkat 10 terbaik dari restoran di kota A adalah :

Hasil akhir untuk 10 restoran terbaik:

	id	pelayanan	makanan	Skor	Infer
68	69	86	10	100.000000	{'reject': 0, 'consider': 0, 'accept': 0.8}
78	79	87	9	100.000000	{'reject': 0, 'consider': 0, 'accept': 0.85}
41	42	94	10	100.000000	{'reject': 0, 'consider': 0, 'accept': 1}
24	25	61	10	100.000000	{'reject': 0, 'consider': 0, 'accept': 0.93333...}
23	24	100	9	100.000000	{'reject': 0, 'consider': 0, 'accept': 1}
5	6	59	10	100.000000	{'reject': 0, 'consider': 0, 'accept': 1}
21	22	79	9	100.000000	{'reject': 0, 'consider': 0, 'accept': 0.45}
53	54	64	10	100.000000	{'reject': 0, 'consider': 0, 'accept': 0.73333...}
30	31	74	9	100.000000	{'reject': 0, 'consider': 0, 'accept': 0.2}
19	20	49	10	86.666667	{'reject': 0, 'consider': 0.4, 'accept': 0.8}

Ketika di plot dapat dilihat seperti gambar dibawah ini



10 peringkat terbaik dari restoran diambil berdasarkan kombinasi dari skor peringkat makanan dan peringkat pelayanan dari restoran tersebut.

Dan hasil restoran yang belum masuk kedalam 10 restoran terbaik adalah:

id pelayanan makanan Skor					Infer
50	51	48	10	82.5	{'reject': 0, 'consider': 0.4666666666666667, ...}
76	77	11	10	60.0	{'reject': 0, 'consider': 0.95, 'accept': 0}
82	83	69	8	60.0	{'reject': 0, 'consider': 0.4, 'accept': 0}
33	34	93	4	60.0	{'reject': 0, 'consider': 0.5, 'accept': 0}
97	98	27	10	60.0	{'reject': 0, 'consider': 0.4, 'accept': 0}
...
58	59	28	4	10.0	{'reject': 0.5, 'consider': 0, 'accept': 0}
18	19	42	3	10.0	{'reject': 0.5, 'consider': 0, 'accept': 0}
60	61	42	4	10.0	{'reject': 0.5, 'consider': 0, 'accept': 0}
63	64	35	2	10.0	{'reject': 1, 'consider': 0, 'accept': 0}
16	17	70	3	10.0	{'reject': 0.3333333333333333, 'consider': 0.0...

90 rows x 5 columns