

Tugas 2 Kecerdasan Buatan

**Analisis, Desain, dan Implementasi Fuzzy Logic dalam Suatu Program Komputer
untuk Memilih 20 Kepala Keluarga yang Layak Menerima Bantuan Langsung
Tunai (BLT)**



Disusun Oleh

Riandi Kartiko

IF-40-02

1301164300

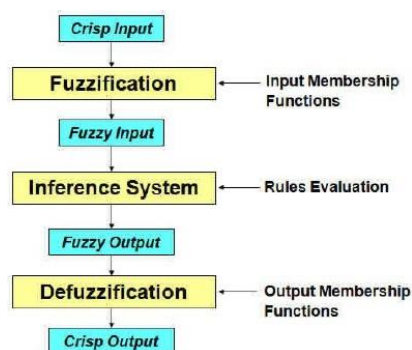
Deskripsi Masalah

Algoritma fuzzy logic merupakan metodologi pemecahan masalah dengan aplikasi / program dalam pengendali yang tersimpan dan pemrosesan informasi. Fuzzy logic memiliki karakteristik:

- Konsep logika fuzzy mudah dimengerti. Konsep matematis yang mendasari penalaran fuzzy sangat sederhana dan mudah dimengerti.
- Logika fuzzy sangat fleksibel.
- Logika fuzzy memiliki toleransi terhadap data-data yang tidak tepat.
- Logika fuzzy mampu memodelkan fungsi-fungsi non linier yang sangat kompleks.
- Pemengaplikasikan pengalaman-pengalaman para pakar secara langsung tanpa harus melalui proses pelatihan.
- Logika fuzzy dapat bekerjasama dengan teknik-teknik kendali secara konvensional.
- Logika fuzzy didasarkan pada bahasa alami.

Diberikan file DataTugas2.csv berupa himpunan data berisi 100 kepala keluarga dengan dua atribut, Penghasilan (Per Bulan) dan Hutang, yang bernilai real (dalam satuan juta rupiah). Bangunlah sebuah sistem berbasis *fuzzy logic* untuk memilih 20 kepala keluarga yang layak menerima Bantuan Langsung Tunai (BLT). Sistem membaca masukan file DataTugas2.csv dan mengeluarkan *output* berupa file TebakanTugas2.csv berupa satu vektor kolom berisi 20 baris angka bernilai *integer*/bulat yang menyatakan nomer baris atau *record* pada file DataTugas2.csv.

Strategi dan Analisis

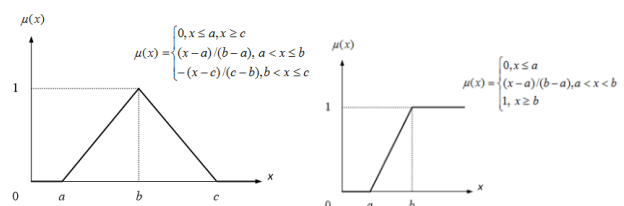


Untuk memecahkan masalah diatas, strategi yang digunakan yaitu dengan menggunakan Bahasa Python 3. Keputusan penerimaan BLT akan ditentukan dengan algoritma fuzzy logic, keputusan mengacu pada peraturan penerima BLT. Parameter input yaitu file DataTugas2.csv yang berupa baris nomor, pendapatan, dan hutang dari 100 kepala keluarga.

Saat program dijalankan, program membaca file dan mengkonversi kedalam record array dan ke bentuk crisp input yang berupa nilai linguistic yang semantiknya ditentukan berdasarkan fungsi keanggotaan. Input berupa 2 yaitu pendapatan dan hutang, berdasarkan input tersebut fungsi keanggotaan yang dapat dibuat yaitu

Fuzzification

1. Untuk pendapatan rendah, 0, pendapatan ≤ 0 dan 1, pendapatan ≥ 0.7
2. Untuk pendapatan menengah 0, pendapatan ≤ 0.7 dan 1, pendapatan ≥ 1.4
3. Untuk pendapatan tinggi 0, pendapatan ≤ 1.4 dan 1, pendapatan ≥ 2
4. Untuk hutang tinggi 0, hutang ≤ 0 dan 1, hutang ≥ 25.0
5. Untuk hutang menengah 0, hutang ≤ 25.0 dan 1, hutang ≥ 75.0
6. Untuk hutang tinggi, 0, hutang ≤ 50 dan 1, hutang > 75.0



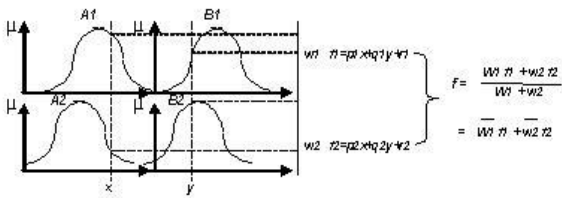
Fungsi tersebut dicari pada proses fuzzification sehingga menghasilkan fuzzy input yang akan diproses pada inference system.

Peraturan:

Pendapatan	Hutang	Score
Tinggi	Rendah	Ditolak
Tinggi	Menengah	Ditolak
Tinggi	Tinggi	Dipertimbangkan
Menengah	Rendah	Dipertimbangkan

Menengah	Menengah	Dipertimbangkan
Menengah	Tinggi	Diterima
Rendah	Rendah	Dipertimbangkan
Rendah	Menengah	Diterima
Rendah	Tinggi	Diterima

Setelah proses inference dilakukan proses defuzzification, proses ini memetakan kembali fungsi fuzzy tunggal kembali ke fungsi crisp. Pada proses defuzzification ini digunakan metode defuzzification constant / Takagi-Sugeno style. Metode sugeno mengeluarkan output berupa konstanta / persamaan linier. Defuzzifikasi pada metode sugeno dilakukan dengan mencari nilai rata-ratanya.



Setelah proses defuzzification dilakukan, crisp output berupa record data yang layak untuk menerima BLT tersebut. Kelayakan tersebut berdasarkan score tiap baris input pada proses defuzzification.

Setelah data berhasil dibuat, program mengeluarkan file .csv yang bersisi 20 kepala keluarga yang layak untuk menerima BLT.

Screenshoot

Mencari fungsi keanggotaan rendah

```
def calcLowIncome(x): #Linear membership function for Low income
    if( x >= (lowIncome.max)):
        return float(0)
    elif(x <= (lowIncome.max + lowIncome.min)/2):
        hasil = 1 - 2*((x-lowIncome.min) / (lowIncome.max-lowIncome.min))**2
        return hasil
    elif(x <= (lowIncome.max)):
        hasil = 2 * ((x-lowIncome.max) / (lowIncome.max-lowIncome.min))**2
        return hasil
    return float(0)
```

Inferensi dan penerapan peraturan

```
inference_data = []
for data in input_crisp:
    membership_result = {
        'lIncome' : calcLowIncome(data['income']),
        'aIncome' : calcAvgIncome(data['income']),
        'hIncome' : calcHghIncome(data['income']),
        'lDebt' : calcLowDebt(data['debt']),
        'aDebt' : calcAvgDebt(data['debt']),
        'hDebt' : calcHghDebt(data['debt'])
    }
    inference = { #apply to rules
        'accepted' : max(
            min(membership_result['lIncome'], membership_result['hDebt']),
            min(membership_result['lIncome'], membership_result['aDebt']),
            min(membership_result['aIncome'], membership_result['hDebt'])
        ),
        'considered' : max(
            min(membership_result['lIncome'], membership_result['lDebt']),
            min(membership_result['aIncome'], membership_result['lDebt']),
            min(membership_result['aIncome'], membership_result['aDebt']),
            min(membership_result['hIncome'], membership_result['hDebt'])
        ),
        'rejected' : max(
            min(membership_result['hIncome'], membership_result['lDebt']),
            min(membership_result['hIncome'], membership_result['aDebt'])
        )
    }
```

Output program

No	Pendapatan	Hutang	Inference
36	0.946	98.0	-9705.4
62	1.159	87.871	-8671.199999999999
77	1.217	82.991	-8177.4
74	0.6	80.864	-8026.400000000001
3	1.273	80.701	-7942.8
61	0.88	79.878	-7899.8
47	0.78	78.054	-7727.4
18	1.345	76.343	-7499.800000000001
6	0.664	63.226	-6256.2
44	0.902	68.021	-6121.756025728963
41	1.159	68.12	-6066.237842087273
95	1.329	71.308	-6044.474323144107
9	0.806	69.192	-6024.768611237591
66	0.666	60.245	-5957.9
88	1.209	67.625	-5880.717669902911
35	1.325	70.009	-5838.139999999999
39	1.332	68.795	-5734.3550000000005
48	1.433	78.028	-5361.65
92	0.817	62.427	-5236.85
57	1.336	62.094	-5164.429999999999

On Github:

<https://github.com/dikoodik/AI-FuzzyLogic>

Referensi:

http://www.academia.edu/22476741/Laporan_Implementasi_Logika_Fuzzy_Pengambilan_Keputusan_Beasi_swa

<https://anditya.staff.telkomuniversity.ac.id/academic/ai/>

