Classify the following data preprocessing techniques

Techniques	Data cleaning	Data integration	Data transformation	Data reduction	Discretization
Binning					
Regression					
Clustering					
Correlation analysis					
Decision tree					
Normalization					
Feature selection					
Histograms					
Sampling					
Concept hierarchy generation					

Classify the following data preprocessing techniques

Techniques	Data cleaning	Data integration	Data transformation	Data reduction	Discretization
Binning	1		1	1	1
Regression	1			1	
Clustering	1			1	1
Correlation analysis		1			
Decision tree	1			1	
Normalization			1		
Feature selection				1	
Histograms				1	1
Sampling				1	
Concept hierarchy generation			1		1

Analisis Cluster

Kuliah 4

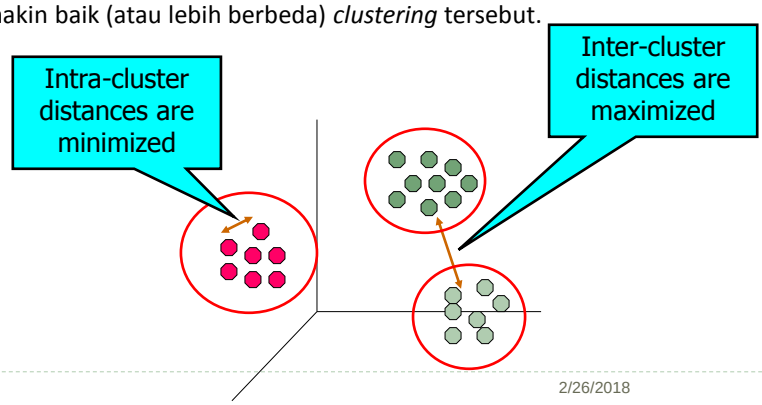
2/26/2018

Pendahuluan

- *Clustering* adalah salah satu teknik *unsupervised learning* di mana kita tidak perlu melatih metoda tersebut atau dengan kata lain, tidak ada fase pembelajaran (l*earning*).
- Analisis *cluster* membagi data ke dalam grup (*cluster*) yang bermakna, berguna, atau keduanya.
- Analisis *cluster* akan mengelompokkan obyek-obyek data hanya berdasarkan pada informasi yang terdapat pada data, yang menjelaskan obyek dan relasinya.

Analisis Cluster

- Tujuan analisis *cluster* adalah agar obyek-obyek di dalam grup adalah mirip (atau berhubungan) satu dengan lainnya, dan berbeda (atau tidak berhubungan) dengan obyek dalam grup lainnya.
- Semakin besar tingkat kemiripan/*similarity* (atau homogenitas) di dalam satu grup dan semakin besar tingkat perbedaan di antara grup, maka semakin baik (atau lebih berbeda) *clustering* tersebut.

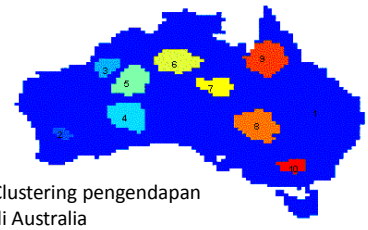


Aplikasi analisis cluster

- **Understanding**

  (Memahami pola atau relasi antar objek data)
  - Pencarian kelompok dokumen yang saling berkaitan, kelompok gen dan protein yang memiliki kemiripan fungsional, atau mengelompokkan stok barang yang memiliki kemiripan fluktuasi harga.

	Discovered Clusters	*Industry Group*
1	Applied-Matl-DOWN,Bay-Network-Down,3-COM-DOWN, Cabletron-Sys-DOWN,CISCO-DOWN,HP-DOWN, DSC-Comm-DOWN,INTEL-DOWN,LSI-Logic-DOWN, Micron-Tech-DOWN,Texas-Inst-Down,Tellabs-Inc-Down, Natl-Semiconduct-DOWN,Oracl-DOWN,SGI-DOWN, Sun-DOWN	Technology1-DOWN
2	Apple-Comp-DOWN,Autodesk-DOWN,DEC-DOWN, ADV-Micro-Device-DOWN,Andrew-Corp-DOWN, Computer-Assoc-DOWN,Circuit-City-DOWN, Compaq-DOWN, EMC-Corp-DOWN, Gen-Inst-DOWN, Motorola-DOWN,Microsoft-DOWN,Scientific-Atl-DOWN	Technology2-DOWN
3	Fannie-Mae-DOWN,Fed-Home-Loan-DOWN, MBNA-Corp-DOWN,Morgan-Stanley-DOWN	Financial-DOWN
4	Baker-Hughes-UP,Dresser-Inds-UP,Halliburton-HLD-UP, Louisiana-Land-UP,Phillips-Petro-UP,Unocal-UP, Schlumberger-UP	Oil-UP



Clustering pengendapan di Australia

- **Summarization (Peringkasan)**
  - Mengurangi ukuran dataset yang besar

Penentuan *cluster* terbaik tergantung dari kondisi data serta hasil yang diinginkan

Berapa banyak kluster ?

enam kluster

dua kluster

Empat kluster

Major Clustering Approaches (I)

- Partitioning approach:
  - Membangun beragam partisi dan kemudian mengevaluasinya melalui beberapa kriteria, contoh: meminimumkan nilai sum of square errors (SSE)
  - Metode yang umum: k-means, k-medoids, CLARANS
- Hierarchical approach:
  - Membuat dekomposisi hirari dari kumpulan data berdasarkan beberapa kriteria
  - Metode yang umum: Diana, Agnes, BIRCH, CAMELEON
- Density-based approach:
  - Berdasarkan pada konektivitas dan fungsi kepadatan
  - Metode yang umum: DBSACN, OPTICS, DenClue
- Grid-based approach:
  - Berdasarkan pada struktur granularitas multi level
  - Metode yang umum: STING, WaveCluster, CLIQUE

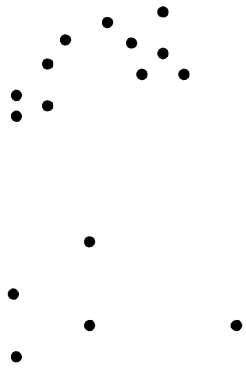
Major Clustering Approaches (II)

- Model-based:
  - Sebuah model dihipotesiskan untuk setiap kelompok dan mencoba mencari model terbaik satu sama lain
  - Metode yang umum: EM, SOM, COBWEB
- Frequent pattern-based:
  - Berdasarkan pada analisis *frequent patterns* (pola yang sering muncul)
  - Metode yang umum: p-Cluster
- User-guided or constraint-based:
  - Clustering dengan mempertimbangkan kendala yang diberikan oleh user atau aplikasi
  - Metode yang umum: COD (obstacles), constrained clustering
- Link-based clustering:
  - Objek seringkali dihubungkan melalui berbagai cara
  - Massive links can be used to cluster objects: SimRank, LinkClus

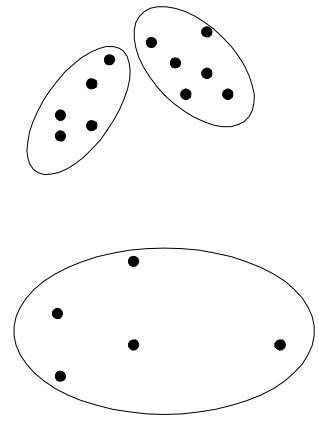
Tipe-tipe clustering

- Hierarchical versus Partitional
  - *Partitional clustering* adalah membagi himpunan obyek data ke dalam sub-himpunan (*cluster*) yang tidak overlap, sehingga setiap obyek data berada dalam tepat satu *cluster.*
  - *Hierarchical clustering*
    - Cluster memiliki subcluster
    - Himpunan cluster bersarang yang diatur dalam bentuk tree.

Partitional Clustering

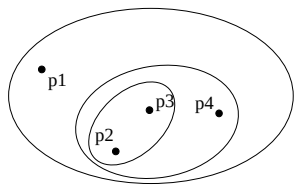


Titik-titik asli
(sebelum dilakukan clustering

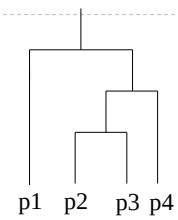


Setalah dilakukan
Partitional Clustering

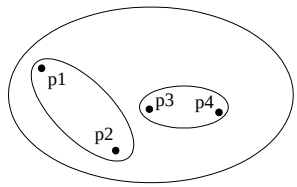
Hierarchical Clustering



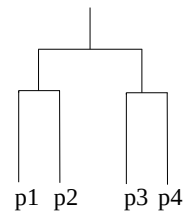
Traditional Hierarchical Clustering



Traditional Dendrogram



Non-traditional Hierarchical Clustering



Non-traditional Dendrogram

Tipe clustering lainnya

- Exclusive versus non-exclusive
  - Dalam non-exclusive clustering, titik-titik dapat menjadi anggota banyak clusters.
  - Dapat merepresentasikan banyak kelas atau titik 'border'
- Fuzzy versus non-fuzzy
  - setiap obyek menjadi milik setiap *cluster* dengan nilai keanggotaan di antara 0 (mutlak bukan anggota *cluster*) dan 1 (mutlak anggota *cluster*)
  - Penjumlahan bobot haruslah 1
- Partial versus complete
  - *Complete clustering* akan menetapkan setiap obyek ke dalam *cluster*, sedangkan *partial clustering* tidak.
  - Alasan *partial clustering* adalah karena beberapa obyek dalam *data set* mungkin bukan anggota kelompok yang telah didefinisikan dengan baik.
  - Banyak obyek dalam *data set* mungkin mewakili *noise* atau *outlier.*

Struktur Data

- Matriks data – merepresentasikan struktur objek versus variabel (baris → objek, kolom → variabel)
  - (two modes)
  - N objects, p variables

$$\begin{bmatrix} x_{11} & \cdots & x_{1f} & \cdots & x_{1p} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{i1} & \cdots & x_{if} & \cdots & x_{ip} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{n1} & \cdots & x_{nf} & \cdots & x_{np} \end{bmatrix}$$

- Matriks disimilaritas
  - mereprsentasikan struktur objek versus objek
  - (one mode)

$$\begin{bmatrix} 0 & & & & \\ d(2,1) & 0 & & & \\ d(3,1) & d(3,2) & 0 & & \\ : & : & : & & \\ d(n,1) & d(n,2) & \cdots & \cdots & 0 \end{bmatrix}$$

Type of data in clustering analysis

- Variabel Interval
- Variabel Biner
- Variabel Nominal, Ordinal, dan Rasio
- Kumpulan berbagai tipe variabel

Variable Interval-valued

- Standardize data (Standarisasi data)
  - Hitung rata-rata simpangan absolut (the mean absolute deviation):

$$s_f = \frac{1}{n}(|x_{1f} - m_f| + |x_{2f} - m_f| + ... + |x_{nf} - m_f|)$$

di mana

$$m_f = \frac{1}{n}(x_{1f} + x_{2f} + ... + x_{nf})$$

  - Hitung ukuran standarisasi (the standardized measurement - *z-score*)

$$z_{if} = \frac{x_{if} - m_f}{s_f}$$

- Penggunaan simpangan absolut, s_f, lebih *robust* dibandingkan menggunakan simpangan baku (standard deviation)

Similarity and Dissimilarity Antar Objek

- Ukuran jarak (*distance*) biasanya digunakan untuk mengukur ukuran kemiripan (*similarity*) atau ketidakmiripan (*dissimilarity*) antara 2 buah objek
- Metode popular lain: *Minkowski distance*:

$$d(i,j) = \sqrt[q]{(|x_{i_1} - x_{j_1}|^q + |x_{i_2} - x_{j_2}|^q + ... + |x_{i_p} - x_{j_p}|^q)}$$

di mana $i = (x_{i1}, x_{i2}, ..., x_{ip})$ dan $j = (x_{j1}, x_{j2}, ..., x_{jp})$ merupakan dua objek berdimensi p, dan q merupakan nilai integer positif

- Jika $q = 1$, akan menjadi jarak Manhattan

$$d(i,j) = |x_{i_1} - x_{j_1}| + |x_{i_2} - x_{j_2}| + ... + |x_{i_p} - x_{j_p}|$$

Similarity and Dissimilarity Antar Objek (Cont.)

- Jika $q = 2$, d merupakan jarak Euclidean:

$$d(i,j) = \sqrt{(|x_{i_1} - x_{j_1}|^2 + |x_{i_2} - x_{j_2}|^2 + ... + |x_{i_p} - x_{j_p}|^2)}$$

  - Sifat
    - $d(i,j) \geq 0$
    - $d(i,i) = 0$
    - $d(i,j) = d(j,i)$
    - $d(i,j) \leq d(i,k) + d(k,j)$

- Ukuran lain yang dapat digunakan adalah weighted distance (jarak terboboti), parametric Pearson product moment correlation, atau ukuran disimilarity lain

Variabel Biner

- Tabel kontingensi untuk data biner

		Object *j*		
		1	0	*sum*
Object *i*	1	*a*	*b*	*a+b*
	0	*c*	*d*	*c+d*
	sum	*a+c*	*b+d*	*p*

- Ukuran jarak untuk variabel simetrik biner:

$$d(i,j) = \frac{b+c}{a+b+c+d}$$

- Ukuran jarak untuk variabel asimetrik biner:

$$d(i,j) = \frac{b+c}{a+b+c}$$

- Jaccard coefficient (ukuran *similarity* untuk variabel asimetrik biner):

$$sim_{Jaccard}(i,j) = \frac{a}{a+b+c}$$

Dissimilarity Antara Variabel Biner

- Contoh

Name	Gender	Fever	Cough	Test-1	Test-2	Test-3	Test-4
Jack	M	Y	N	P	N	N	N
Mary	F	Y	N	P	N	P	N
Jim	M	Y	P	N	N	N	N

  - Gender merupakan atribut simetrik
  - Atribut yang lain merupakan asimetrik biner
  - Ubah nilai Y and P menjadi 1, dan nilai N menjadi 0

$$d(jack, mary) = \frac{0+1}{2+0+1} = 0.33$$

$$d(jack, jim) = \frac{1+1}{1+1+1} = 0.67$$

$$d(jim, mary) = \frac{1+2}{1+1+2} = 0.75$$

Nominal Variables

- Generalisasi dari variabel biner yaitu dapat mengambil lebih dari 2 nilai, contoh, red, yellow, blue, green
- Metode 1: Simple matching (pencocokan sederhana)
  - *m*: # of matches, *p*: total # of variables

$$d(i, j) = \frac{p - m}{p}$$

- Metode 2: Gunakan sejumlah besar variabel biner
  - Membuat variabel biner baru untuk setiap nilai nominal *M*

Ordinal Variables

- Variabel Ordinal dapat berupa nilai diskret atau kontinu
- Urutan merupakan aspek penting, contoh: rank
- Dapat diperlakukan layaknya variabel interval
  - Ubah x_{if} berdasarkan tingkatannya $r_{if} \in \{1, \ldots, M_f\}$
  - Petakan range dari setiap variabel dalam rentang [0, 1] dengan mengganti objek ke-*i* pada variabel ke-*f* menggunakan

$$z_{if} = \frac{r_{if} - 1}{M_f - 1}$$

  - Hitung nilai dissimilarity menggunakan metode yang biasa digunakan untuk variabel interval-scaled

Ratio-Scaled Variables

- Ratio-scaled variable: ukuran positif pada skala non-linier, sekitar skala eksponensial, contoh Ae^{Bt} atau Ae^{-Bt}
- Methods:
  - Perlakukan variabel tersebut seperti variabel interval-scaled—*not a good choice!* (why?—the scale can be distorted)
  - Gunakan transformasi logaritmik

$$y_{if} = log(x_{if})$$

  - Perlakukan variabel tersebut sebagai data ordinal kontinu, perlakukan ranknya sebagai variabel interval-scaled

Variables of Mixed Types

- Sebuah database variabelnya mungkin mengandung semua tipe data
  - symmetric binary, asymmetric binary, nominal, ordinal, interval and ratio
- One may use a weighted formula to combine their effects

$$d(i,j) = \frac{\sum_{f=1}^{p} \delta_{ij}^{(f)} d_{ij}^{(f)}}{\sum_{f=1}^{p} \delta_{ij}^{(f)}}$$

  - f merupakan biner atau nominal:
    $d_{ij}^{(f)} = 0$ if $x_{if} = x_{jf}$, or $d_{ij}^{(f)} = 1$ selainnya
  - f merupakan interval-based: gunakan normalized distance
  - f is ordinal or ratio-scaled
    - hitung ranks r_{if} $\quad z_{if} = \frac{r_{if} - 1}{M_f - 1}$
    - dan perlakukan z_{if} sebagai interval-scaled

Vector Objects

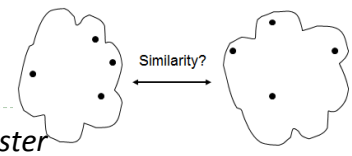
- Objek vektor (Vector objects): kata kunci pada documents, fitur gen dalam micro-arrays, dll.
- Aplikasi yang lebih luas: temu kembali informasi, taksonomi biologi, dll.
- Ukuran Cosine $$s(\vec{X},\vec{Y}) = \frac{\vec{X}^t \cdot \vec{Y}}{|\vec{X}||\vec{Y}|},$$

$\vec{X}^t$ is a transposition of vector $\vec{X}$, $|\vec{X}|$ is the Euclidean normal of vector $\vec{X}$,

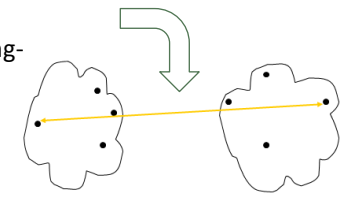
- Variannya: koefisien Tanimoto

$$s(\vec{X},\vec{Y}) = \frac{\vec{X}^t \cdot \vec{Y}}{\vec{X}^t \cdot \vec{X} + \vec{Y}^t \cdot \vec{Y} - \vec{X}^t \cdot \vec{Y}},$$

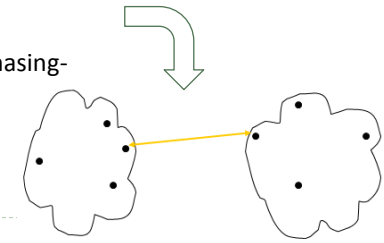
Ukuran Jarak Antar *Cluster*



- Jarak maksimum antar elemen dalam *cluster* (*complete linkage clustering*):
  - d(A,B) = max {$S_{x,y}$}, x ∈ A, y ∈ A
  - di mana $S_{x,y}$ adalah jarak dua data x dan y masing-masing dari *cluster* A dan B.

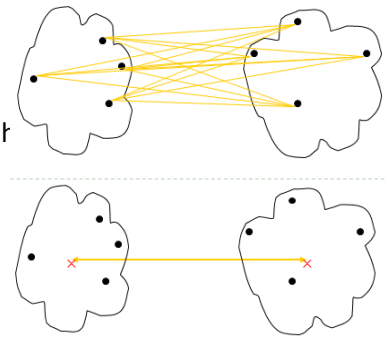


- Jarak minimum antara elemen dari setiap *cluster* (*single linkage clustering*)
  - d(A,B) = min{$S_{x,y}$}, x ∈ A, y ∈ A
  - di mana $S_{x,y}$ adalah jarak dua data x dan y masing-masing dari *cluster* A dan B.



Ukuran Jarak Antar *Cluster*

- Rata-rata jarak antara elemen dari setiap *cluster* (*average linkage clustering*)
  - d(A, B) = $\frac{1}{n_A n_B}\sum_{x \in A}\sum_{y \in B} s(x, y)$
  - di mana n_A dan n_B masing-masing adalah banyaknya data dalam *cluster* A dan B.
- *Centroid Linkage*
  - d(A, B) = $s(\bar{x}, \bar{y})$
- *Ward Linkage*
  - d(A, B) = $\frac{n_A n_B s^2_{AB}}{n_A + n_B}$
  - s^2_{AB} adalah jarak antara *cluster* A dan B dengan menggunakan formula *centroid linkage*.



K-means Clustering

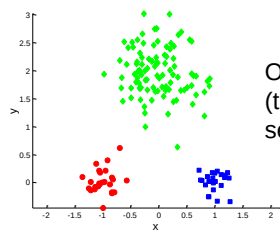
- Partitional clustering approach
- Setiap cluster terasosiasi dengan sebuah centroid (titik tengah)
- Setiap titik dikumpulkan pada cluster dengan centroid terdekat
- Jumlah cluster, K, harus ditentukan
- Algoritme dasarnya sangat sederhana

1: Select K points as the initial centroids.
2: **repeat**
3: Form K clusters by assigning all points to the closest centroid.
4: Recompute the centroid of each cluster.
5: **until** The centroids don't change

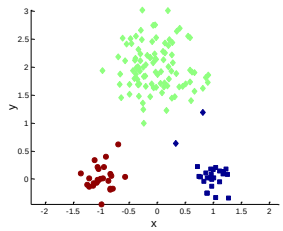
K-means Clustering – Details

- Centroid awal seringkali ditentukan secara random.
  - Hal ini mengakibatkan kluster yang dihasilkan berbeda-beda setiap program dieksekusi
- centroid umumnya dipilih dari nilai rata-rata (mean) dari titik-titik dalam suatu kluster
- 'Closeness' (kedekatan) diukur dengan menggunakan Euclidean distance, cosine similarity, correlation, etc.
- K-means akan konvergen dengan menggunakan ukuran similaritasi yang umum yang disebutkan di atas
- Sebagian besar konvergensi terjadi pada beberapa iterasi saja
  - Sering kali kondisi berhenti dari iterasi berubah menjadi 'Until relatively few points change clusters'
- Complexity is O(n * K * I * d)
  - n = jumlah titik sampel, K = jumlah kluster, I = jumlah iterasi, d = jumlah atribut

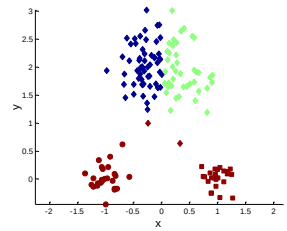
Perbedaan K-means Clusterings



Original Points
(titik-titik awal sebelum klustering)

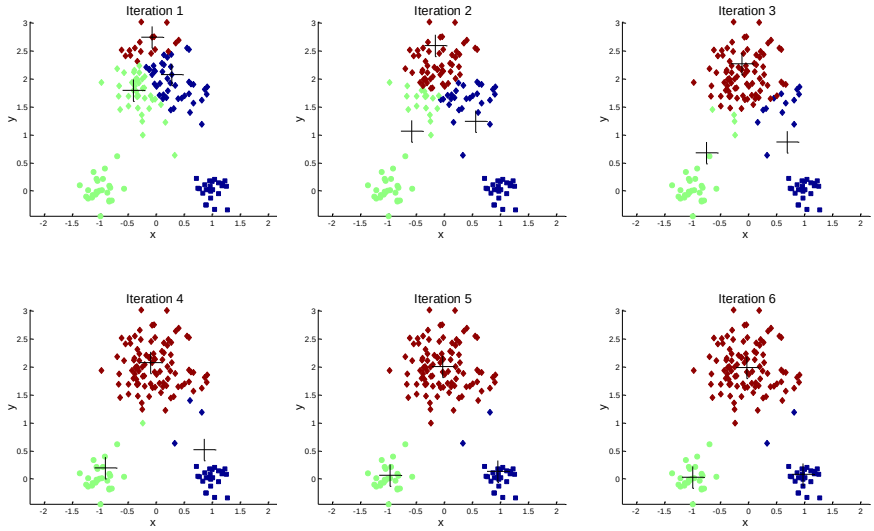


Optimal Clustering



Sub-optimal Clustering

Pentingnya Memilih Centroid Awal



Evaluasi pada K-means Clusters

- Pada umumnya ukuran yang digunakan untuk mengevaluasi adalah Sum of Squared Error (SSE)
  - Untuk setiap titik, error adalah jarak ke cluster terdekat
  - Untuk mendapat SSE, kita kuadratkan dan jumlahkan.

$$SSE = \sum_{i=1}^{K} \sum_{x \in C_i} dist^2(m_i, x)$$

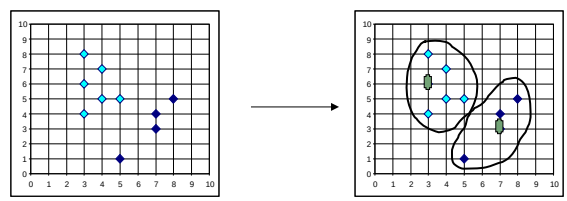
  - x merupakan titik data dalam clusterC_i dan m_i merupakan titik representatif untuk cluster C_i
    - dapat menunjukkan bahwa m_i berkorespondensi pada titik tengah (rataan/mean) dari cluster
  - Diberikan 2 cluster, kita dapat memilih satu yang memiliki error minimal.
  - Satu cara mudah untuk mengurangi nilai SSE yaitu dengan meningkatkan nilai K, jumlah cluster
    - Sebuah clustering yang baik dengan nilai K yang lebih kecil dapat memiliki SSE yang lebih kecil dibanding cluster yang buruk dengan nilai K yang lebih tinggi

Variasi dari metode *K-Means*

- Sebagian kecil dari varian *k-means* berbeda dalam hal:
  - Pemilihan initial *k* means
  - Perhitungan disimilaritas
  - Strategi untuk menghitung nilai rata-rata kluaster (cluster means)
- Penanganan data yang bersifat kategorikal *k-modes* (Huang'98)
  - Ganti nilai rata-rata (means) dari kluster dengan modus (nilai yang sering muncul
  - Gunakan ukuran disimilaritas baru yang sesuai dengan objek data kategorikal
  - Gunakan metode berbasis frekuensi (frequency-based method) untuk memperbarui nilai modus dari kluster
  - Untuk data yang mengandung data kategorical dan numerik: *k-prototype* method

Apa Masalah pada Metode k-Means?

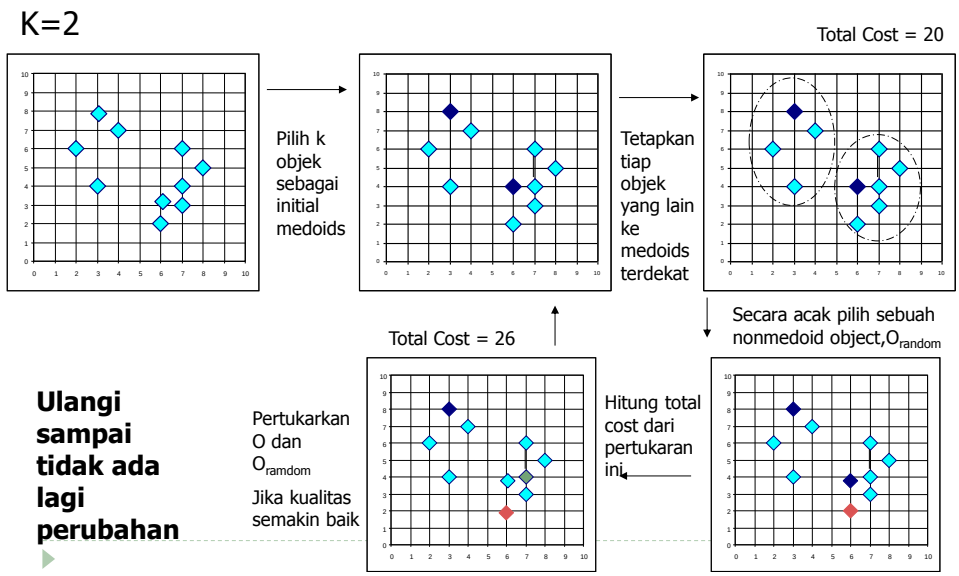
- Algoritme k-means sensitif terhadap outliers !
  - Karena sebuah objek yang memiliki nilai yang sangat berbeda secara substansial dapat mendistorsi distribusi data.
- K-Medoids: Dibandingkan menggunakan nilai **mean** dari objek pada suatu cluster sebagai titik referensi, **medoids** dapat digunakan, yang merupakan objek yang terletak paling tengah di cluster.



The *K-Medoids* Clustering Method

- Cari objek yang representatif, disebut medoids, di dalam cluster
- *PAM* (Partitioning Around Medoids, 1987)
  - Dimulai dari suatu himpunan awal medoids dan selanjutnya secara iteratif mengganti satu medoids dengan satu objek yang bukan medoid jika dengan penggantian ini akan memperbaiki nilai total jarak dari clustering yang dihasilkan
  - *PAM* bekerja secara efektif untuk dataset kecil namun tidak dapat bekerja dengan baik untuk dataset yang besar
- *CLARA* (Kaufmann & Rousseeuw, 1990)
- *CLARANS* (Ng & Han, 1994): Penarikan sampel secara acak (Randomized sampling)
- Focusing + spatial data structure (Ester et al., 1995)

Typical k-medoids algorithm (PAM)

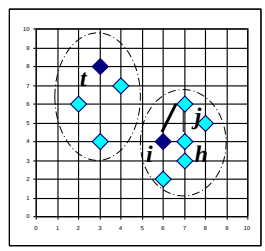


PAM (Partitioning Around Medoids) (1987)

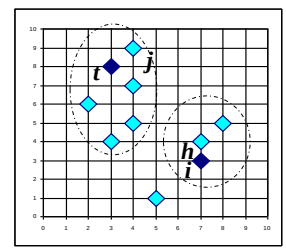
- PAM (Kaufman and Rousseeuw, 1987), dibangun menggunakan Splus (versi komersial bahasa pemrograman S)
- Gunakan objek (titik/sampel/objek data) yang real (dari dataset) untuk merepresentasikan pusat kluster (medoid): (catatan: ini berbeda dengan k-means yang memilih nilai acak (bukan dari data/sampel) sebagai pusat kluster.
  - Pilih ***k*** objek yang representatif secara acak
  - Untuk tiap pasangan objek yang tidak dipilih ***h*** dan objek yang dipilih ***i***, hitung total cost untuk melakukan pertukaran (swapping) TC_{ih}
  - Untuk tiap pasangan dari ***i*** dan ***h***,
    - If $TC_{ih} < 0$, ***i*** digantikan ***h***
    - Selanjutnya tetapkan tiap-tiap objek yang tidak dipilih ke objek representatif yang paling mirip
  - Ulangi langkah 2-3 sampai tidak ada lagi perubahan

PAM Clustering: Total swapping cost

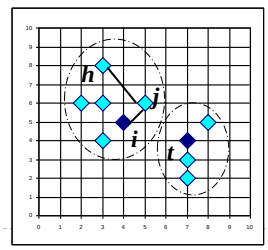
$TC_{ih}=\sum_j C_{jih}$



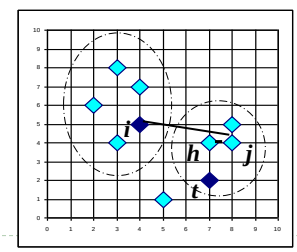
$C_{jih} = d(j, h) - d(j, i)$



$C_{jih} = 0$



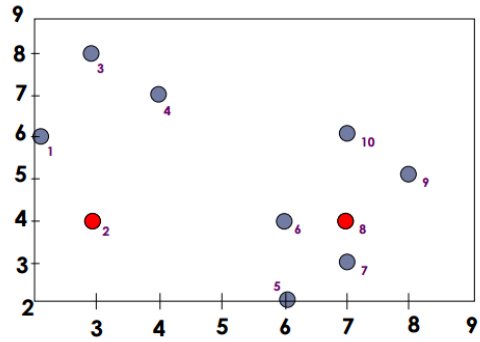
$C_{jih} = d(j, t) - d(j, i)$



$C_{jih} = d(j, h) - d(j, t)$

K-Medoids Example

	A1	A2
O1	2	6
O2	3	4
O3	3	8
O4	4	7
O5	6	2
O6	6	4
O7	7	3
O8	7	4
O9	8	5
O10	7	6



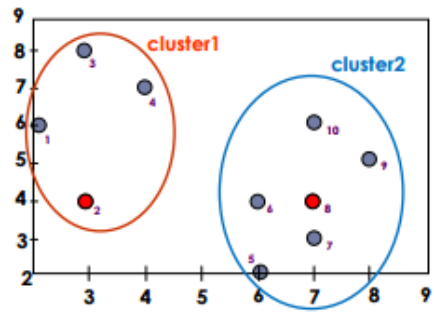
K = 2

Pilih secara acak 2 medoid
O2 = (3,4)
O8 = (7,4)

K-Medoids Example

K = 2

	A1	A2	C1 (O2)	C2 (O8)
O1	2	6	3	7
O2	3	4	0	4
O3	3	8	4	8
O4	4	7	4	6
O5	6	2	5	3
O6	6	4	3	1
O7	7	3	5	1
O8	7	4	4	0
O9	8	5	6	2
O10	7	6	6	2



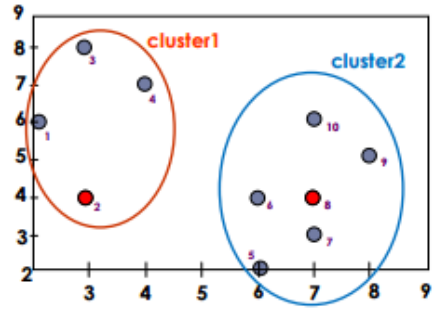
→ Tetapkan setiap objek pada objek representatif terdekat
→ Gunakan L1 Metric (Manhattan), maka terbentuklah:

Cluster1 = {O1, O2, O3, O4}
Cluster2 = {O5, O6, O7, O8, O9, O10}

K-Medoids Example

K = 2

	A1	A2	C1 (O2)	C2 (O8)
O1	2	6	3	7
O2	3	4	0	4
O3	3	8	4	8
O4	4	7	4	6
O5	6	2	5	3
O6	6	4	3	1
O7	7	3	5	1
O8	7	4	4	0
O9	8	5	6	2
O10	7	6	6	2



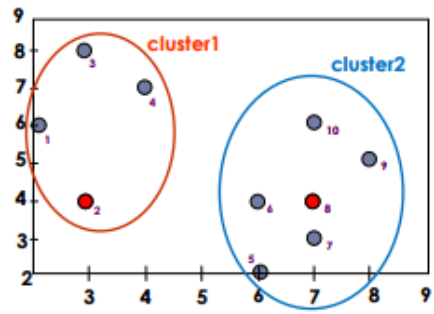
→ Hitung kriteria absolute error
[untuk himpunan **Medoids (O2,O8)**]

$$E=\sum_{i=1}^{k}\sum_{p\in C_i}|p-o_i|=|o_1-o_2|+|o_3-o_2|+|o_4-o_2|$$
$$+|o_5-o_8|+|o_6-o_8|+|o_7-o_8|+|o_9-o_8|+|o_{10}-o_8|$$

K-Medoids Example

K = 2

	A1	A2	C1 (O2)	C2 (O8)
O1	2	6	3	7
O2	3	4	0	4
O3	3	8	4	8
O4	4	7	4	6
O5	6	2	5	3
O6	6	4	3	1
O7	7	3	5	1
O8	7	4	4	0
O9	8	5	6	2
O10	7	6	6	2



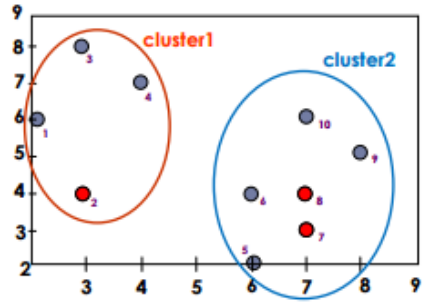
→ Kriteria absolute error
[**untuk himpunan Medoids (O2,O8)**]

$$E=(3+4+4)+(3+1+1+2+2)=20$$

K-Medoids Example

K = 2

	A1	A2	C1 (O2)	C2 (O7)
O1	2	6	3	8
O2	3	4	0	5
O3	3	8	4	9
O4	4	7	4	7
O5	6	2	5	2
O6	6	4	3	2
O7	7	3	5	0
O8	7	4	4	1
O9	8	5	6	3
O10	7	6	6	3



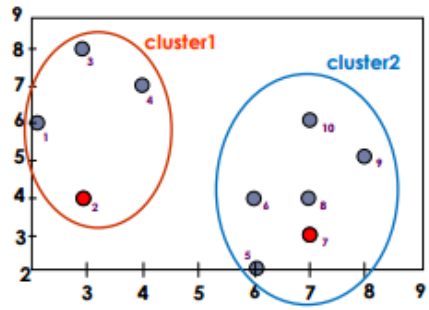
→ Pilih objek secara acak: O7
→ Tukar O8 and O7
→ Hitung kriteria absolute error
[**untuk himpunan Medoids (O2,O7)**]

$$E = (3+4+4)+(2+2+1+3+3) = 22$$

K-Medoids Example

K = 2

	A1	A2
O1	2	6
O2	3	4
O3	3	8
O4	4	7
O5	6	2
O6	6	4
O7	7	3
O8	7	4
O9	8	5
O10	7	6



→ Hitung nilai cost dari fungsi Swap O8 and O7

S = Absolute error [O2,O7] – Absolute error [O2,O8]

S = 22-20

S > 0 => Ÿbukan ide yang baik menggantikan O8 dengan O7

What is the problem with PAM?

- PAM lebih kokoh (robust) daripada k-means (dengan adanya noise dan outlier) karena medoid kurang dipengaruhi oleh outlier atau nilai ekstrim lainnya tidak seperti halnya mean.
- PAM bekerja secara efisien untuk himpunan data kecil namun tidak sesuai dengan himpunan data yang besar.
  - $O(k(n-k)^2)$ untuk tiap iterasi

    di mana n dalah jumlah data,k adalah jumlah clusters

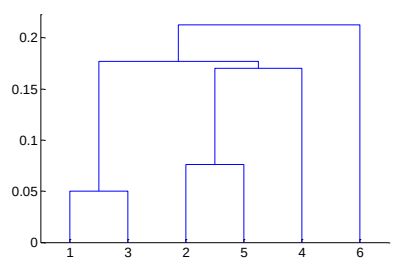
➔ Metode berbasis penarikan sampel (Sampling)

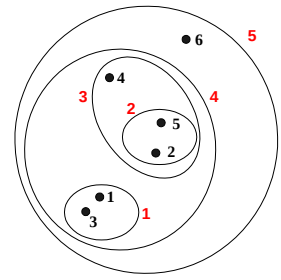
CLARA(Clustering LARge Applications)

Hierarchical Clustering

- Menghasilkan kumpulan cluster yang bersarang yang dapat direpresentasikan dalam hierarchical tree (pohon yang berhirarki)
- Dapat divisualisasikan sebagai dendrogram
  - Sebuah pohon seperti diagram yang mencatat urutan penggabungan atau pemisahan

Skala similaritas





Kelebihan Hierarchical Clustering

- Tidak perlu mengasumsikan jumlah cluster
  - Setiap jumlah cluster yang diinginkan dapat diperoleh dengan 'memotong' dendogram pada tingkat yang tepat
- Dapat menghasilkan taksonomi yang bermakna
  - Contoh pada ilmu biologi (e.g., kingdom hewan, perekonstruksian phylogeny, …)

Hierarchical Clustering

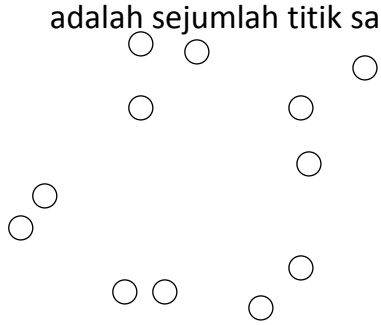
- Dua tipe utama dari hierarchical clustering
  - Agglomerative:
    - Mulai dari titik sebagai kluster individu (setiap individu dianggap sebagai satu kluster)
    - Untuk setiap langkah, gabungkan pasangan yang terdekat sampai hanya tersisa 1 cluster (or k clusters)
  - Divisive:
    - Mulai dengan 1 cluster, all-inclusive cluster (semua titik/sampel disatukan dalam satu kluster)
    - Untuk setiap tahapan, pisahkan suatu cluster sampai setiap cluster mengandung 1 titik (atau terdapat sebanyak k clusters)
- Traditional hierarchical algorithms menggunakan matriks similaritas atau matriks jarak
  - Gabungkan atau pisahkan satu kluster dalam satu waktu (bersamaan)

Agglomerative Clustering Algorithm

- Teknik hierarchical clustering yang lebih populer
- Algoritme dasarnya mudah
  1. Hitung matriks kedekatan (proximity matrix)
  2. Tetapkan setiap titik data sebagai sebuah kluster
  3. **Repeat**
  4. Gabungkan dua buah kluster terdekat
  5. Perbarui matriks kedekatan (proximity matrix)
  6. **Until** hanya satu kluster yang tersisa
- Operasi kuncinya adalah kedekatan dari 2 cluster
  - Perbedaan pendekatan untuk mendefinisikan jarak antar kluster menjadi pembeda dari tiap algoritme

Situasi Awal

- Mulai dengan cluster dari setiap titik individu dan matriks kedekatan (proximity matrix)
- Artinya jumlah kluster awal adalah sejumlah titik sampel

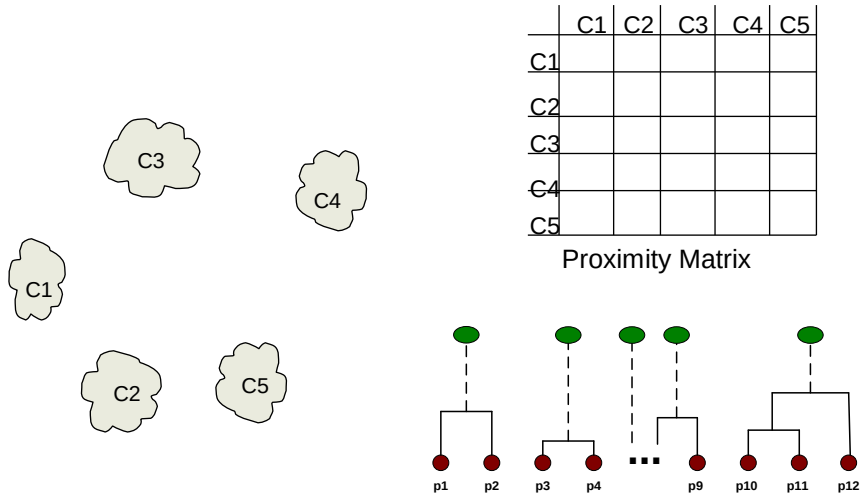


	p1	p2	p3	p4	p5	. . .
p1						
p2						
p3						
p4						
p5						
.						
.						
.						

Proximity Matrix

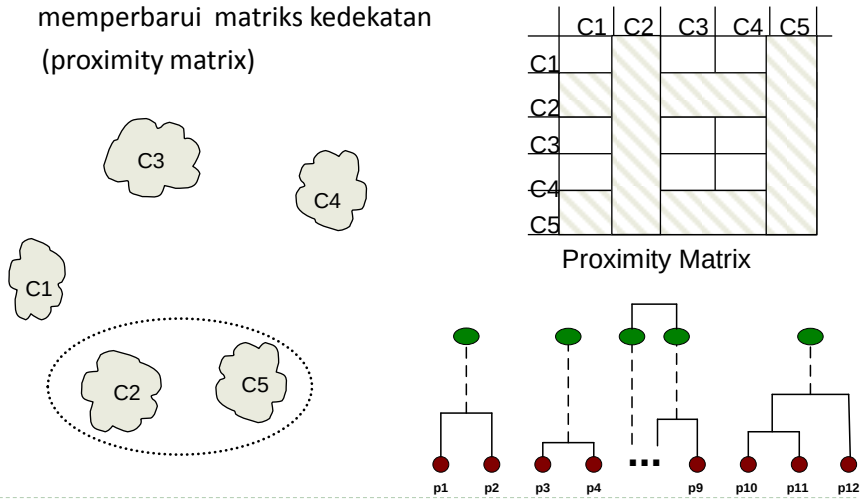
Situasi Pertengahan

- Setelah langkah penggabungan, kita memiliki beberapa cluster



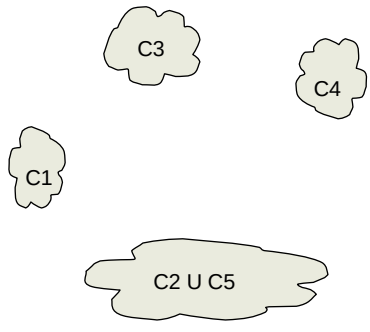
Situasi Pertengahan

- Kita ingin menggabungkan dua cluster yang terdekat dan memperbarui matriks kedekatan (proximity matrix)



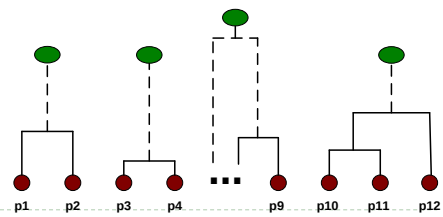
Setelah Penggabungan

- Pertanyaan: “Bagaimana cara memperbarui proximity matrix?”

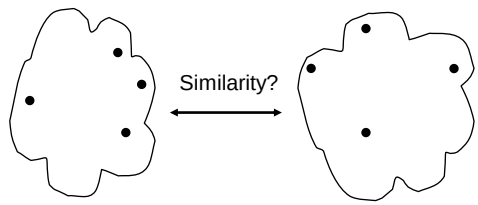


	C1	C2 U C5	C3	C4
C1		?		
C2 U C5	?	?	?	?
C3		?		
C4		?		

Proximity Matrix



Bagaimana mendefinisikan similaritas antar kluster?
→ menentukan bagaimana memperbarui proximity matrix

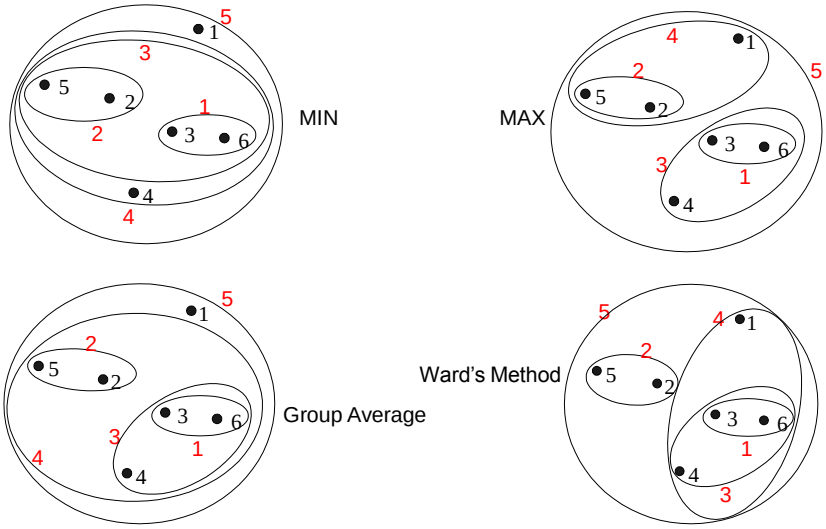


- MIN
- MAX
- Group Average
- Distance Between Centroids
- Other methods driven by an objective function
  - Ward’s Method uses squared error

	p1	p2	p3	p4	p5	. . .
p1						
p2						
p3						
p4						
p5						
.						
.						
.						

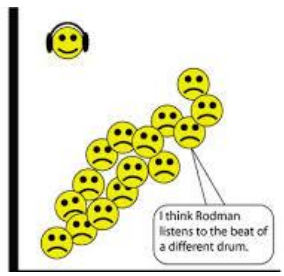
Proximity Matrix

Hierarchical Clustering: Perbandingan



Reference

- Tan P., Michael S., & Vipin K. 2006. Introduction to Data mining. Pearson Education, Inc.
- Han J & Kamber M. 2006. Data mining – Concept and Techniques. Morgan-Kauffman, San Diego



Topik selanjutnya: outlier detection