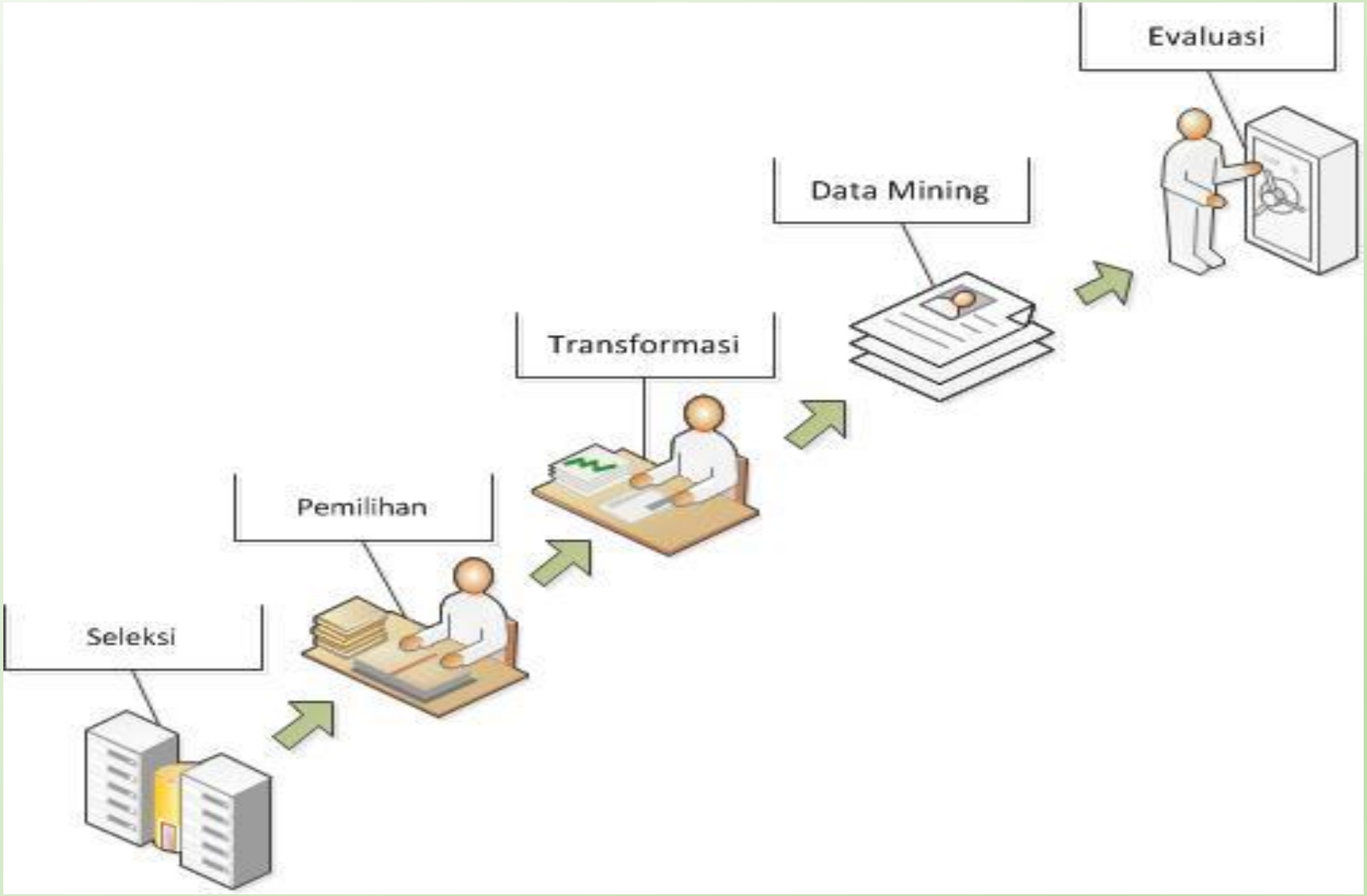
Pemrosesan Data

Kuliah : 30-September-2021

Knowledge Discovery in Database (KDD)



Cosine Similarity

- Cosine similarity adlah ukuran jarak ynag digunakan untuk data yang berupa vector dokumen
- Vektor dokumen→sebuah dokumen bisa dipandang sebagai sebuah data yang berisi ratusan atau ribuan atribute.
- Setiap atribute menyatakan sebuah term atau istilah (kata) yang nilainya berup frekuensi kemunculan kata dalam sebuah dokumen.
- Contoh : Table-5, terdapat 10 atribute (kata), pada dokumen 1 kata agama muncul 3x dan kata aksi muncul 4x dst.

- Contoh : Table-5

Dokumen	Agama	Aksi	Bela	Calon	Gubernur	Islam	Monas	Pemilihan	Presiden
Dok-1	3	4	2	0	0	1	1	0	0
Dok-2	1	5	2	0	0	4	3	0	0
Dok-3	0	3	2	2	2	2	0	0	0
Dok-4	0	0	0	4	0	0	0	3	2
Dok-5	0	0	0	4	0	0	0	5	6

- $sim(x, y = \frac{x.y^T}{\|x\|\|y\|}$
- $\|x\| = \sqrt{x_1{}^2 + x_2{}^2 + .. + x_p{}^2}$
- Sim(x,y)=0, berarti tidak memiliki kesamaan
- Sim(x,y)=1, berarti sama persis
- **Hitung cosine similarity dokumen dok-1 dan dok-2**

Pembersihan Data

- Tujuan pemrosesan data
  - Untuk mempermudah memahami data, sehingga mempermudah dalam pemilihan Teknik dan metode data mining
  - Meningkatkan kualitas data sehingga hasil data mining mnejadi lebih baik
  - Meningkatkan efisiensi dan kemudahan proses penambangan data
- Pemrosesan data dapat dilakukan dengan :
  - Pembersihan data, dan atau
  - Reduksi data, dan atau
  - Penambahan data, dan atau
  - Transformasi data

Pembersihan data

- Data seperti apa yang disebut kotor?
- Sekotor apa data yang ada sehingga perlu dibersihkan?
- Bisakah data kotor langsung dibuang?
- Adakah Teknik data mining yang bisa digunakan untuk data kotor namun memberikan hasil yang baik?

- Sebuah data dikatakan tidak bersih→jika mengandung kotoran yang berupa :
  - Nilai kosong, dan atau
  - Derau, dan atau
  - Pencilan, dan atau
  - Inkonsistensi
- Semakin tinggi kandungan kotoran pada suatu data semakin tinggi pula tingkat kekotoran pada suatu data.
- Teknik data mining, jika diterapkan dalam data yang kotor, maka umumnya memberikan kinerja yang tidak baik
- Sebelum menggunakan data mining, pada data yang kotor harus dibersihkan terlebih dahulu

- Membersihkan nilai kosong
  - Jika kita memiliki sebuah data yang mengandung tuple dengan satu atau lebih attribute tanpa nilai, kita dapat membersihkannya dengan cara :
    - Abaikan tuple
    - Isi attribute kosong secara manual
    - Gunakan konstanta global untuk mengisi attribute kosong
    - Gunakan sebuah nilai tendensi sentral (missal rata-rata atau median) untuk mengisi attribute kosong
    - Gunakan rata-rata atau median dari suatu attribute untuk mengisi sample dalam kelas yang sama dengan tuple tersebut.
    - Gunakan nilai yang paling mungkin untuk mengisi atribute kosong→nilai yang paling mungkin dapat ditentukan dengan menggunakan regresi atau inferensi

Menghaluskan data berderau

- Pembersihan data berderau dilakukan dengan cara :
  - Binning atau wadahan adalah metode ini sangat mudah dilakukan, yaitu dengan cara mengurutkan nilai nilai pada suatu atribut lalu membaginya ke dalam sejumlah (bin) secara merata dan akhirnya pengalusan dapat dilakukan mengunakan tiga cara yaitu rata-rata, media atau batas nilai minimum dan maksimum.
  - Regresi→Suatu regresi linier biasanya mencari persamaan garis terbaik yang paling mendekati nilai nilai dari dua buah atribut sedemikian hingga suatu atribut dapat digunakan untuk memprediksi kan atribut yang lain
  - Clustering→Teknik ini memungkinkan anda dapat mempartisi data secara lebih baik, tidak harus merata berdasarkan frekuensi seperti pada teknik binning.

Membuang Pencilan

- Data-data pencilan dapat ditemukan menggunakan tendensi sentral, grafik statistik boxplot, berbagai teknik visualisasi data atau clustering
- Jika data pencilan diperoleh, maka kita dapat membuang tuple tersebut.

Memperbaiki inkonsistensi

- Inkonsistensi data dapat disebabkan oleh beberapa faktor diantaranya :
  - kurang bagusnya desain formulir pemasukan data
  - kesalahan operator dalam memasukkan data
  - kesalahan yang disengaja oleh pengguna data
  - Data kedaluwarsa
  - representasi data yang inkonsisten
  - penggunaan kode yang inkonsisten
  - kesalahan dalam perangkat perkam data
  - kesalahan sistem
  - integrasi data yang inkonsisten
- Data yang inkonsistensi mungkin saja bisa dikoreksi secara manual→tapi membutuhkan energi yang lebih
- Kesalahan yang mengakibatkan inkonsistensi data memerlukan transformasi data.
- Dapat menggunakan alat bantu komersial untuk transformasi data, misalnya ETL (extraction/transformation/loading)

Integrasi Data

- Dalam data mining secara praktis integrasi atau penggabungan sejumlah basis data berbeda seringkali harus dilakukan sebagai
- misal perusahaan operator telekomunikasi memiliki seduah cabang di berbagai kota dengan sistem informasi dan basis data yang berbeda-beda.
- Integrasi data yang baik akan menghasilkan data gabungan dengan sedikit redudansi dan atau inkonsistensi sehingga meningkatkan akurasi dan kecepatan proses data mining
- Permasalan utama dalam integrasi data adalah heterogenitas semantik dan struktur dari semua data yang diintegrasikan

Feature Selection

- Feature Selection adalah suatu kegiatan yang umumnya bisa dilakukan secara preprocessing dan bertujuan untuk memilih feature yang berpengaruh dan mengesampingkan feature yang tidak berpengaruh dalam suatu kegiatan pemodelan atau penganalisaan data.
- Secara garis besar ada dua kelompok besar dalam pelaksanaan feature selection:
  - Ranking Selection
  - Subset Selection

Ranking selection

- Ranking selection secara khusus memberikan ranking pada setiap feature yang ada dan mengesampingkan feature yang tidak memenuhi standar tertentu.
- Ranking selection menentukan tingkat ranking secara independent antara satu feature dengan feature yang lainnya.
- Feature yang mempunyai ranking tinggi akan digunakan dan yang rendah akan dikesampingkan.
- Ranking selection ini biasanya menggunakan beberapa cara dalam memberikan nilai ranking pada setiap feature misalnya regression, correlation, mutual information dll

Subset selection

- Subset selection adalah metode selection yang mencari suatu set dari features yang dianggap sebagai optimal feature.
- Ada tiga jenis metode yang bisa digunakan yaitu selection dengan tipe :
  - Selection type Wrapper,
  - selection dengan tipe filter
  - selection dengan tipe embedded.

Feature Selection Tipe Wrapper

- Feature Selection Tipe Wrapper: feature selection tipe wrapper ini melakukan feature selection dengan melakukan pemilihan bersamaan dengan pelaksanaan pemodelan.
- Selection tipe ini menggunakan suatu criterion yang memanfaatkan classification rate dari metode pengklasifikasian/pemodelan yang digunakan.
- Untuk mengurangi computational cost, proses pemilihan umumnya dilakukan dengan memanfaatkan classification rate dari metode pengklasifikasian untuk pemodelan dengan nilai terendah (misalnya dalam kNN, menggunakan nilai k terendah).

- Untuk tipe wrapper, perlu untuk terlebih dahulu melakukan feature subset selection sebelum menentukan subset mana yang merupakan subset dengan ranking terbaik.
- Feature subset selection bisa dilakukan dengan memanfaatkan metode sequential forward selection (dari satu menjadi banyak feature), sequential backward selection (dari banyak menjadi satu), sequential floating selection (bisa dari mana saja), GA, Greedy Search, Hill Climbing, Simulated Annealing, among others.

Feature Selection Tipe Filter

- *Feature Selection Tipe Filter*: feature selection dengan tipe filter hampir sama dengan selection tipe wrapper dengan menggunakan intrinsic statistical properties dari data.
- Tipe filter berbeda dari tipe wrapper dalam hal pengkajian feature yang tidak dilakukan bersamaan dengan pemodelan yang dilakukan.
- Selection ini dilakukan dengan memanfaatkan salah satu dari beberapa jenis filter yang ada.
- Metode filter ini memilih umumnya dilakukan pada tahapan preprocessing dan mempunyai computational cost yang rendah.

Feature Selection Tipe Embedded

- Feature Selection Tipe Embedded: feature selection jenis ini memanfaatkan suatu learning machine dalam proses feature selection.
- Dalam sistem selection ini, feature secara natural dihilangkan, apabila learning machine menganggap feature tersebut tidak begitu berpengaruh.
- Beberapa learning machine yang bisa digunakan antara lain: Decision Trees, Random Forests dan lain-lain.

Klasifikasi (kNN)

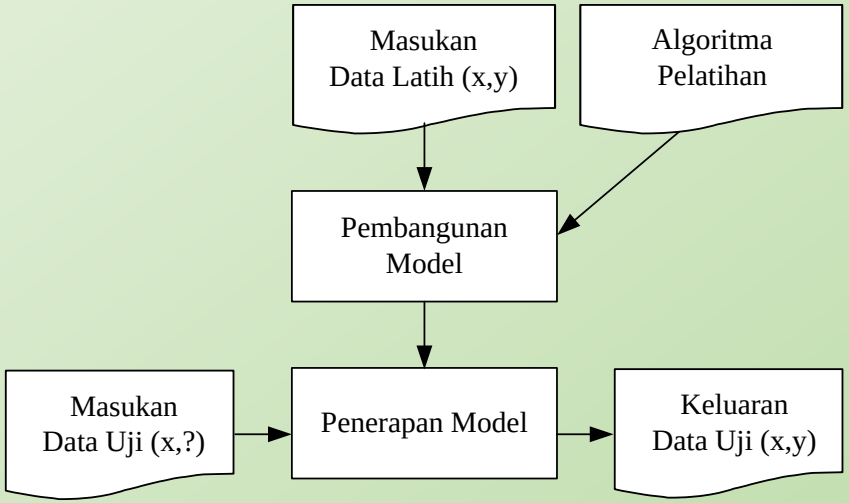
Kuliah : 30-September-2021

Konsep Klasifikasi

- Klasifikasi merupakan suatu pekerjaan yang melakukan penilaian terhadap suatu obyek data untuk masuk dalam suatu kelas tertentu dari sejumlah kelas yang tersedia.
- Ada dua pekerjaan utama yang dilakukan:
  - Pembangunan model sebagai prototype untuk disimpan sebagai memori,
  - Menggunakan model tersebut untuk melakukan pengenalan/ klasifikasi/prediksi pada suatu obyek data lain untuk dinilai bahwa obyek data tersebut masuk pada kelas mana dalam model yang sudah disimpannya.
- Contoh, pengklasifikasian jenis hewan
  - dimana hewan mempunyai sejumlah atribut sehingga dari atribut tersebut dapat diketahui jika ada hewan baru maka bisa diketahui hewan tersebut masuk dalam kelas yang mana sesuai dengan kelas yang sudah dipelajari/diketahui.

Konsep Klasifikasi

- Klasifikasi adalah pekerjaan yang melakukan pelatihan/ pembelajaran terhadap fungsi target *f* yang memetakan setiap set atribut (fitur) *x* ke satu dari sejumlah label kelas *y* yang tersedia.
- Pekerjaan pelatihan akan menghasilkan suatu model yang kemudian disimpan sebagai memori.
- Model dalam klasifikasi mempunyai arti yang sama dengan black box,
  - Suatu model yang menerima masukan kemudian mampu melakukan pemikiran terhadap masukan tersebut dan memberikan jawaban sebagai keluaran dari hasil pemikirannya.



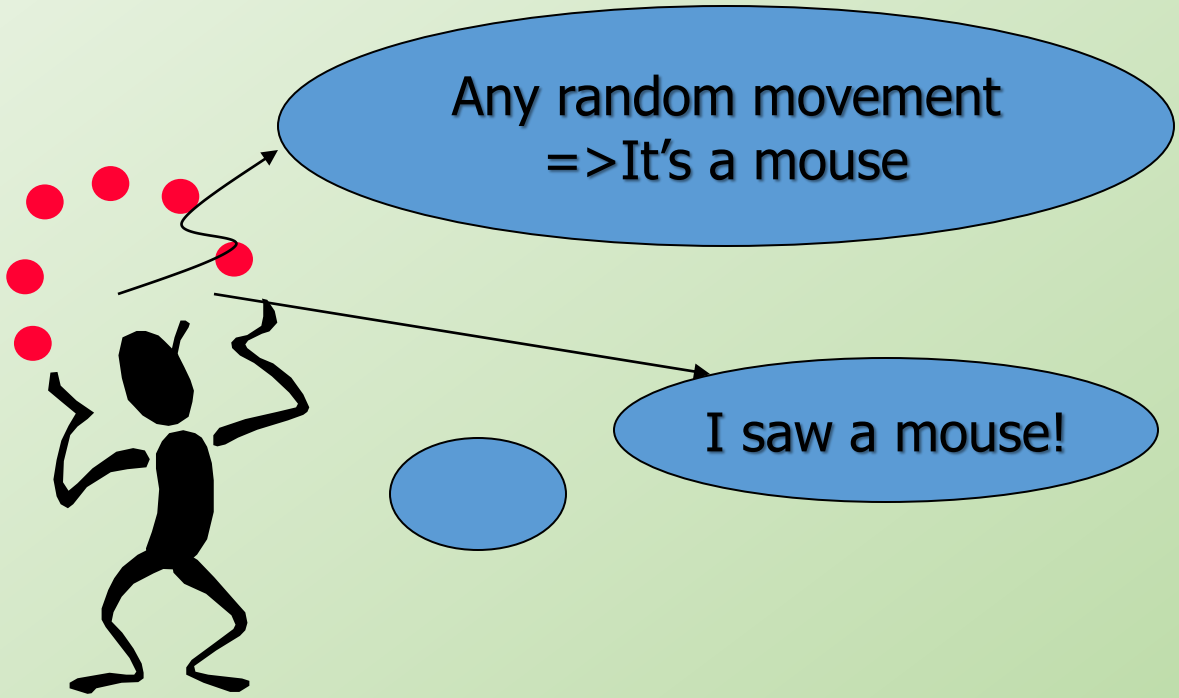
Algoritma pelatihan yang sudah dikembangkan

Metode Pembelajaran (Pelatihan)

- Eager Learning
  - Secara eksplisit mendeskripsikan fungsi target pada semua bagian training set (data latih).
- Instance-based Learning
  - Learning = Menyimpan semua training instances
  - Prediksi = Menggunakan fungsi tujuan (model) pada instansi baru (data uji)
  - Disebut juga “Lazy” learning.

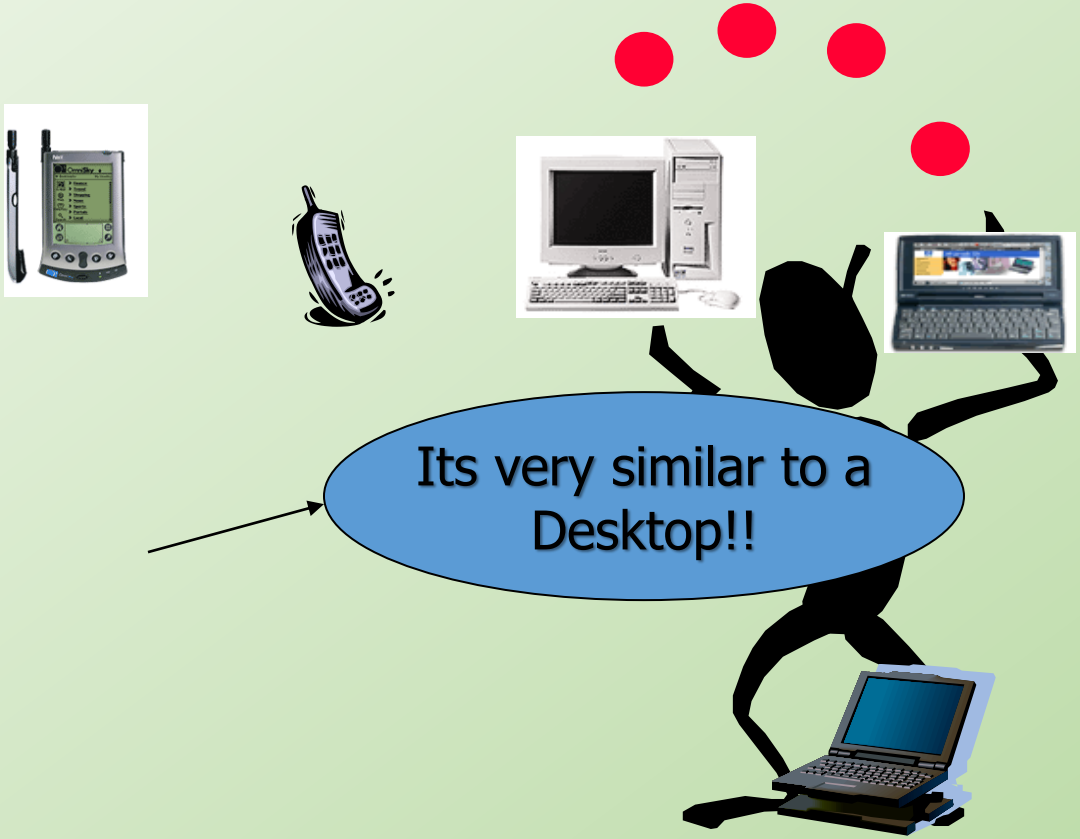
Metode Pembelajaran

- Eager Learning
- Misal: ANN, SVM, Decision Tree, Bayesian, dsb.



Metode Pembelajaran

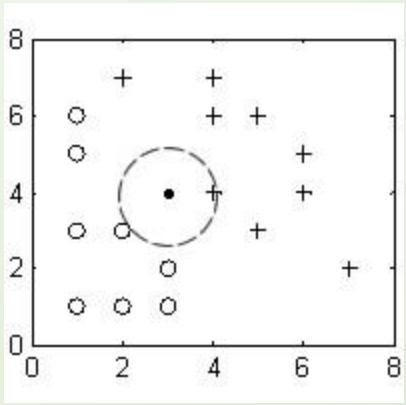
- Lazy Learning
- Misal: K-NN, Fuzzy K-NN, Fuzzy K-NNC, Weighted Regression, Case-based reasoning, dsb.



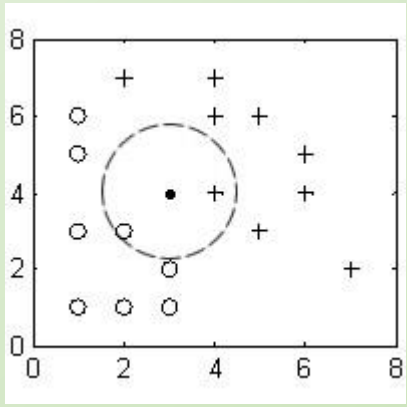
K-Nearest Neighbor

- Algoritma yang melakukan klasifikasi berdasarkan kedekatan lokasi (jarak) suatu data dengan data yang lain.
- Prinsip sederhana yang diadopsi oleh algoritma K-NN adalah: "*Jika suatu hewan berjalan seperti bebek, bersuara kwek-kwek seperti bebek, dan penampilannya seperti bebek, maka hewan itu mungkin bebek*".
- Pada algoritma K-NN, data berdimensi *q*, dapat dihitung jarak dari data tersebut ke data yang lain,
  - Nilai jarak ini yang digunakan sebagai nilai kedekatan/kemiripan antara data uji dengan data latih.

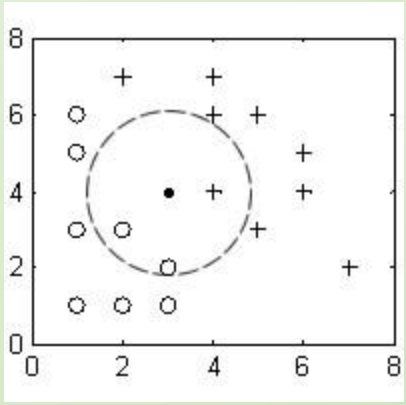
K-Nearest Neighbor



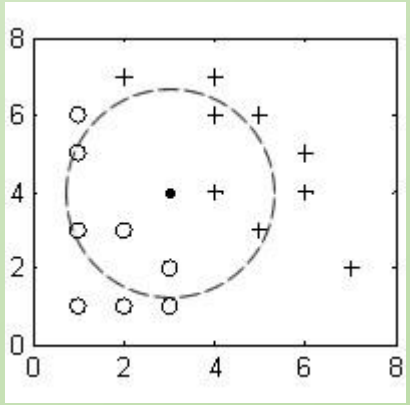
1 tetangga terdekat (1-NN)



2 tetangga terdekat (2-NN)



3 tetangga terdekat (3-NN)



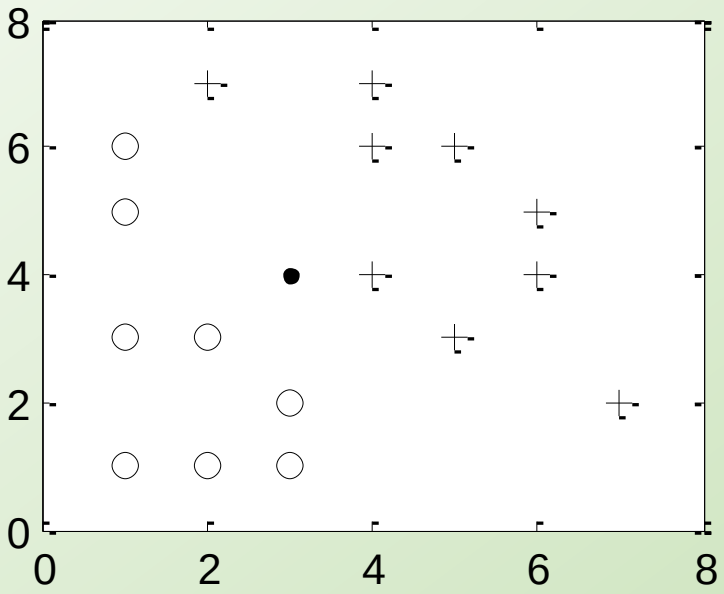
7 tetangga terdekat (7-NN)

Algoritma K-NN

- z = (x',y'), adalah data uji dengan vektor x' dan label kelas y' yang belum diketahui
- Hitung jarak d(x',x), jarak diantara data uji z ke setiap vektor data latih, simpan dalam D
- Pilih $D_z \subseteq D$, yaitu K tetangga terdekat dari z

$$y' = \arg\max_{v} \sum_{(x_i, y_i) \in D_z} I(v = y_i)$$

Contoh



Data uji adalah data (3,4), fitur X=3, Y=4.

Akan dilakukan prediksi, masuk dalam kelas yang manakah seharusnya ?

Gunakan jarak Euclidean !

Data latih

Data	X	Y	Kelas
1	1	1	0
2	2	1	0
3	3	1	0
4	3	2	0
5	7	2	1
6	1	3	0
7	2	3	0
8	5	3	1
9	4	4	1
10	6	4	1
11	1	5	0
12	6	5	1
13	1	6	0
14	4	6	1
15	5	6	1
16	2	7	1
17	4	7	1

Prediksi dengan K-NN

Jarak data uji (3,4) ke 17 data latih

Nomor data	x	y	Kelas asli	Jarak data uji ke data latih	1-NN	3-NN	7-NN
1	1	1	0	3.6055	0	0	0
2	2	1	0	3.1622	0	0	0
3	3	1	0	3	0	0	0
4	3	2	0	2	0	1	1
5	7	2	1	4.4721	0	0	0
6	1	3	0	2.2360	0	0	1
7	2	3	0	1.4142	0	1	1
8	5	3	1	2.2360	0	0	1
9	4	4	1	1	1	1	1
10	6	4	1	3	0	0	0
11	1	5	0	2.2360	0	0	1
12	6	5	1	3.1622	0	0	0
13	1	6	0	2.8284	0	0	0
14	4	6	1	2.2360	0	0	1
15	5	6	1	2.8284	0	0	0
16	2	7	1	3.1622	0	0	0
17	4	7	1	3.1622	0	0	0

Prediksi dengan K-NN

Untuk K=1

Data latih yang terdekat adalah data nomor 9 (4,4) dengan kelas 1, maka data uji (3,4) diprediksi masuk kelas 1.

Untuk K=3

Data latih yang terdekat adalah data nomor 9 (4,4) dengan kelas 1, data nomor 7 (2,3) dan data nomor 4 (3,2) dengan kelas 0, karena kelas 1 berjumlah 1 dan kelas 0 berjumlah 2 (***lebih banyak kelas 0 daripada kelas 1***) maka data uji (3,4) diprediksi masuk kelas 0.

Untuk K=7

Data latih yang terdekat adalah data nomor 8 (5,3), 9 (4,4), 14 (4,6) dengan kelas 1, data nomor 4 (3,2), 6 (1,3), 7 (2,3), dan 11 (1,5) dengan kelas 0, karena kelas 1 berjumlah 3 dan kelas 0 berjumlah 4 (***lebih banyak kelas 0 daripada kelas 1***) maka data uji (3,4) diprediksi masuk kelas 0.

Evaluasi K-NN

- Algoritma yang menggunakan seluruh data latih untuk melakukan proses klasifikasi (*complete storage*).
  - Mengakibatkan untuk data dalam jumlah yang sangat besar, proses prediksi menjadi sangat lama.
- Tidak membedakan setiap fitur dengan suatu bobot
  - Pada ANN (Artificial Neural Network) yang berusaha menekan fitur yang tidak punya kontribusi terhadapklasifikasi menjadi 0 pada bagian bobot,
  - NN tidak ada bobot untuk masing-masing fitur.
- Menyimpan sebagian atau semua data dan hampir tidak ada proses pelatihan,
  - maka K-NN sangat cepat dalam proses training (karena memang tidak ada) tetapi sangat lambat dalam proses prediksi.
- Hal yang rumit adalah menentukan nilai K yang paling sesuai
- K-NN pada prinsipnya memilih tetangga terdekat,
  - Parameter jarak juga penting untuk dipertimbangkan sesuai dengan kasus datanya. Euclidean sangat cocok untuk menggunakan jarak terdekat (lurus) antara dua data, tetapi Manhattan sangat *robust* untuk mendeteksi outlier dalam data.

FUZZY K-NEAREST NEIGHBOR (FK-NN)

Fuzzy K-NN

- K-NN melakukan prediksi secara tegas pada uji berdasarkan perbandingan K tetangga terdekat.
- Fuzzy K-Nearest Neighbor (FK-NN) diperkenalkan oleh Keller et al (1985) dengan mengembangkan K-NN yang digabungkan dengan teori fuzzy dalam memberikan definisi pemberian label kelas pada data uji yang diprediksi.
- Pada teori fuzzy, sebuah data mempunyai nilai keanggotaan pada setiap kelas,
  - yang artinya sebuah data bisa dimiliki oleh kelas yang berbeda dengan nilai derajat keanggotaan dalam interval [0,1].
- Teori himpunan fuzzy men-generalisasi teori K-NN klasik dengan mendefinisikan nilai keanggotaan sebuah data pada masing-masing kelas.

Nilai keanggotaan

$$u(x,c_i) = \frac{\sum_{k=1}^{K} u(x_k,c_i) * d(x,x_k)^{\frac{-2}{(m-1)}}}{\sum_{k=1}^{K} d(x,x_k)^{\frac{-2}{(m-1)}}}$$

$u(x,c_i)$ adalah nilai keanggotaan data x ke kelas c_i

K adalah jumlah tetangga terdekat yang digunakan

$u(x_k,c_i)$ adalah nilai keanggotaan data tetangga dalam K tetangga pada kelas c_i, nilainya 1 jika data latih x_k milik kelas c_i atau 0 jika bukan milik kelas c_i

$d(x, x_k)$ adalah jarak dari data x ke data x_k dalam K tetangga terdekat

m adalah bobot pangkat (*weight exponent*) yang besarnya m > 1

Nilai keanggotaan suatu data pada kelas sangat dipengaruhi oleh jarak data itu ke tetangga terdekatnya,

semakin dekat ke tetangganya maka semakin besar nilai keanggotaan data tersebut pada kelas tetangganya, begitu pula sebaliknya.

Jarak tersebut diukur dengan N dimensi (fitur) data

Jarak yang digunakan

$$d(x_i, x_j) = \left(\sum_{l=1}^{N} \left| x_{il} - x_{jl} \right|^p \right)^{\frac{1}{p}}$$

N adalah dimensi (jumlah fitur) data.
Untuk p adalah penentu jarak yang digunakan,

jika p=1 maka jarak yang digunakan adalah Manhattan,
jika p=2 maka jarak yang digunakan adalah Euclidean,
jika p=∞ maka jarak yang digunakan adalah Chebyshev.

Meskipun FK-NN menggunakan nilai keanggotaan untuk menyatakan keanggotaan data pada setiap kelas, tetapi untuk memberikan keluaran akhir, FK-NN tetap harus memberikan kelas akhir hasil prediksi, untuk keperluan ini, FK-NN memilih ***kelas dengan nilai keanggotaan terbesar*** pada data tersebut

Algoritma FK-NN

- Normalisasikan data menggunakan nilai terbesar dan terkecil data pada setiap fitur.
- Cari K tetangga terdekat untuk data uji x menggunakan persamaan:

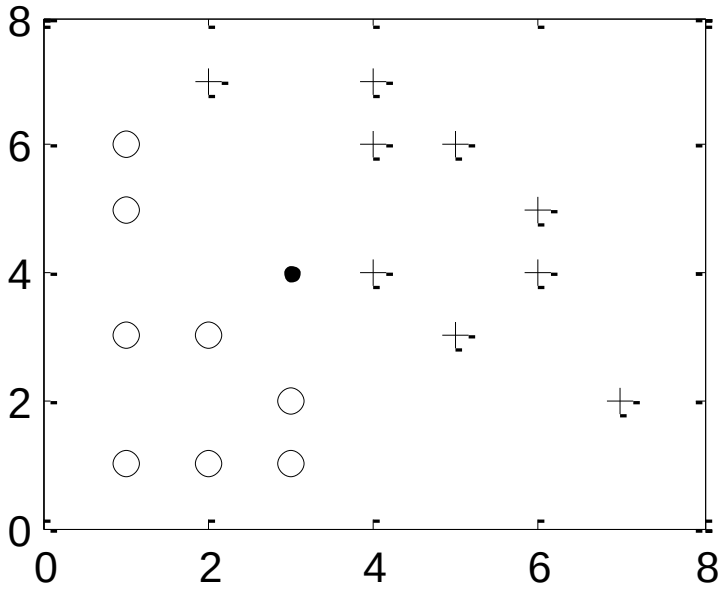
$$d(x_i, x_j) = \left(\sum_{l=1}^{N} \left| x_{il} - x_{jl} \right|^p \right)^{\frac{1}{p}}$$

- Hitung nilai keanggotaan u(x, c_i) menggunakan persamaan dibawah ini untuk setiap i, dimana $1 \leq i \leq C$.

$$u(x, c_i) = \frac{\sum_{k=1}^{K} u(x_k, c_i) * d(x, x_k)^{\frac{-2}{(m-1)}}}{\sum_{k=1}^{K} d(x, x_k)^{\frac{-2}{(m-1)}}}$$

- Ambil nilai terbesar c=u(x, c_i) untuk semua $1 \leq i \leq C$.
- Berikan label kelas c ke data uji x.

Contoh



Data uji adalah data (3,4), fitur X=3,Y=4.

Akan dilakukan prediksi, masuk dalam kelas yang manakah seharusnya ?

Gunakan w=2, dan jarak Euclidean !

Data latih

Data	X	Y	Kelas
1	1	1	0
2	2	1	0
3	3	1	0
4	3	2	0
5	7	2	1
6	1	3	0
7	2	3	0
8	5	3	1
9	4	4	1
10	6	4	1
11	1	5	0
12	6	5	1
13	1	6	0
14	4	6	1
15	5	6	1
16	2	7	1
17	4	7	1

Prediksi dengan K-NN

Jarak data uji (3,4) ke 17 data latih

Nomor data	x	y	Kelas asli	Jarak data uji ke data latih	1-NN	3-NN	7-NN
1	1	1	0	3.6055	0	0	0
2	2	1	0	3.1622	0	0	0
3	3	1	0	3	0	0	0
4	3	2	0	2	0	1	1
5	7	2	1	4.4721	0	0	0
6	1	3	0	2.2360	0	0	1
7	2	3	0	1.4142	0	1	1
8	5	3	1	2.2360	0	0	1
9	4	4	1	1	1	1	1
10	6	4	1	3	0	0	0
11	1	5	0	2.2360	0	0	1
12	6	5	1	3.1622	0	0	0
13	1	6	0	2.8284	0	0	0
14	4	6	1	2.2360	0	0	1
15	5	6	1	2.8284	0	0	0
16	2	7	1	3.1622	0	0	0
17	4	7	1	3.1622	0	0	0

Untuk K=1
Data uji (3,4) diprediksi masuk kelas 1.

Untuk K=3
Data uji (3,4) diprediksi masuk kelas 1.

Untuk K=7
Data uji (3,4) diprediksi masuk kelas 1.

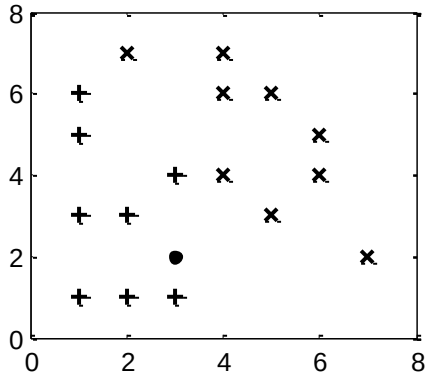
Nomor data	x	y	Kelas asli	Jarak data uji ke data latih	1-NN	$\frac{2}{d^{m-1}}$	3-NN	$\frac{2}{d^{m-1}}$	7-NN	$\frac{2}{d^{m-1}}$
1	1	1	0	3.6055	0		0		0	
2	2	1	0	3.1622	0		0		0	
3	3	1	0	3	0		0		0	
4	3	2	0	2	0		1	0.25	1	0.2500
5	7	2	1	4.4721	0		0		0	
6	1	3	0	2.236	0		0		1	0.2000
7	2	3	0	1.4142	0		1	0.5	1	0.5000
8	5	3	1	2.236	0		0		1	0.2000
9	4	4	1	1	1	1	1	1	1	1.0000
10	6	4	1	3	0		0		0	
11	1	5	0	2.236	0		0		1	0.2000
12	6	5	1	3.1622	0		0		0	
13	1	6	0	2.8284	0		0		0	
14	4	6	1	2.236	0		0		1	0.2000
15	5	6	1	2.8284	0		0		0	
16	2	7	1	3.1622	0		0		0	
17	4	7	1	3.1622	0		0		0	
Jumlah kelas 0						0		0.8		1.1500
Jumlah kelas 1						1		1.00		1.4000
Jumlah						1		1.75		2.5501
Nilai keanggotaan di kelas 0						0		0.4286		0.4510
Nilai keanggotaan di kelas 1						1		0.5714		0.5490

FUZZY K-NEAREST NEIGHBOR IN EVERY CLASS (FK-NNC)

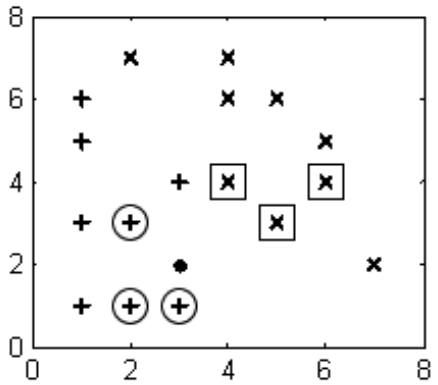
Framework FK-NNC

- Diperkenalkan oleh Prasetyo (2012).
- FK-NNC menggunakan sejumlah K tetangga terdekat pada setiap kelas dari sebuah data uji, bukan K tetangga terdekat seperti pada K-NN dan FK-NN.
- FK-NNC menggunakan FK-NN sebagai basis kerangka kerja, dimana sebuah data uji mempunyai nilai keanggotaan pada setiap kelas dalam interval [0.1].
  - Jumlah nilai keanggotaan sebuah data pada semua kelas sama dengan 1

$$\sum_{j=1}^{C} u_{ij} = 1, \quad 0 \leq u_{ij} \leq 1$$



Tanda dot hitam (solid) adalah data uji



Tiga tetangga dikelas + dan tiga tetangga dikelas x

Framework FK-NNC – *Cont'd*

- Jarak data uji x_i ke semua K tetangga dari setiap kelas ke-*j* dijumlahkan, formula yang digunakan:

$$S_{ij} = \sum_{r=1}^{K} d(x_i, x_r) \qquad (4)$$

- akumulasi jarak data uji x_i ke setiap kelas digabungkan, disimbolkan *D*, formula yang digunakan:

$$D_i = \sum_{j=1}^{C} (S_{ij})^{\frac{-2}{m-1}} \qquad (5)$$

  - Nilai *m* disini merupakan pangkat bobot (*weight exponent*) seperti pada FK-NN, nilai $m > 1$.

- Untuk mendapatkan nilai keanggotaan data uji x_i pada setiap kelas ke-*j* (ada *C* kelas), menggunakan formula:

$$u_{ij} = \frac{S_{ij}}{D_i} \qquad (6)$$

- Untuk menentukan kelas hasil prediksi data uji x_i, dipilih kelas dengan nilai keanggotaan terbesar dari data x_i. Formula yang digunakan:

$$y' = \arg\max_{j=1}^{C} (u_{ij}) \qquad (7)$$

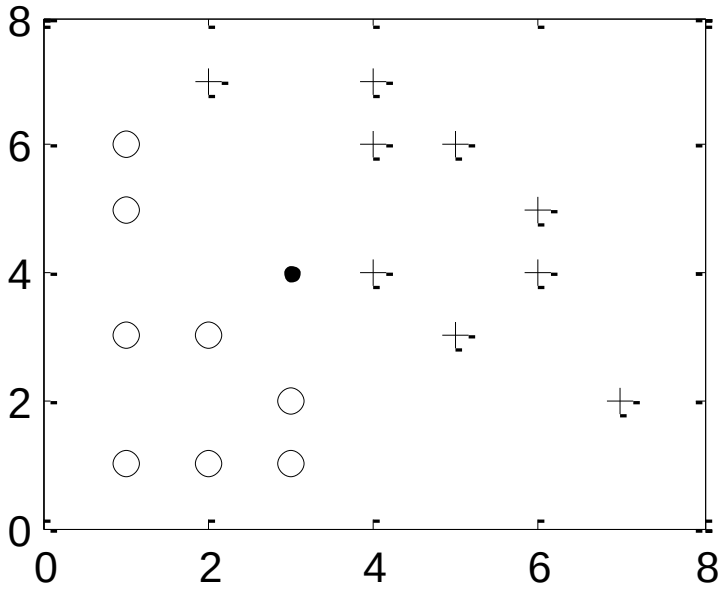
Algoritma FK-NNC

- Cari K tetangga terdekat pada setiap kelas, menggunakan formula

$$d(x_i, x_j) = \left(\sum_{l=1}^{N} \left| x_{il} - x_{jl} \right|^p \right)^{\frac{1}{p}}$$

- Hitung S sebagai akumulasi jarak K tetangga pada setiap kelas, menggunakan formula (4)
- Hitung J sebagai akumulasi semua jarak dari C×K tetangga, menggunakan formula (5)
- Hitung u sebagai nilai keanggotaan data pada setiap kelas, menggunakan formula (6)
- Pilih nilai keanggotaan terbesar menggunakan formula (7), kelas dengan nilai keanggotaan terbesar menjadi kelas hasil prediksi untuk data uji tersebut.

Contoh



Data uji adalah data (3,4), fitur X=3,Y=4.

Akan dilakukan prediksi, masuk dalam kelas yang manakah seharusnya ?

Gunakan w=2, dan jarak Euclidean !

Data latih

Data	X	Y	Kelas
1	1	1	0
2	2	1	0
3	3	1	0
4	3	2	0
5	7	2	1
6	1	3	0
7	2	3	0
8	5	3	1
9	4	4	1
10	6	4	1
11	1	5	0
12	6	5	1
13	1	6	0
14	4	6	1
15	5	6	1
16	2	7	1
17	4	7	1

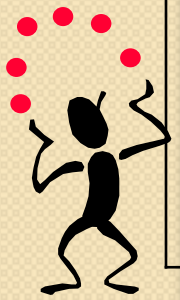
Prediksi dengan K-NN

Jarak data uji (3,4) ke 17 data latih

Nomor data	x	y	Kelas asli	Jarak data uji ke data latih
1	1	1	0	3.6055
2	2	1	0	3.1622
3	3	1	0	3
4	3	2	0	2
5	7	2	1	4.4721
6	1	3	0	2.2360
7	2	3	0	1.4142
8	5	3	1	2.2360
9	4	4	1	1
10	6	4	1	3
11	1	5	0	2.2360
12	6	5	1	3.1622
13	1	6	0	2.8284
14	4	6	1	2.2360
15	5	6	1	2.8284
16	2	7	1	3.1622
17	4	7	1	3.1622

Setelah diurutkan

Nomor data	x	y	Kelas asli	Jarak data uji ke data latih
7	2	3	0	1.414
4	3	2	0	2
6	1	3	0	2.236
11	1	5	0	2.236
13	1	6	0	2.828
3	3	1	0	3
2	2	1	0	3.162
1	1	1	0	3.606
9	4	4	1	1
8	5	3	1	2.236
14	4	6	1	2.236
15	5	6	1	2.828
10	6	4	1	3
12	6	5	1	3.162
16	2	7	1	3.162
17	4	7	1	3.162
5	7	2	1	4.472



Untuk K=1
Data uji (3,4) diprediksi masuk kelas 1.

Untuk K=3
Data uji (3,4) diprediksi masuk kelas 1.

Untuk K=3
Data uji (3,4) diprediksi masuk kelas 0.

Untuk K=7
Data uji (3,4) diprediksi masuk kelas 0.

Nomor data	x	y	Kelas asli	Jarak data uji ke data latih	K=1	K=3	K=5	K=7
7	2	3	0	1.414	1	1	1	1
4	3	2	0	2	0	1	1	1
6	1	3	0	2.236	0	1	1	1
11	1	5	0	2.236	0	0	1	1
13	1	6	0	2.828	0	0	1	1
3	3	1	0	3	0	0	0	1
2	2	1	0	3.162	0	0	0	1
1	1	1	0	3.606	0	0	0	0
9	4	4	1	1	1	1	1	1
8	5	3	1	2.236	0	1	1	1
14	4	6	1	2.236	0	1	1	1
15	5	6	1	2.828	0	0	1	1
10	6	4	1	3	0	0	1	1
12	6	5	1	3.162	0	0	0	1
16	2	7	1	3.162	0	0	0	1
17	4	7	1	3.162	0	0	0	0
5	7	2	1	4.472	0	0	0	0
Jumlah kelas 0					1.414	5.65	10.71	16.88
Jumlah kelas 1					1	5.472	11.3	17.62
S_0					0.5	0.031	0.009	0.004
S_1					1	0.033	0.008	0.003
Jumlah (D)					1.50	0.06	0.02	0.01
Nilai keanggotaan di kelas 0 (u_0)					0.333	0.484	0.527	0.522
Nilai keanggotaan di kelas 1 (u_1)					0.667	0.516	0.473	0.478

ANY QUESTIONS ?