Clustering: Models and Algorithms

Shikui Tu

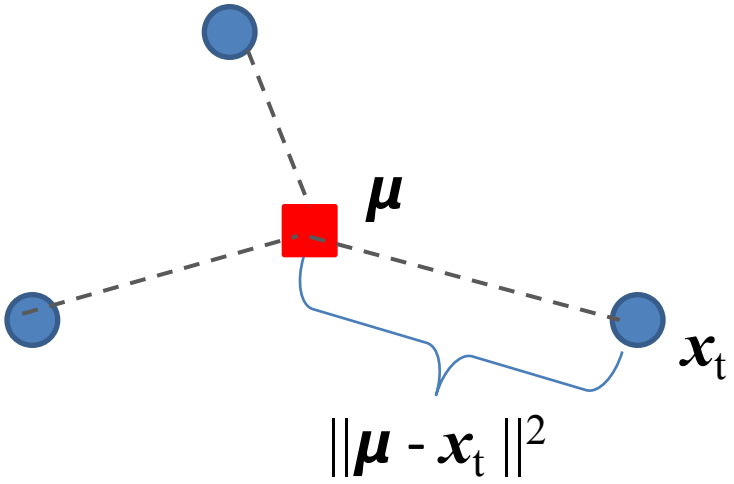
2019-02-28

Outline

- Clustering
  - K-mean clustering, hierarchical clustering
- Adaptive learning (online learning)
  - CL, FSCL, RPCL
- Gaussian Mixture Models (GMM)
- Expectation-Maximization (EM) for maximum likelihood

How to define error?

Square distance:

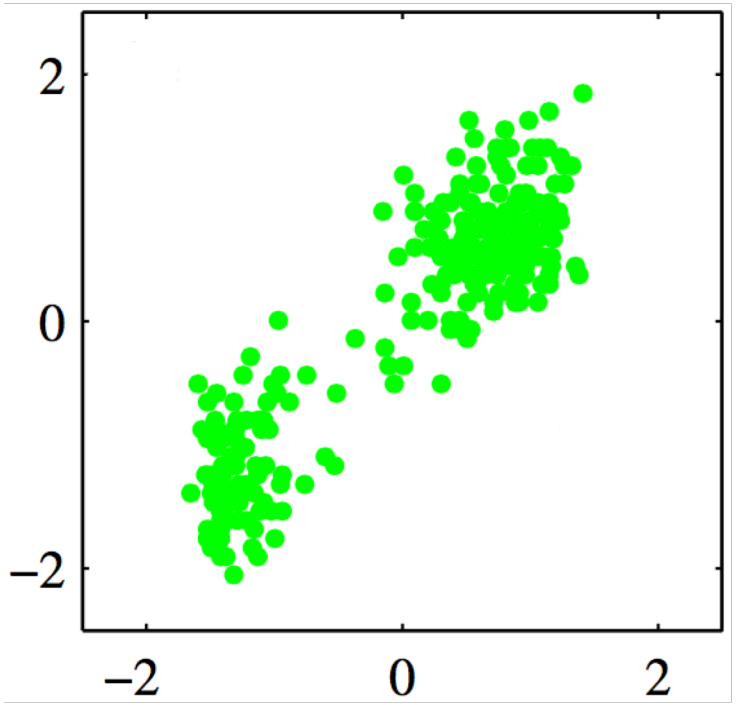


$$||\boldsymbol{\mu} - \boldsymbol{x}_1||^2 + ||\boldsymbol{\mu} - \boldsymbol{x}_2||^2 + ||\boldsymbol{\mu} - \boldsymbol{x}_3||^2$$

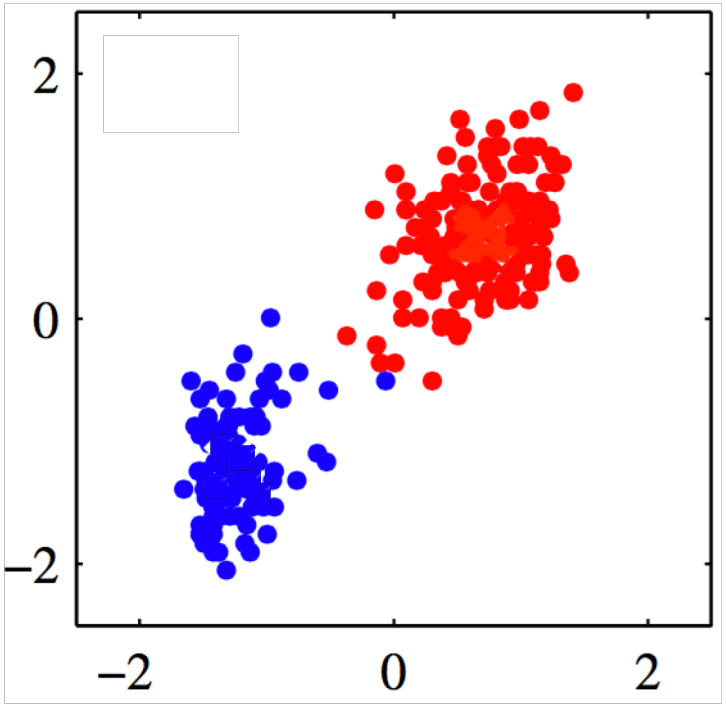
可以证明：当$\boldsymbol{\mu}$是所有数据点的均值时，平方距离和最小

Clustering the data

We have the following data:



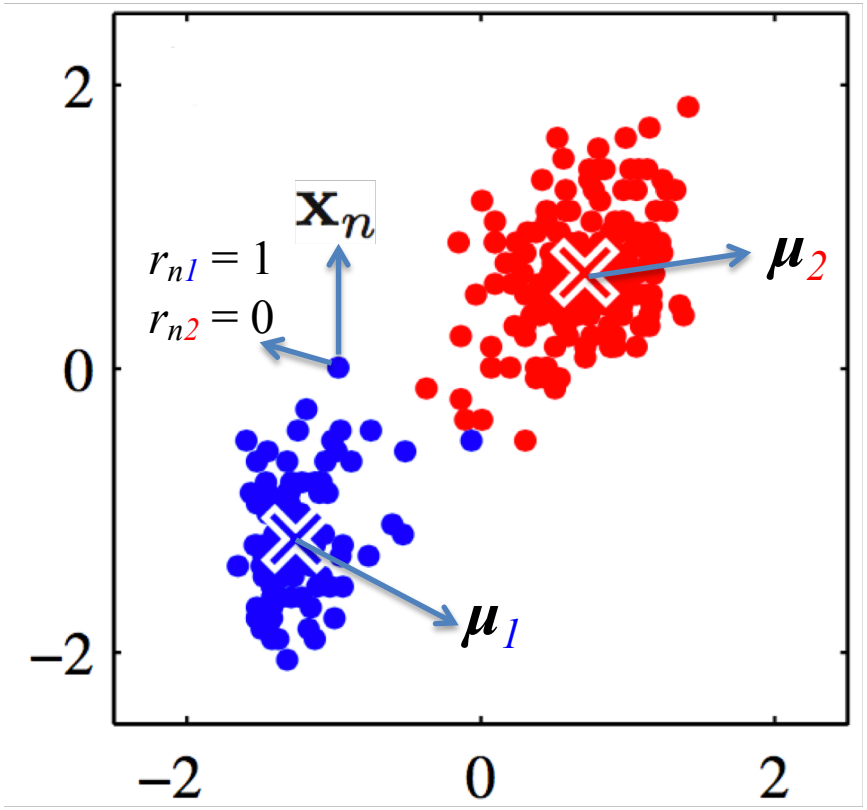
We want to cluster the data into two clusters (red and blue)



How?

Minimize the sum of square distances J

minimize $$J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} \|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2$$



$r_{nk} = 1$ if and only if data point $\mathbf{x}_n$ is assigned to cluster k;
otherwise $r_{nk} = 0$.

$k = 1, 2$; K = 2 clusters

$n = 1, \ldots, N$;
N: the total number of points.

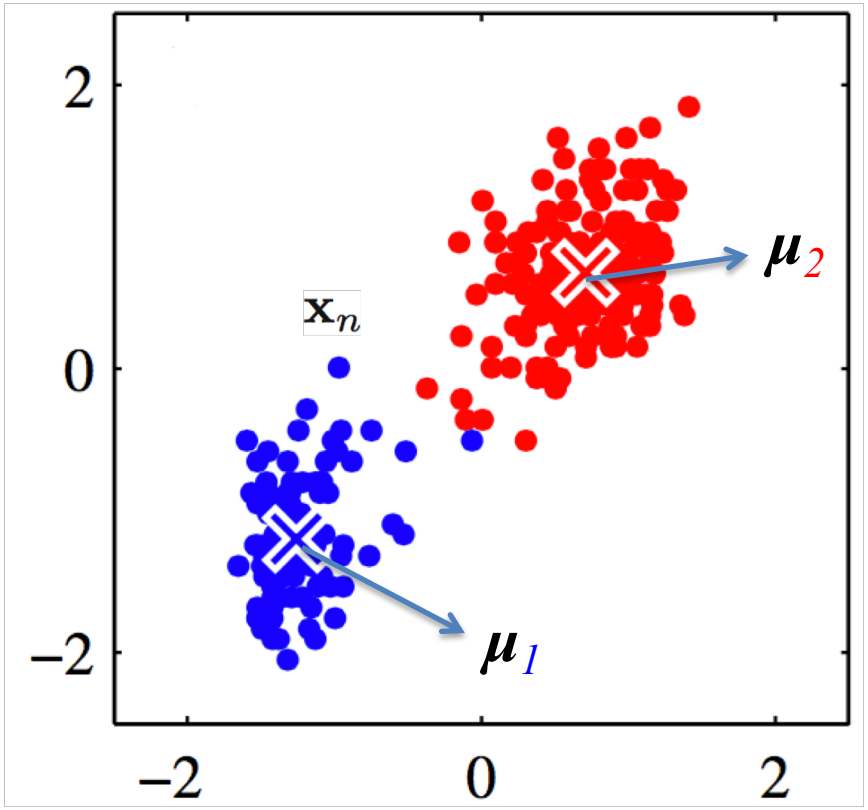
We need to calculate $\{ r_{nk} \}$ and $\{ \boldsymbol{\mu}_k \}$.

If we know r_{n1} , r_{n2} for all n=1,…,N

Since the points have been assigned to cluster 1 or cluster 2, we calculate

$\boldsymbol{\mu}_1$ = mean of the points in cluster 1

$\boldsymbol{\mu}_2$ = mean of the points in cluster 2



Or formally

$$\boldsymbol{\mu}_k = \frac{\sum_n r_{nk}\mathbf{x}_n}{\sum_n r_{nk}}$$

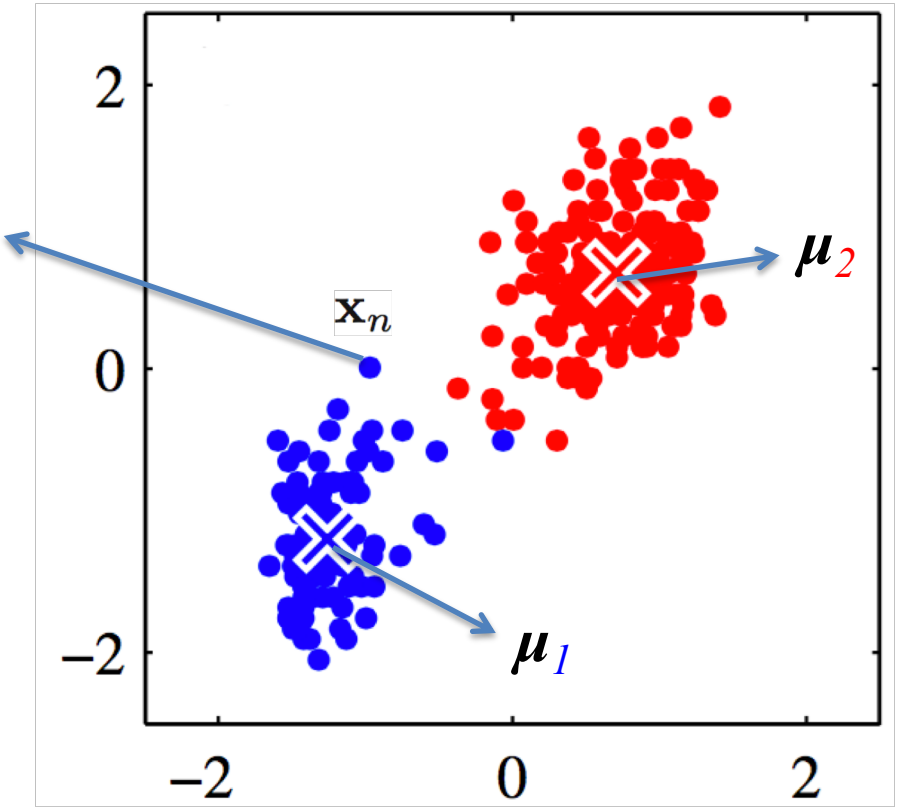
We call it the **M Step.**

If we know $\boldsymbol{\mu}_1, \boldsymbol{\mu}_2$

We should assign point $\mathbf{x}_n$ to cluster 1, because

$$\| \mathbf{x}_n - \boldsymbol{\mu}_1 \|^2 < \| \mathbf{x}_n - \boldsymbol{\mu}_2 \|^2$$

Then, $r_{n1} = 1$
$r_{n2} = 0$

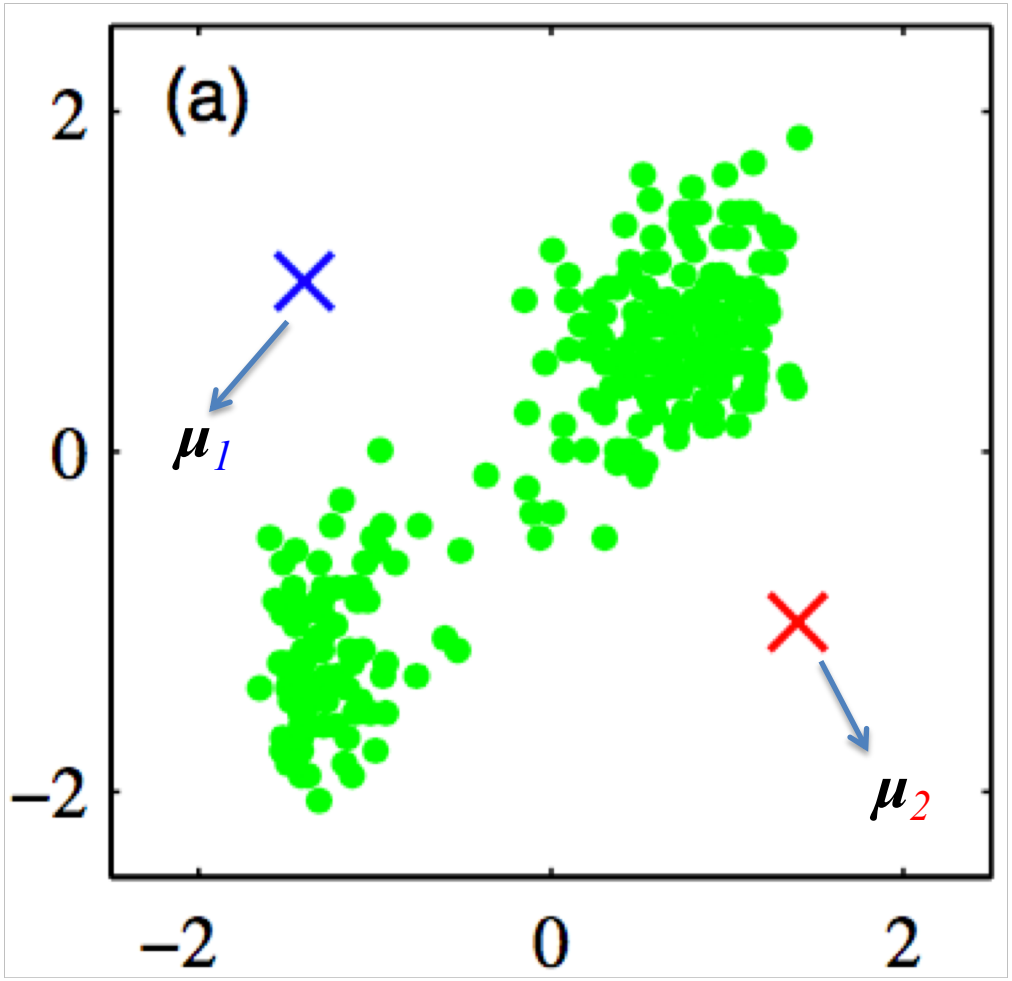


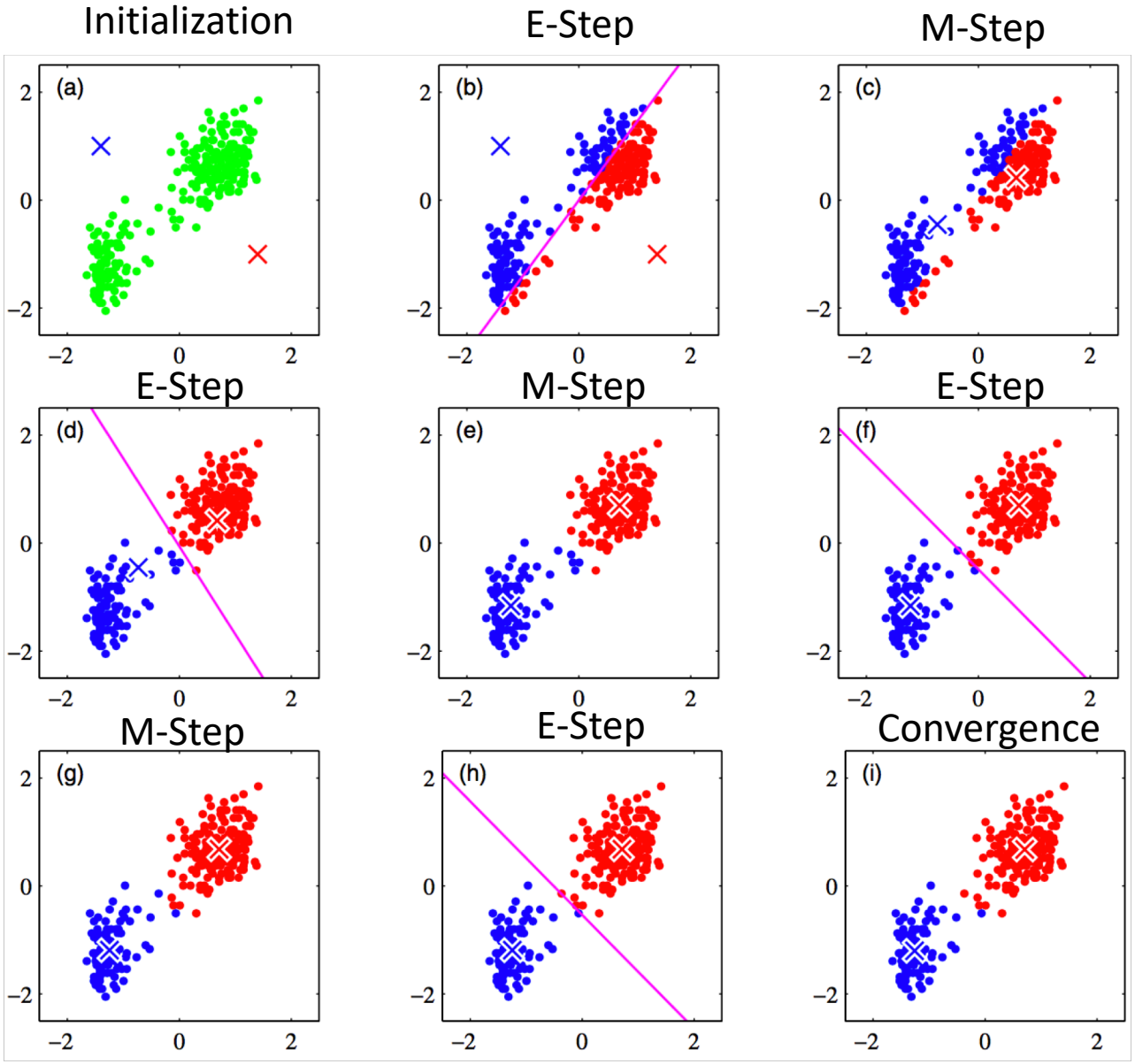
Or formally

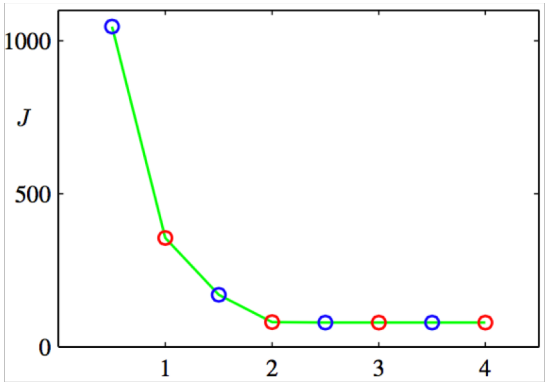
$$r_{nk} = \begin{cases} 1 & \text{if } k = \arg\min_j \|\mathbf{x}_n - \boldsymbol{\mu}_j\|^2 \\ 0 & \text{otherwise.} \end{cases}$$

We call it the **E Step**

Initialization







If J does not change, or $\{ \boldsymbol{\mu}_1, \boldsymbol{\mu}_2 \}$ do not change, then the algorithm converges.

K均值法小结

- 初始化均值点$\boldsymbol{\mu}_1,\ldots,\boldsymbol{\mu}_k$
- 迭代如下
  - 把每个数据点按照就近原则分配给相应的$\boldsymbol{\mu}_i$
  - 把$\boldsymbol{\mu}_i$更新为所分配的数据点的均值
- 迭代停止，如果聚类分配不变

Initialize $\boldsymbol{m}_i, i = 1, \ldots, k$, for example, to k random $\boldsymbol{x}^t$

Repeat

For all $\boldsymbol{x}^t \in \mathcal{X}$

$$b_i^t \leftarrow \begin{cases} 1 & \text{if } \|\boldsymbol{x}^t - \boldsymbol{m}_i\| = \min_j \|\boldsymbol{x}^t - \boldsymbol{m}_j\| \\ 0 & \text{otherwise} \end{cases}$$

For all $\boldsymbol{m}_i, i = 1, \ldots, k$

$$\boldsymbol{m}_i \leftarrow \sum_t b_i^t \boldsymbol{x}^t / \sum_t b_i^t$$

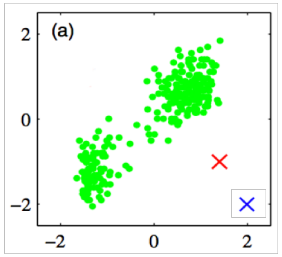
Until $\boldsymbol{m}_i$ converge

Basic ingredients

- Model or structure
- Objective function
- Algorithm
- Convergence

Questions for K-mean algorithm

- Does it find the global optimum of J?
  - No, the nearest local optimum, depending on initialization



- If Euclidean distance is not good for some data, do we have other choices?

- Can we assign each data point to the clusters probabilistically?

- If K (the total number of clusters) is unknown, can we estimate it from the data?

Outline

- Clustering
  - K-mean clustering, **hierarchical clustering**
- Adaptive learning (online learning)
  - CL, FSCL, RPCL
- Gaussian Mixture Models (GMM)
- Expectation-Maximization (EM) for maximum likelihood

Hierarchical Clustering

- *k*-means clustering requires
  - *k*
  - Positions of initial centers
  - A distance measure between points (*e.g.* Euclidean distance)
- Hierarchical clustering requires a measure of distance between *groups* of data points

Adapted from Blei, D. *Hierarchial Cluster* [PwerPoint slides]. www.cs.princeton.edu/courses/archive/spr08/cos424/slides/clustering-2.pdf

Hierarchical Clustering

- Agglomerative clustering
- A very simple procedure:
  - Assign each data point into its own group
  - Repeat: look for the two closest groups and merge them into one group
  - Stop when all the data points are merged into a single cluster

Adapted from Blei, D. *Hierarchial Cluster* [PwerPoint slides]. www.cs.princeton.edu/courses/archive/spr08/cos424/slides/clustering-2.pdf

Distance Measure

- Distance between data points *a* and *b*:
  - $d(a, b)$
- Group A and B
  - Single-linkage

$$d(A, B) = \min_{a \in A, b \in B} d(a, b)$$

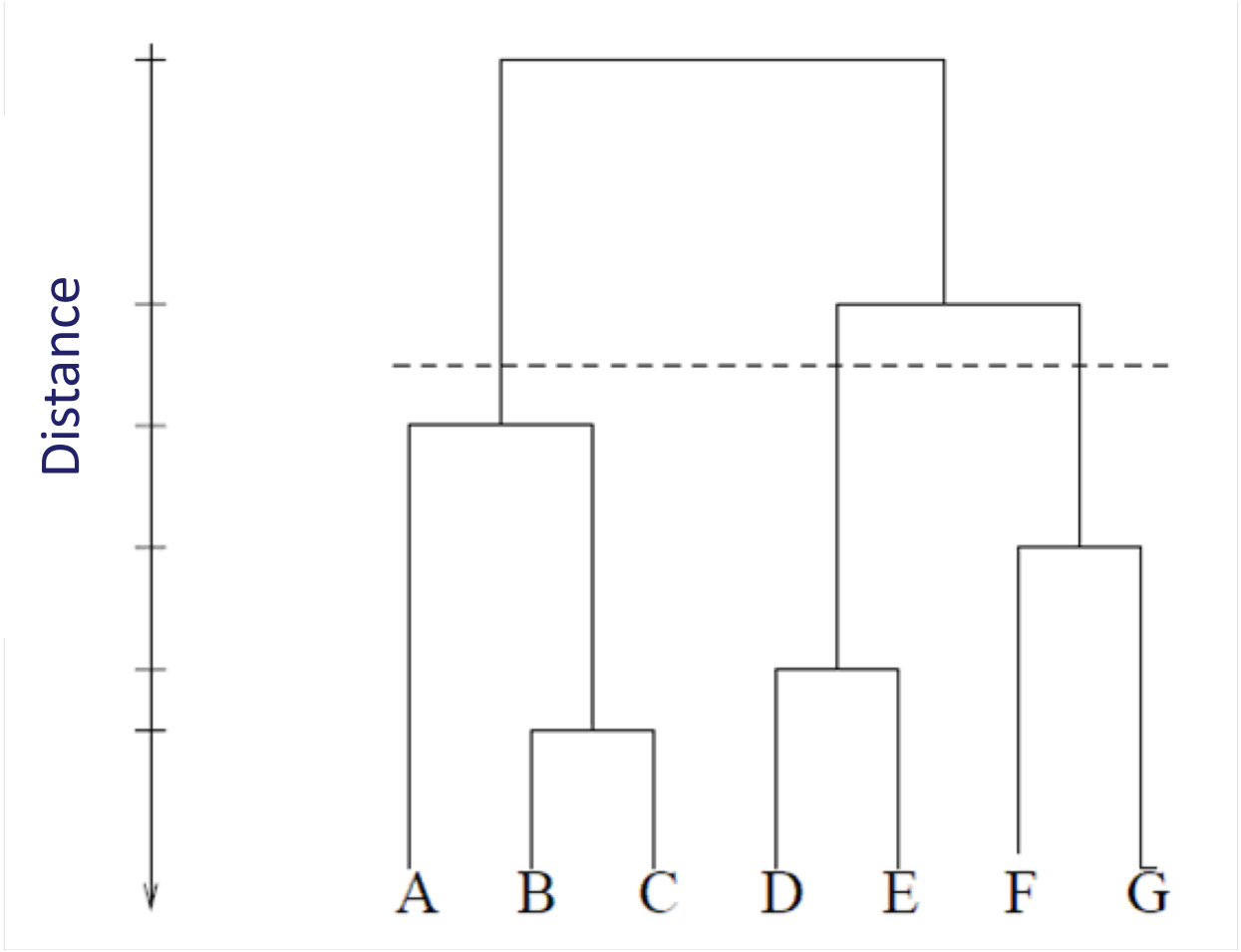
  - Complete-linkage

$$d(A, B) = \max_{a \in A, b \in B} d(a, b)$$

  - Average-linkage

$$d(A, B) = \frac{\sum_{a \in A, b \in B} d(a, b)}{|A| \cdot |B|}$$

Dendrogram



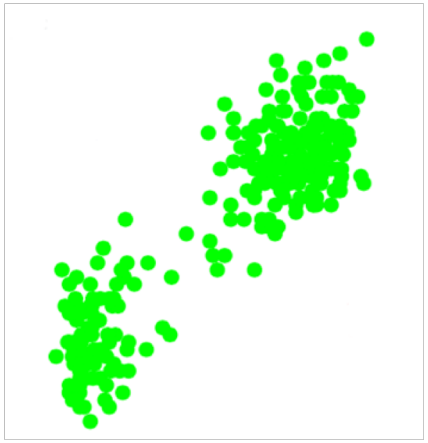
Jain, A. K., Murty, M. N., Flynn, P. J. (1999) "Data Clustering: A Review". ACM Computing Surveys (CSUR), 31(3), p.264-323, 1999.

Outline

- Clustering
  - K-mean clustering, hierarchical clustering
- Adaptive learning (online learning)
  - CL, FSCL, RPCL
- Gaussian Mixture Models (GMM)
- Expectation-Maximization (EM) for maximum likelihood

From batch to adaptive

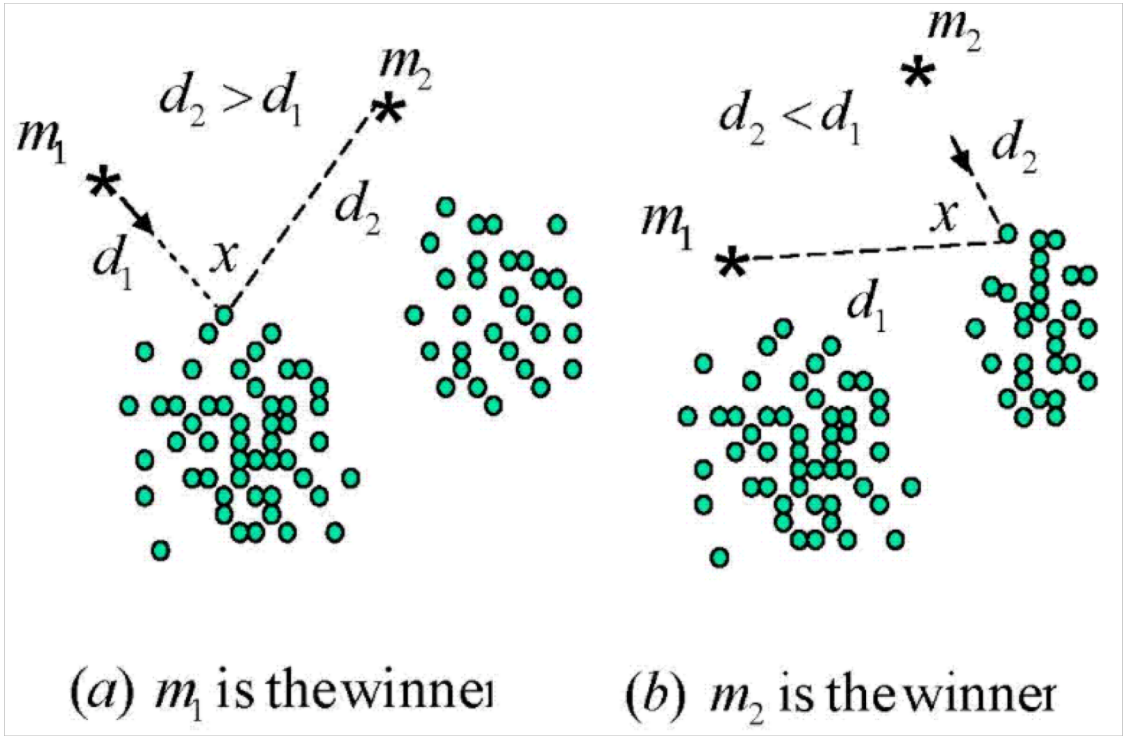
- Given a batch of data points



- Data points come one by one:

Competitive learning

- Data points come one by one:



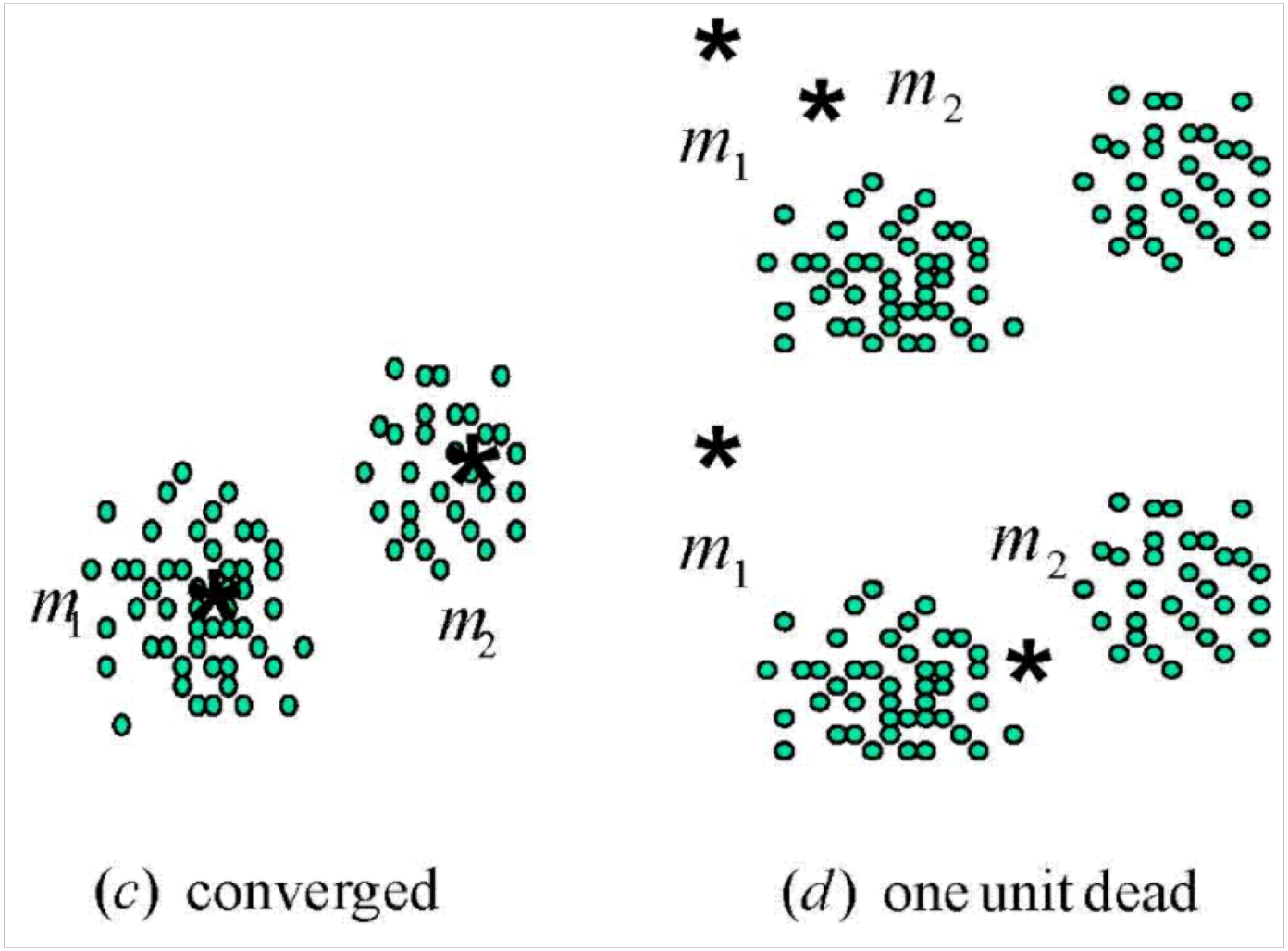
$$\varepsilon_t(\theta_j) = \|x_t - m_j\|^2$$

$$p_{j,t} = \begin{cases} 1, & \text{if } j = c, \\ 0, & \text{otherwise;} \end{cases}$$

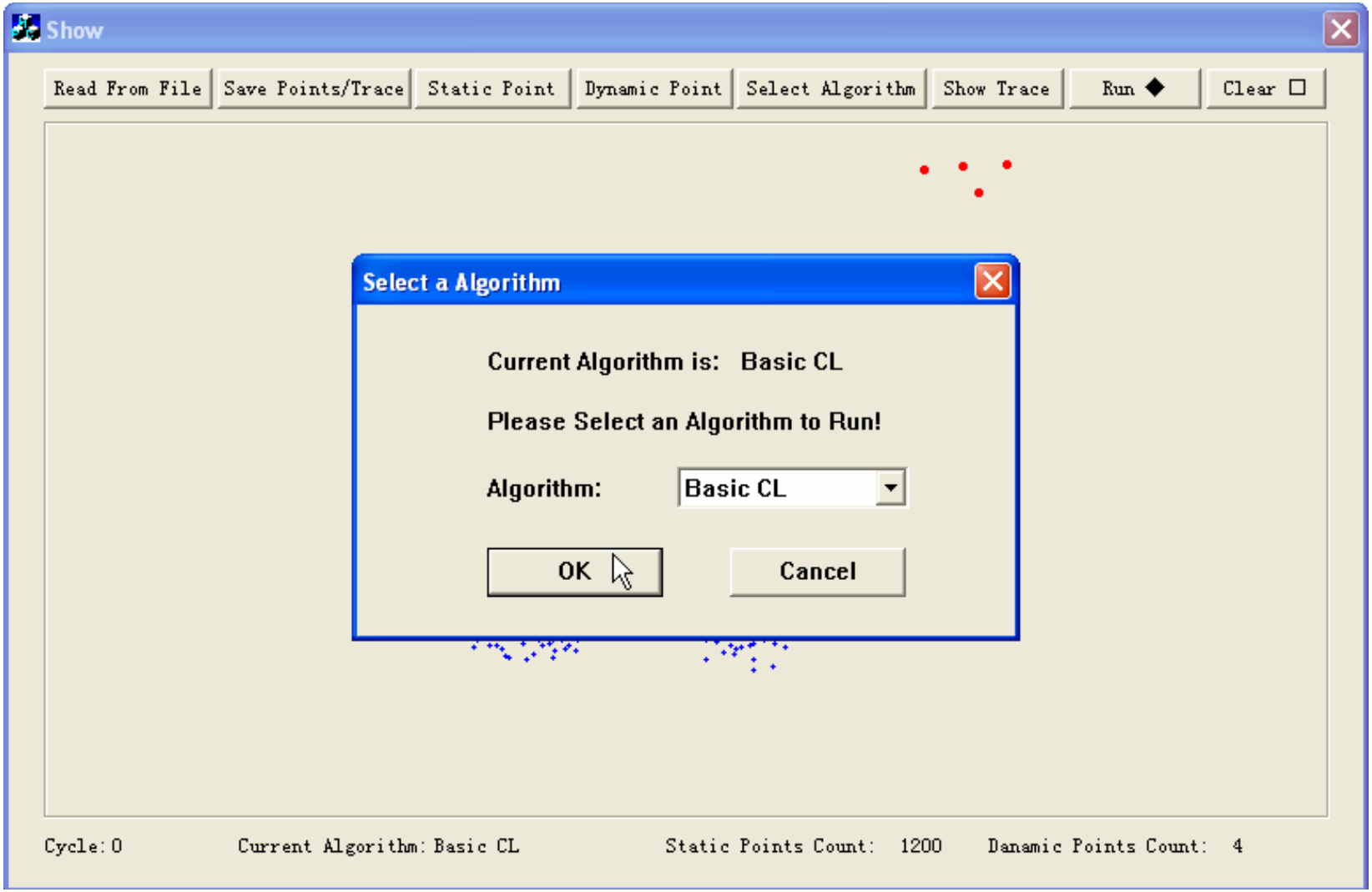
$$c = arg\ min_j \varepsilon_t(\theta_j).$$

$$m_j^{new} = m_j^{old} + \eta p_{j,t}(x_t - m_j^{old}).$$

When starting with “bad initializations”



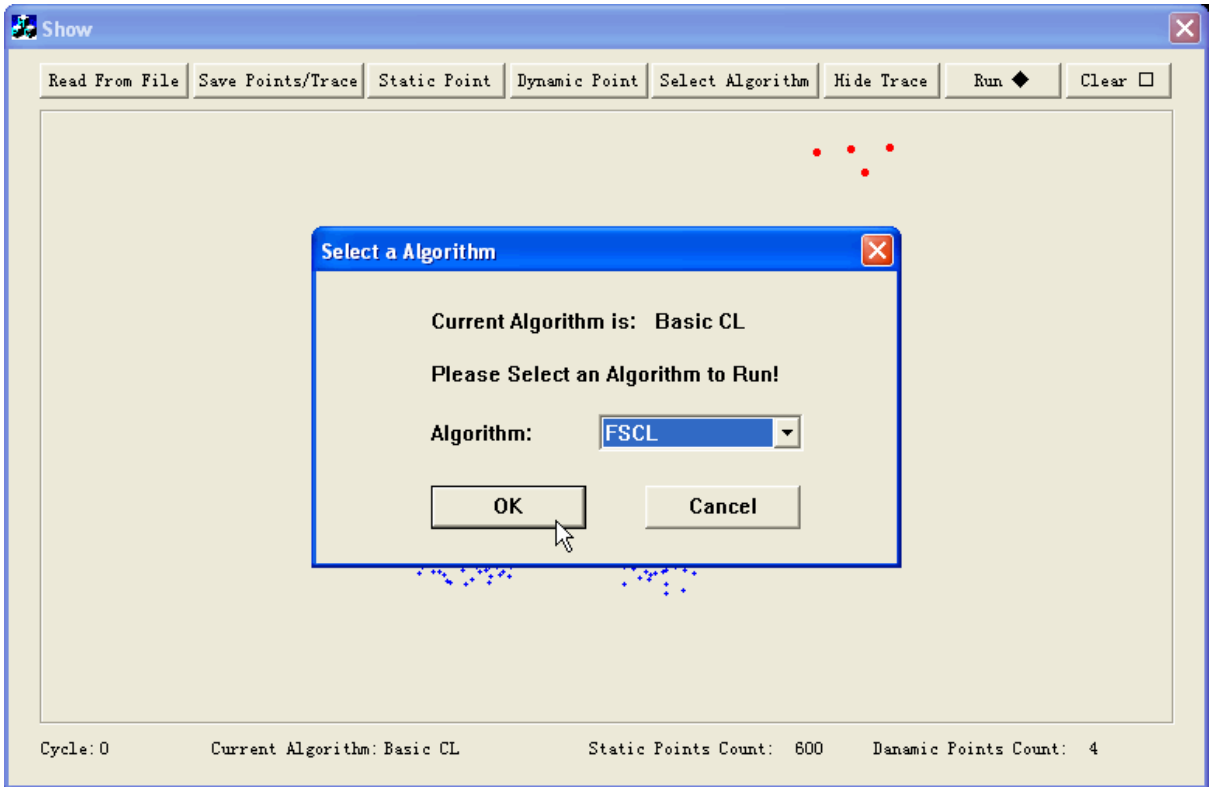
A four-cluster case



frequency sensitive competitive learning (FSCL) [Ahalt et al., 1990]

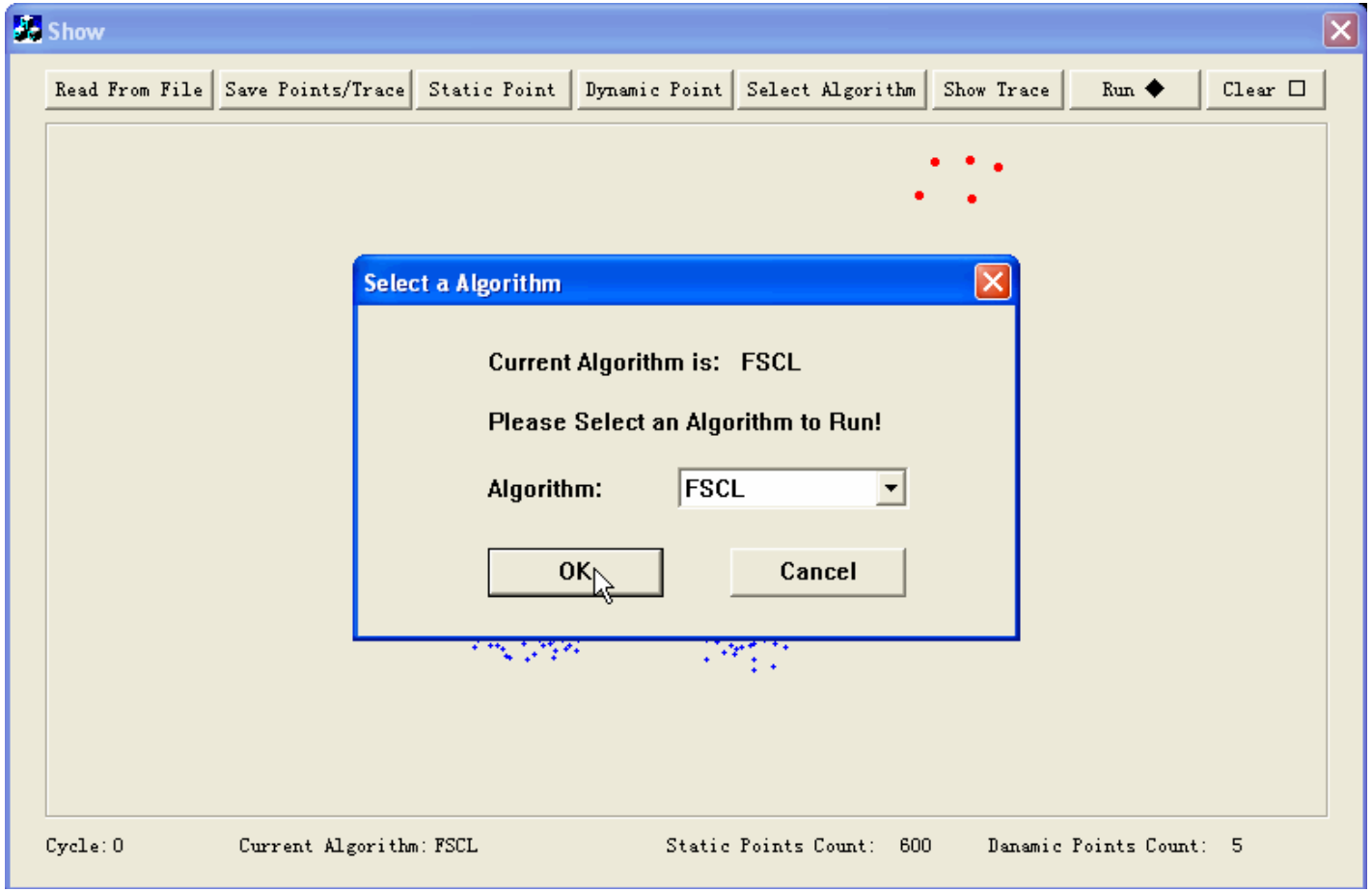
The idea is to penalize the frequent winners:

$$\varepsilon_t(\theta_j) = \alpha_j \|x_t - m_j\|^2$$



FSCL is not good when there are extra centers

When k is pre-assigned to 5. the frequency sensitive mechanism also brings the extra one into data to disturb the correct locations of others



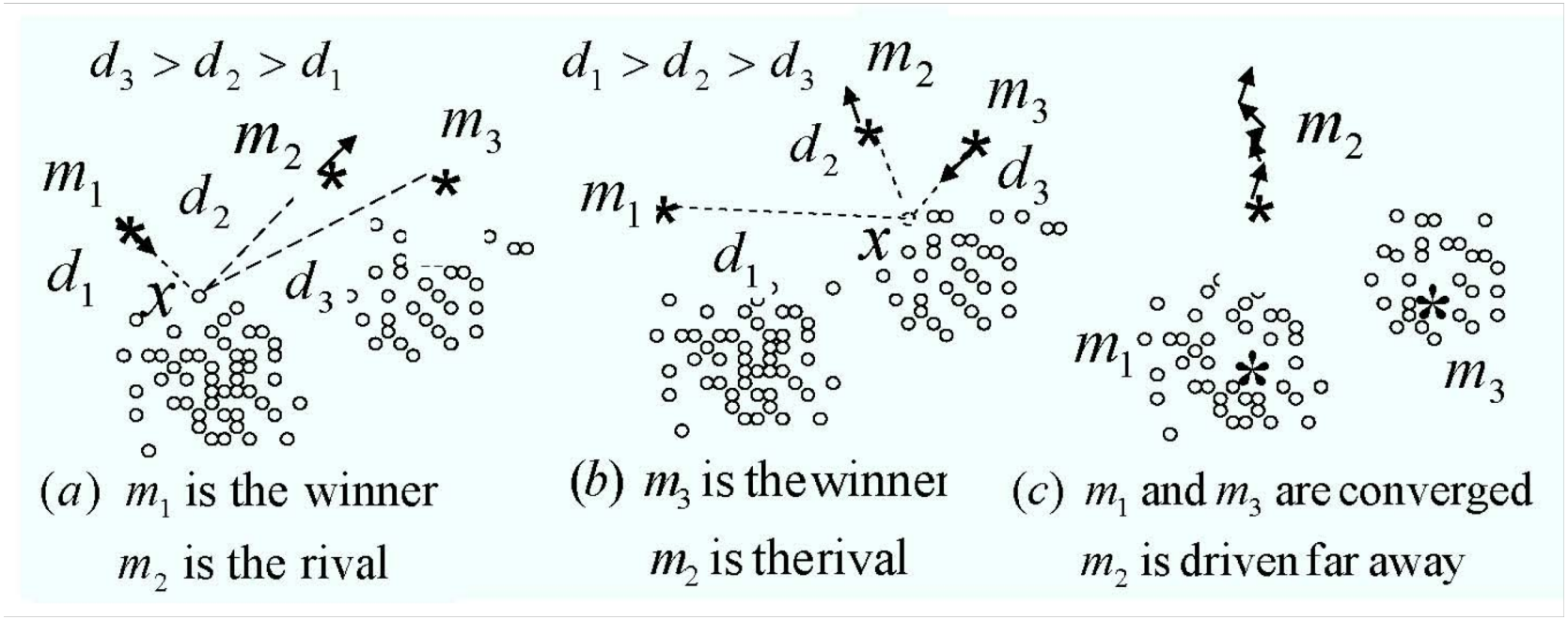
Rival penalized competitive learning (RPCL)

(Xu, Krzyzak, & Oja, 1992 , 1993)

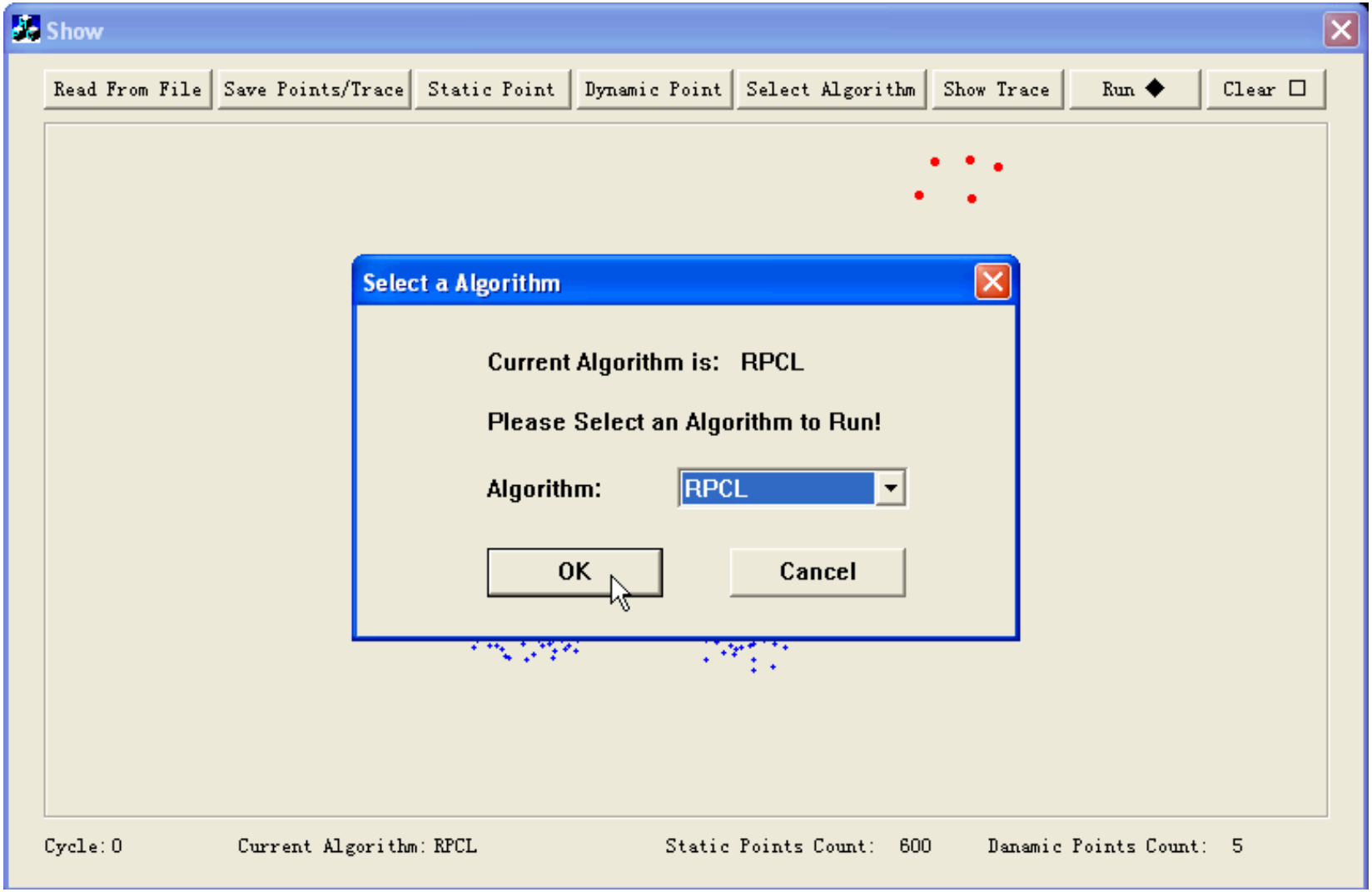
The RPCL differs from FSCL by implementing $p_{j,t}$ as follows:

$$p_{j,t} = \begin{cases} 1, & \text{if } j = c, \\ -\gamma, & \text{if } j = r, \\ 0, & \text{otherwise,} \end{cases} \quad \begin{cases} c = arg\ min_j \varepsilon_t(\theta_j), \\ r = arg\ min_{j \neq c} \varepsilon_t(\theta_j), \end{cases}$$

where γ approximately takes a number between 0.05 and 0.1 for controlling the penalizing strength.



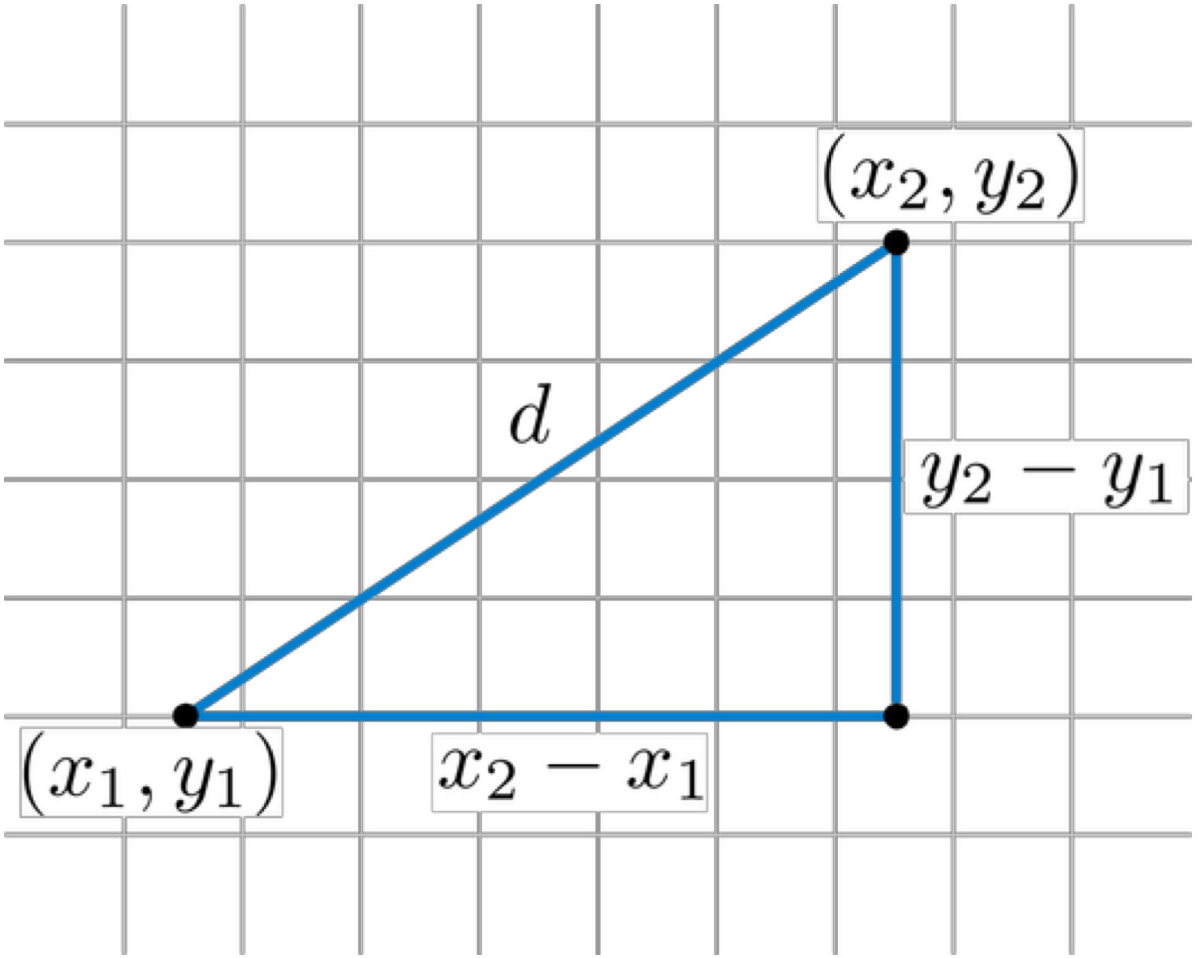
Rival penalized mechanism makes extra agents driven far away.



Outline

- Clustering
  - K-mean clustering, hierarchical clustering
- Adaptive learning (online learning)
  - CL, FSCL, RPCL
- Gaussian Mixture Models (GMM)
- Expectation-Maximization (EM) for maximum likelihood

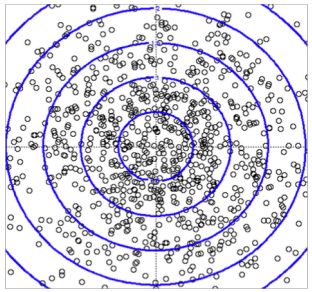
Euclidean Distance



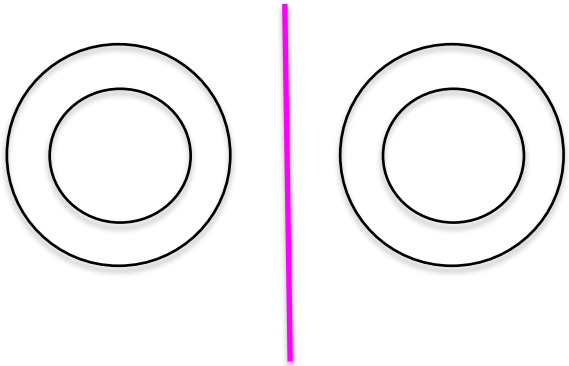
Glossary. (n.d.). Retrieved February 13, 2015, from http://rosalind.info/glossary/euclidean-distance/

Euclidian distance may not be a good measure for some data

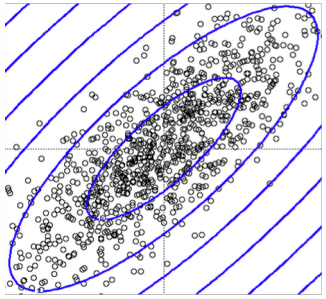
Euclidean distance

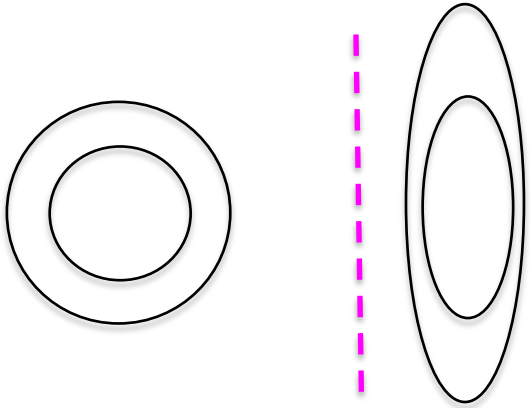


Equal distance line

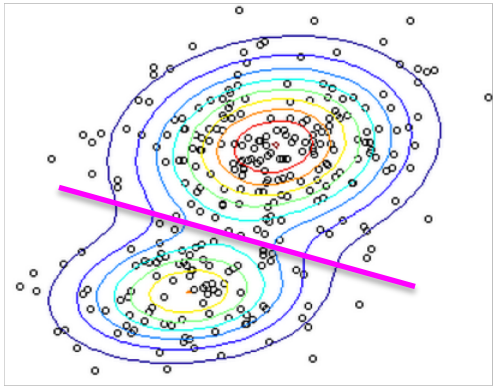


Mahalanobis distance





In general



Distances at different directions could be different!

$$d(\vec{x},\vec{y}) = \sqrt{(\vec{x}-\vec{y})^T \Sigma^{-1} (\vec{x}-\vec{y})}.$$

Σ is the covariance matrix

More Distance Measures

Table 1 Gene expression similarity measures

Manhattan distance (city-block distance, L1 norm)	$d_{fg} = \sum_c \lvert e_{fc} - e_{gc} \rvert$
Euclidean distance (L2 norm)	$d_{fg} = \sqrt{\sum_c (e_{fc} - e_{gc})^2}$
Mahalanobis distance	$d_{fg} = (e_f - e_g)' \Sigma^{-1} (e_f - e_g)$, where Σ is the (full or within-cluster) covariance matrix of the data
Pearson correlation (centered correlation)	$d_{fg} = 1 - r_{fg}$, with $r_{fg} = \frac{\sum_c (e_{fc} - \bar{e}_f)(e_{gc} - \bar{e}_g)}{\sqrt{\sum_c (e_{fc} - \bar{e}_f)^2 \sum_c (e_{gc} - \bar{e}_g)^2}}$
Uncentered correlation (angular separation, cosine angle)	$d_{fg} = 1 - r_{fg}$, with $r_{fg} = \frac{\sum_c e_{fc} e_{gc}}{\sqrt{\sum_c e_{fc}^2 \sum_c e_{gc}^2}}$
Spellman rank correlation	As Pearson correlation, but replace $\boldsymbol{e_{gc}}$ with the rank of $\boldsymbol{e_{gc}}$ within the expression values of gene g across all conditions $\boldsymbol{c = 1 \ldots C}$
Absolute or squared correlation	$d_{fg} = 1 - \lvert r_{fg} \rvert$ or $d_{fg} = 1 - r_{fg}^2$

d_{fg}, distance between expression patterns for genes f and g. e_{gc}, expression level of gene g under condition c.

D'haeseleer, P. (2005). How does gene expression clustering work? Nat Biotech *23*, 1499–1501.

From distance to probability

distance

likely

$$\| x - \mu \|^2 \longrightarrow \exp\{-\lambda \| x - \mu \|^2\}$$

"The closer, the more likely."

Sum or integral to be one

Probability

It is more powerful to consider everything in probability framework!

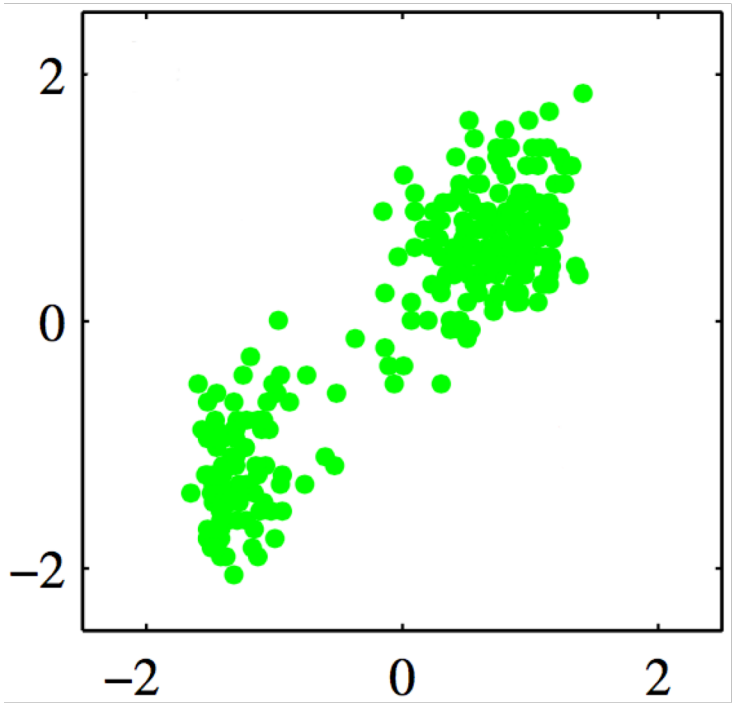
$$\frac{1}{\sqrt{(2\pi)^k|\Sigma|}} \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right)$$

Gaussian distribution with the Mahalanobis distance

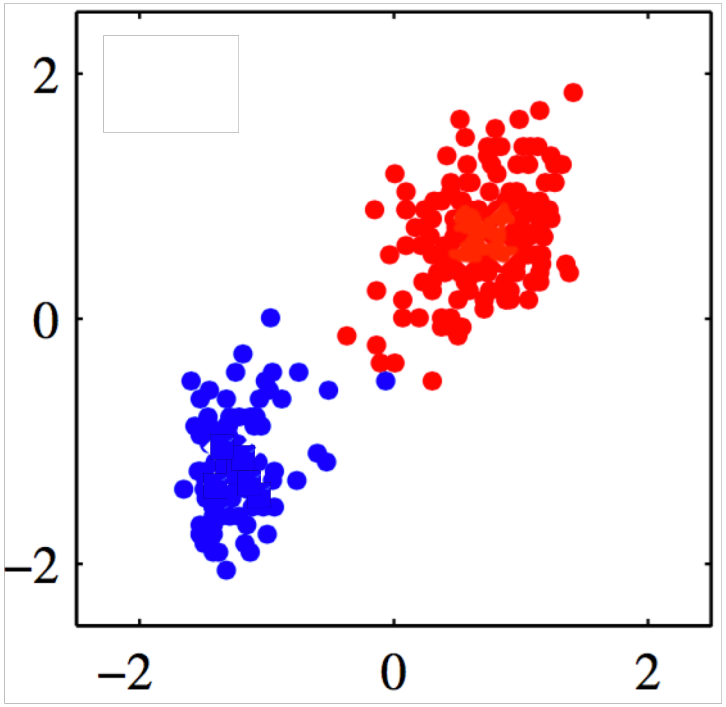
$$D_M(x) = \sqrt{(x-\mu)^T \Sigma^{-1} (x-\mu)}.$$

Review the clustering problem again

We have the following data:

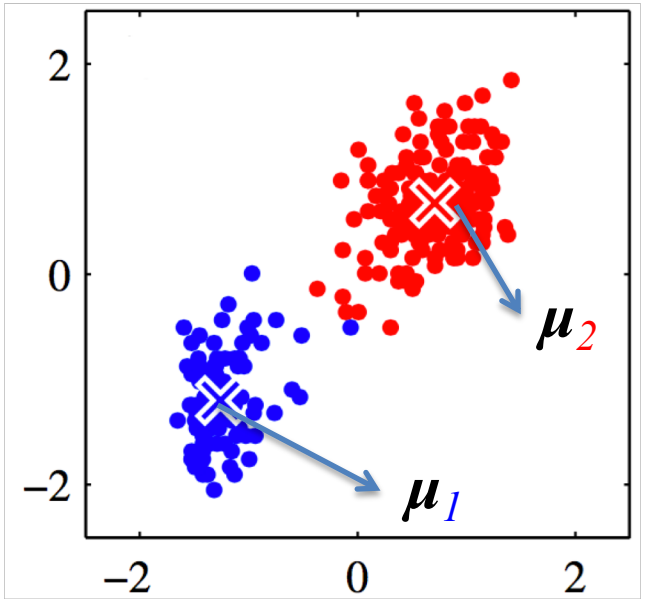


We want to cluster the data into two clusters (red and blue)

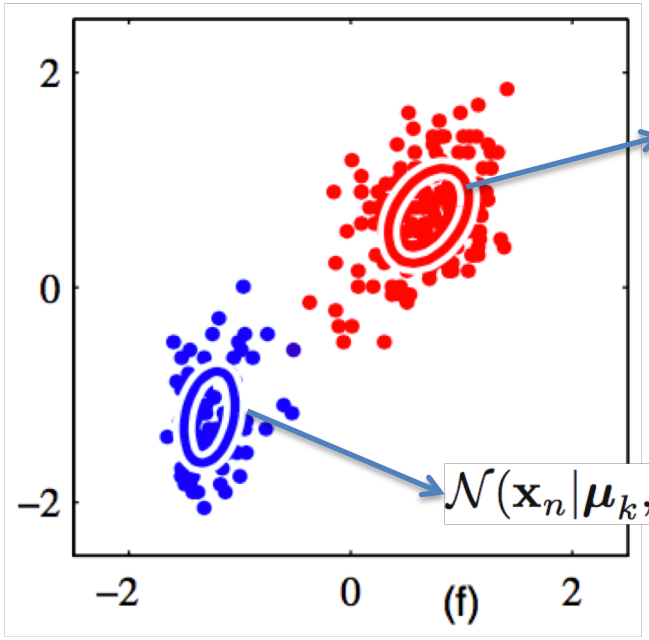


Instead if using $\{\mu_1, \mu_2\}$, each cluster is represented as a Gaussian distribution

K-means



Gaussian Mixture Model (GMM)

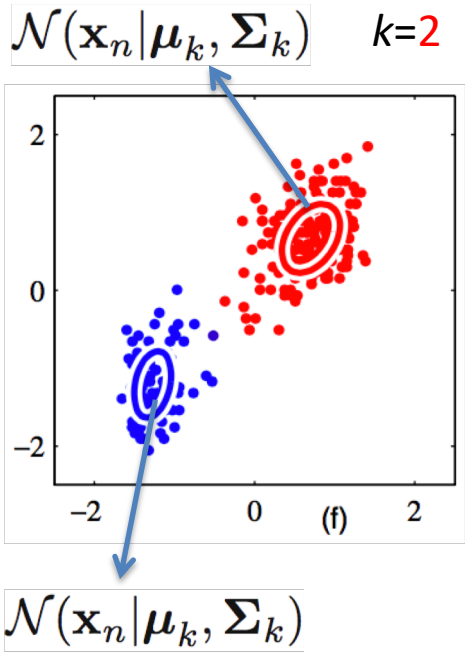


k=2

k=1

$$\mathcal{N}(\mathbf{x}\ |\boldsymbol{\mu}\ ,\boldsymbol{\Sigma}\ ) = \frac{1}{\sqrt{(2\pi)^k|\boldsymbol{\Sigma}|}} \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right)$$

Gaussian Mixture Model (GMM)



We use $z_k = 1$ to indicate a point $\mathbf{x}$ belongs to cluster k

$$\mathbf{z} = (z_1, \ldots, z_K) \qquad z_k \in \{0, 1\} \qquad \sum_k z_k = 1$$

Assume the points in the same cluster follow a **Gaussian distribution**

$$p(\mathbf{x}|z_k = 1) = \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

A mixing weight for each cluster:

$$p(z_k = 1) = \pi_k \qquad 0 \leqslant \pi_k \leqslant 1 \qquad \sum_{k=1}^{K} \pi_k = 1$$

prior probability of point belonging to a cluster

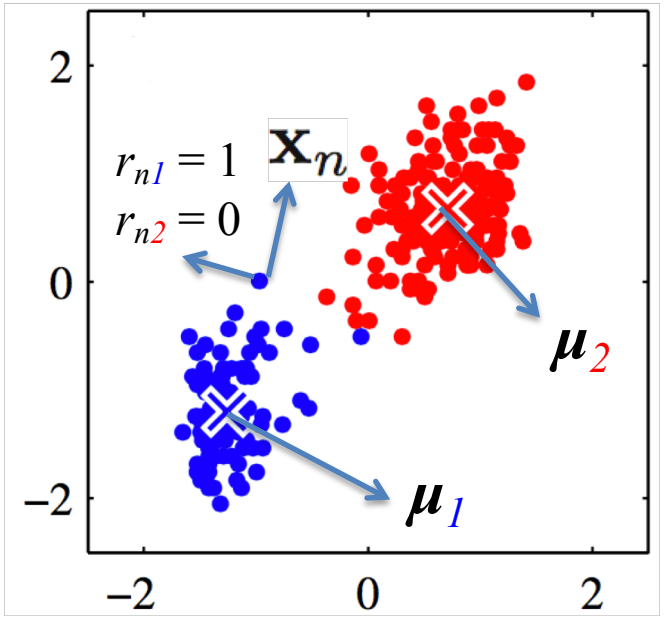
So, we get a distribution for the data point $\mathbf{x}$:

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

From minimizing sum of square distances to finding maximum likelihood

minimize

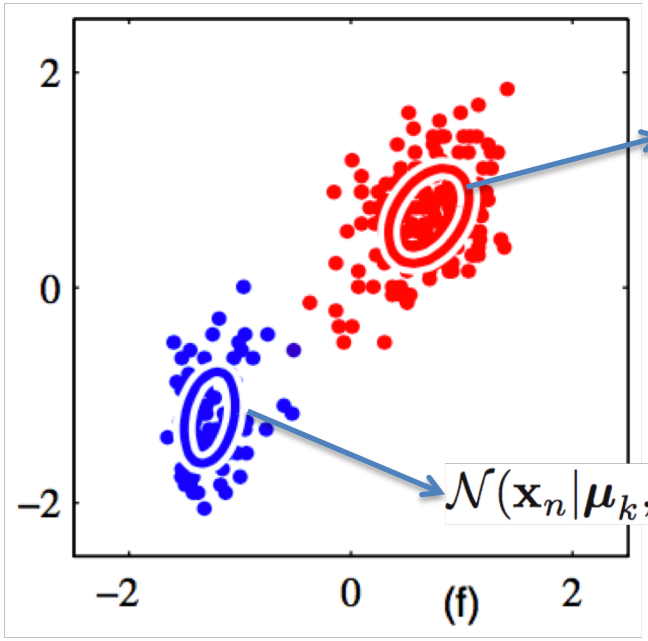
$$J = \sum_{n=1}^{N}\sum_{k=1}^{K} r_{nk}\|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2$$



maximize likelihood

$$p(\mathbf{X}|\boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\pi})$$

$$X = \{x_1, \ldots, x_N\}$$
$$\pi = \{\pi_1, \ldots, \pi_K\}$$
$$\mu = \{\mu_1, \ldots, \mu_K\}$$
$$\Sigma = \{\Sigma_1, \ldots, \Sigma_K\}$$



Remember: **The closer the distance, the more likely the probability.**

Maximum likelihood

Given a data set $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_N)^{\mathrm{T}}$ in which the observations $\{\mathbf{x}_n\}$ are assumed to be drawn independently from a multivariate Gaussian distribution, we can estimate the parameters of the distribution by maximum likelihood. The log likelihood function is given by

$$\ln p(\mathbf{X}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = -\frac{ND}{2}\ln(2\pi) - \frac{N}{2}\ln|\boldsymbol{\Sigma}| - \frac{1}{2}\sum_{n=1}^{N}(\mathbf{x}_n - \boldsymbol{\mu})^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{x}_n - \boldsymbol{\mu})$$

Maximizing the log-likelihood function:

$$\frac{\partial}{\partial \boldsymbol{\mu}}\ln p(\mathbf{X}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \sum_{n=1}^{N}\boldsymbol{\Sigma}^{-1}(\mathbf{x}_n - \boldsymbol{\mu}) = 0 \quad \longrightarrow \quad \boldsymbol{\mu}_{\mathrm{ML}} = \frac{1}{N}\sum_{n=1}^{N}\mathbf{x}_n$$

Similarly we get $\boldsymbol{\Sigma}_{\mathrm{ML}} = \frac{1}{N}\sum_{n=1}^{N}(\mathbf{x}_n - \boldsymbol{\mu}_{\mathrm{ML}})(\mathbf{x}_n - \boldsymbol{\mu}_{\mathrm{ML}})^{\mathrm{T}}$

$\boldsymbol{\mu}_{\mathrm{ML}}$ and $\boldsymbol{\Sigma}_{\mathrm{ML}}$ are the maximum likelihood estimates of the mean and the co-variance matrix.

Outline

- Clustering
  - K-mean clustering, hierarchical clustering
- Adaptive learning (online learning)
  - CL, FSCL, RPCL
- Gaussian Mixture Models (GMM)
- Expectation-Maximization (EM) for maximum likelihood

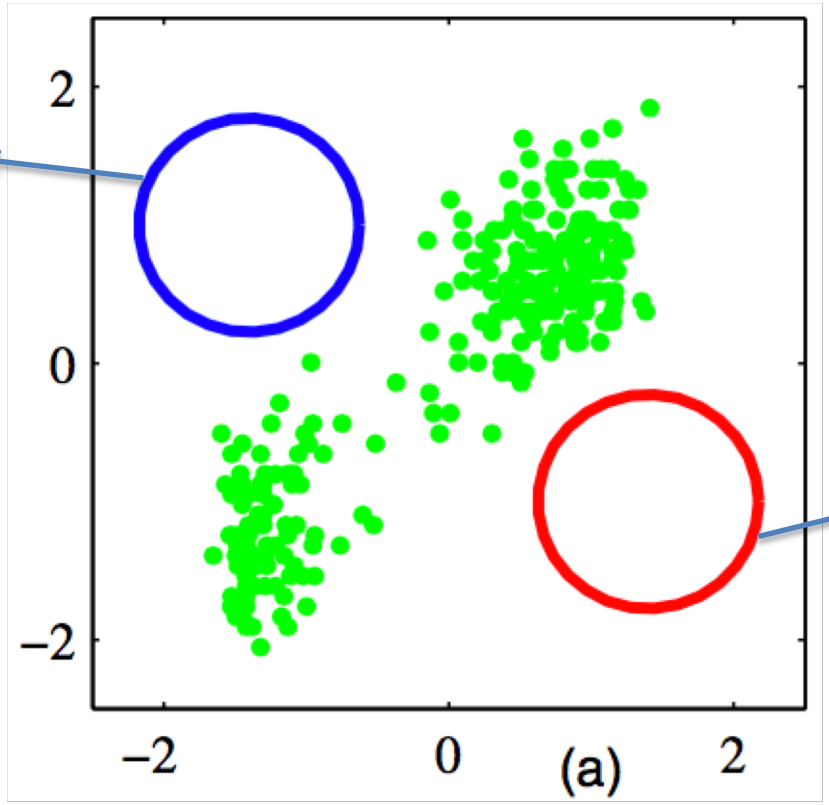
Expectation-Maximization (EM) algorithm for maximum likelihood

$$\ln p(\mathbf{X}|\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \sum_{n=1}^{N} \ln \left\{ \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \right\}$$

Initialization

$\mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$

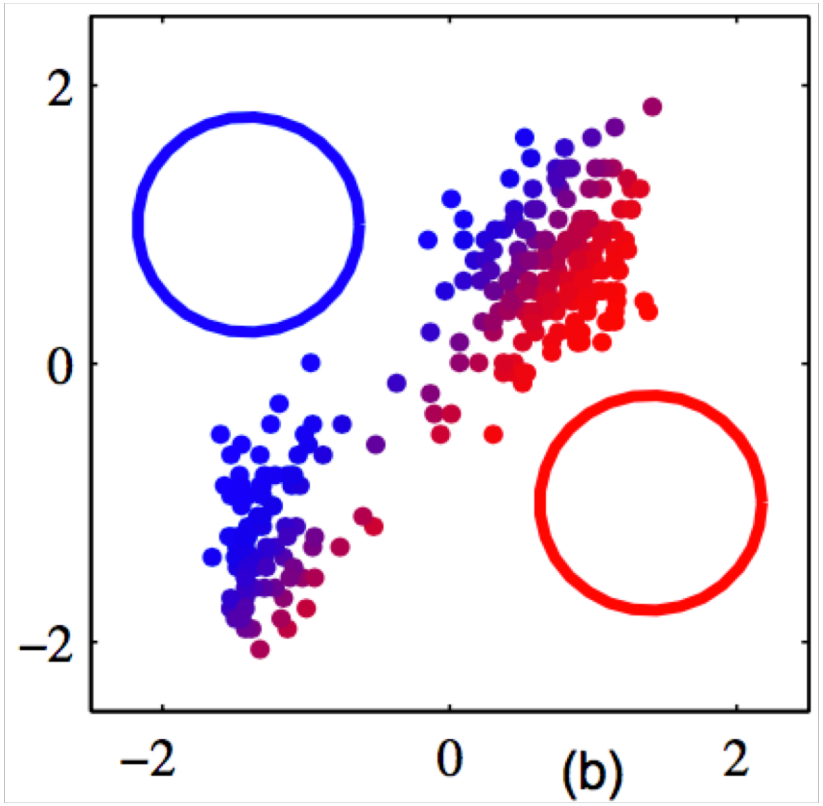
k=1



$\mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$

k=2

E Step



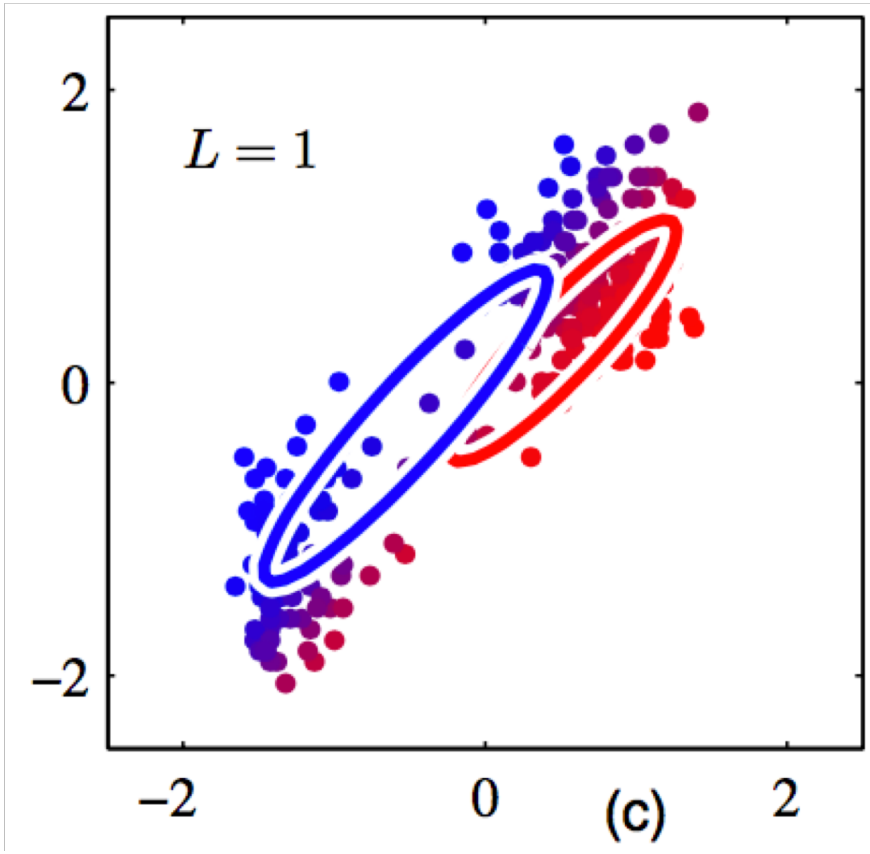
When the parameters are given, the assignments of the points can be calculated by the posterior probability, i.e., the probability of a data point belonging to a cluster once we have observed the data point.

$$\gamma(z_{nk}) = \frac{\pi_k \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}$$

Soft assignment:
A point fractionally belongs to two clusters.

For example,
0.2 belong to cluster 1
0.8 belong to cluster 2

M Step



When the assignments $\gamma(z_{nk})$ of the points to the clusters are known, parameters could be calculated for each cluster (Gaussian) separately.

Mixing weight π_k: the proportion of number of points in cluster k within all data points

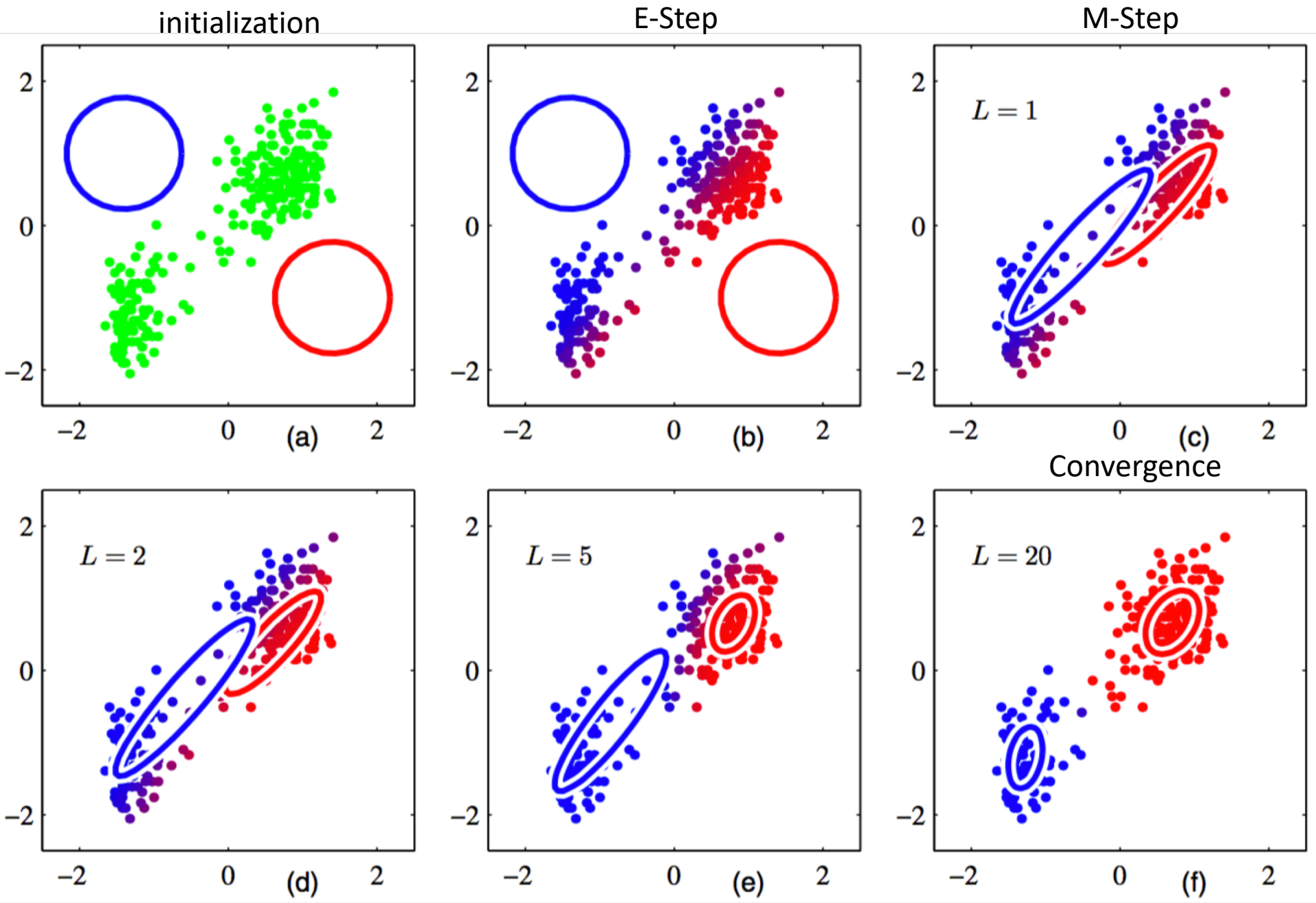
$$\pi_k = \frac{N_k}{N} \quad ; \quad N_k = \sum_{n=1}^{N} \gamma(z_{nk}).$$

$\boldsymbol{\mu}_k$, $\boldsymbol{\Sigma}_k$: the mean and the covariance matrix are calculated for each cluster

$$\boldsymbol{\mu}_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nk}) \mathbf{x}_n$$

$$\boldsymbol{\Sigma}_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nk})(\mathbf{x}_n - \boldsymbol{\mu}_k)(\mathbf{x}_n - \boldsymbol{\mu}_k)^{\mathrm{T}}$$

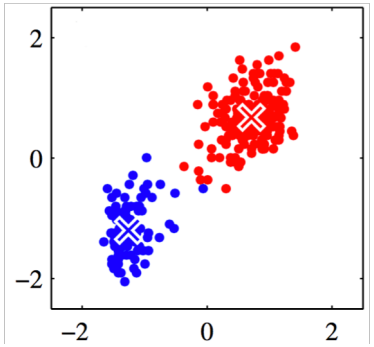
L denotes the number of cycles of the EM algorithm.



L denotes the number of cycles of E-Step and M-Step.

Relation to K-means

$\|\mathbf{x} - \boldsymbol{\mu}_k\|^2$

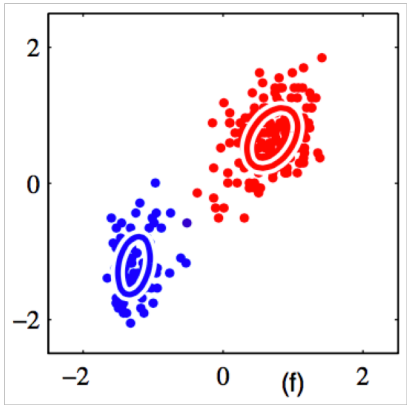


$\{\boldsymbol{\mu}_k\}$

One-in-K assignment

$$r_{nk} = \begin{cases} 1 & \text{if } k = \arg\min_j \|\mathbf{x}_n - \boldsymbol{\mu}_j\|^2 \\ 0 & \text{otherwise.} \end{cases}$$

$\boldsymbol{\Sigma}_k = \varepsilon \mathbf{I}$



GMM considers covariance and mixing weights.

$$p(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) = \frac{1}{(2\pi\epsilon)^{1/2}} \exp\left\{-\frac{1}{2\epsilon}\|\mathbf{x} - \boldsymbol{\mu}_k\|^2\right\}$$

Soft assignment

$$\gamma(z_{nk}) = \frac{\pi_k \exp\left\{-\|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2/2\epsilon\right\}}{\sum_j \pi_j \exp\left\{-\|\mathbf{x}_n - \boldsymbol{\mu}_j\|^2/2\epsilon\right\}}$$

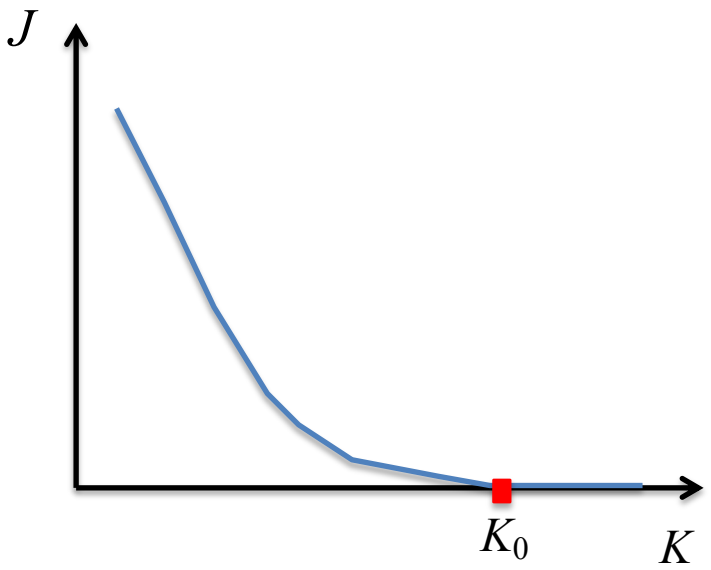
Summary for the EM algorithm for GMM

- Does it find the global optimum?
  - No, like K-means, EM only finds the nearest local optimum and the optimum depends on the initialization
- GMM is more general then K-means by considering mixing weights, covariance matrices, and soft assignments.
- Like K-means, it does not tell you the best K.

How to determine the cluster number K?

K-mean

$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} \|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2$$

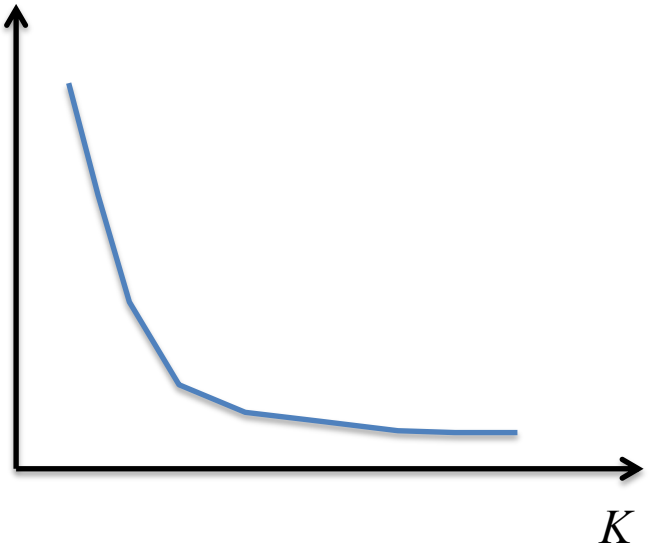


J does not tell which K is better.

GMM

$$\ln p(\mathbf{X}|\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \sum_{n=1}^{N} \ln \left\{ \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \right\}$$

- Log-likelihood



Negative log-likelihood also decreases as K increases.

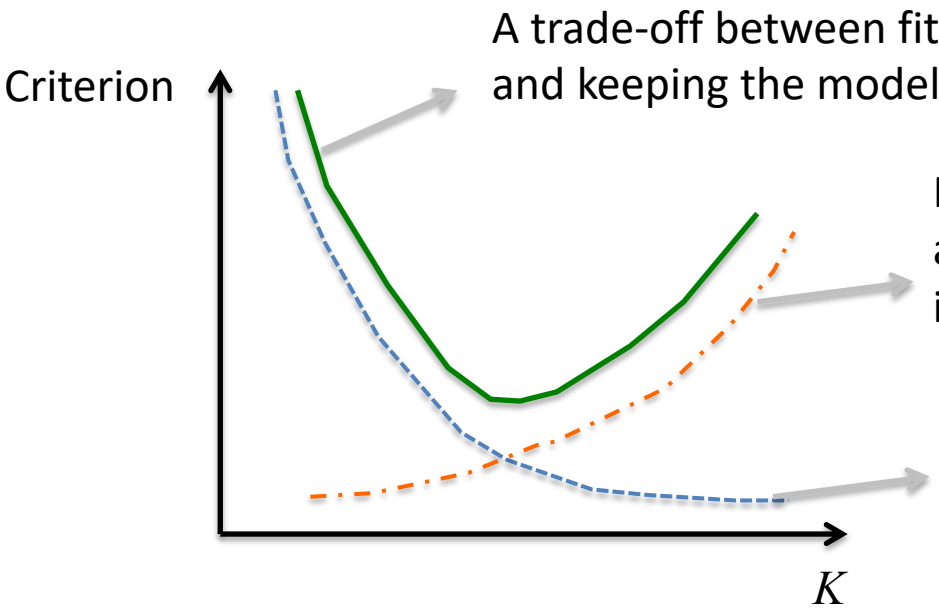
Model selection in general

Probabilistic model

$$p(X_N \mid \Theta_K)$$

Candidate models:

$$\Theta_1 \subseteq \Theta_2 \subseteq \cdots \subseteq \Theta_K \subseteq \cdots$$



A trade-off between fitting the data well and keeping the model simple

Models become more and more complex as K increases.

Negative log-likelihood (or fitting error) to make sure the model fit the data well.

Akaike's Information Criterion (AIC) $\ln p(X_N \mid \hat{\Theta}_K) - d_k$

Bayesian Information Criterion (BIC) $\ln p(X_N \mid \hat{\Theta}_K) - \frac{1}{2} d_k \ln N$

d_k: number of free parameters

N: sample size

Thank you!