

**LAPORAN TUGAS ARTIFICIAL INTELLIGENCE**  
**Reasoning CSI-2E3 Kecerdasan Buatan Semester Ganjil 2023/2024**

**Oleh Kelompok 9**



**Devin Mairhan ( 1305220073 )**

**Valentino Fredrick Albert Mamesah (1305223104)**

**Alfathur Rizki Hermawan (1305223113)**

**PROGRAM STUDI STRATA SAINS DATA**  
**FAKULTAS**  
**UNIVERSITAS TELKOM**  
**BANDUNG**

# PENDAHULUAN

## Latar Belakang

Logika kabur atau logika fuzzy merupakan cabang ilmu matematika yang mempunyai fungsi untuk memberikan pemodelan pemecahan masalah seperti yang dilakukan manusia dengan bantuan teknologi komputer. Fuzzy sendiri memiliki arti samar. Oleh karena itu, maksud dari logika kabur sendiri berarti nilai yang bisa benar atau bisa memiliki salah secara bersamaan.

## Rumusan Masalah

Dalam masalah ini kita disuruh membuat sebuah sistem berbasis Fuzzy Logic Inference untuk memilih 5 bengkel terbaik dan menampilkan output berupa 5 supplier terbaik yang dilengkapi informasi berupa: Id bengkel, info kualitas servis dan harganya, serta skor kelayakan supplier untuk dipilih (hasil dari proses defuzzification).

Poin-poin yang harus Anda desain dan analisis:

- Jumlah dan Nama Linguistik setiap atribut input
- Bentuk dan Batas Fungsi Keanggotaan Input
- Aturan Inferensi
- Metode Defuzzification
- Bentuk dan Batas Fungsi Keanggotaan Output (sesuai metode Defuzzification)

## Tujuan

Tujuannya adalah untuk mengidentifikasi lima bengkel terbaik dengan menerapkan fuzzy logic.

## PEMBAHASAN

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

# inisiasi dataset bengkel
df = pd.read_excel('/content/bengkel.xlsx')

id = df['id']
servis = df['servis']
harga = df['harga']

df
```

	id	servis	harga
0	1	58	7
1	2	54	1
2	3	98	2
3	4	52	4
4	5	11	4
...	...	...	...
95	96	30	1
96	97	25	3
97	98	27	10
98	99	8	6
99	100	11	8

100 rows x 3 columns

Import pandas, numpy, dan matplotlib.pyplot digunakan untuk membantu mengerjakan tugas yang diberikan. Lalu buat variable id untuk data id nya, servis untuk data servisnya, dan harga untuk data harganya.

### Linguistik Jumlah Nama Atribut serta Bentuk dan Batas Fungsi Keanggotaan

Terdapat 2 variabel yaitu servis dan harga dan untuk setiap variabel memiliki 3 nama linguistik,

variabel servis : good, average, poor

variabel harga : pricey, mediocre, affordable

dan atribut outputnya juga terdapat 3 nama linguistik yaitu

- loved, artinya bengkel ini memiliki nilai servis sera harga yang baik/istimewa
- b\_aja, artinya bengkel ini memiliki nilai servis servis harga yang biasa saja
- tutup, artinya bengkel ini memiliki nilai servis servis harga yang buruk

## Bentuk dan fungsi keanggotaan variabel

Untuk bentuk serta fungsi keanggotaan yang kami pakai terdapat 3 bentuk yaitu fungsi linier naik, linier turun, dan trapesium. Berikut rumus fungsi keanggotaannya.

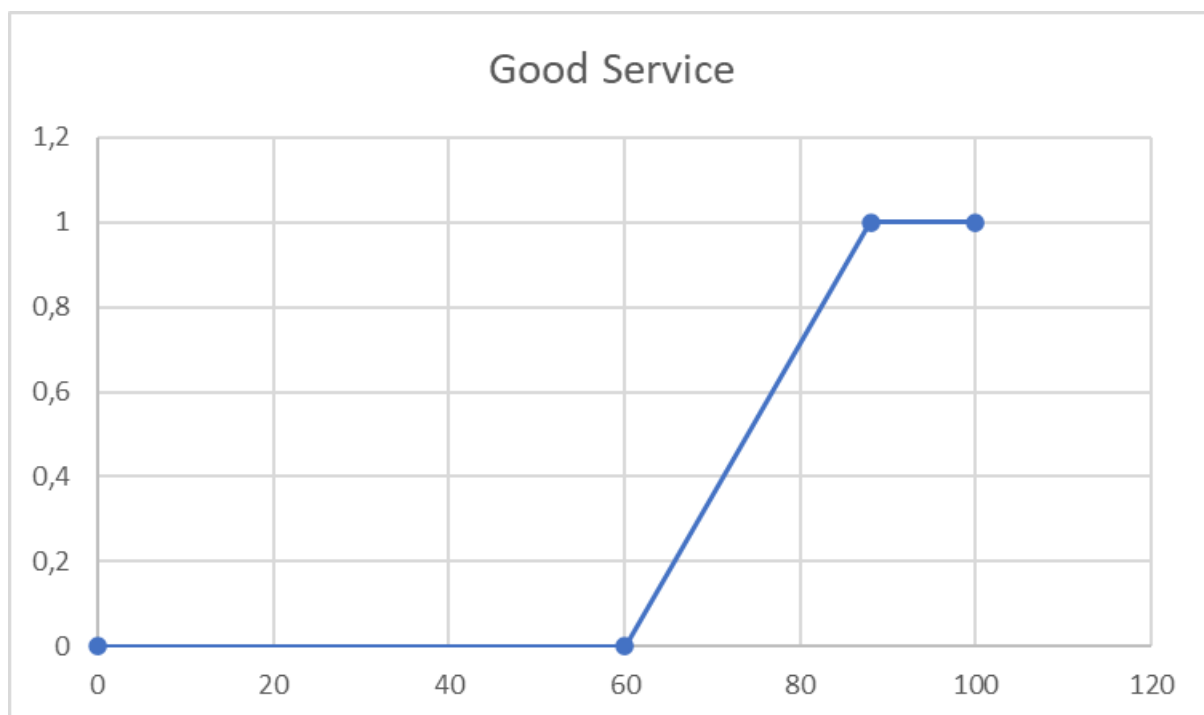
$$\text{LinierNaik}(x, a, b) = \begin{cases} 0, & x \leq a \\ (x - a) / (b - a), & a < x \leq b \end{cases}$$

$$\text{LinierTurun}(x, a, b) = \begin{cases} (b - x) / (b - a), & a \leq x < b \\ 0, & x \geq b \end{cases}$$

$$\text{Trapesium}(x, a, b, c, d) = \begin{cases} 0, & x \leq a, x \geq d \\ (x - a) / (b - a), & a < x < b \\ 1, & b \leq x \leq c \\ -(x - d) / (d - c), & c < x \leq d \end{cases}$$

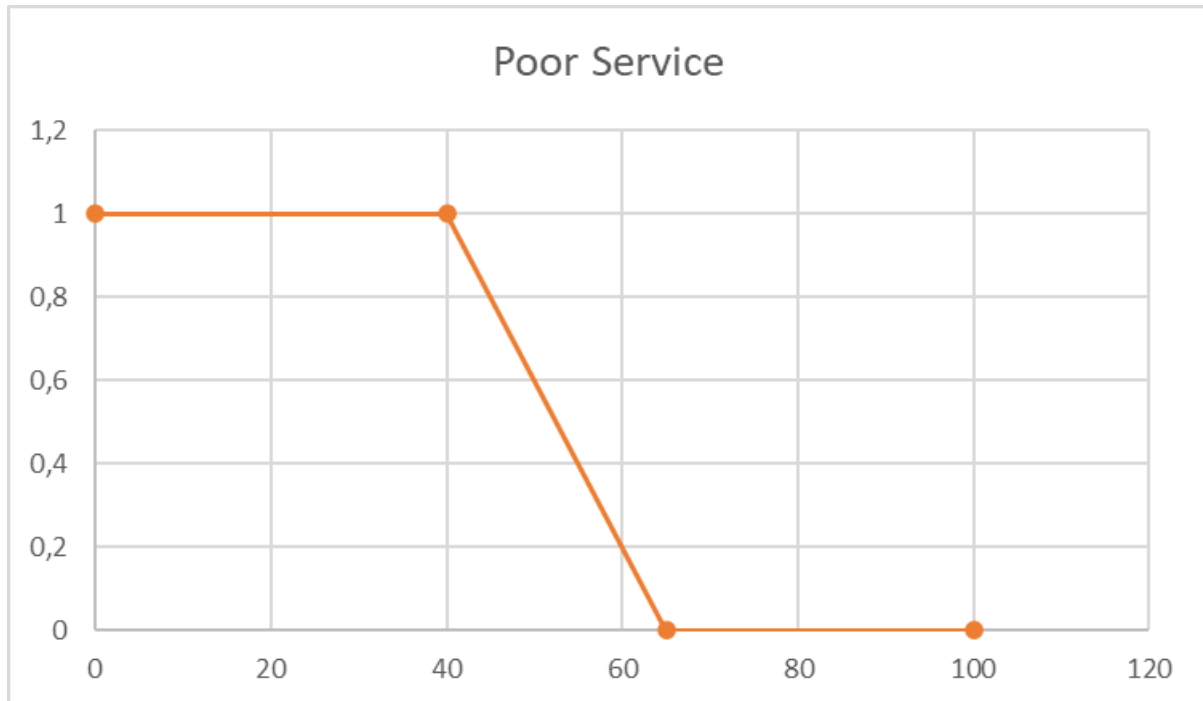
## Grafik Fungsi Variabel Servis

Untuk sumbu y merupakan derajat keanggotaan sedangkan x merupakan nilai linguistiknya.



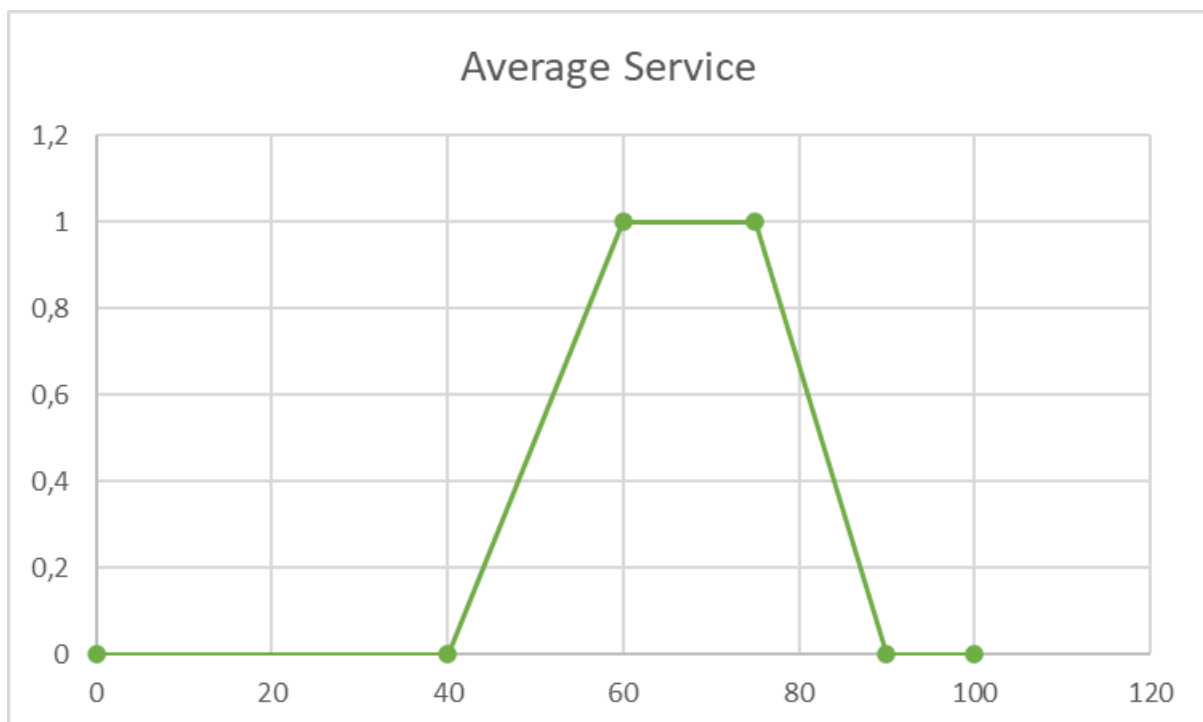
Penjelasan:

Untuk nama lingusitik good akan bernilai 1/pasti terjadi setelah batas keanggotaan melewati angka 88, 0 di saat kurang dari 60, dan terdapat grey area/ grafik naik pada saat nilai x 60 hingga 88.



Penjelasan:

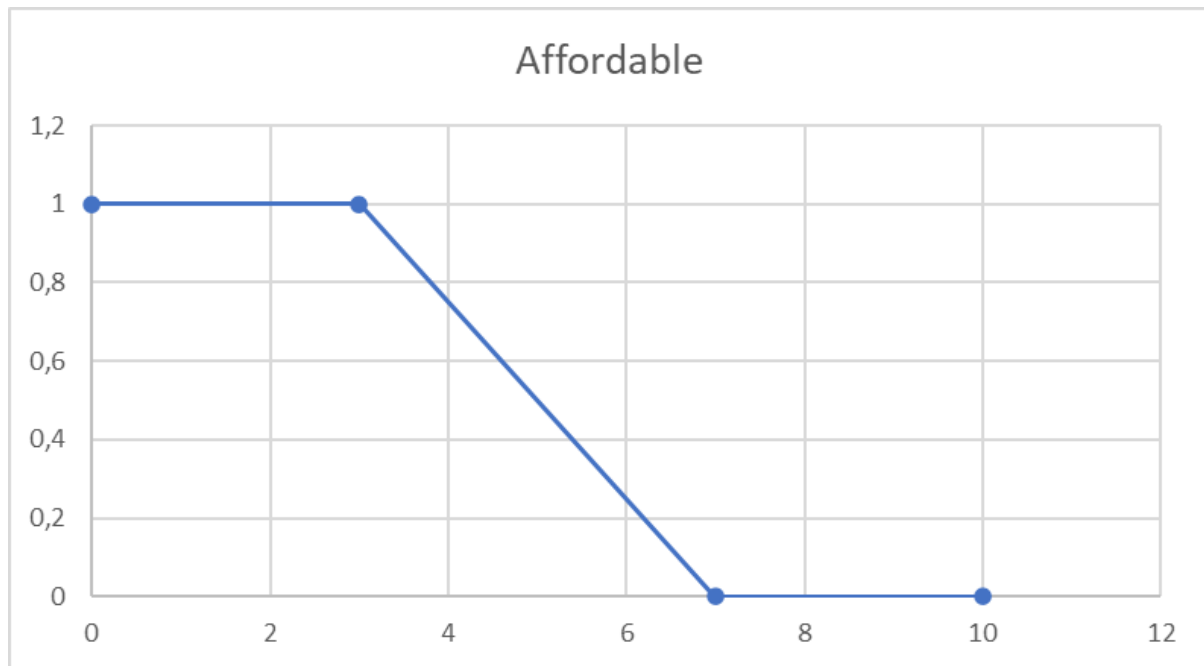
Untuk nama lingusitik poor akan bernilai 1/pasti terjadi saat batas keanggotaan bernilai kurang dari angka 40, 0 di saat lebih dari 60, dan terdapat grey area/ grafik turun pada saat nilai xnya 40 hingga 65.



### Penjelasan

Untuk nama lingusitik poor akan bernilai 1/pasti terjadi saat batas keanggotaan bernilai dari 60 sampai 75, 0 di saat kurang dari 40 dan lebih dari 90, dan terdapat dua grey area/ grafik naik pada saat nilai xnya 40 hingga 60 dan turun pada saat nilai 75 hingga 90 dimana grafik diatas berbentuk bangun datar trapesium.

### Grafik Fungsi Variabel Harga



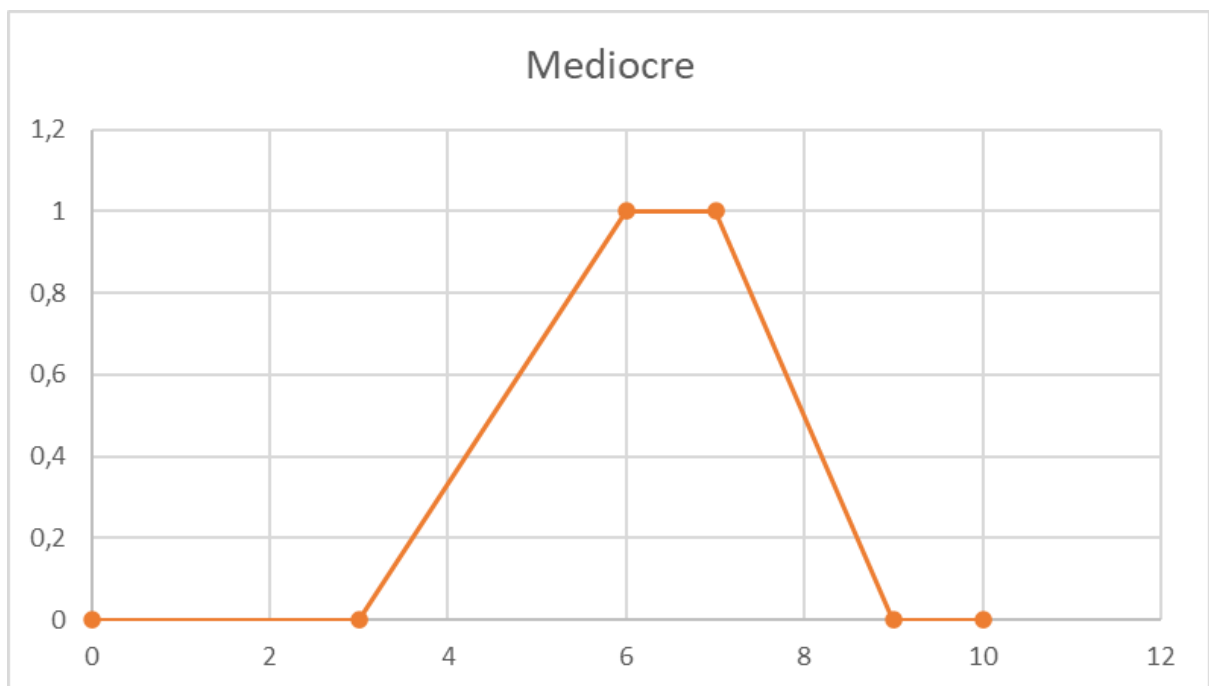
### Penjelasan:

Untuk nama lingusitik poor akan bernilai 1/pasti terjadi saat batas keanggotaan bernilai kurang dari angka 3, 0 di saat lebih dari 7, dan terdapat grey area/ grafik turun pada saat nilai xnya 3 hingga 7.



Penjelasan:

Untuk nama lingusitik good akan bernilai 1/pasti terjadi setelah batas keanggotaan melewati angka 8, 0 di saat kurang dari 6, dan terdapat grey area/ grafik naik pada saat nilai x 6 hingga 8.



Penjelasan:

Untuk nama lingusitik poor akan bernilai 1/pasti terjadi saat batas keanggotaan bernilai dari 6 sampai 7,5, 0 di saat kurang dari 3 dan lebih dari 9, dan terdapat dua grey area/ grafik naik pada

saat nilai xnya 3 hingga 6 dan turun pada saat nilai 75 hingga 9 dimana grafik diatas berbentuk bangun datar trapesium.

Berikut implementasi kodenya,

```
# member function untuk servis dan harga

def rulesService(x):
    # linier naik
    if x > 88:
        good = 1
    elif x <= 60:
        good = 0
    else:
        good = (x-60)/(88-60)
    # linier turun
    if x < 40:
        poor = 1
    elif x >= 65:
        poor = 0
    else:
        poor = (65-x)/(65-40)
    # trapesium
    if x > 60 and x <= 75:
        average = 1
    elif x <= 40 and x > 90:
        average = 0
    elif 40 < x <= 60:
        average = (x-40)/(60-40)
    elif 75 < x <= 90:
        average = -(x-90)/(90-75)
    else:
        average = 0

    return [good, average, poor] #print(f"good = {good:.2f}\naverage = {average:.2f}\npoor = {poor:.2f}")
```

Fungsi diatas digunakan untuk mengetahui jenis service nya. Ada *poor*, *good*, dan *average*.

```
def rulesPrice(x):
    # linier naik
    if x >= 8:
        pricey = 1
    elif x < 6:
        pricey = 0
    elif 6 < x <= 8.5:
        pricey = (x-6)/(8.5-6)
    else:
        pricey = 0
    # linier turun
    if x >= 7:
        affordable = 0
    elif x < 3:
        affordable = 1
    else:
        affordable = (7-x)/(7-3)
    # trapesium
    if 6 < x <= 7:
        mediocre = 1
    elif x <= 3 and x > 9:
        mediocre = 0
    elif 3 < x <= 6:
        mediocre = (x-3)/(6-3)
    elif 7.1 < x <= 9:
        mediocre = -(x-9)/(9-7)
    else:
        mediocre = 0

    return [pricey, mediocre, affordable] #print(f"affordable = {affordable:.2f}\nmediocre = {mediocre:.2f}\npricey = {pricey:.2f}")
```

Fungsi diatas digunakan untuk mengetahui jenis harganya. Ada *pricey*, *affordable*, dan *mediocre*.



```
# contoh inputan dan output untuk servis dan price
print("Rules Service")
print(rulesService(45))

print("\n")
print("Rules Price")
rulesPrice(5)
```

Rules Service  
[0, 0.25, 0.8]

Rules Price  
[0, 0.6666666666666666, 0.5]

ini adalah contoh input dan output untuk service dan price

### Aturan Inferensi

Servis	Harga	Nilai Inferensi
Good	Pricey	b_aja
Good	Mediocre	loved
Good	Affordable	loved
Average	Pricey	tutup!
Average	Mediocre	b_aja
Average	Affordable	loved
Poor	Pricey	tutup!
Poor	Mediocre	tutup!
Poor	Affordable	b_aja

Berikut kode implementasi aturan inferensi yang telah dibuat

```

def rulesInference(a,b):
    # [loved, b aja, tutup]
    # tambahkan aturan inferensi berdasarkan tabel inferensi yang dibuat
    loved = [min(a[0],b[2]),min(a[0],b[1]),min(a[1],b[2])]
    b_aja = [min(a[1],b[0]),min(a[1],b[1]),min(a[2],a[2])]
    tutup = [min(a[1],b[0]),min(a[2],b[1]),min(a[2],a[0])]

    # cari nilai paling besar diantara score yang dihasilkan aturan inference
    SHloved = max(loved)
    SHb_aja = max(b_aja)
    SHtutup = max(tutup)

    return [SHloved, SHb_aja, SHtutup] #print(f"loved = {SHloved:.2f}\nb_aja = {SHb_aja:.2f}\ntutup! = {SHtutup:.2f}")

```

Fungsi diatas digunakan untuk menghitung *Rules Inference*. Pertama ditambahkan dahulu aturan table inferensinya lalu cari nilai paling besar diantara score yang dihasilkan aturan yang dibuat.

```

[6] # contoh inputan dan output untuk inference
print("Inference Rules")
rulesInference(rulesService(45),rulesPrice(5))

Inference Rules
[0, 0.8, 0.6666666666666666]

```

Contoh input dan output inference

## Defuzzification

Untuk mendapatkan crisp outputnya kami menggunakan Takagi-Sugeno-Style dengan rumus,

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

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

Berikut kodenya,

```

[7] # rumus defuzzy
def defuzzy(i):
    if (i[0]+i[1]+i[2]) != 0:
        return ((i[0]*100)+(i[1]*65)+(i[2]*30))/ (i[0]+i[1]+i[2]))
    else:
        return print("")

```

kode diatas merupakan implementasi rumus sugeno style serta terdapat aturan dimana jika bilangan penyebutnya bernilai tidak sama dengan 0 maka akan dimasukkan kedalam rumus jika penyebut bernilai 0, dimana berdasarkan aturan matematika nilai tersebut akan bernilai *undefined* .

```

#print seluruh data beserta nilai fuzzynya
for i in range(100):
    iR = rulesInference(rulesService(servis[i]),rulesPrice(harga[i]))
    print (
        "Data Ke      :",id[i],
        "\nService Quality:", rulesService(servis[i]),
        "\nPrice Quality   :", rulesPrice(harga[i]),
        "\nHasil Inference:", iR,
        "\nNilai Kelayakan:", defuzzy(iR),
        "\n"
    )

```

kode ini digunakan untuk meng-outputkan data dan nilai hasil *defuzzification* sebanyak 100x. yang ditampilkan berupa data ke berapa, service quality, price quality, hasil inference, dan nilai kelayakannya/hasil *defuzzification* .

```

# Tambah kolom baru untuk nilai defuzzy
for i in range(len(df)):
    iR = rulesInference(rulesService(servis[i]),rulesPrice(harga[i]))
    df.loc[i, "Nilai Kelayakan"] = defuzzy(iR)
# buat sorting mencari nilai kelayakan paling bagus(top 10)
df = df.sort_values("Nilai Kelayakan")
df

```



	id	servis	harga	Nilai Kelayakan
53	54	64	10	47.5
26	27	59	8	47.5
35	36	32	6	47.5
24	25	61	10	47.5
21	22	79	9	47.5
...	...	...	...	...
51	52	94	3	100.0
2	3	98	2	100.0
12	13	80	3	100.0
23	24	100	9	NaN
41	42	94	10	NaN

100 rows × 4 columns

Fungsi ini digunakan untuk menambahkan kolom baru yaitu kolom “Nilai Kelayakan”.

```

[38] # ganti nilai kelayakan yang hasilnya anomaly/NaN menjadi rata rata data
for col in df.columns:
    if df[col].dtype == 'float64' or df[col].dtype == 'int64':
        df[col] = df[col].replace(np.nan, df[col].mean())
# lakukan sorting secara descending
sortingData = df.sort_values(by = "Nilai Kelayakan", ascending=False)
sortingData

```

	id	servis	harga	Nilai Kelayakan
90	91	98	3	100.0
91	92	83	3	100.0
12	13	80	3	100.0
2	3	98	2	100.0
51	52	94	3	100.0
...	...	...	...	...
68	69	86	10	47.5
71	72	26	6	47.5
11	12	38	7	47.5
78	79	87	9	47.5
53	54	64	10	47.5

100 rows × 4 columns

Fungsi ini digunakan untuk mengganti Nilai Kelayakan yang datanya anomaly atau NaN menjadidi rata-rata data agar tidak mempengaruhi kualitas datanya dan dilakukan sorting berdasarkan Nilai Kelayakannya secara descending atau dari Nilai Kelayakan yang terbesar ke Nilai Kelayakan terkecil.

```

[39] #print 10 bengkel terbaik berdasarkan nilai kelayakannya
sortingData.head(10)

```

	id	servis	harga	Nilai Kelayakan
90	91	98	3	100.000000
91	92	83	3	100.000000
12	13	80	3	100.000000
2	3	98	2	100.000000
51	52	94	3	100.000000
33	34	93	4	100.000000
16	17	70	3	100.000000
74	75	61	1	93.225806
43	44	63	2	92.758621
47	48	64	4	87.121662

ini adalah data untuk 10 bengkel terbaik berdasarkan Nilai Kelayakannya

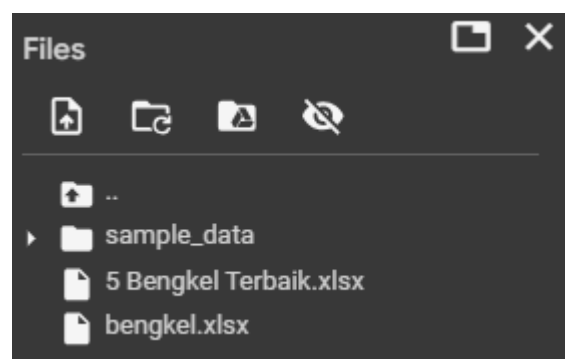
## Output

```
[40] #print 5 bengkel terbaik berdasarkan nilai kelayakannya
      sortingData.head(5)
```

	id	servis	harga	Nilai	Kelayakan
90	91	98	3		100.0
91	92	83	3		100.0
12	13	80	3		100.0
2	3	98	2		100.0
51	52	94	3		100.0

```
[41] BengkelTerbaik = sortingData.head(5).reset_index(drop=True)
      #output bengkel terbaik menjadi sebuah file excel
      BengkelTerbaik.to_excel('5 Bengkel Terbaik.xlsx')
      BengkelTerbaik
```

	id	servis	harga	Nilai	Kelayakan
0	91	98	3		100.0
1	92	83	3		100.0
2	13	80	3		100.0
3	3	98	2		100.0
4	52	94	3		100.0



Fungsi yang atas berguna untuk menampilkan 5 bengkel terbaik dan Fungsi yang dibawahnya berguna untuk meng convert data 5 bengkel terbaik ke dalam excel denga nama dan format file “5 bengkel terbaik.xlsx”.

## Keimpulan

Setelah menerapkan logika fuzzy pada data set bengkel.xlsx, kami dapat menyimpulkan bahwa lima bengkel terbaik adalah sebagai berikut: bengkel dengan ID 91 memiliki skor servis sebesar 98 dan skor harga 3, dengan nilai kelayakan mencapai 100. Selanjutnya, bengkel dengan ID 92 memiliki skor servis 83 dan skor harga 3, juga dengan nilai kelayakan 100. Bengkel dengan ID 13 memiliki skor servis 80 dan skor harga 3, serta nilai kelayakan 100. Posisi berikutnya ditempati oleh bengkel dengan ID 3, memiliki skor servis 98 dan skor harga 2, dengan nilai kelayakan 100. Terakhir, bengkel dengan ID 52 memiliki skor servis 94 dan skor harga 3, juga dengan nilai kelayakan 100.

	id	servis	harga	Nilai Kelayakan
0	91	98	3	100.0
1	92	83	3	100.0
2	13	80	3	100.0
3	3	98	2	100.0
4	52	94	3	100.0