**LAPORAN PRAKTIKUM**

**KECERDASAN BUATAN**

**“STUDI KASUS”**

|  |
| --- |
|  |

|  |  |  |
| --- | --- | --- |
| NAMA | : | **AFRIDHO IKHSAN** |
| NPM | : | **2210631170002** |
| KELAS | : | **4A** |

**PROGRAM STUDI INFORMATIKA**

**FAKULTAS ILMU KOMPUTER**

**UNIVERSITAS SINGAPERBANGSA KARAWANG**

**2023**

# DAFTAR ISI

[DAFTAR ISI i](#_Toc162463204)

[SOAL 2](#_Toc162463205)

[JAWABAN 6](#_Toc162463206)

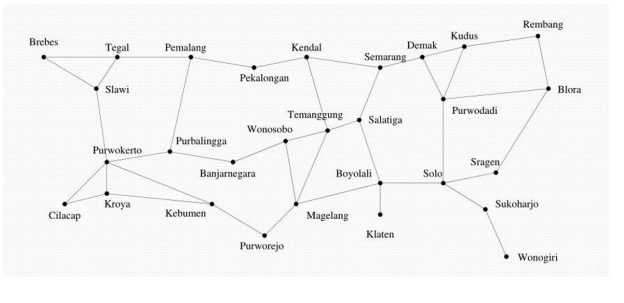
# SOAL

**Latihan: Uji Coba Menggunakan Berbagai Jenis Fungsi Keanggotaan (Membership Function)**

1. **Kasus 1 – Metode Pencarian**

Andri adalah seorang karyawan yang bekerja di salah satu perusahaan di Kawasan Industri Wijayakusuma, kota Semarang. Pada saat libur hari raya idul fitri nanti, Andri berencana untuk mengunjungi keluarganya di Cilacap. Namun, ia tidak tahu rute perjalanan yang harus ia lalui untuk sampai tujuan. Andri hanya memiliki peta jawa tengah yang berbentuk graph berisi nama nama kota di jawa tengah. Namun, dia belum memiliki jarak masing masing kota tersebut. Bantulah andri untuk menentukan rute perjalanan terbaiknya!

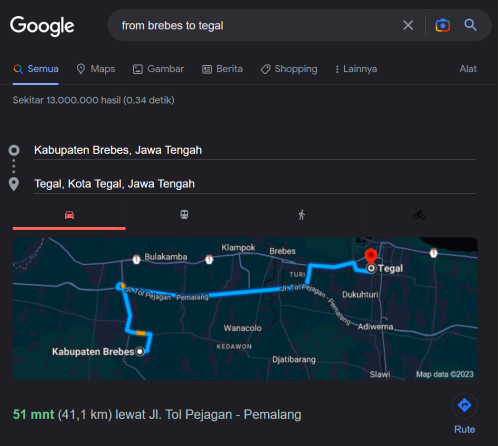
**PETA JAWA TENGAH**



1. Carilah jarak masing masing kota menggunakan bantuan google!

Kamu dapat mencarinya di google untuk jarak masing masing kota dengan

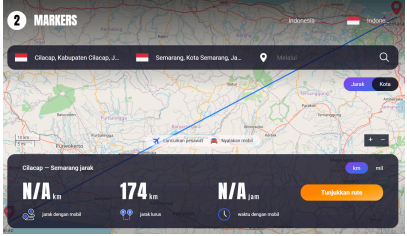
menggunakan keyword : “From city1 to city2”. Example :



Didapatkan jarak antara kota 1 dan kota 2 adalah 41,1 km.

1. Carilah jarak garis lurus ke lokasi tujuan!

Kamu dapat menggunakan website <https://id.2markers.com/> masukan cilacap sebagai kota asal dan nama kota lainnya sebagai tujuan untuk mengukur jarak lurus dari cilacap ke kota kota lainnya. Example :



1. Gunakan Teknik Best First Search untuk menentukan rute perjalanan untuk Andri!
2. **Kasus 1 – Ketidakpastian**

Seorang penjual baju online ingin memprediksi apakah seorang pembeli akan membeli produknya atau tidak. Penjual tersebut memiliki dataset pembeli yang telah membeli dan tidak membeli produk. Dataset tersebut terdiri dari variabel independen berupa usia dan penghasilan, serta variabel dependen berupa pembelian (1=ya, 0=tidak). Gunakan metode

Naive Bayes untuk memprediksi apakah seseorang akan membeli produk dari toko online tersebut berdasarkan usia dan penghasilan.

1. **Kasus 1 – Fuzzy Logic**

Seorang petani ingin mengetahui kapan waktu yang tepat untuk melakukan penyiraman tanaman jagung di lahan pertaniannya. Ia telah memasang sebuah sensor yang mengukur kelembaban tanah pada lahan tersebut, tetapi ia masih kesulitan menentukan waktu yang tepat untuk melakukan penyiraman.

Dalam kasus ini, Fuzzy Logic dapat digunakan untuk membantu petani menentukan waktu yang tepat untuk melakukan penyiraman. Fuzzy Logic digunakan untuk mengukur kelembaban tanah pada lahan pertanian, dan menghasilkan keluaran berupa "kering", "lembab", atau "basah", yang kemudian digunakan untuk menentukan waktu penyiraman yang tepat.

Untuk menjawab permasalahan di atas gunakan library Scikit-fuzzy lalu menggunakan control dari scikit-fuzzy untuk membuat nilai variabel penampung input, dengan range 0 sampai dengan 100 dengan increment sebesar 1. Artinya, himpunan dari variabel "Kelembaban" terdiri dari 101 elemen yang dimulai dari 0 dan berakhir pada 100. seperti berikut ini

**ctrl.Antecedent(np.arange(0, 101, 1), 'kelembapan')**

buat juga variabel untuk menampung nilai outputnya dengan range 0 sampai dengan 30 dengan increment sebesar 1. Artinya, himpunan dari variabel "waktu\_siram" terdiri dari 31 elemen yang dimulai dari 0 dan berakhir pada 30, seperti berikut :

**ctrl.Consequent(np.arange(0, 31, 1), 'waktu\_siram')**

**Note** :

Nilai **kelembaban** dapat di set menjadi 3 kondisi, berdasarkan rentang nilai kelembaban

0-100 yaitu: kering (0-30), lembab (31-60) dan basah (61-101).

Nilai **waktu\_siram** dapat di set menjadi 3 kondisi, berdasarkan rentang nilai waktu\_siram

0-30 yaitu: cepat (0-10), sedang (11-20) dan lambat (21-31)

Gunakan urutan langkah-langkah berikut untuk menjawab studi kasus ini :

1. Membuat variabel input dan output menggunakan Antecedent dan Consequent

2. Membuat fungsi keanggotaan untuk variabel kelembaban dan waktu\_siram

menggunakan trimf.

3. Membuat aturan fuzzy menggunakan ctrl.Rule

4. Membuat sistem kontrol fuzzy menggunakan ctrl.ControlSystem

5. Menampilkan grafik fungsi keanggotaan menggunakan view()

# JAWABAN

1. Berikut langkah yang perlu dilakukan untuk melakukan pencarian rute menggunakan teknik Best First Search:
   1. Inisialisasi variable graph untuk menampung informasi mengenai jarak antar kota.

Data mengenai jarak antar kota yaitu:

* Cilacap = Kroya(29,1 km), Purwokerto (49,6 km)
* Kroya = Cilacap(29,1 km), Purwokerto (32, 2 km), Kebumen (53, 6 km)
* Kebumen = Kroya(53,6 km), Purworejo (42 km), Purwokerto (71,2 km)
* Purworejo = Kebumen(42 km), Magelang (43 km)
* Magelang = Purworejo(43 km), Wonosobo (60,4 km), Temanggung (25,3 km), Boyolali (64,3 km)
* Boyolali = Magelang(64,3 km), Klaten (34,9 km), Salatiga (24,4 km), Solo (36,6 km)
* Klaten = Boyolali(34,9 km
* Solo = Boyolali(36,6 km), Purwodadi (67,4 km), Sragen (34,8 km), Sukoharjo (14,2 km)
* Sukoharjo = Solo(14,2 km), Wonogiri (40,7 km)
* Wonogiri = Sukoharjo(40,7 km)
* Sragen = Solo(34,8 km), Blora (83,7 km
* Blora = Purwodadi(65,3 km), Rembang(36,1 km), Sragen (83,7 km)
* Rembang = Blora(36,1 km), Kudus(60,9 km)
* Kudus = Rembang(60,9 km), Purwodadi (47,9 km), Demak (42,5 km)
* Purwodadi = Demak(39 km), Kudus (47,9 km), Blora (65,3 km), Solo (67,4 km)
* Demak = Semarang(31,4 km), Purwodadi (39 km), Kudus (42,5 km)
* Semarang = Demak(31,4 km), Salatiga(51,5 km), Kendal (31,7 km)
* Salatiga = Semarang(51,5 km), Boyolali(24,4 km), Temanggung(55,6 km)
* Temanggung = Kendal(71,2 km), Salatiga(55,6 km), Magelang(25,3 km), Wonosobo (40 km)
* Wonosobo = Temanggung(40 km), Magelang(60,4 km), Banjarnegara(42,5 km)
* Banjarnegara = Wonosobo(42,5 km), Purbalingga(33,4 km)
* Purbalingga= Banjarnegara(33,4 km), Purwokerto(16,8 km),
* Purwokerto = Purbalingga(16,8 km), Kebumen(71,2 km), Kroya(32,2 km), Cilacap (49,6 km), Slawi (89 km)
* Slawi = Purwokerto(89 km), Brebes(25,3 km), Tegal(12 km)
* Brebes = Slawi(25,3 km), Tegal(40,9 km)
* Tegal = Brebes(40,9 km), Slawi(12 km)
* Pemalang = Tegal(32,4 km), Purbalingga(68,8 km), Pekalongan(47,5 km)
* Pekalongan = Pemalang(47,5 km), Kendal(74,7 km)
* Kendal = Pekalongan(74,7 km), Temanggung(71,2 km), Semarang (31,7 km)



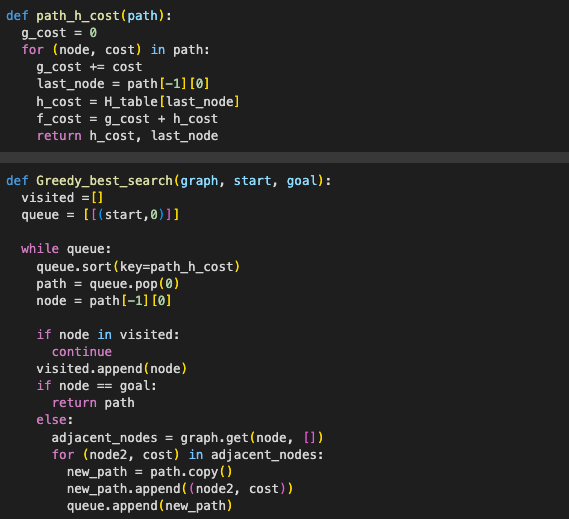
* 1. Inisialisasi variable H\_table untuk menampung data mengenai jarak garis lurus dari setiap kota ke kota tujuan.

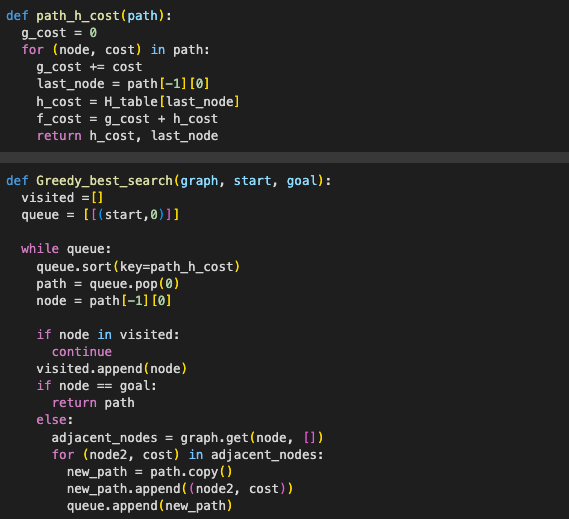
Berikut data mengenai jarak garis lurus dari setiap kota ke kota cilacap :

* Brebes: 94 KM
* Tegal: 92 KM
* Pemalang: 95 KM
* Kendal: 156 KM
* Semarang: 174 KM
* Demak: 199 KM
* Kudus: 224 KM
* Rembang: 278 KM
* Slawi: 80 KM
* Pekalongan: 115 KM
* Purwokerto: 39 KM
* Purbalingga: 51 KM
* Wonosobo: 105 KM
* Temanggung: 134 KM
* Salatiga: 169 KM
* Purwodadi: 219 KM
* Blora: 276 KM
* Banjarnegara: 65 KM
* Cilacap: 0 KM
* Kroya: 23 KM
* Kebumen: 71 KM
* Magelang: 134 KM
* Boyolali: 174 KM
* Solo: 200 KM
* Sragen: 223 KM
* Purworejo: 109 KM
* Klaten: 181 KM
* Sukoharjo: 197 KM
* Wonogiri: 211 KM



* 1. Pendefinisian fungsi yang dibutuhkan dalam pencarian rute





* 1. Memanggil fungsi Greedy\_best\_search dan menampilkan hasil nya.

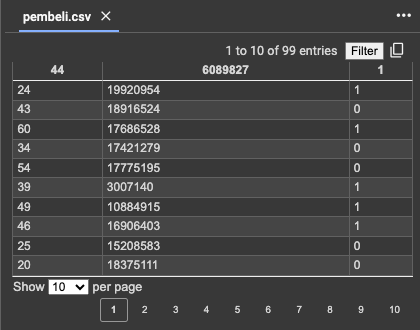


**Output:**

Rute menggunakan teknik Greedy Best First Search adalah [('Semarang', 0), ('Kendal', 31.7), ('Pekalongan', 76.7), ('Pemalang', 47.2), ('Purbalingga', 68.8), ('Purwokerto', 20.2), ('Cilacap', 49.6)]

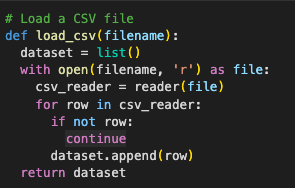
1. Berikut langkah yang dilakukan untuk dapat memprediksi apakah seseorang akan membeli produk dari toko online berdasarkan usia dan penghasilan menggunakan algoritma Naive Bayes
   1. Melakukan import data pembeli



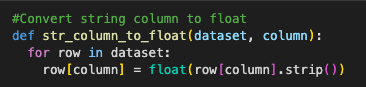


Ada tiga kolom dalam tabel tersebut, dengan kolom pertama menunjukkan usia pelanggan, kolom kedua menunjukkan pendapatan mereka, dan kolom ketiga memuat informasi apakah mereka telah membeli produk dari penjual baju online tersebut (1) atau tidak (0).

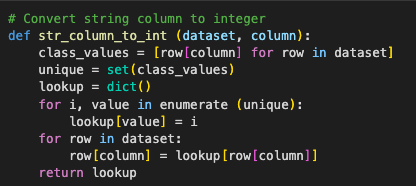
* 1. Pendefinisian fungsi yang dibutuhkan dalam algoritma Naive Bayes, yang diantaranya:
* **load\_csv**: membaca file CSV dan mengembalikan dataset dalam bentuk daftar.



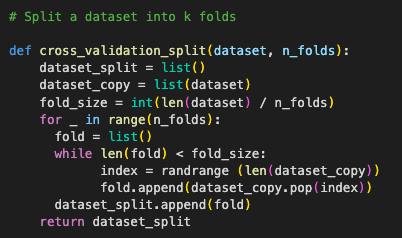
* **str\_column\_to\_float**: Untuk mengonversi kolom dataset dari string menjadi float.



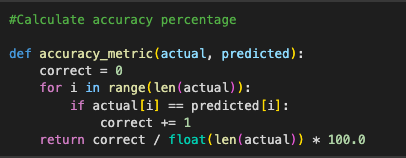
* **str\_column\_to\_int**: mengonversi kolom dataset dari string menjadi integer dan membuat kamus lookup untuk nilai kelas.



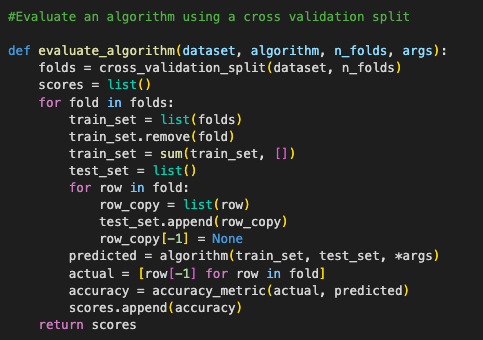
* **cross\_validation\_split**: Untuk membagi dataset menjadi k bagian untuk melakukan cross-validation.



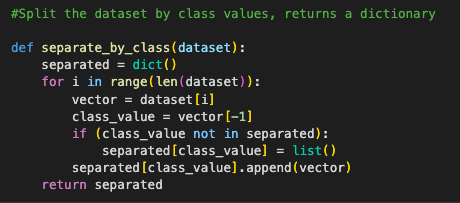
* **accuracy\_metric**: Untuk menghitung akurasi prediksi dengan membandingkan nilai sebenarnya dengan nilai yang diprediksi.



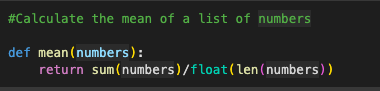
* **evaluate\_algorithm**: Untuk mengevaluasi algoritma dengan menggunakan validasi silang dan menghitung akurasi untuk setiap lipatan.



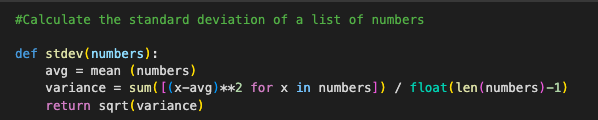
* **separate\_by\_class**: Untuk memisahkan dataset berdasarkan kelas.



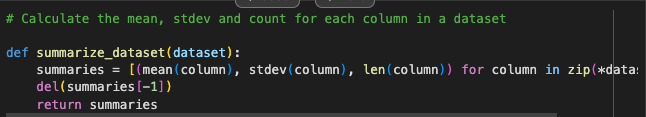
* **mean**: Untuk menghitung rata-rata dari suatu daftar angka.



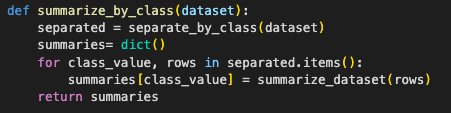
* **stdev**: Untuk menghitung standar deviasi dari sebuah daftar angka.



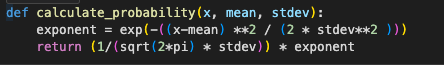
* **summarize\_dataset**: Untuk menghitung rata-rata, standar deviasi, dan jumlah data untuk setiap kolom dalam dataset.



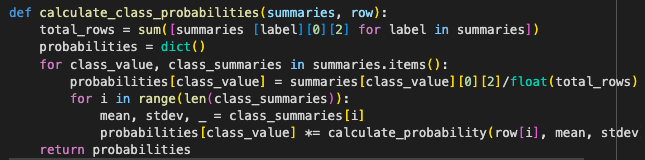
* **summarize\_by\_class**: Untuk memisahkan dataset berdasarkan kelas dan menghitung statistik ringkasan untuk setiap kelas.



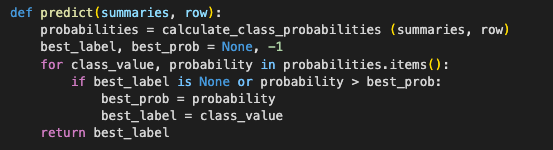
* **calculate\_probability**: Untuk menghitung probabilitas distribusi Gaussian pada suatu nilai.



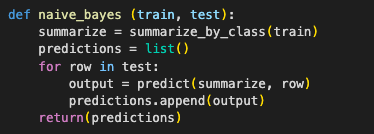
* **calculate\_class\_probabilities**: Untuk menghitung probabilitas memprediksi setiap kelas untuk sebuah baris.



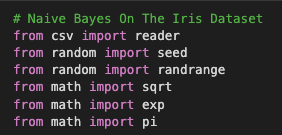
* **predict**: Untuk memprediksi kelas untuk sebuah baris berdasarkan probabilitas terbesar.



* **naive\_bayes**: Fungsi penerapan algoritma Naive Bayes untuk klasifikasi, menggunakan ringkasan statistik yang dihitung dari data pembeli.

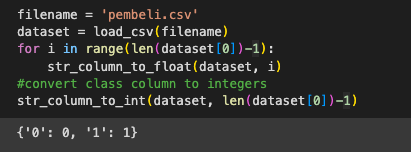


Sebelum mendefinisikan fungsi-fungsi di atas, perlu juga melakukan import library eksternal yang pastinya akan digunakan pada fungsi-fungsi tersebut.



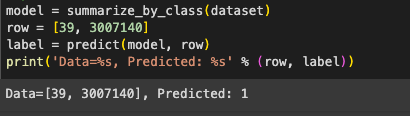
* 1. Memanggil fungsi naivce\_bayes dan menampilkan hasil prediksi

Setelah menyelesaikan semua fungsi yang diperlukan untuk mengimplementasikan algoritma Naïve-Bayes, langkah berikutnya adalah memasukkan file .csv ke dalam program untuk diolah dengan menggunakan perhitungan Naive-Bayes.





Lalu, kita akan mencoba mengevaluasi peluang bahwa salah satu individu dalam dataset akan melakukan pembelian produk dari toko online. Prediksi (1) akan menunjukkan bahwa individu tersebut akan membeli produk dari toko online, sementara prediksi (0) akan menunjukkan sebaliknya, bahwa individu tersebut tidak akan melakukan pembelian produk dari toko online.



Pada kode di atas, dapat terlihat, saya mencoba memprediksi pembeli jika usia nya 60 tahun dan pendapatan Rp.3.007.140,00 pada variable "row", apakah akan melakukan pembelian produk.

Dan hasilnya:

"Data=[39, 10884915], Predicted: 0."

Yang artinya pembeli dengan data tersebut, jika diprediksi menggunakan metode Naive Bayes tidak melakukan pembelian.

1. Kita dapat menggunakan pustaka Scikit-fuzzy dan mengontrolnya dengan menggunakan kontrol dari Scikit-fuzzy untuk membuat variabel penampung input dan output. Berikut adalah langkah-langkah yang perlu dilakukan:
   1. Install *library scikit-fuzzy*



* 1. Melakukan import library eksternal yang diperlukan



* 1. Membuat variabel input dan output menggunakan Antecedent dan Consequent



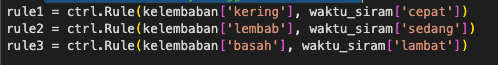
* 1. Membuat fungsi keanggotaan untuk variabel kelembaban dan waktu\_siram menggunakan trimf
* Variabel kelembaban



* Variabel waktu\_siram



* 1. Membuat aturan fuzzy dengan ctrl.Rule



* 1. Membuat sistem kontrol fuzzy dengan ctrl.ControlSystem



* 1. Menampilkan grafik fungsi keanggotaan



Output:

