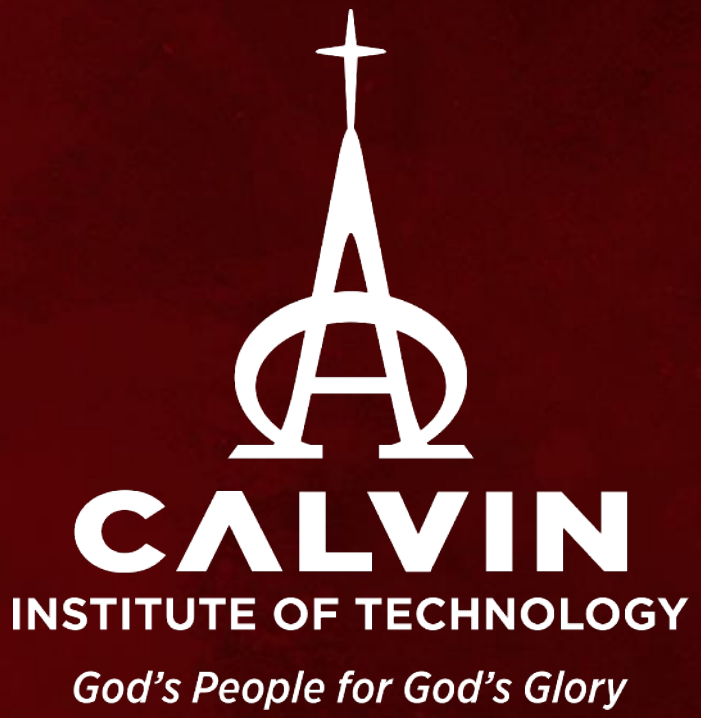


Clustering

Hendrik Santoso Sugiarto

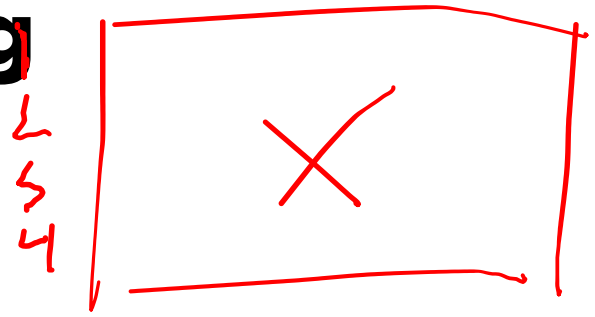
IBDA2032 – *Artificial Intelligence*

Capaian Pembelajaran

- Konsep Clustering
- Hard Clustering
- Soft Clustering

Clustering

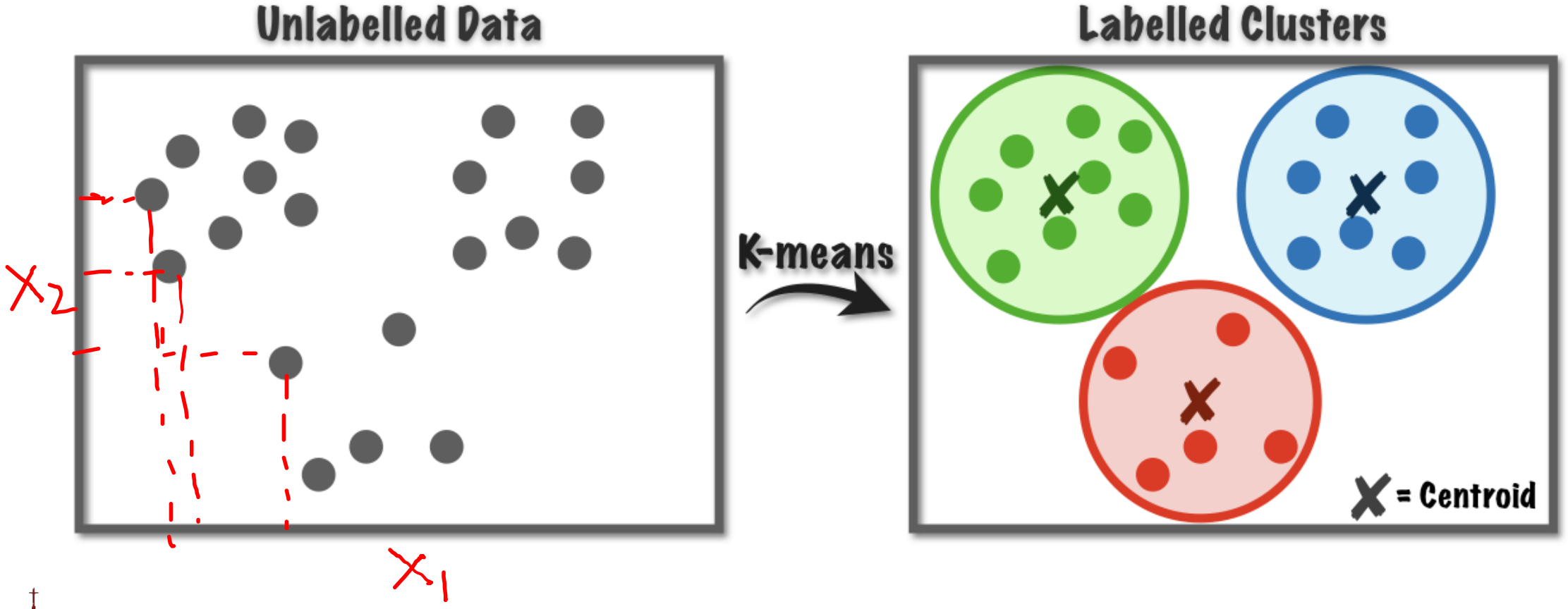
Unsupervised Learning



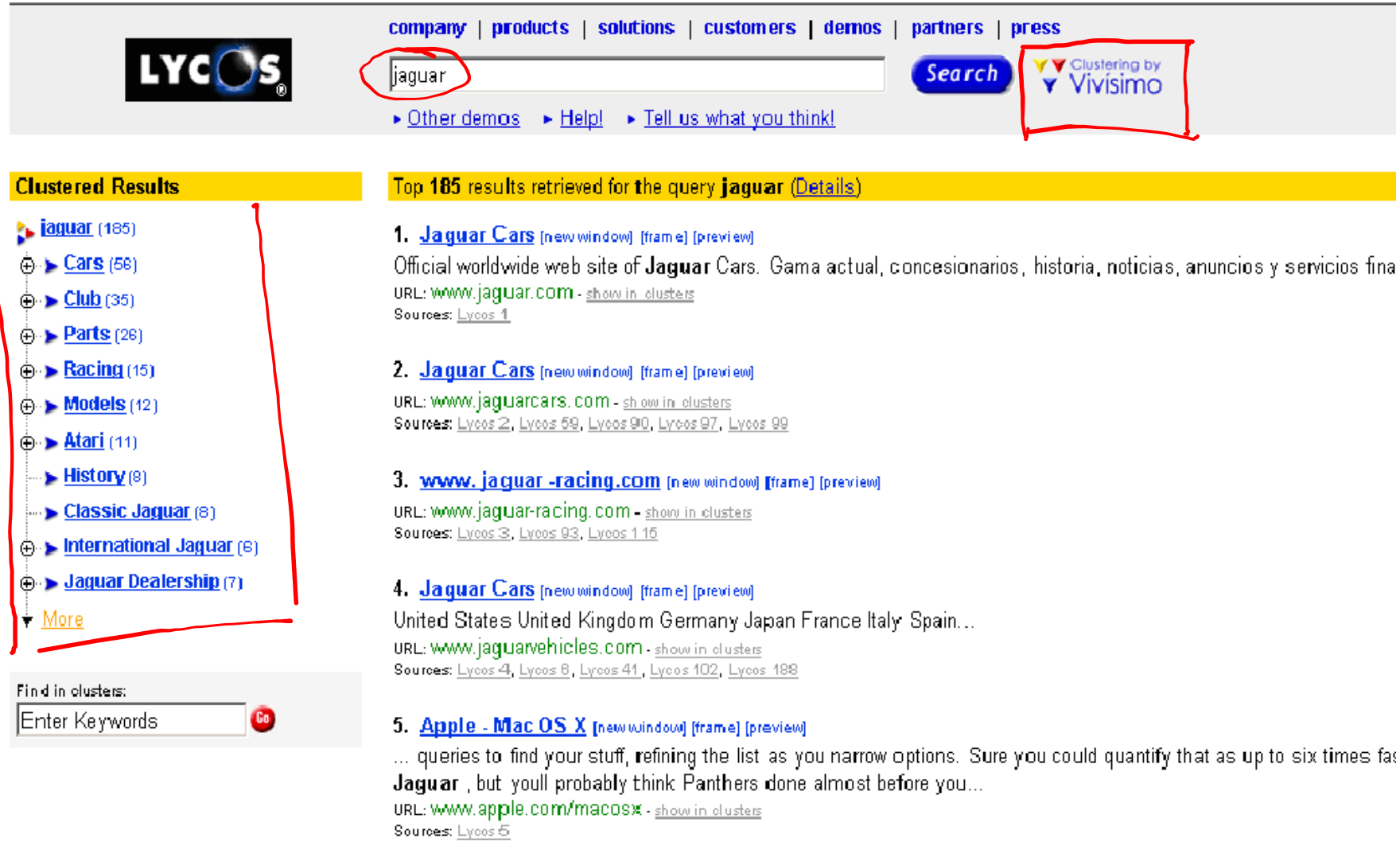
- Bentuk formal:
  - Input: $\boldsymbol{x} \in \mathcal{X} \in \mathbb{R}^n$
  - Output: $y \in \mathcal{Y} \in \begin{cases} \{1,2,\dots,K\} \rightarrow clustering\ K\ grup \\ \mathbb{R}^K \rightarrow embedding\ K\ dimension \end{cases}$
  - Target function: $f: \mathcal{X} \rightarrow \mathcal{Y}\ (unknown)$
  - Training data: $D = \{(\boldsymbol{x}^{(1)}), (\boldsymbol{x}^{(2)}), \dots, (\boldsymbol{x}^{(m)})\}$
  - Hypothesis: $h: \mathcal{X} \rightarrow \mathcal{Y}$
  - Hypothesis space: $h \in \mathcal{H}$

Clustering

- Menemukan struktur tersembunyi pada data
- Unsupervised learning: tidak ada informasi mengenai label



Contoh Aplikasi Clustering: Search Result



CALVIN INSTITUTE OF TECHNOLOGY

Contoh aplikasi clustering: Image Compression

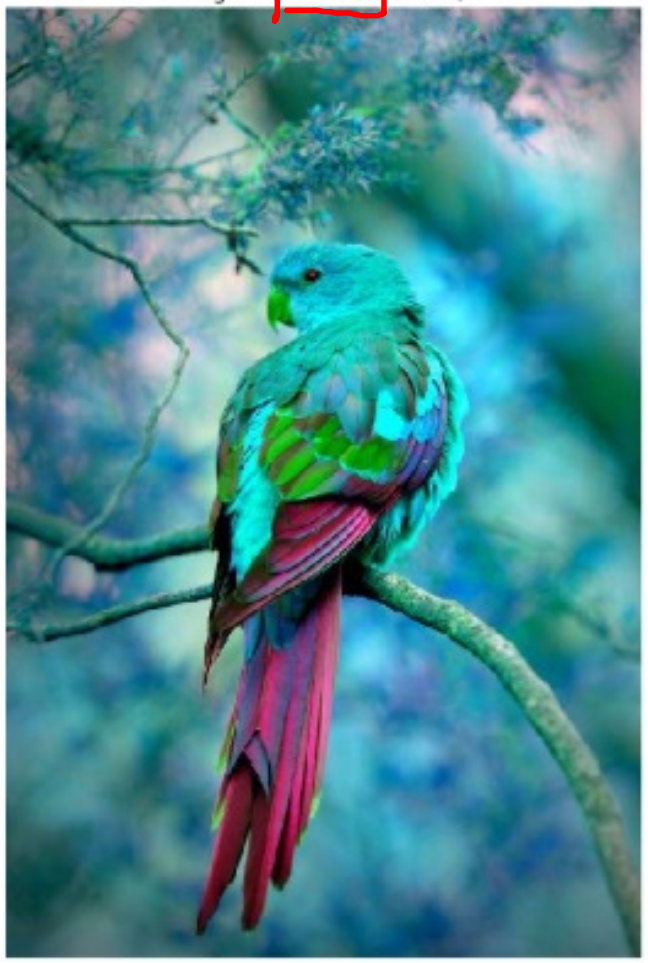
image Bird.jpg

k 160

K-means Image Compressor

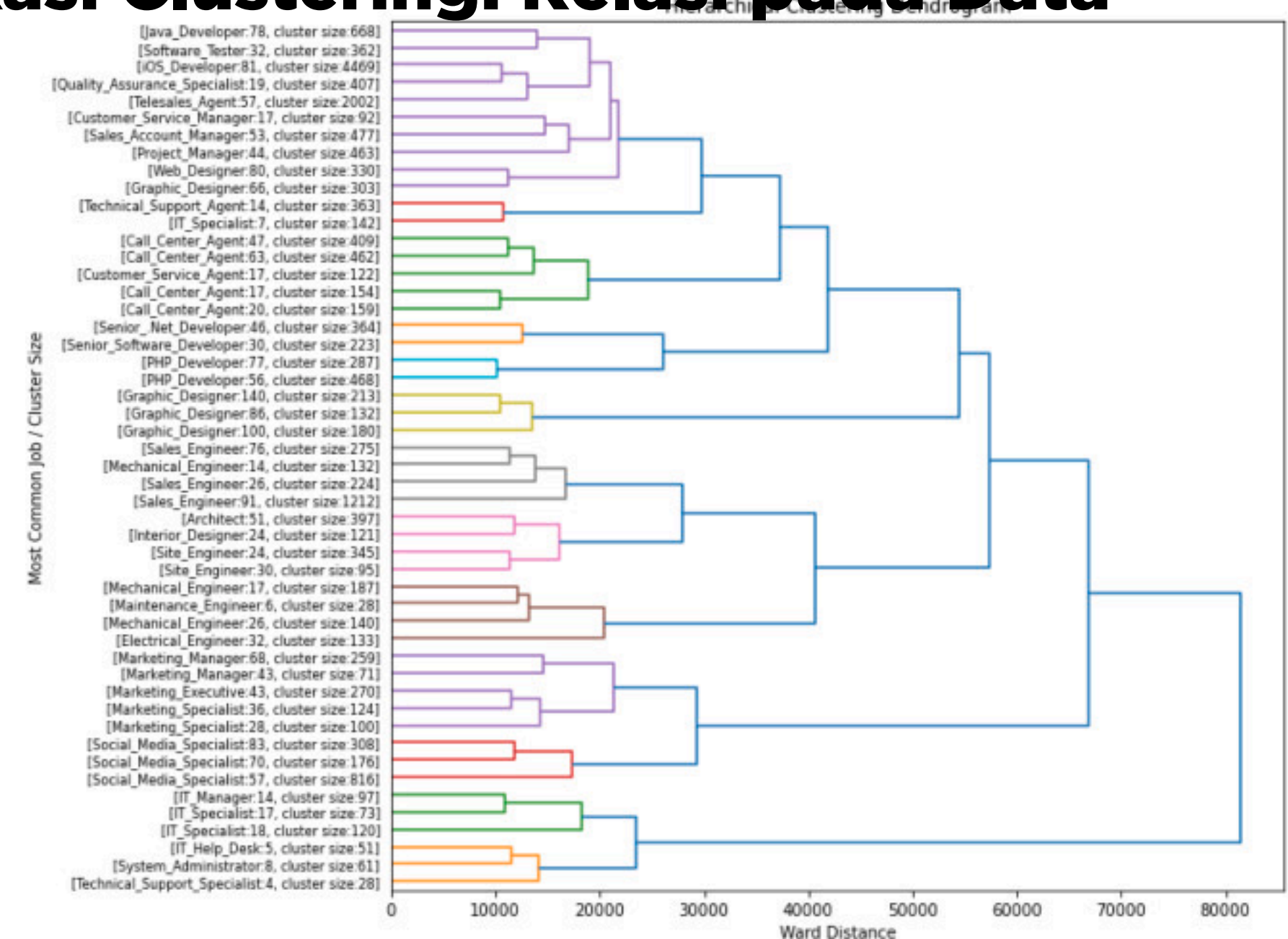
Compressed Image

Original (16Million Colors)



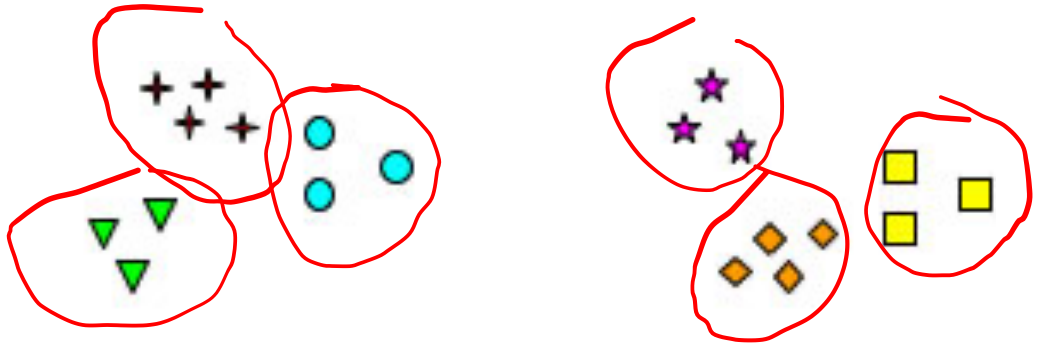
Contoh Aplikasi Clustering: Relasi pada Data

- www.sciencedirect.com/science/article/abs/pii/S1877750322002149

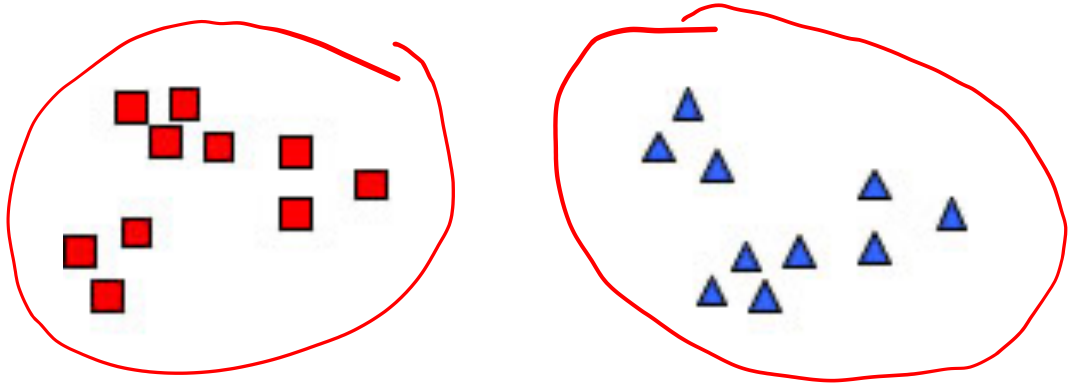


Cluster bersifat ambigu

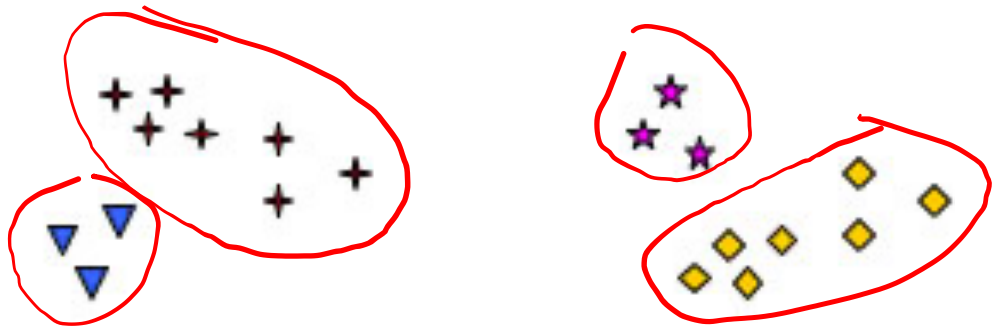
How many clusters?



Six Clusters



Two Clusters



Four Clusters

Jenis Clustering

- Hard Clustering:
  - Partitional clustering: pembagian data menjadi subset yang tidak overlap sehingga setiap data terletak pada suatu cluster
  - Hierarchical clustering: himpunan nested cluster yang disusun menurut sebuah hierarchical tree
  - Spectral clustering: pembagian data menjadi subset yang mengikuti struktur lokal
- Soft Clustering:
  - Mixture clustering: pembagian data menjadi subset yang saling overlap sehingga setiap data dapat terletak pada beberapa cluster sekaligus

Hard Clustering

K-means Clustering

- Konsep: minimisasi total jarak intra-cluster
- Cost function:

$$J(\boldsymbol{\theta}) = \sum_{i=1}^{M}\sum_{k=1}^{K} A_{ik} \left|\left|\theta_k - x^{(i)}\right|\right|^2$$

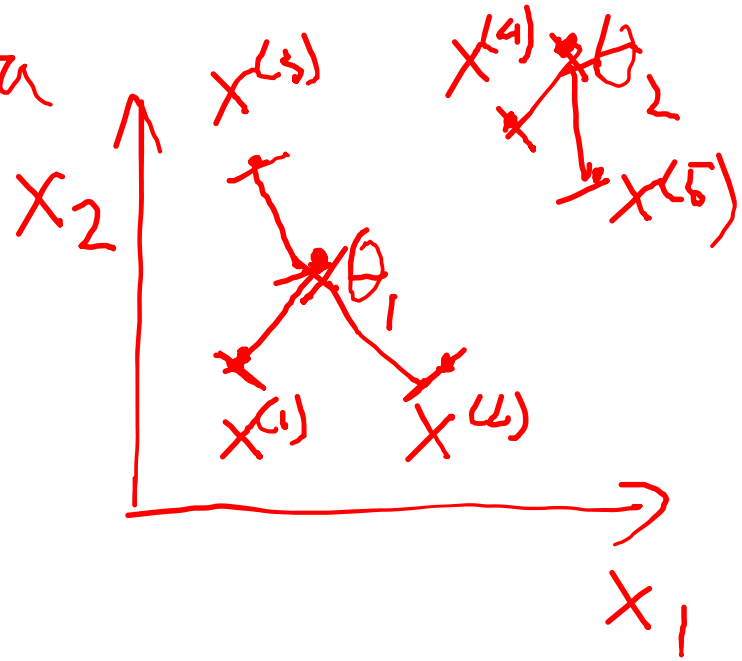
- Dimana $A_{ik} \in \{0,1\} \forall i,k$ dan $\sum_{k=1}^{K} A_{ik} = 1 \, \forall i$
- Gradient descent:

$$\theta_k := \theta_k - \alpha \frac{\partial J(\boldsymbol{\theta})}{\partial \theta_k}$$

- Solusi analitik:

$$0 = \frac{\partial J(\boldsymbol{\theta})}{\partial \theta_k} = 2\sum_{i=1}^{M} A_{ik}\left(\theta_k - x^{(i)}\right) \rightarrow \sum_{i=1}^{M} A_{ik}(\theta_k) = \sum_{i=1}^{M} A_{ik}\left(x^{(i)}\right)$$

$$\theta_k = \frac{\sum_{i=1}^{M} A_{ik}\left(x^{(i)}\right)}{\sum_{i=1}^{M} A_{ik}}$$

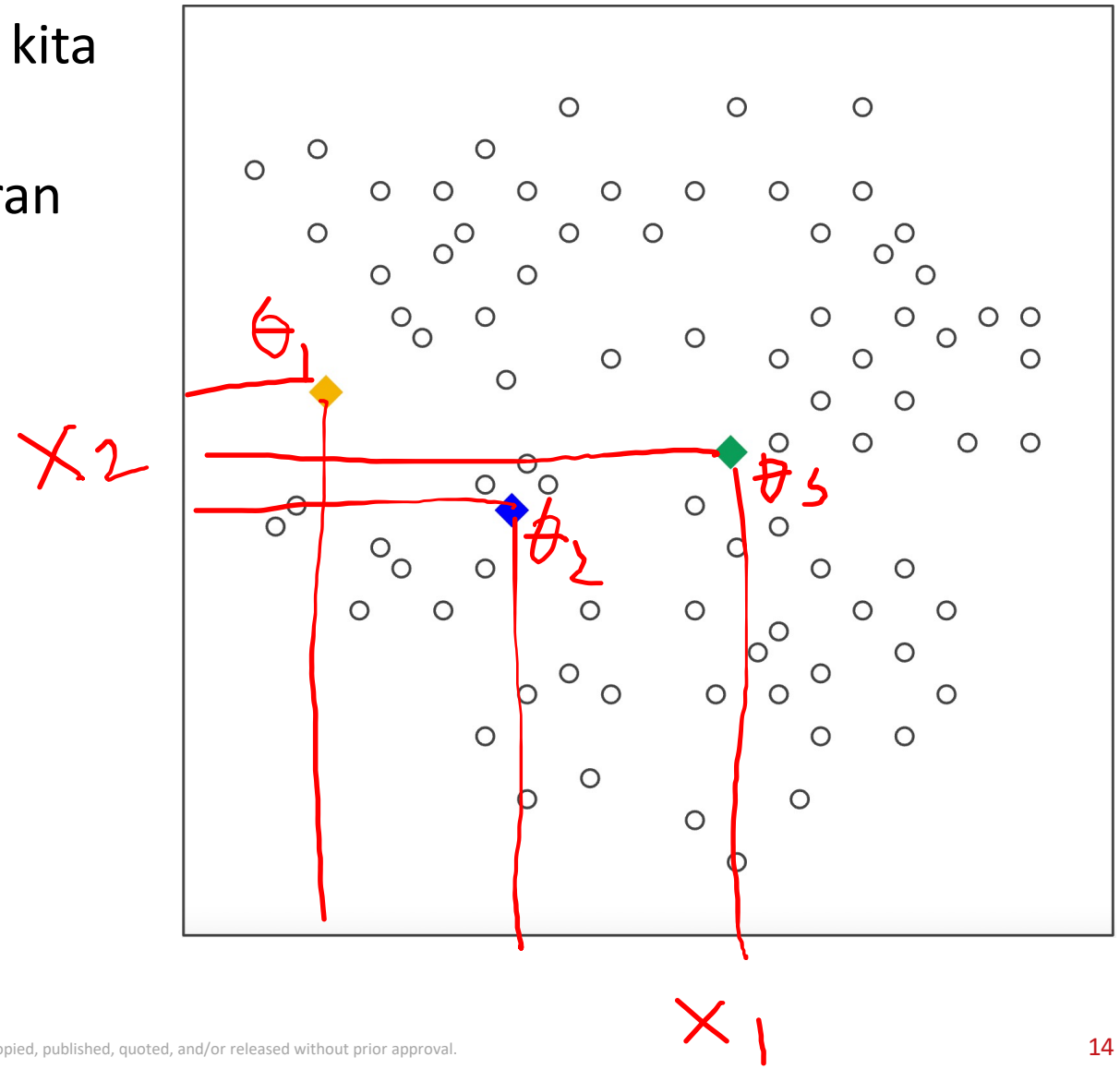


Algoritma K-Means

- https://developers.google.com/machine-learning/clustering/algorithm/run-algorithm
- 0. Tentukan jumlah cluster
- 1. Algoritma memilih secara acak centroid (pusat) dari setiap cluster
- 2. Algoritma menetapkan setiap data pada centroid terdekat untuk memperoleh k cluster awal
- 3. Untuk setiap cluster, algoritma akan menghitung ulang centroid sebagai rata-rata dari setiap data cluster tersebut
- 4. Algoritma mengulangi perhitungan centroid dan penetapan cluster sampai konvergen

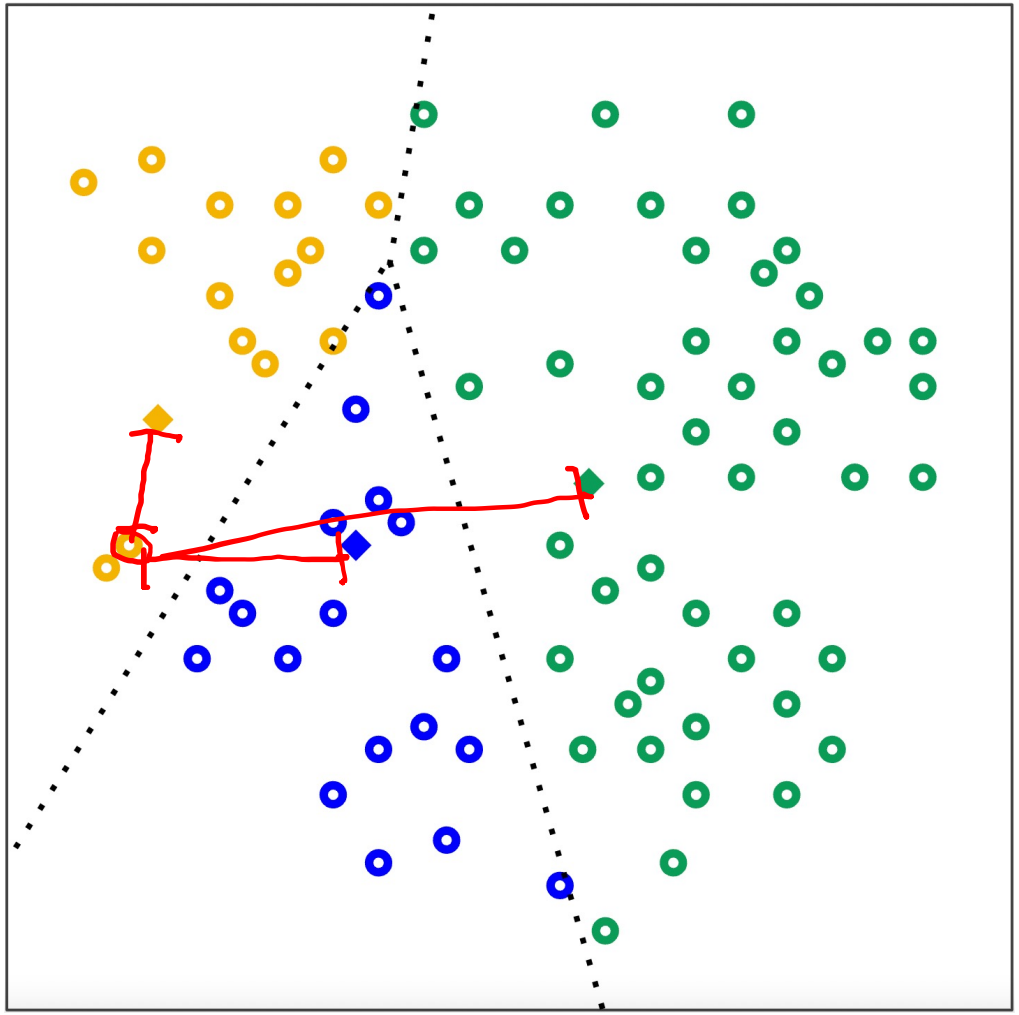
Pilih acak centroid

- Tentukan jumlah cluster (dalam kasus ini kita tentukan ada 3 clusters) k=3
- Lalu pilih acak posisi centroid pada sebaran data (centroid berada pada posisi yang berwarna kuning, biru, hijau)



Penentuan cluster sesuai jarak centroid terdekat

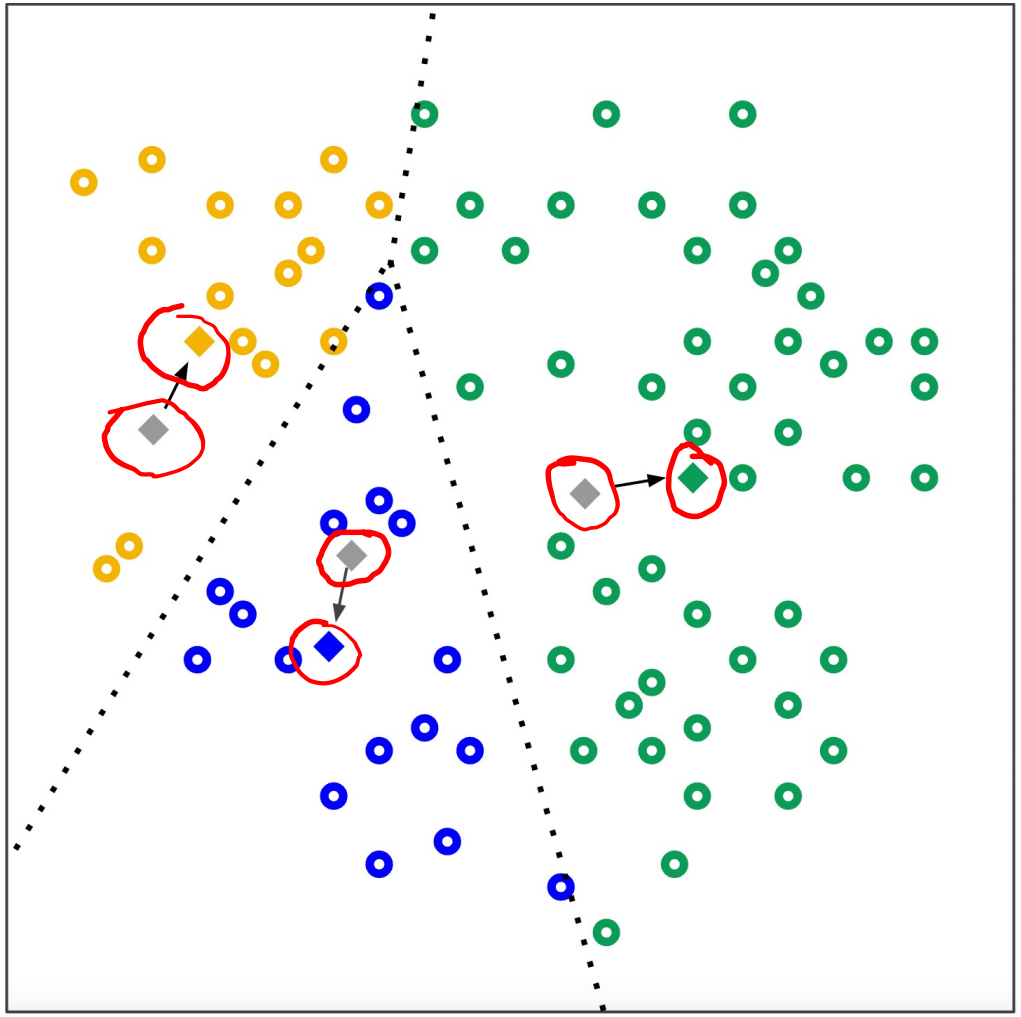
- Untuk setiap data, tentukan anggota clusternya dengan cara menghitung jaraknya dengan setiap centroid
- lalu pilih cluster dengan centroid terdekat (setiap data digolongkan menuju cluster kuning, hijau, dan biru menurut kedekatan lokasinya)



CALVIN INSTITUTE OF TECHNOLOGY

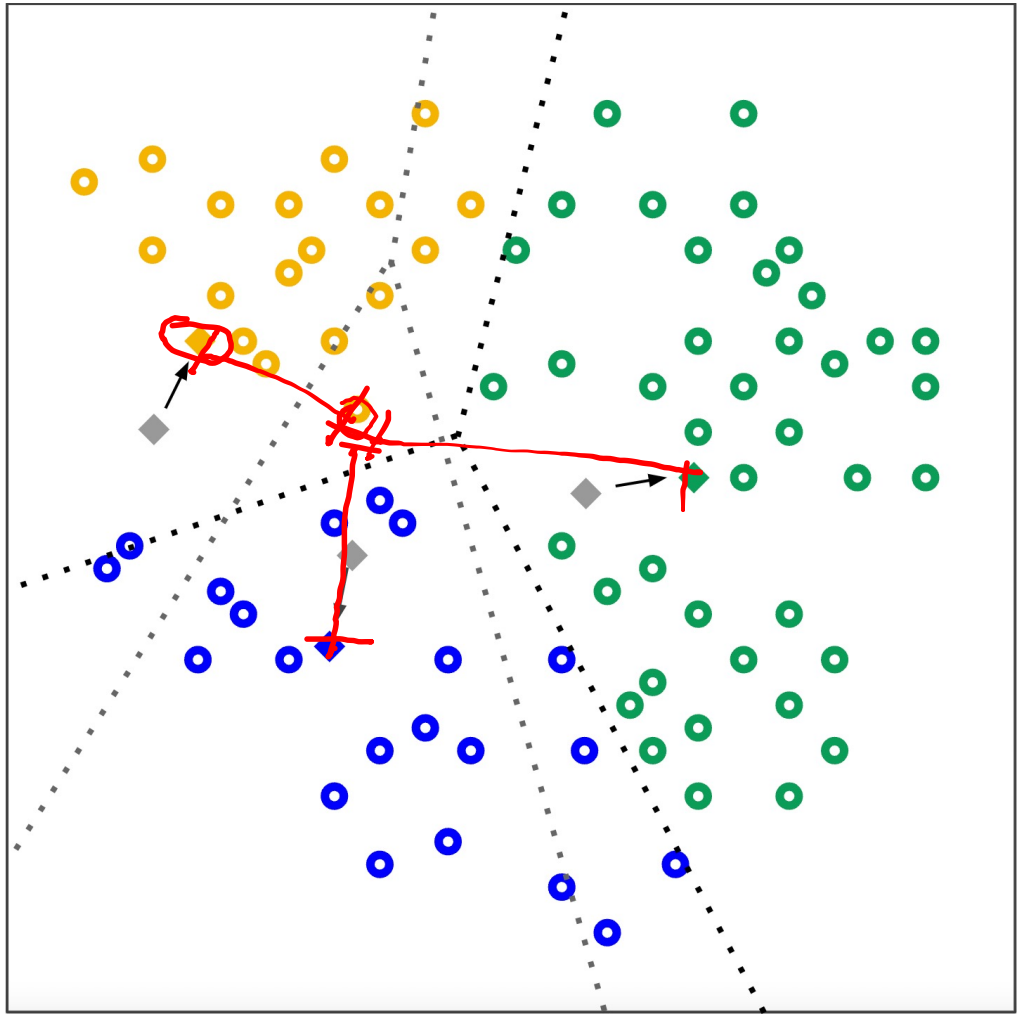
Hitung ulang posisi centroid

- Untuk setiap cluster, hitung ulang posisi centroid dengan cara menghitung rata-rata dari seluruh data pada cluster tersebut
- Langkah ini akan merubah posisi centroid lama ke baru (centroid bergeser dari titik abu-abu ke titik yang baru)



Cluster baru sesuai centroid baru

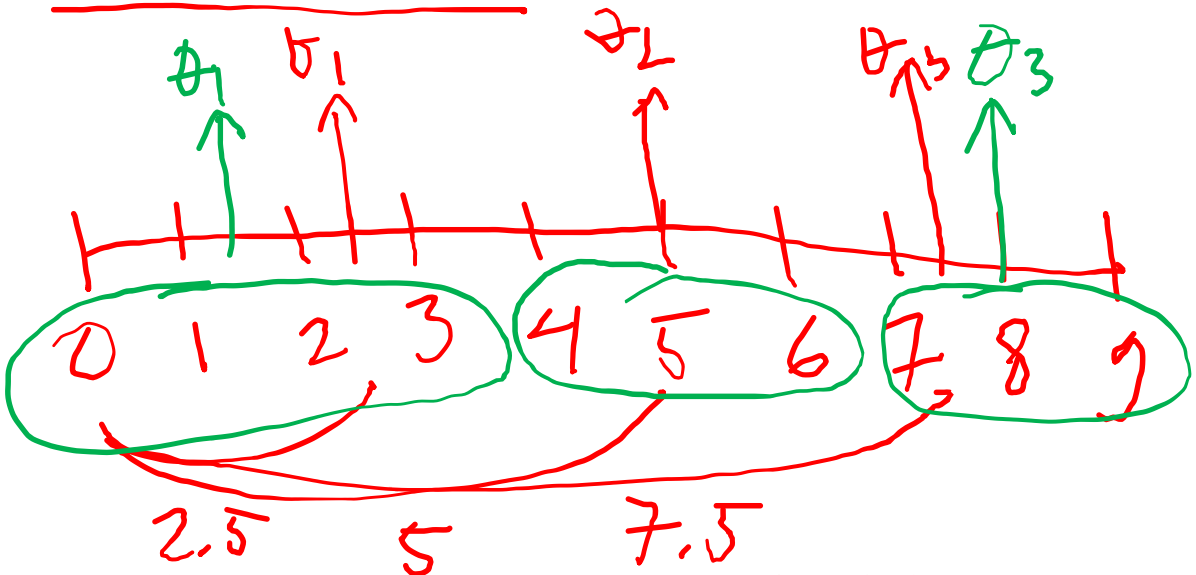
- Karena centroid telah berubah, maka posisi seluruh data terhadap cluster baru juga berubah
- Anggota cluster dihitung ulang menurut centroid terdekat (pembagian cluster disesuaikan menurut posisi centroid yang baru)



CALVIN INSTITUTE OF TECHNOLOGY

Uji Pemahaman

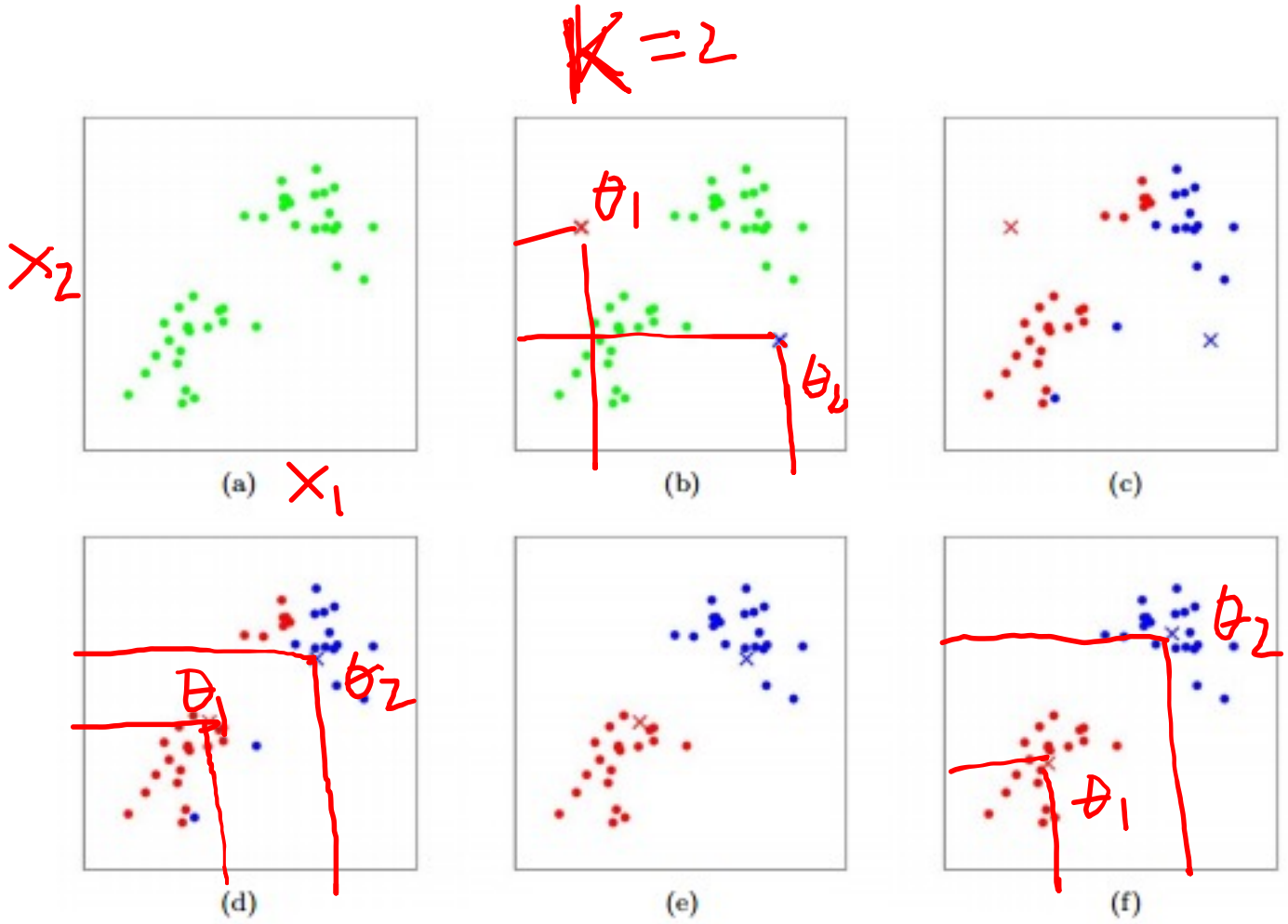
- Terdapat data 1 fitur $X = [0,1,2,3,4,5,6,7,8,9]$
- Untuk k-means clustering 3 cluster dengan centroid ($m_1 = 0.25, m_2 = 0.5, m_3 = 0.75$), manakah yang termasuk cluster 1, 2, dan 3?
- Berapakah nilai centroid yang baru untuk arrangement cluster tersebut?



$\theta_1 = \frac{0+1+2+3}{4} = 1.5$ $\theta_2 = \frac{4+5+6}{3} = 5$ $\theta_3 = \frac{7+8+9}{3} = 8$

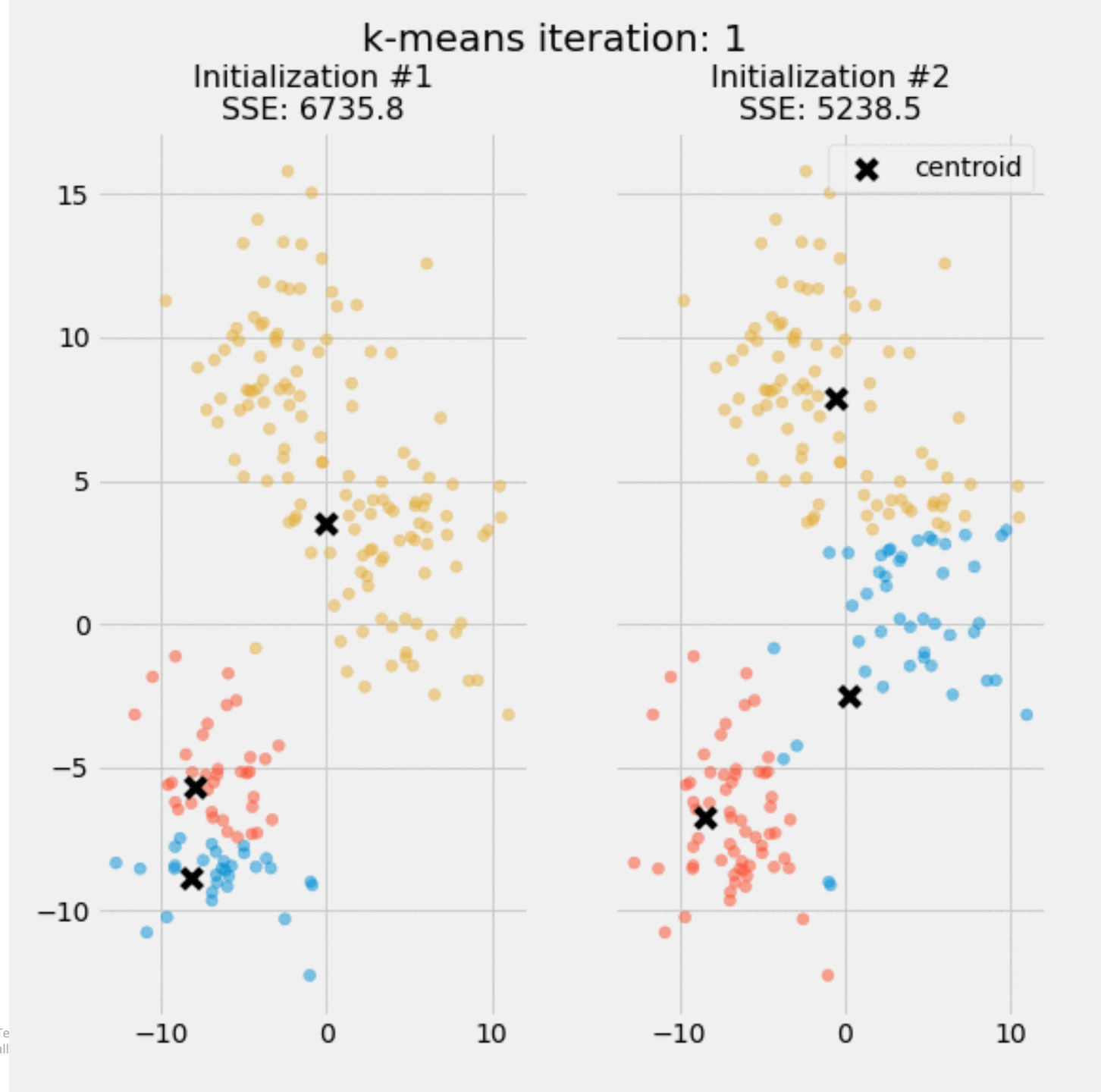
Iterasi K-means

- https://stanford.edu/~cpiech/cs221/handouts/kmeans.html
- *(a) Original dataset. (b) Random initial cluster centroids. (c-f) Illustration of running two iterations of k-means. In each iteration, we assign each training example to the closest cluster centroid*
- *Images courtesy of Michael Jordan.*



K-means konvergen

- K-means sensitive terhadap posisi awal centroids
- K-means tidak selalu konvergen pada hasil yang sama

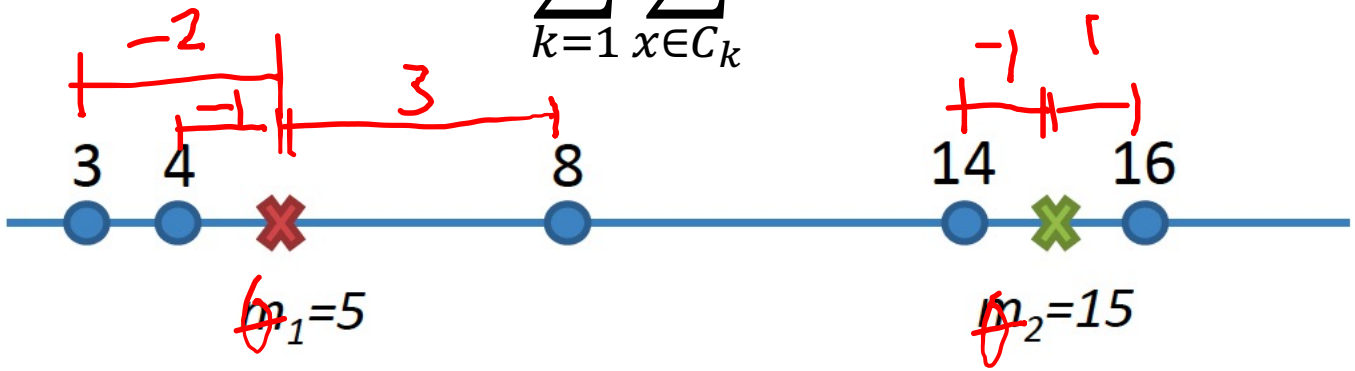


Evaluasi

- Sum of square error (SSE)
  - Untuk setiap titik data, error (jarak dari cluster terdekat) dihitung
  - Semua error dikuadratkan lalu dijumlah

$$SSE = \sum_{k=1}^{K} \sum_{x \in C_k} (m_k - x)^2$$

- Contoh:



$$SSE = (3-5)^2 + (4-5)^2 + (8-5)^2 + (14-15)^2 + (16-15)^2$$
$$SSE = (4+1+9) + (1+1) = 16$$

Uji Pemahaman

- Berapa nilai SSE dari data 1 fitur $X = [0,1,2,3,4,5,6,7,8,9]$ dengan centroid ($m_1 = 0.25, m_2 = 6.5, m_3 = 8.75$)?

$(0-2.5)^2 + (1-2.5)^2 + (2-2.5)^2 + (3-2.5)^2 + (4-5)^2 + (5-5)^2 + (6-5)^2 +$
$(7-7.5)^2 + (8-7.5)^2 + (9-7.5)^2$

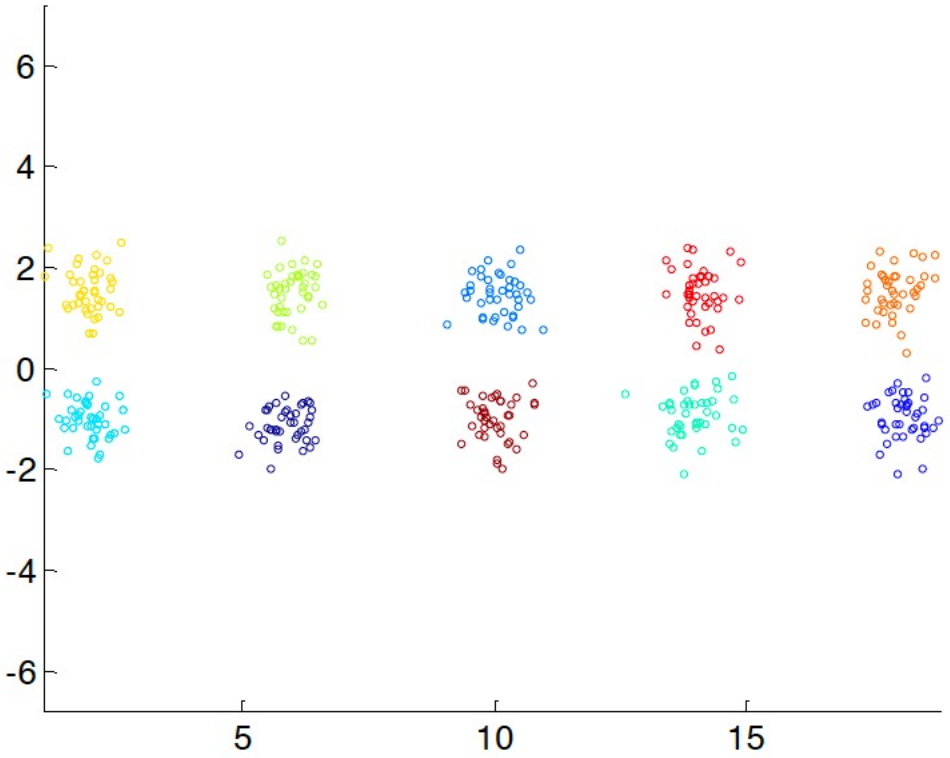
$6.25 + 2.25 + 0.25 + 0.25 + 1 + 0 + 1 + 0.25 + 0.25 + 2.25$

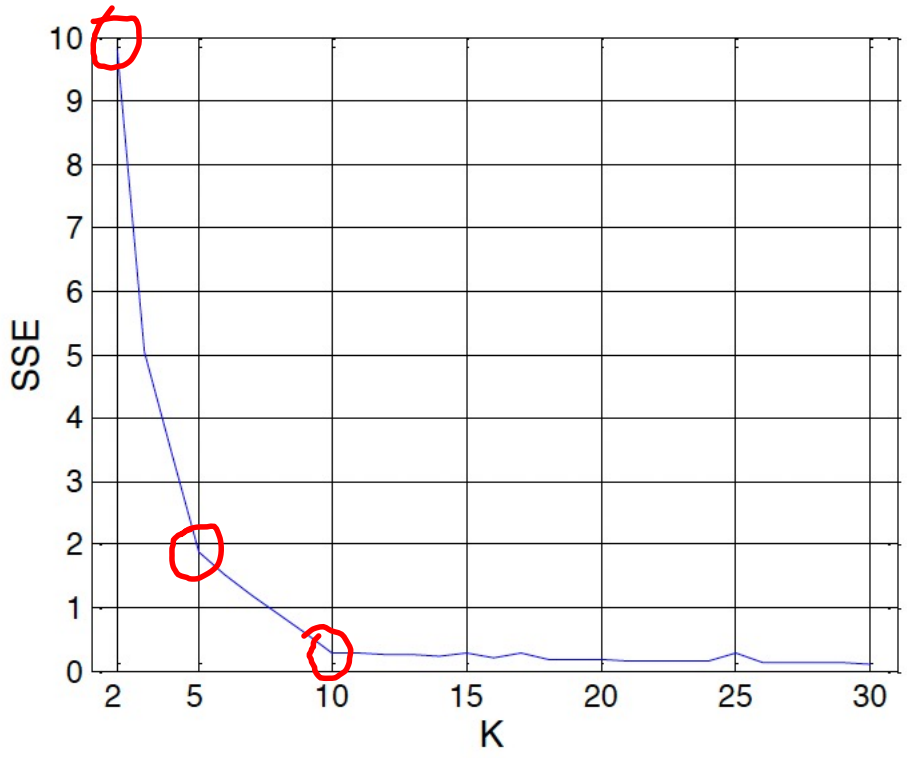
$4 \times 0.25 + 2 \times 1 + 2 \times 2.25 + 6.25 = 13.75$

1 2 4.5

Menentukan K optimal

- Plot SSE untuk setiap pilihan jumlah cluster K
- Elbow method (pilih nilai K pada siku kurva)

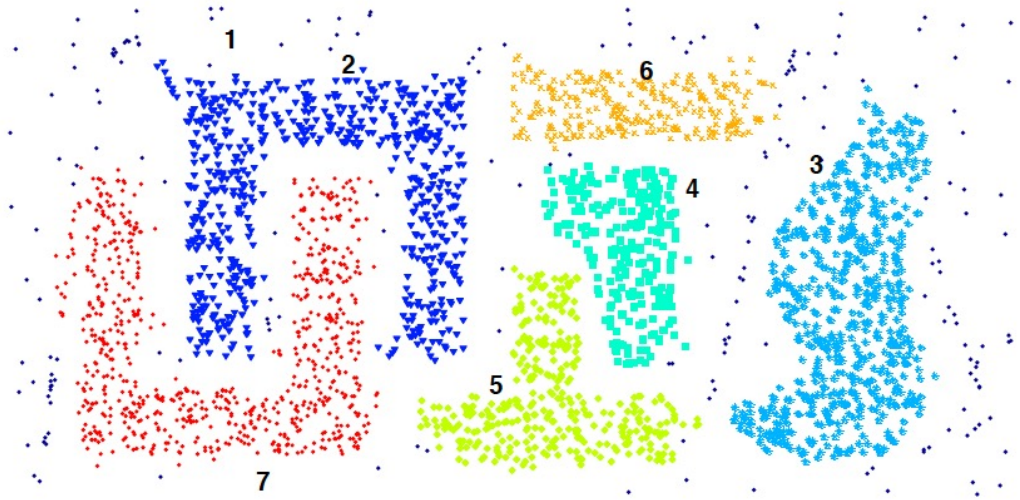


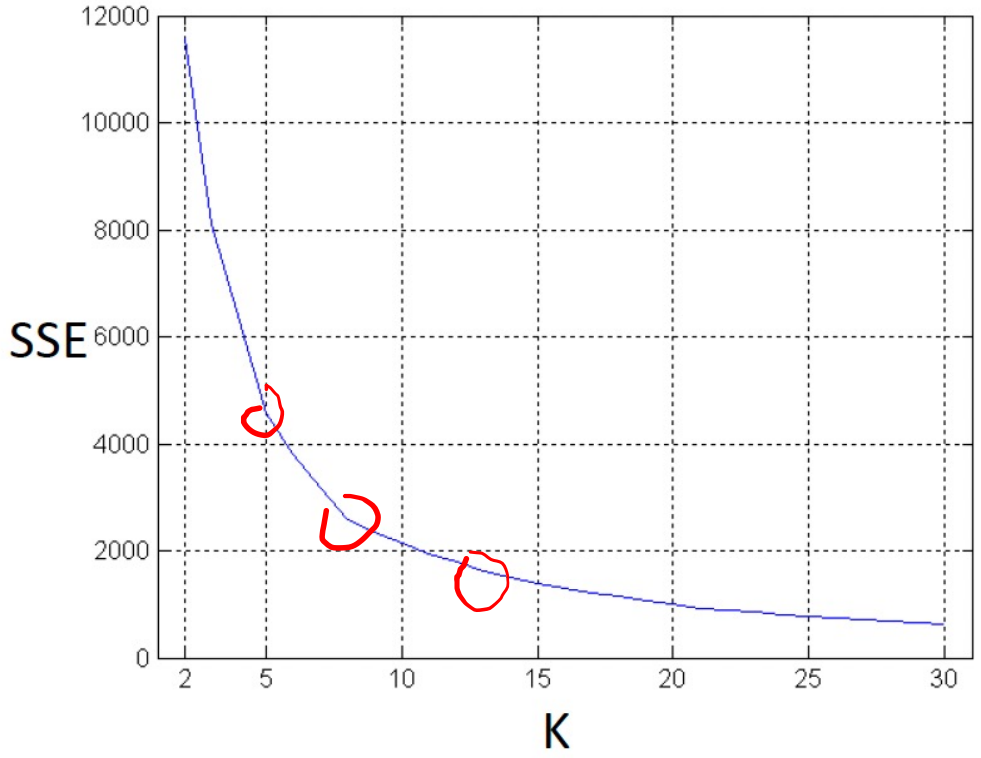


CALVIN INSTITUTE OF TECHNOLOGY

Nilai K ambigu

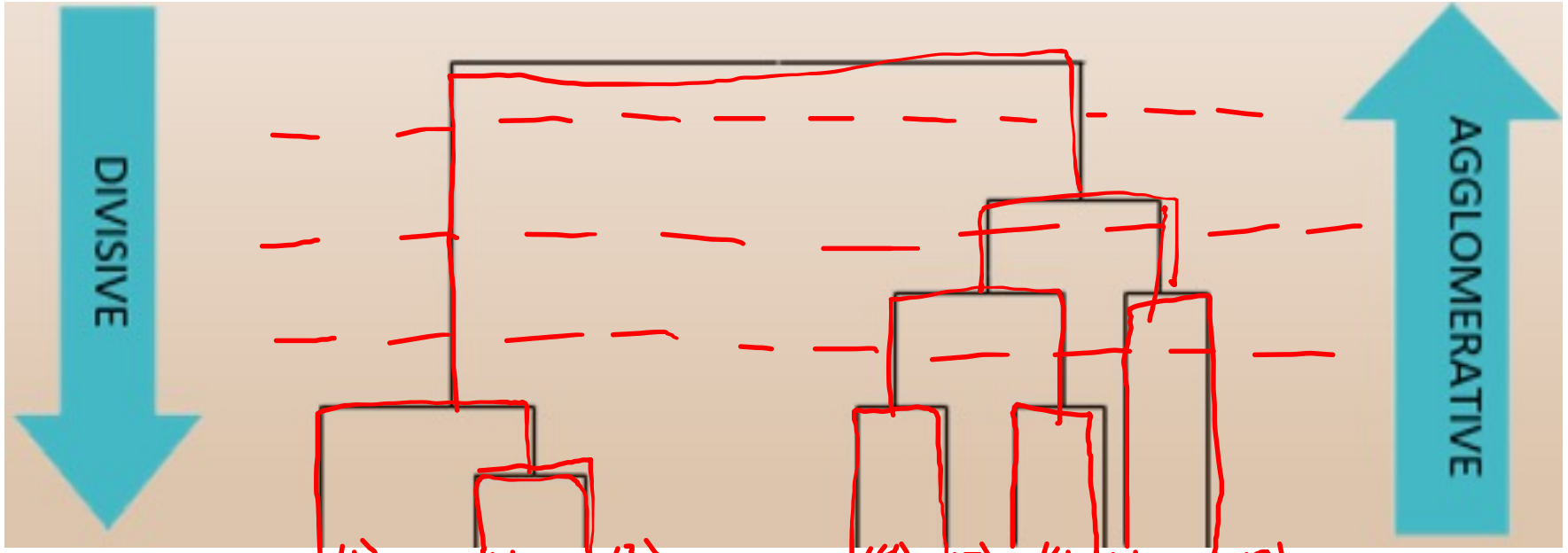
- Terkadang sulit menentukan siku kurva SSE
- Gunakan metode lain seperti metode silhouette
- Gunakan algoritma lain seperti DBScan, hierarchical clustering, spectral clustering, dll





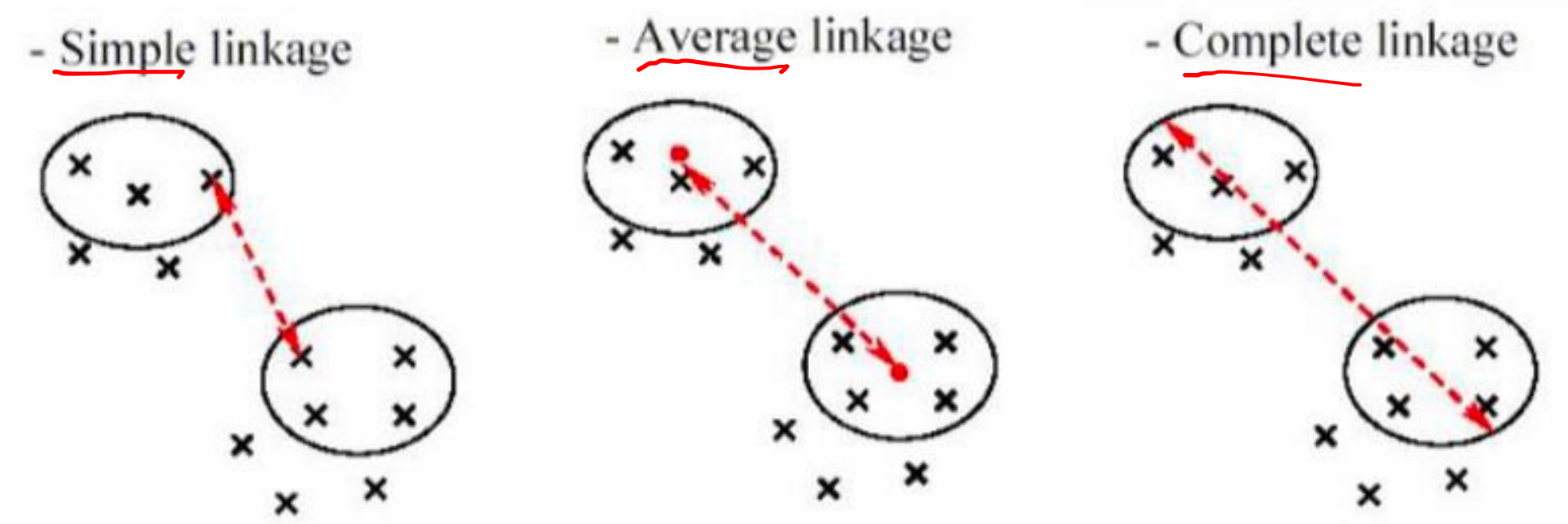
Hierarchical Clustering

- Membangun hirarki cluster dari tunggal menjadi banyak atau sebaliknya
- Terdapat 2 jenis strategi pengelompokan:
  - Agglomerative: dimulai dengan setiap data sebagai sebuah cluster kemudian membentuk cluster yang semakin membesar
  - Divisive: dimulai dengan semua data dikumpulkan dalam satu cluster kemudian dibagi sampai setiap data berada pada cluster tersendiri



Agglomerative Hierarchical Clustering

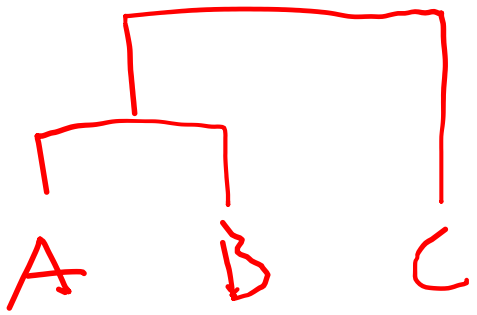
- Teknik pengelompokan:
  - Single linkage (jarak terdekat): menggabungkan cluster menurut jarak 2 data terdekat antar cluster
  - Average linkage (jarak rata-rata): menggabungkan cluster menurut jarak rata-rata tiap pasangan data antar cluster
  - Complete linkage (jarak terjauh): menggabungkan cluster menurut jarak 2 data terjauh antar cluster



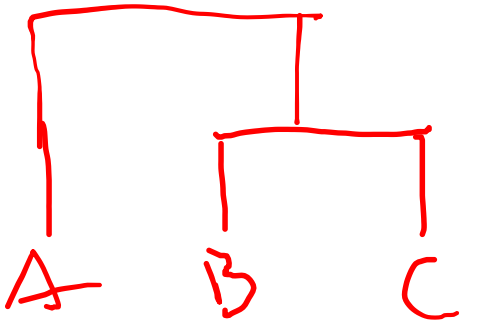
CALVIN INSTITUTE OF TECHNOLOGY

Uji Pemahaman

- Terdapat data 1 fitur yang sudah membentuk clusters:
  - $A = [0,1,2,3,4]$
  - $B = [5,6,7]$
  - $C = [8,9]$
- Manakah bentuk pengelompokan yang tepat menurut:
  - Single linkage
  - Average linkage
  - Complete linkage

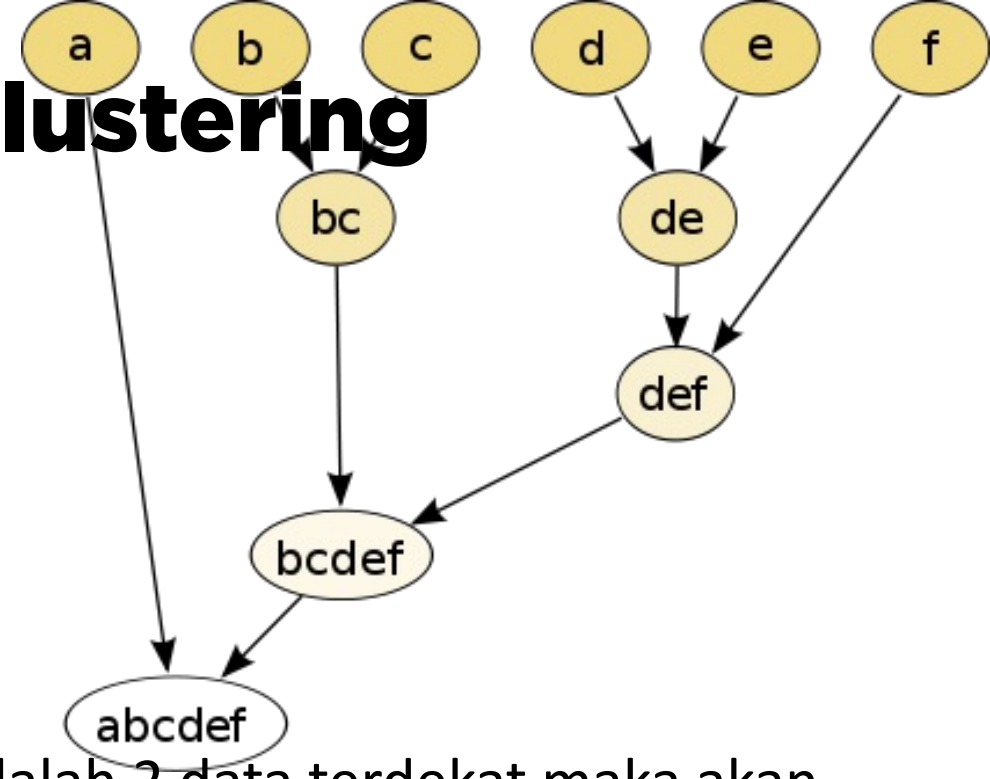


Single



Complete

Agglomerative Hierarchical Clustering



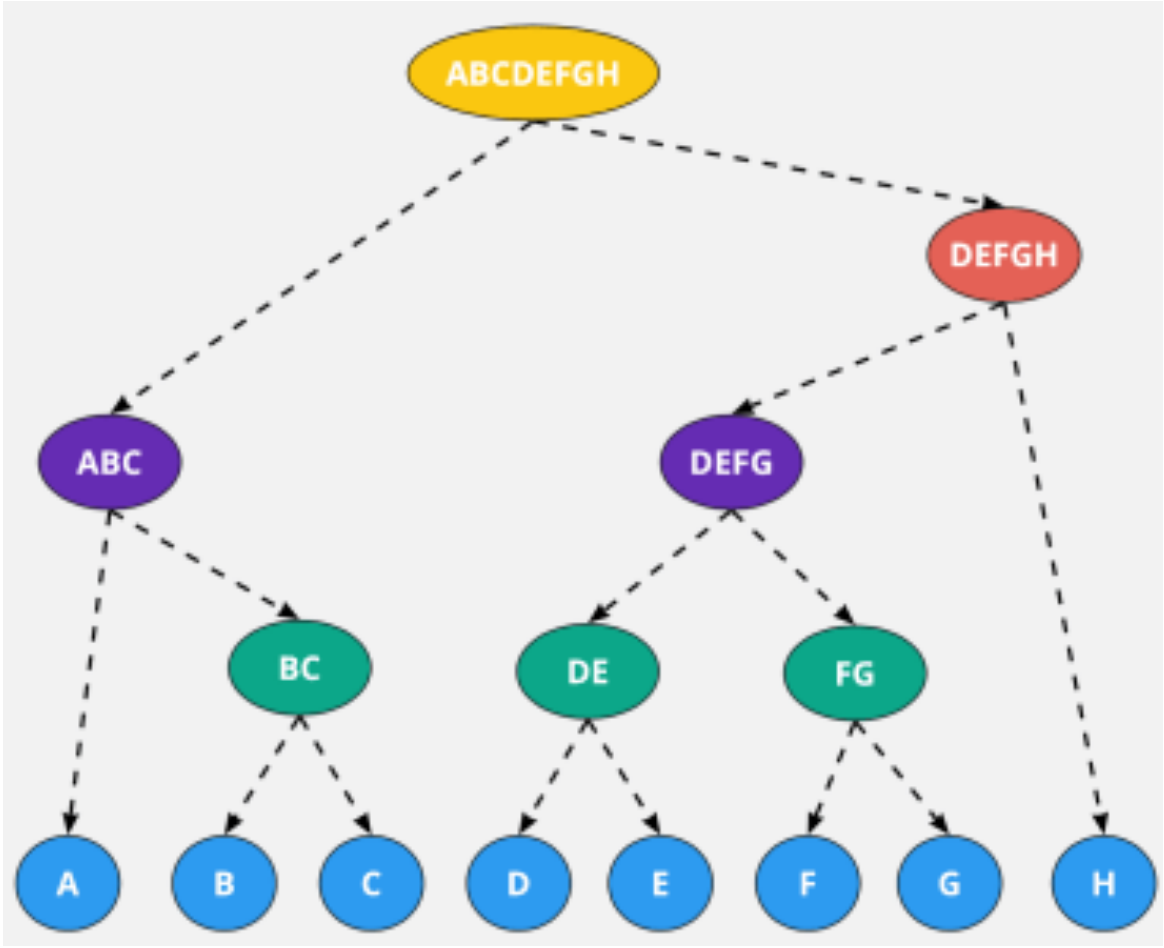
- Algoritma:
  - Buat matriks jarak dengan menghitung jarak
    - Misalnya jarak Euclidean:

$$d_{ii'} = \sqrt{\sum_{j=1}^{N} \left(x_j^{(i)} - x_j^{(i')}\right)^2}$$

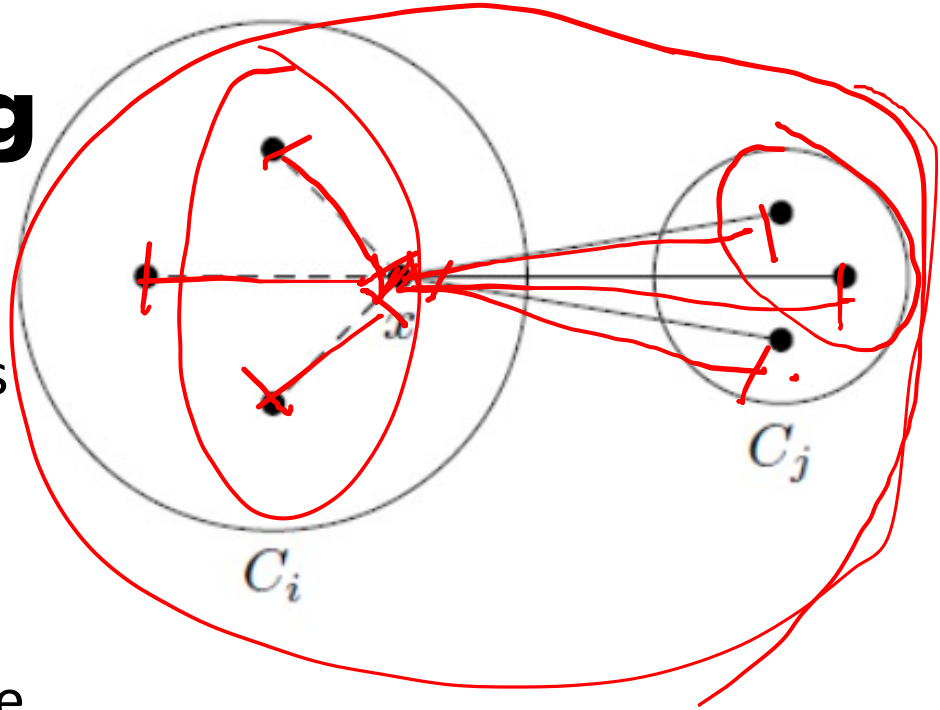
  - Gabungkan 2 cluster terdekat: jika data a dan b adalah 2 data terdekat maka akan membentuk cluster gabungan pertama
  - Perbaharui matriks jarak sesuai dengan teknik pengelompokan (single, average, complete)
  - Ulangi Langkah 2-3 hingga tersisa 1 cluster yang memuat seluruh data
  - Buat dendrogram (“dendron”=pohon, “gramma”=gambar)

Divisive Hierarchical Clustering

- DIANA (Divisive ANAlysis):
  - membagi cluster tunggal menjadi 2 clusters yang lebih kecil, dan seterusnya hingga setiap data adalah 1 cluster tersendiri



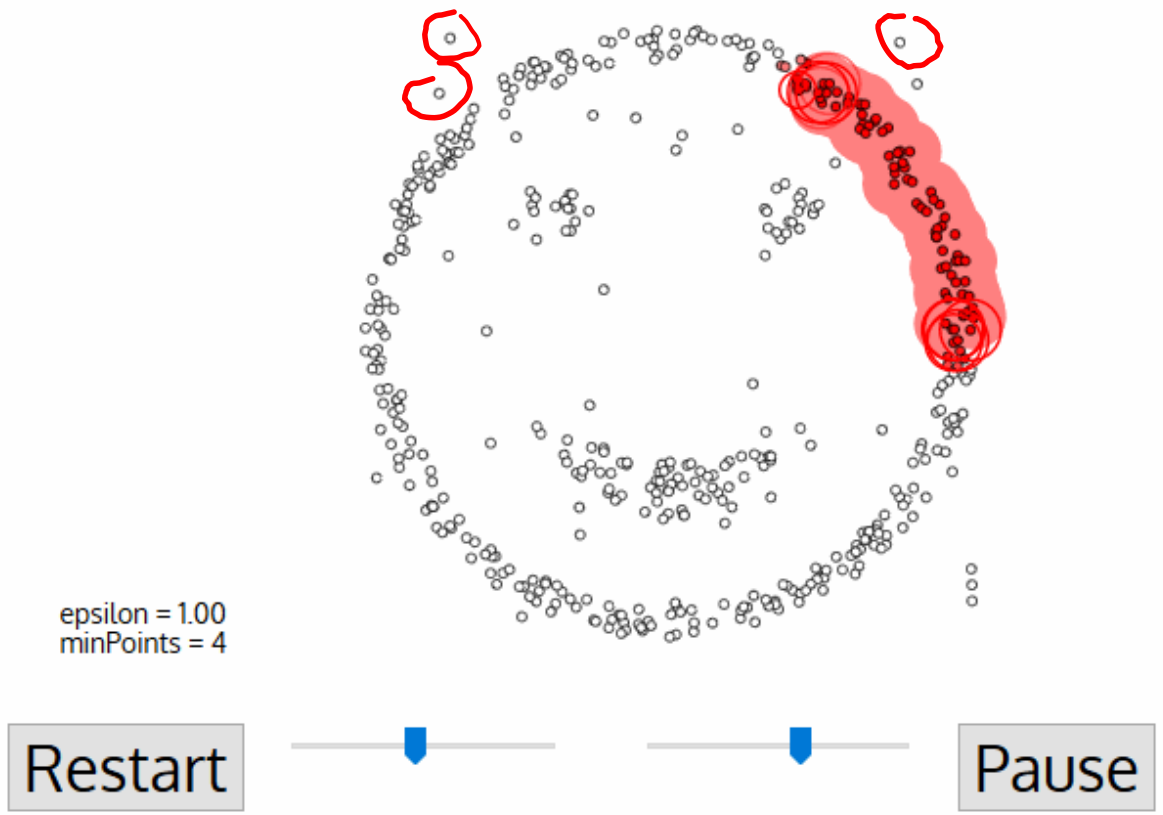
Divisive Hierarchical Clustering



- Algoritma:
  - Suatu cluster tunggal C akan dibagi menjadi clusters C_i dan C_j
  - Anggaplah $C_i = C$ sedangkan $C_j = \emptyset$
  - Untuk setiap data $x \in C_i$:
    - Untuk iterasi pertama: hitung jarak rata-rata x ke semua data→pindah data dengan jarak rata-rata maksimum ke C_j
    - Untuk iterasi selanjutnya: hitung $D_x = avg\{d(x,y), y \in C_i\} - avg\{d(x,y), y \in C_j\}$ → tentukan data di C_i yang memiliki D_x paling besar
    - Ulangi sampai semua nilai D_x negative
  - Tentukan cluster terkecil dengan diameter terbesar.
  - Ulangi hingga setiap cluster hanya ada 1 data

DBScan

- Density Based Spatial Clustering of Applications with Noise
- https://www.kdnuggets.com/2020/04/dbsca-clustering-algorithm-machine-learning.html
- Mengelompokan data yang berdekatan satu sama lain (high density)
- Data di daerah low density (tetangga jauh)→outliers / noise
- Cluster: wilayah terhubung dengan kepadatan tinggi
- Bagaimana menentukan anggota cluster? Epsilon? Minpoints?

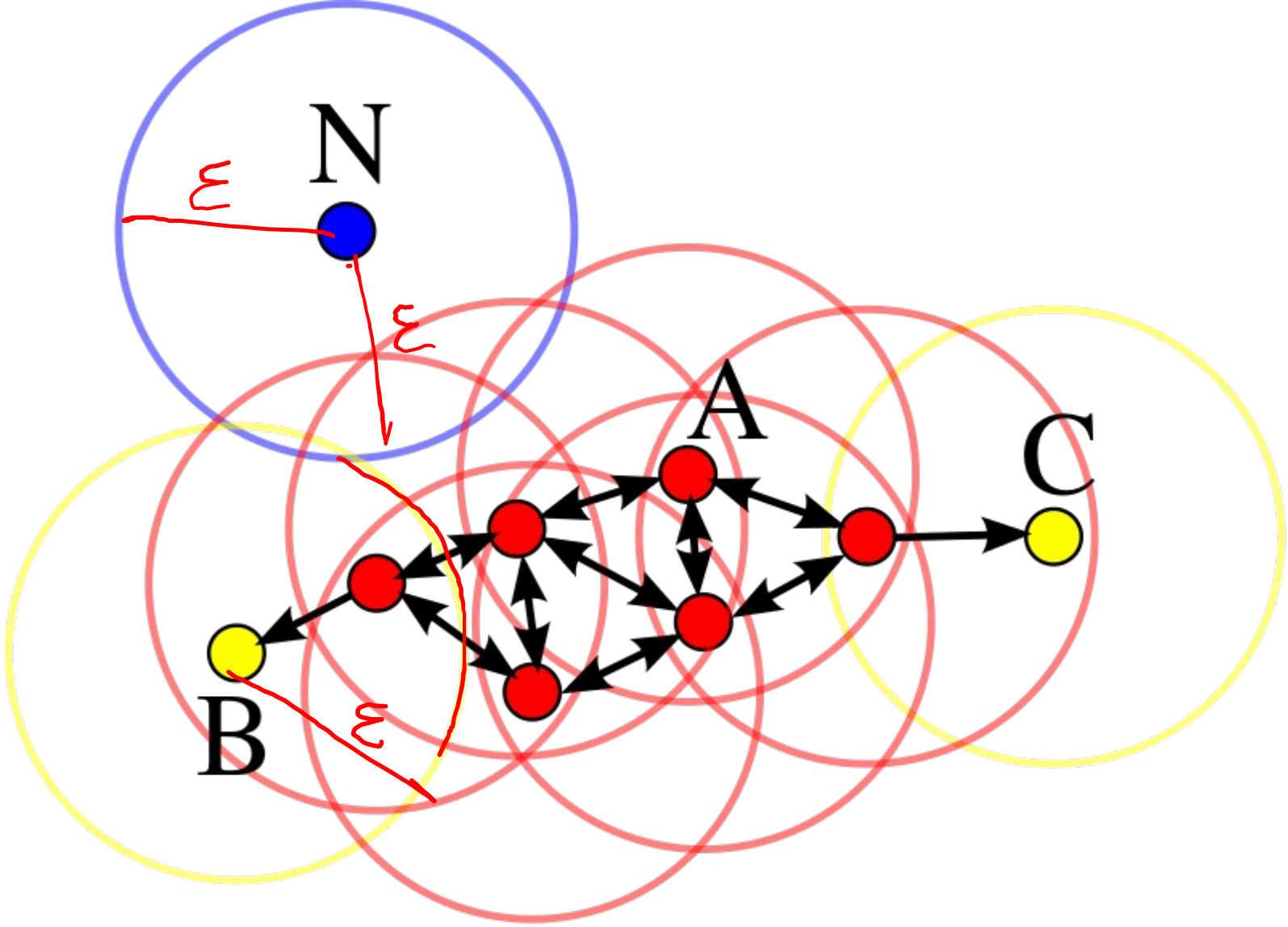


DBScan

- Definisi:
  - ε: radius dari suatu titik p
  - MinPts: jumlah minimal data untuk membentuk daerah padat
- Penggolongan data:
  - Core Point→titik p disebut core point jika ada minimal MinPts titik yang berjarak $\leq \varepsilon$ dari p (termasuk p sendiri)→dense region
  - Directly reachable point→titik q directly reachable jika berjarak $\leq \varepsilon$ dari p
  - Reachable point→titik q reachable dari p jika ada jalur $p_1, p_2, \ldots, p_n$ dimana di awal jalur $p_1 = p$, di akhir jalur $p_n = q$, dan di tengah-tengah jalur p_{i+1} directly reachable dari p_i
  - Outliers/noise points→ titik yang tidak reachable dari titik lain manapun

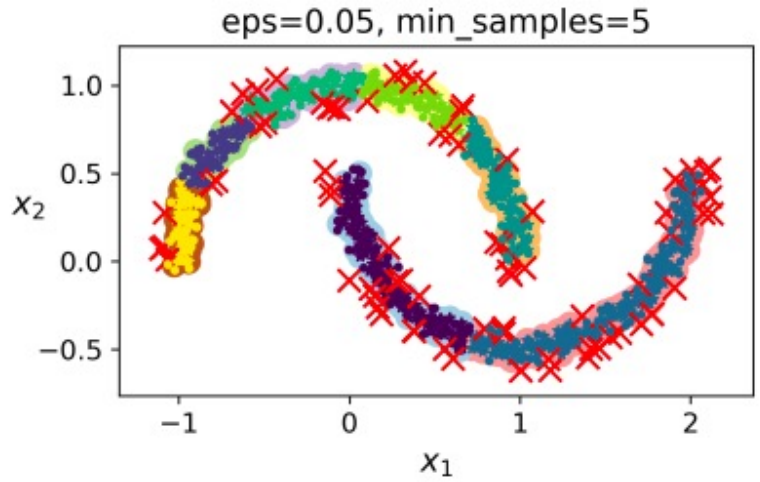
DBScan

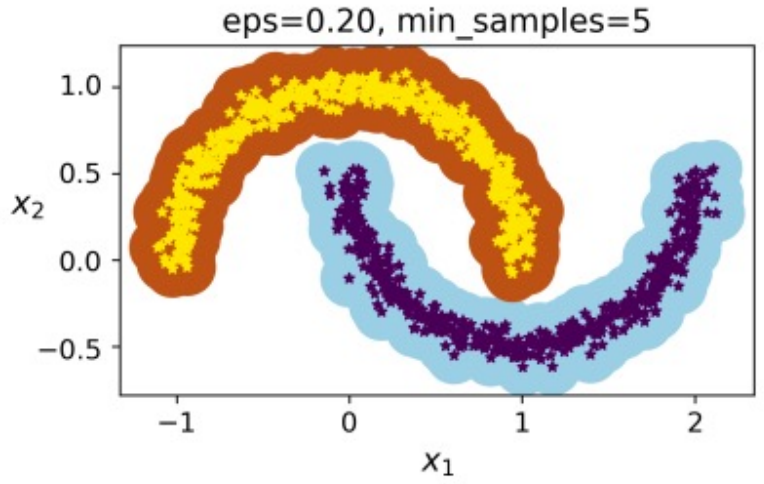
- Lingkaran = radius ε
- MinPts = 4
- Merah = core points
- Kuning = reachable points
- Biru = outliers / noise points



DBScan

- Algoritma bekerja baik jika semua clusters cukup padat dan dipisahkan oleh wilayah low density secara jelas
- Kelebihan:
  - Implementasi sederhana
  - Identifikasi clusters berapapun dan bentuk apapun
  - Robust terhadap outliers
  - Hyperparameter hanya 2 (ε,minPts)
- Limitasi:
  - Jika kepadatan bervariasi antar clusters
  - Sulit pilih hyperparameter

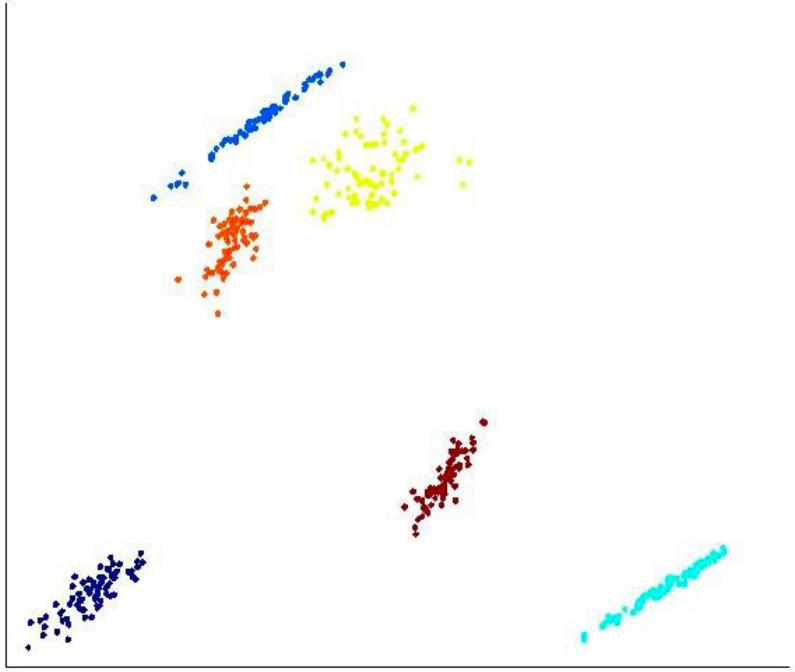




CALVIN INSTITUTE OF TECHNOLOGY

Spectral Clustering

- Clustering biasa dapat dipakai tetapi model parametrik dan jarak Euclidean tidak selalu cocok



Spectral Clustering

- Affinity Matrix:

$$A_{ii'} = e^{-\frac{\left\|x^{(i)} - x^{(i')}\right\|^2}{2\sigma^2}}$$

- Diagonal Matrix:

$$D_{ii} = \sum_{i'} A_{ii'}$$

- Graph Laplacian Matrix:

$$L = D^{-\frac{1}{2}} A D^{-\frac{1}{2}}$$

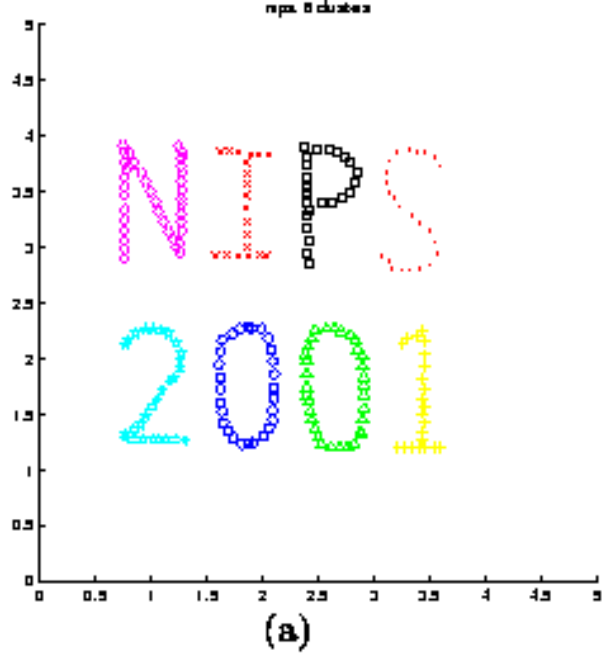
- Lakukan eigen-decomposition terhadap Graph Laplacian Matrix dan kumpulkan k largest eigenvectors:

$$v_1, v_2, \ldots, v_k$$

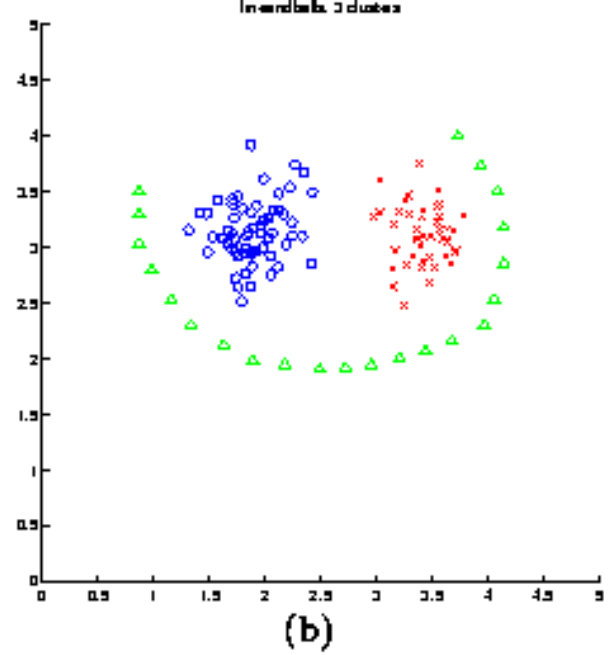
- Renormalisasi eigenvector:

$$Y_{ij} = \frac{V_{ij}}{\left(\sum_j V_{ij}^2\right)^2}$$

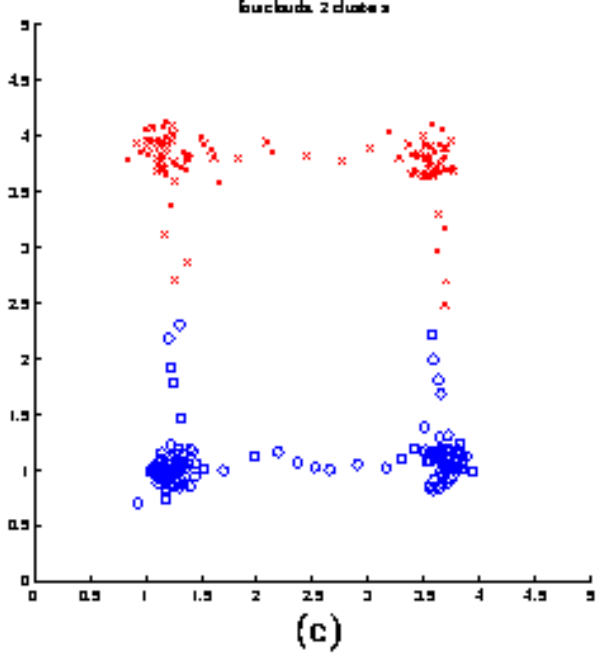
- Cluster $Y \in \mathbb{R}^{N \times k}$ dengan menggunakan K-means



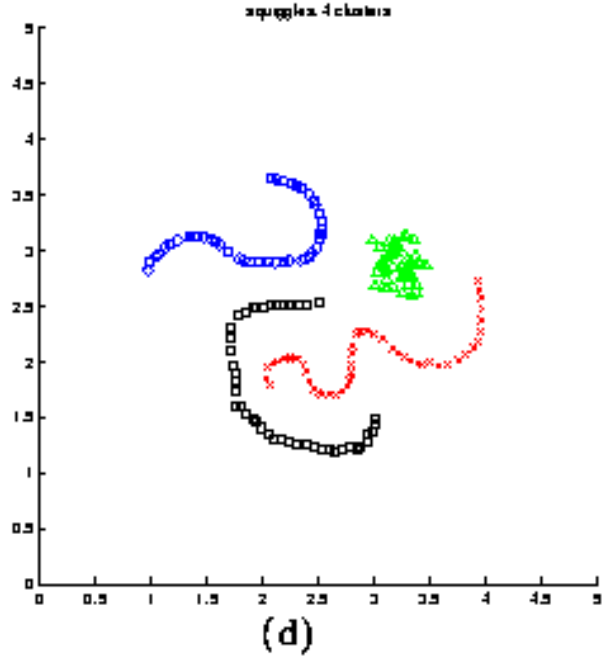
(a)



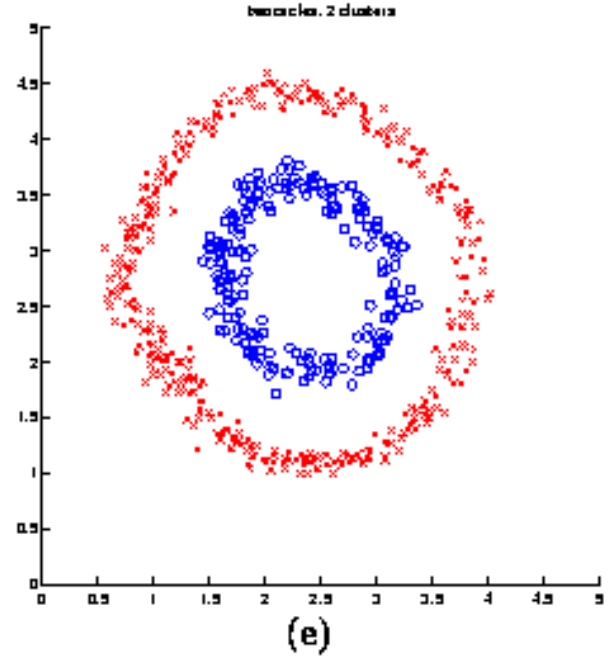
(b)



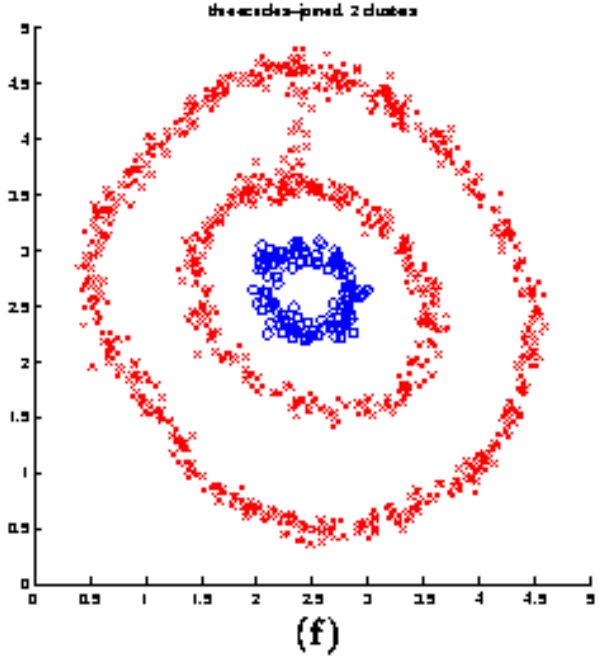
(c)



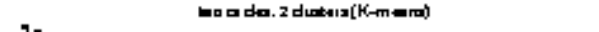
(d)



(e)



(f)

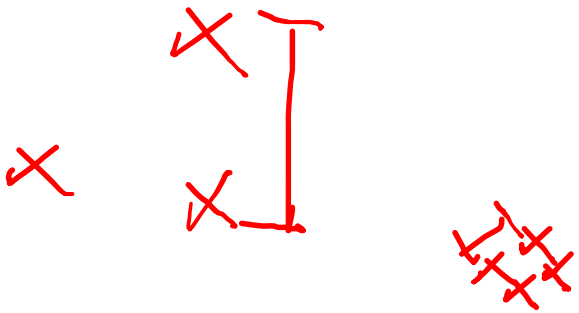


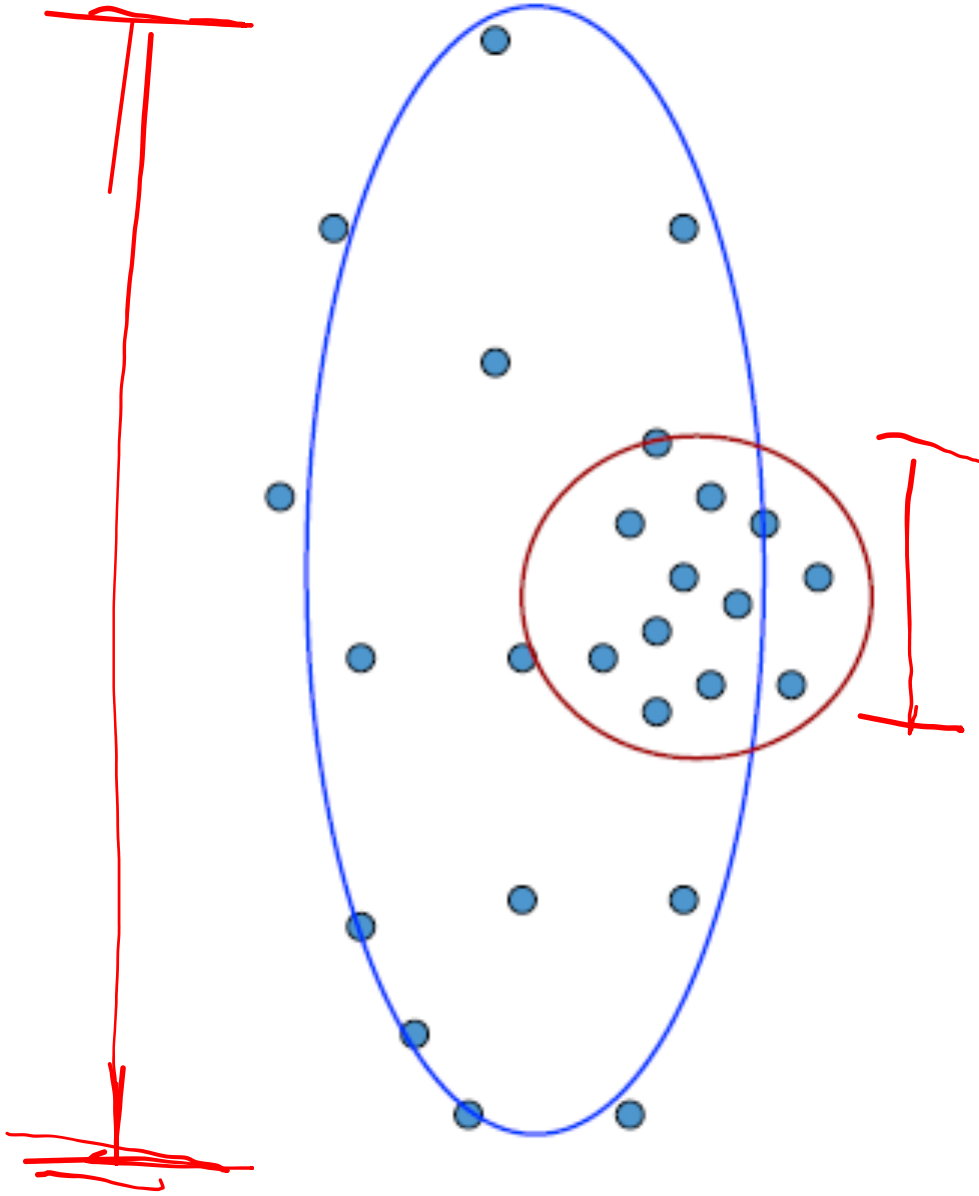
CALVIN
INSTITUTE OF TECHNOLOGY

Soft Clustering

Kelemahan Hard Clustering

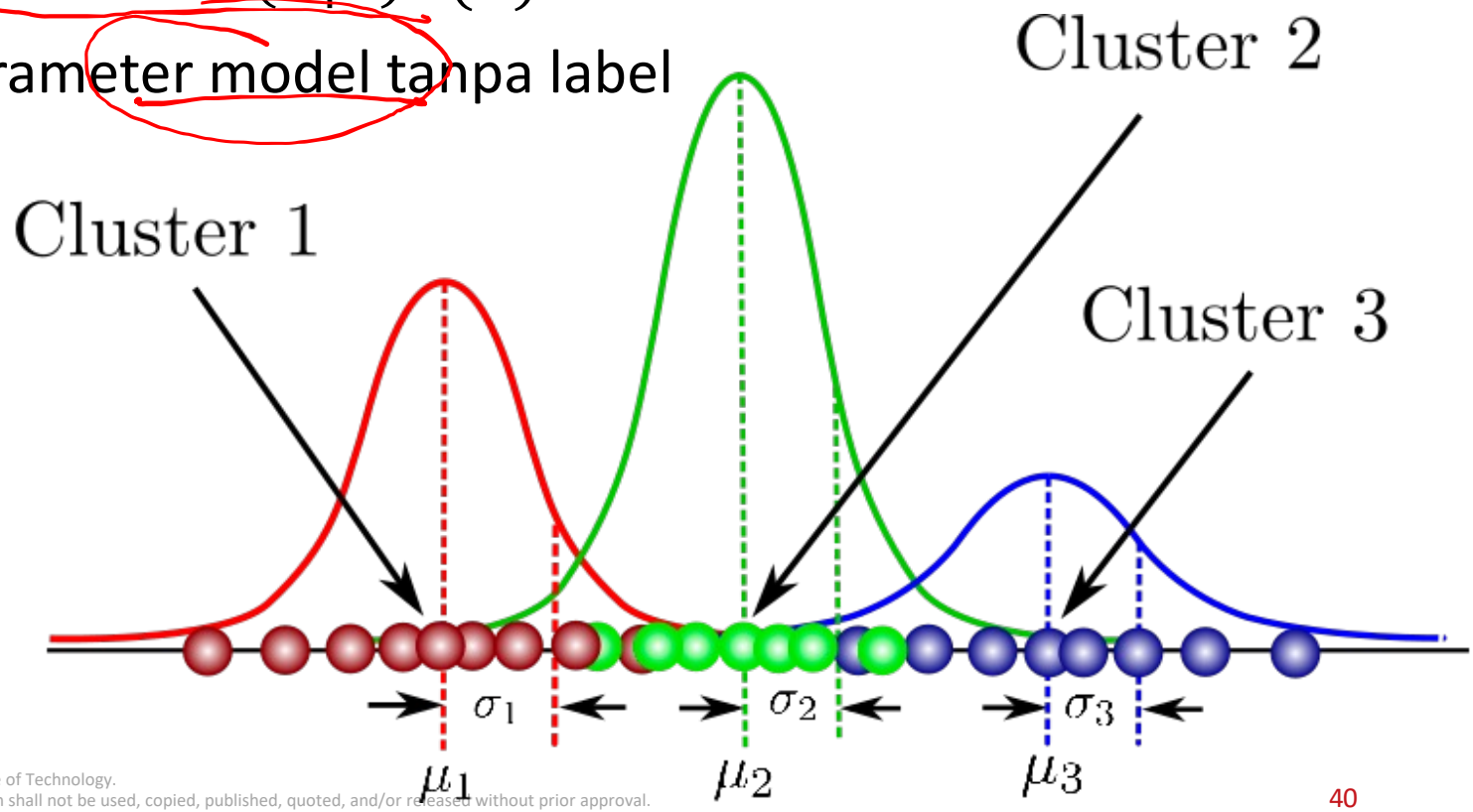
- Cluster bisa overlap
- Beberapa cluster lebih lebar dari yang lain
- Jarak dapat menipu





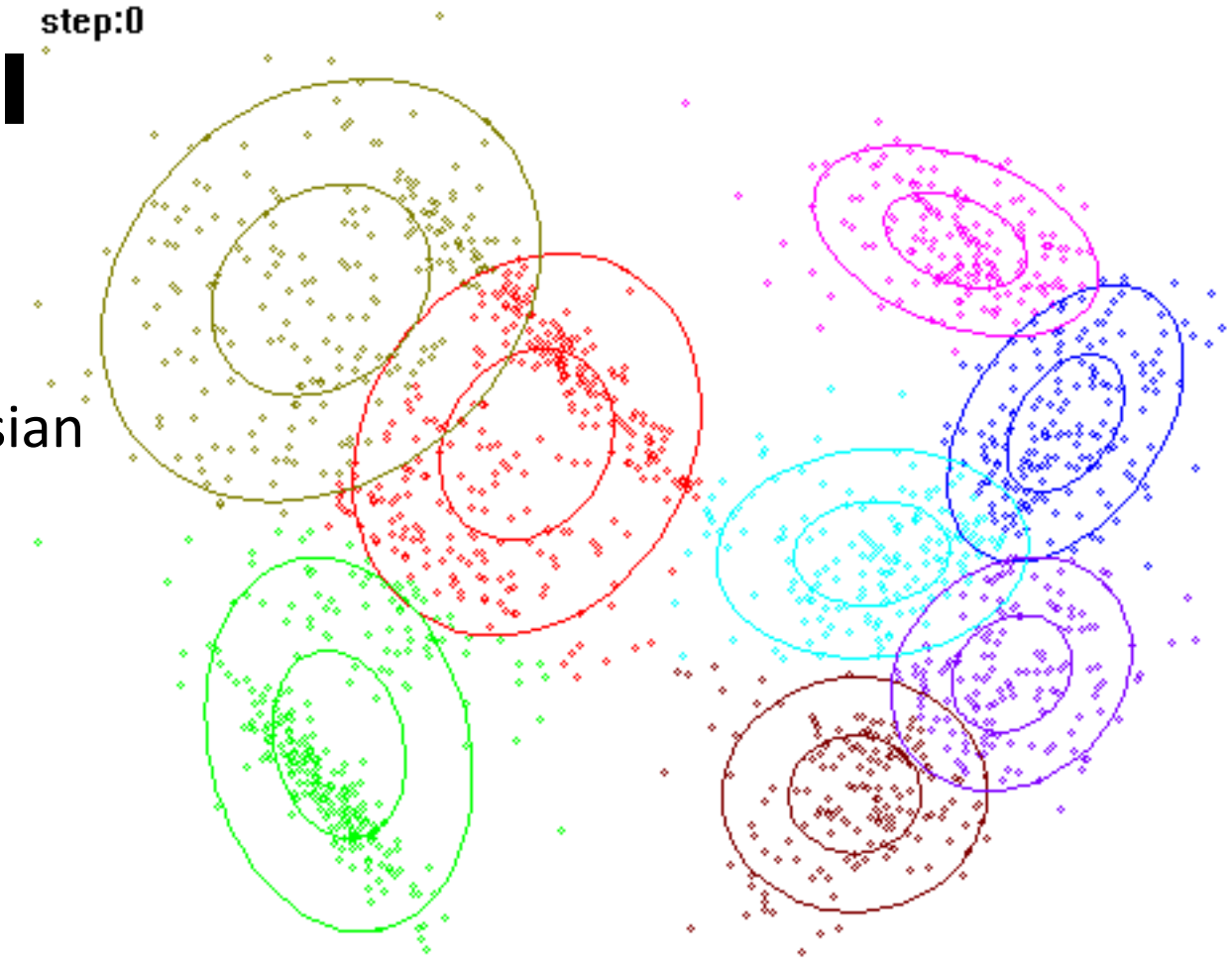
Probabilistic Clustering

- Asumsi: setiap data dapat dibentuk dari mixture beberapa clusters dengan distribusi gaussian yang parameternya dapat dipelajari
- Mengijinkan overlaps clusters berbeda ukuran
- Dapat digunakan untuk generative model: $P(X|Y)P(Y)$
- Challenge: bagaimana estimasi parameter model tanpa label
- Contoh: GMM, PSLI, LDA



Gaussian Mixture Model

- $P(Y)$: terdapat k komponen
- $P(X|Y)$: masing-masing komponen menghasilkan data dari multivariate gaussian dengan mean μ_i dan covariance Σ_i

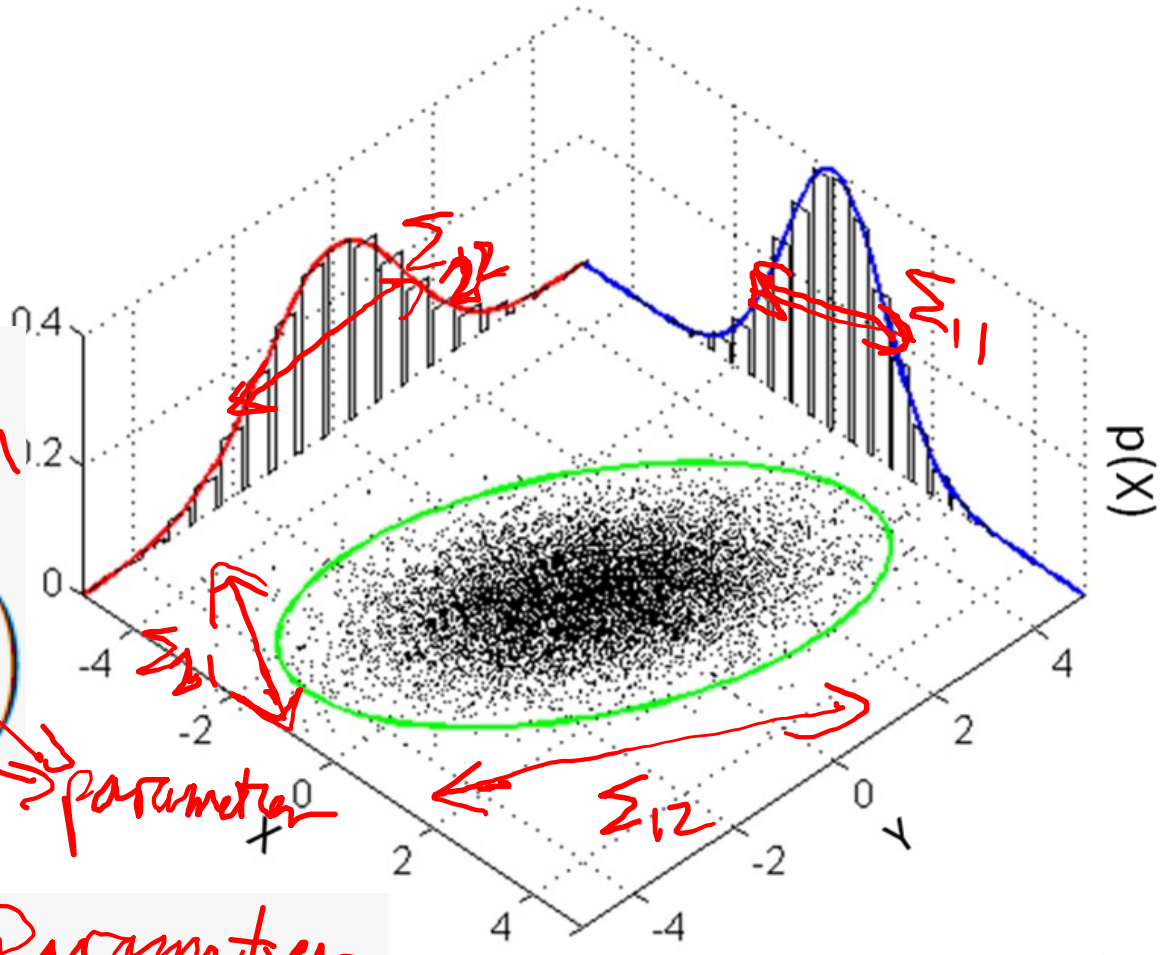


Multivariate Gaussian

- Kasus 1 dimensi

$$p(x) = \sum_{i=1}^{K} \phi_i \mathcal{N}(x \mid \mu_i, \sigma_i)$$

$$\mathcal{N}(x \mid \mu_i, \sigma_i) = \frac{1}{\sigma_i \sqrt{2\pi}} \exp\left(-\frac{(x-\mu_i)^2}{2\sigma_i^2}\right)$$



- Kasus multi dimensi

$$p(\vec{x}) = \sum_{i=1}^{K} \phi_i \mathcal{N}(\vec{x} \mid \vec{\mu}_i, \Sigma_i)$$

$$\mathcal{N}(\vec{x} \mid \vec{\mu}_i, \Sigma_i) = \frac{1}{\sqrt{(2\pi)^K |\Sigma_i|}} \exp\left(-\frac{1}{2}(\vec{x}-\vec{\mu}_i)^{\mathrm{T}} \Sigma_i^{-1} (\vec{x}-\vec{\mu}_i)\right)$$

Mixture of Gaussian

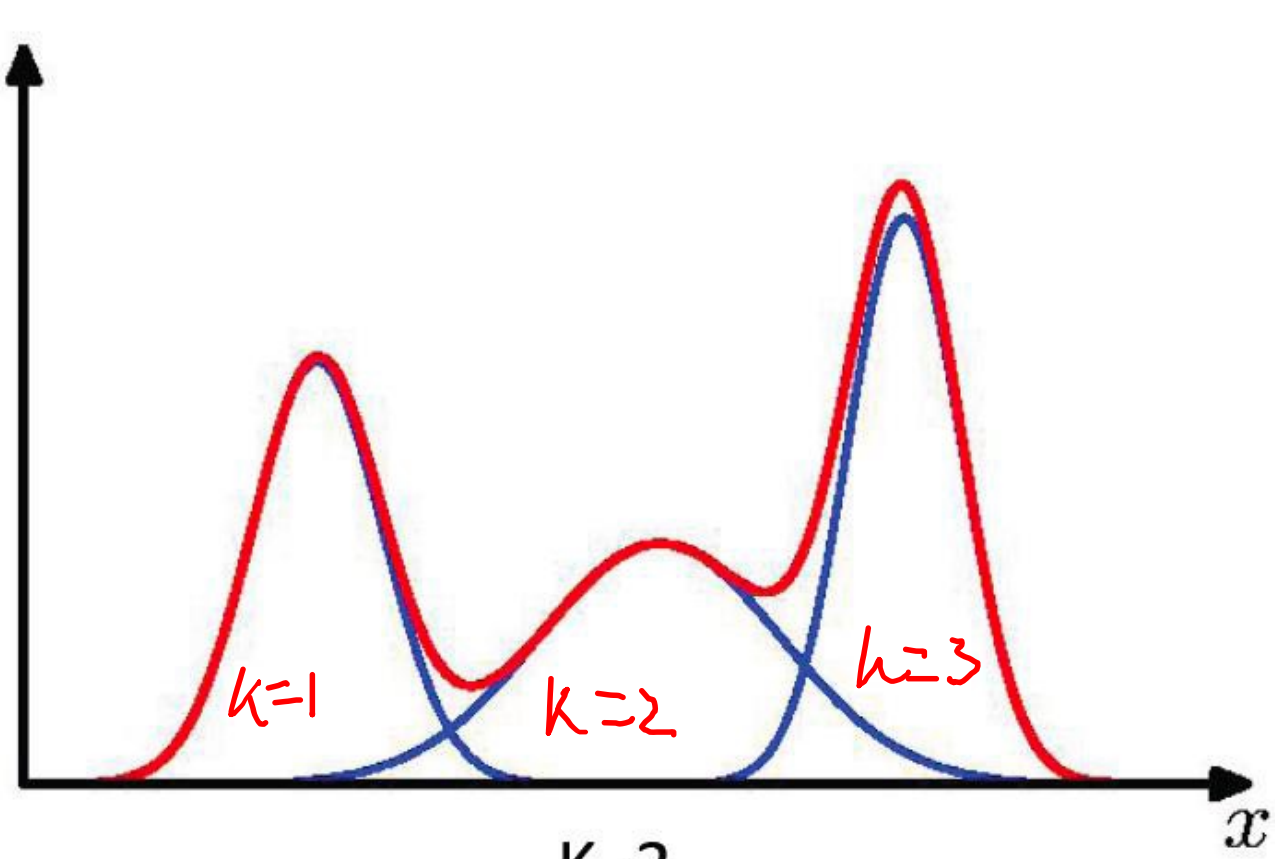
- Menjumlahkan beberapa gaussian

$$p(\mathbf{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

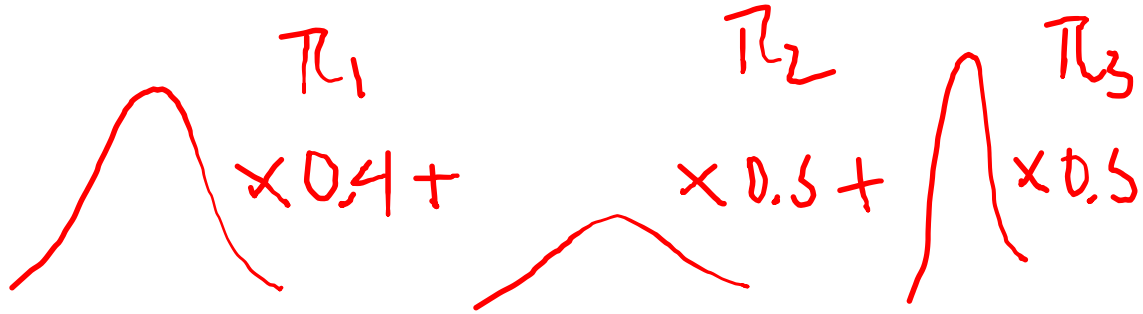
Component

Mixing coefficient

$$\forall k : \pi_k \geqslant 0 \qquad \sum_{k=1}^{K} \pi_k = 1$$



K=3

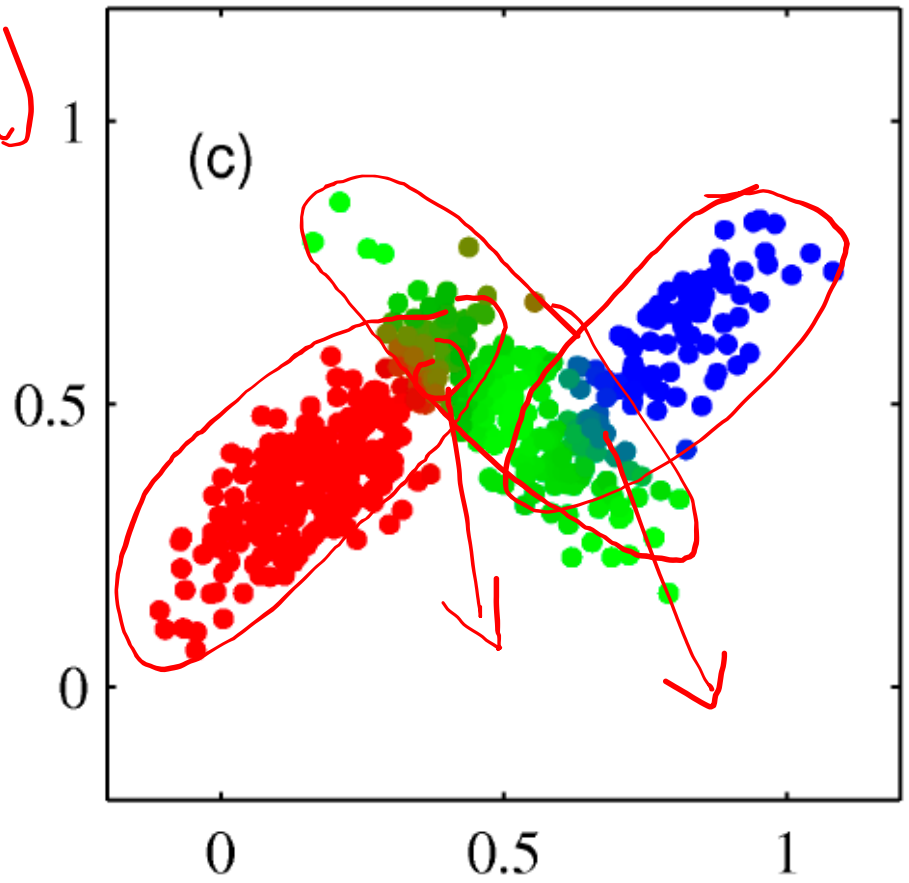


Soft Clustering

- Setiap data merupakan campuran (mixture) dari beberapa multivariate gaussians

$$p(x) = \sum_{k=1}^{3} \pi_k \mathcal{N}(x \mid \mu_k, \Sigma_k)$$

$$\sum_{k=1}^{3} \pi_k = 1$$



$D = 2$

$K = 3$

π^r, π^g, π^b

$\mu_1^r, \mu_1^g, \mu_1^b$

$\mu_2^r, \mu_2^g, \mu_2^b$

$\Sigma_{11}^r, \Sigma_{12}^r, \Sigma_{21}^r, \Sigma_{22}^r$

$\Sigma_{11}^g, \Sigma_{12}^g, \Sigma_{21}^g, \Sigma_{22}^g$

$\Sigma_{11}^b, \Sigma_{12}^b, \Sigma_{21}^b, \Sigma_{22}^b$

CALVIN INSTITUTE OF TECHNOLOGY

EM Algorithm

- Tahap Expectation (E):
  - Hitung ekspektasi kelas setiap data

$$\hat{\gamma}_{ik} = \frac{\hat{\phi}_k \mathcal{N}(x_i \mid \hat{\mu}_k, \hat{\sigma}_k)}{\sum_{j=1}^{K} \hat{\phi}_j \mathcal{N}(x_i \mid \hat{\mu}_j, \hat{\sigma}_j)}$$

Maximization

- Tahap ~~Expectation (M)~~:
  - Hitung nilai parameter dengan ekspektasi diatas

$$\hat{\phi}_k = \sum_{i=1}^{N} \frac{\hat{\gamma}_{ik}}{N}$$

$$\hat{\mu}_k = \frac{\sum_{i=1}^{N} \hat{\gamma}_{ik} x_i}{\sum_{i=1}^{N} \hat{\gamma}_{ik}}$$

$$\hat{\sigma}_k^2 = \frac{\sum_{i=1}^{N} \hat{\gamma}_{ik} (x_i - \hat{\mu}_k)^2}{\sum_{i=1}^{N} \hat{\gamma}_{ik}}$$

EM Algorithm

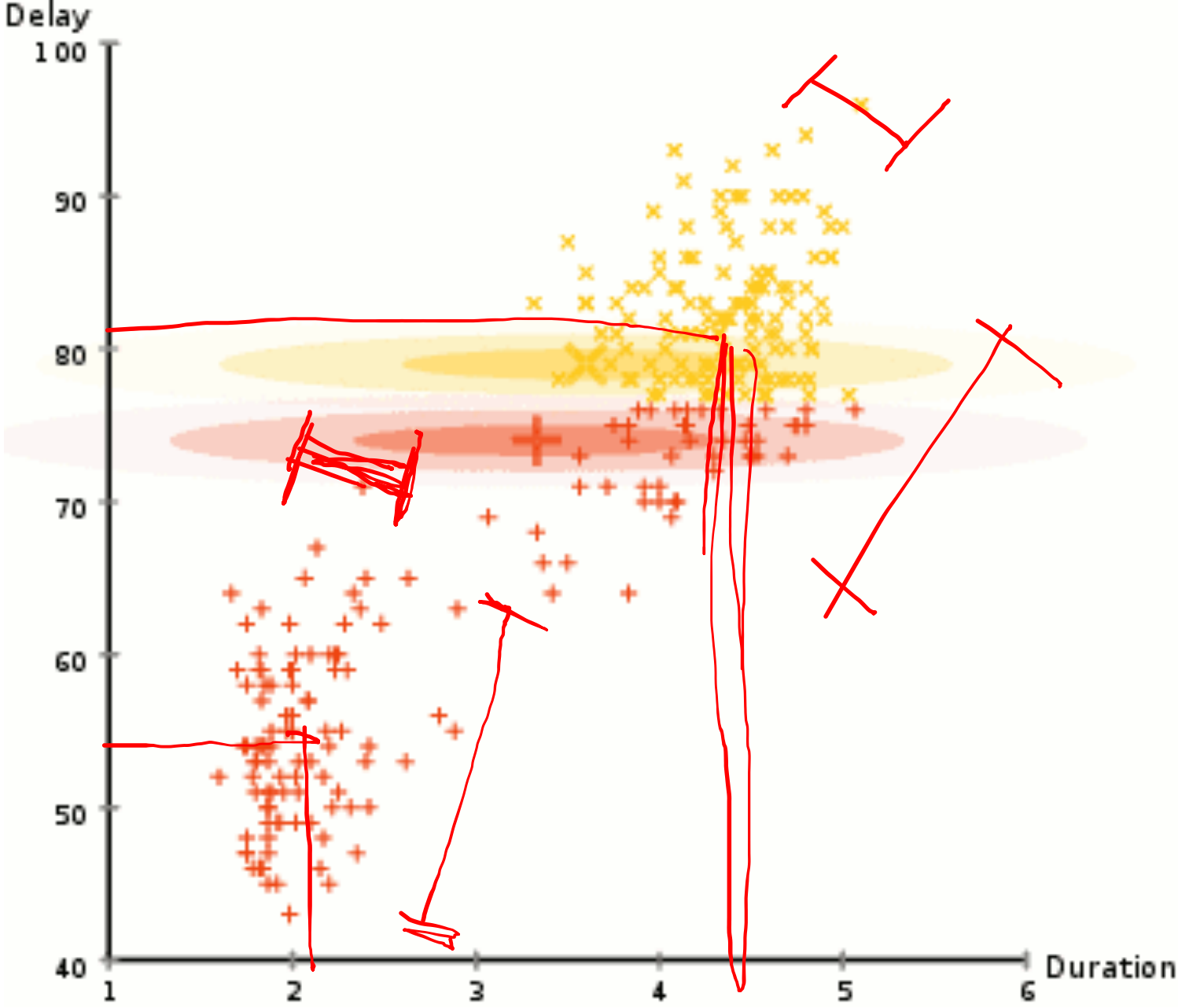
$D=2, K=2$

$\mu_1^r, \mu_2^r, \mu_1^y, \mu_2^y$

$\Sigma_{11}^r, \Sigma_{12}^r, \Sigma_{21}^r, \Sigma_{22}^r$

$\Sigma_{11}^y, \Sigma_{12}^y, \Sigma_{21}^y, \Sigma_{22}^y$

π^r, π^y



K-means sebagai kasus khusus

- Tahap Expectation (E):
  - Hitung ekspektasi kelas setiap data

$$\hat{\gamma}_{ik} = \frac{\hat{\phi}_k \mathcal{N}(x_i \mid \hat{\mu}_k, \hat{\sigma}_k)}{\sum_{j=1}^{K} \hat{\phi}_j \mathcal{N}(x_i \mid \hat{\mu}_j, \hat{\sigma}_j)}$$

- Tahap Expectation (M):
  - Hitung nilai parameter dengan ekspektasi diatas

$$\hat{\phi}_k = \sum_{i=1}^{N} \frac{\hat{\gamma}_{ik}}{N}$$

$$\hat{\mu}_k = \frac{\sum_{i=1}^{N} \hat{\gamma}_{ik} x_i}{\sum_{i=1}^{N} \hat{\gamma}_{ik}}$$

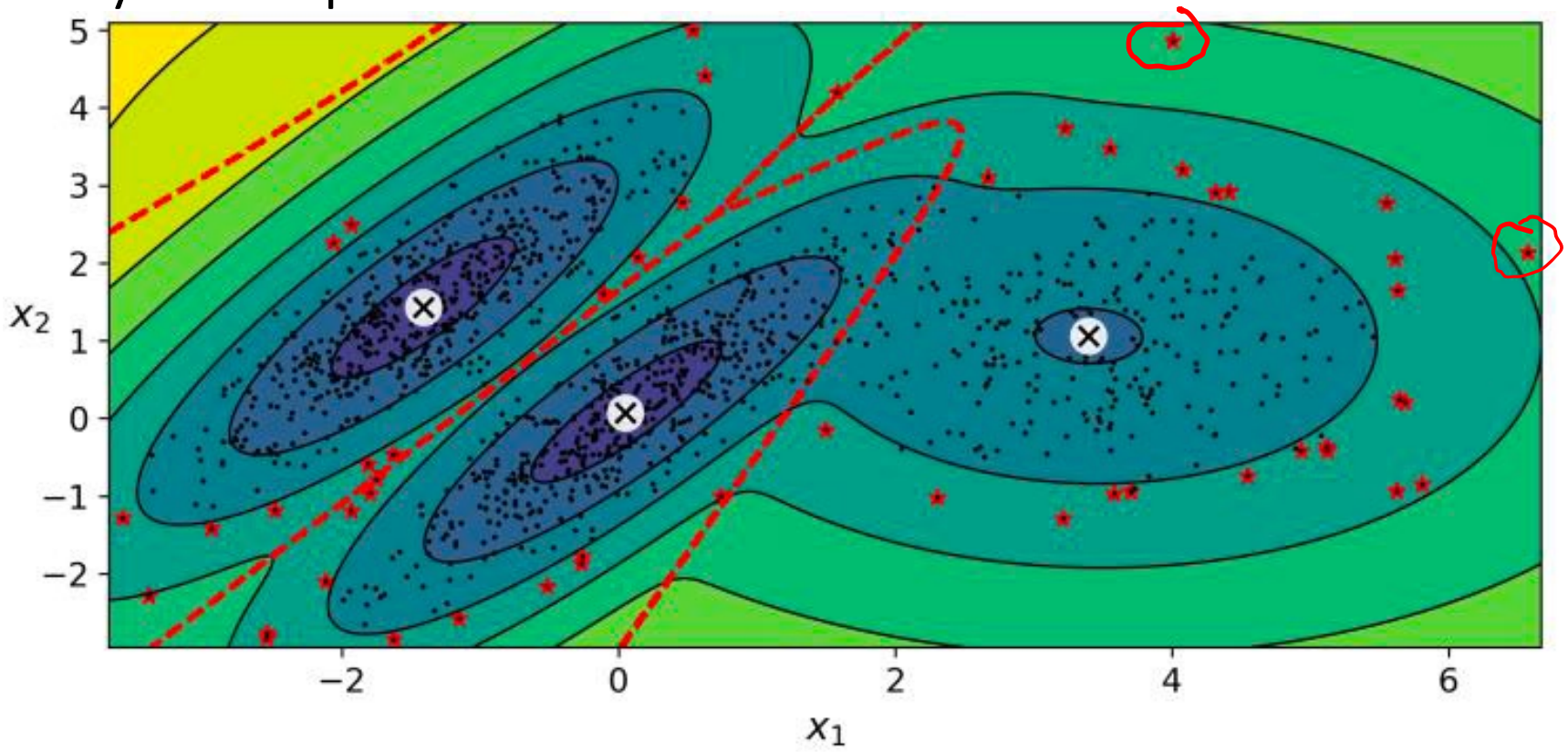
Centroid

$$\hat{\sigma}_k^2 = \frac{\sum_{i=1}^{N} \hat{\gamma}_{ik} (x_i - \hat{\mu}_k)^2}{\sum_{i=1}^{N} \hat{\gamma}_{ik}}$$

SSE

GMM untuk deteksi anomali

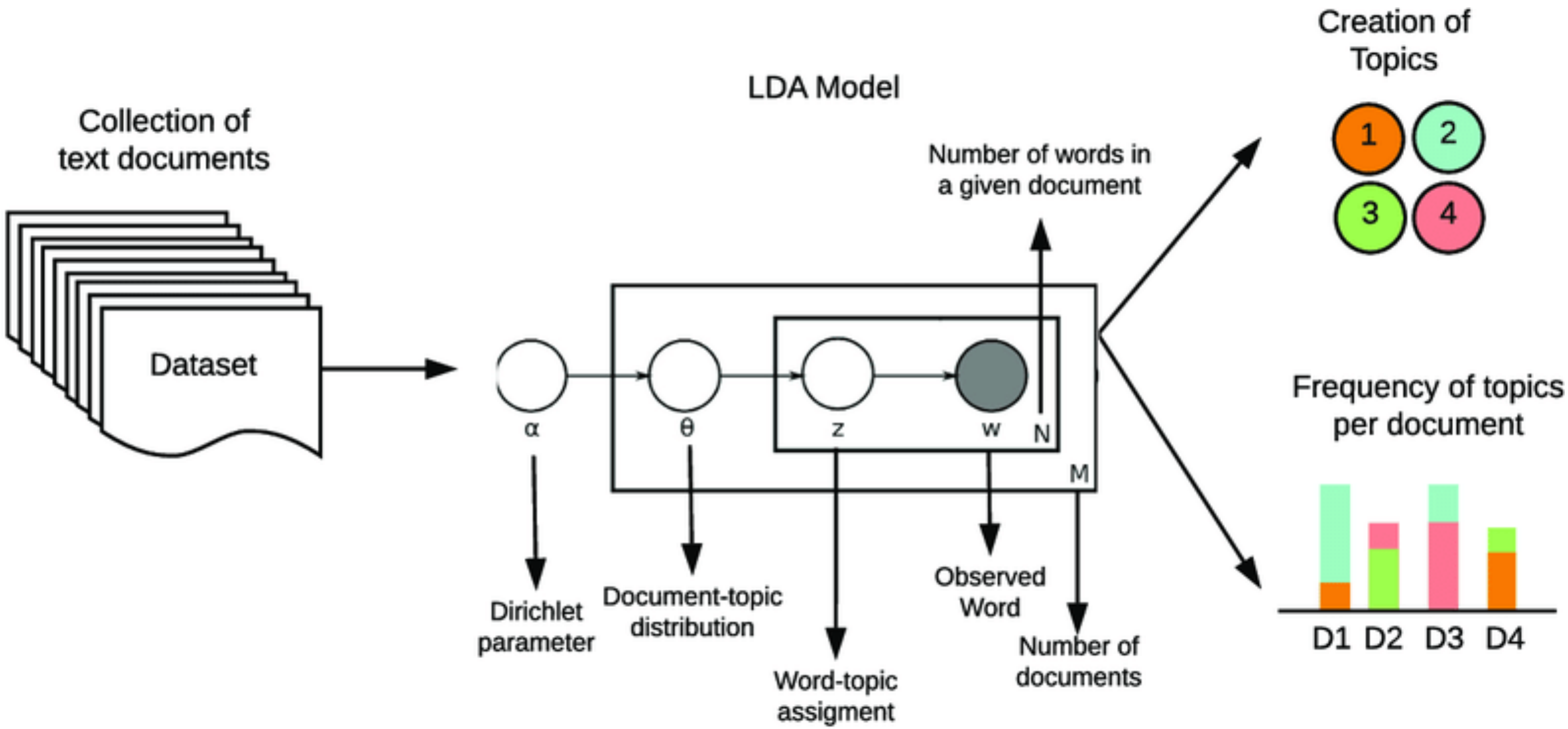
- Anomaly: data di area yang tidak padat
- Tentukan density threshold, misal: area kepadatan < 4%→ outliers
- Jika terlalu banyak false positive → turunkan threshold



CALVIN INSTITUTE OF TECHNOLOGY

Latent Dirichlet Allocation

- Latent Dirichlet Allocation (LDA) adalah algoritma machine learning dari kelompok probabilistic model untuk mempelajari topik tersembunyi dari sebuah teks
- Latent Dirichlet Allocation berbeda dengan linear discriminant analysis
- Latent Dirichlet Allocation mereduksi bag-of-words menjadi 2 matriks yang bersifat probabilistic yaitu: document-topic matrix dan topic-word matrix
- Matriks probabilistik berarti jumlah total dari setiap nilai elemen adalah satu
- https://en.wikipedia.org/wiki/Latent_Dirichlet_allocation



Topics

gene	0.04
dna	0.02
genetic	0.01
...	

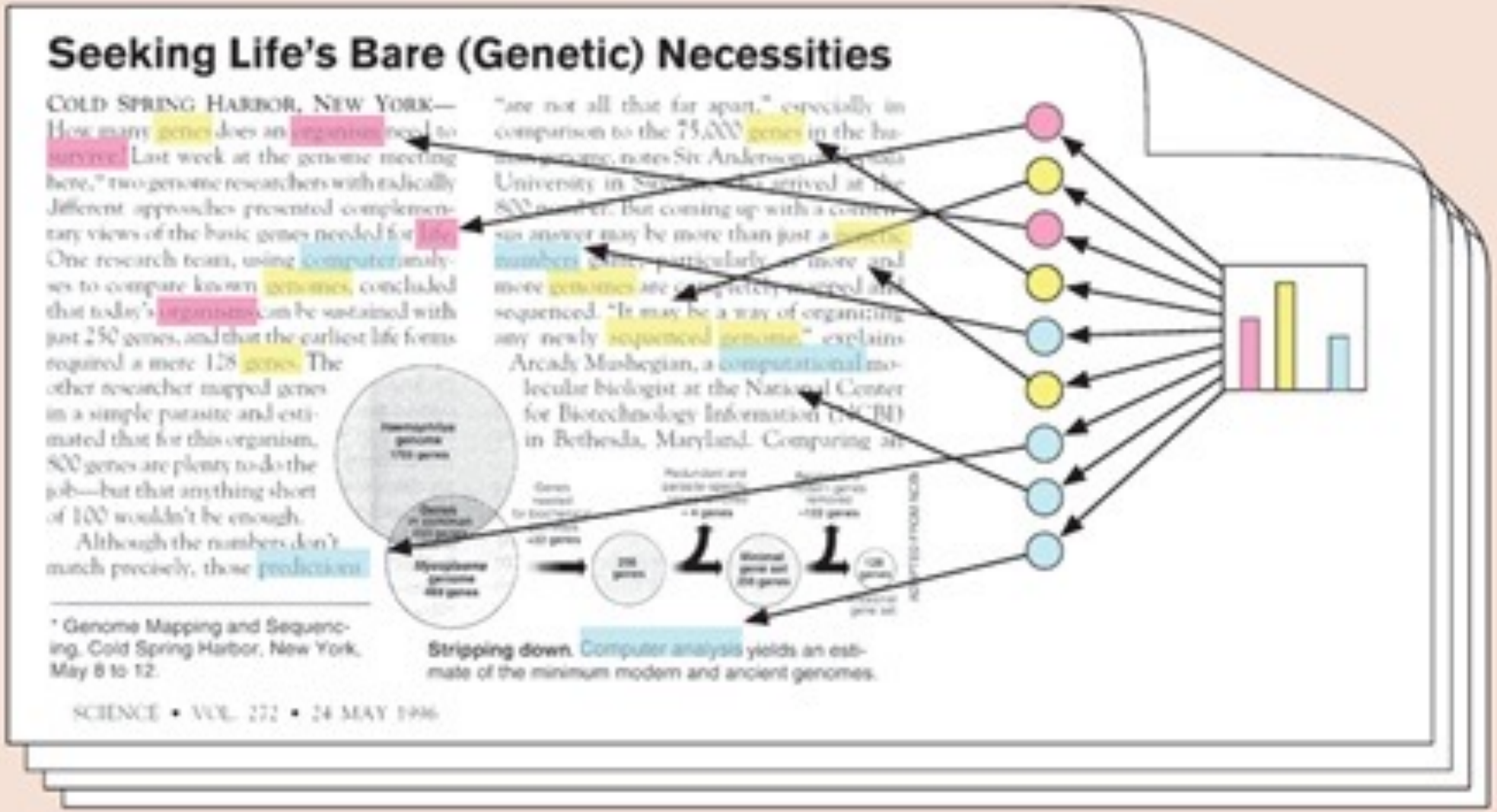
life	0.02
evolve	0.01
organism	0.01
...	

brain	0.04
neuron	0.02
nerve	0.01
...	

data	0.02
number	0.02
computer	0.01
...	

Documents

Topic proportions and assignments



Seeking Life's Bare (Genetic) Necessities

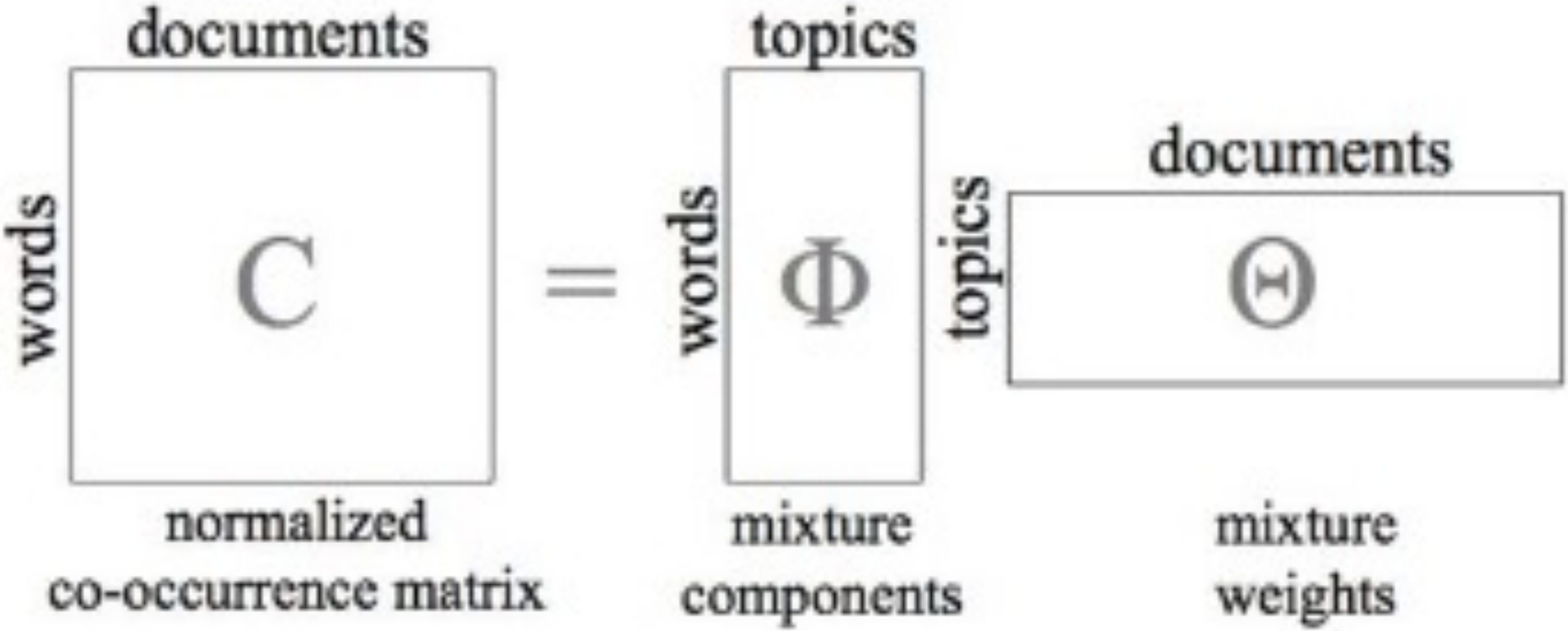
COLD SPRING HARBOR, NEW YORK—How many genes does an organism need to survive? Last week at the genome meeting here,* two genome researchers with radically different approaches presented complementary views of the basic genes needed for life. One research team, using computer analyses to compare known genomes, concluded that today's organisms can be sustained with just 250 genes, and that the earliest life forms required a mere 128 genes. The other researcher mapped genes in a simple parasite and estimated that for this organism, 800 genes are plenty to do the job—but that anything short of 100 wouldn't be enough.

Although the numbers don't match precisely, those predictions "are not all that far apart," especially in comparison to the 75,000 genes in the human genome, notes Sir Anderson [illegible] University in [illegible] arrived at the 800 number. But coming up with a concrete answer may be more than just a genetic numbers game, particularly as more and more genomes are completely mapped and sequenced. "It may be a way of organizing any newly sequenced genome," explains Arcady Mushegian, a computational molecular biologist at the National Center for Biotechnology Information (NCBI) in Bethesda, Maryland. Comparing an

* Genome Mapping and Sequencing, Cold Spring Harbor, New York, May 8 to 12.

Stripping down. Computer analysis yields an estimate of the minimum modern and ancient genomes.

SCIENCE • VOL. 272 • 24 MAY 1996


documents
words
C
normalized
co-occurrence matrix
=
topics
words
Φ
mixture
components
topics
documents
Θ
mixture
weights

Topic 1	Topic 2	Topic 3	Topic 4
"Technology"	"Entertainment"	"Health"	"U.S. Politics"
company	film	drug	republican
mobile	show	drugs	house
technology	music	cancer	senate
facebook	year	fda	president
google	television	patients	state
apple	singer	reuters	republicans
online	years	disease	political
industry	movie	treatment	campaign
video	band	virus	party
business	actor	health	democratic

Tuhan Memberkati

God's People for God's Glory
CALVIN
INSTITUTE OF TECHNOLOGY