UVA CS 4501: Machine Learning

Lecture 24: Unsupervised Clustering (III)

Dr. Yanjun Qi

University of Virginia

Department of
Computer Science

Where are we ? → major sections of this course

- ❑ Regression (supervised)
- ❑ Classification (supervised)
  - ❑ Feature selection
- ❑ Unsupervised models
  - ❑ Dimension Reduction (PCA)
  - ❑ Clustering (K-means, GMM/EM, Hierarchical)
- ❑ Learning theory
- ❑ Graphical models

An unlabeled Dataset X

	X_1	X_2	X_3
s_1			
s_2			
s_3			
s_4			
s_5			
s_6			

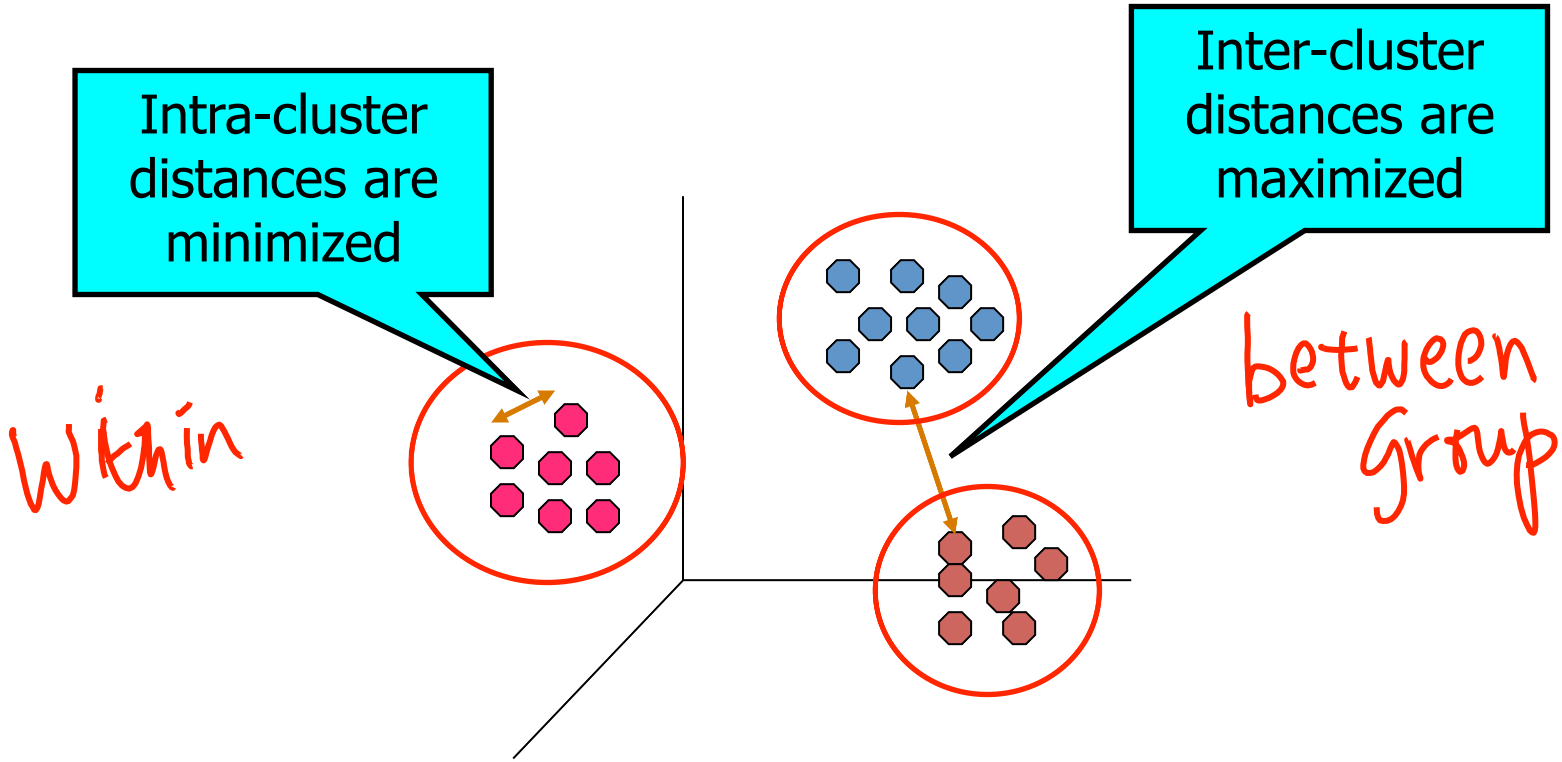
a data matrix of n observations on p variables $x_1, x_2, \ldots x_p$

Unsupervised learning = learning from raw (unlabeled, unannotated, etc) data, as opposed to supervised data where a classification label of examples is given

- **Data**/*points/instances/examples/samples/records*: [rows]
- **Features**/*attributes/dimensions/independent variables/covariates/predictors/regressors*: [columns]

What is clustering?

- Find groups (clusters) of data points such that data points in a group will be similar (or related) to one another and different from (or unrelated to) the data points in other groups

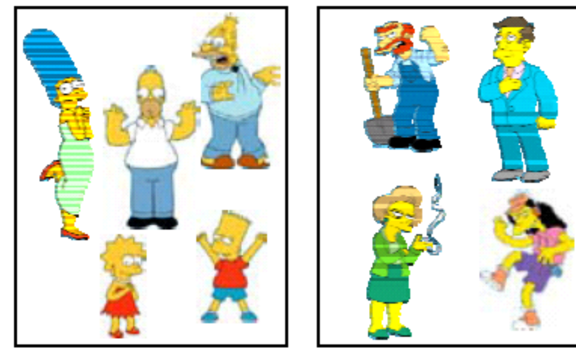


Roadmap: clustering

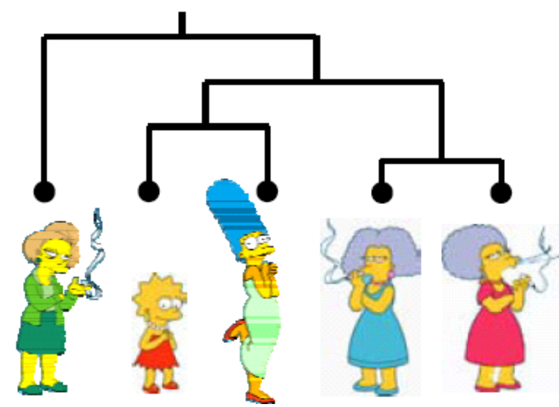
- Definition of "groupness”
- Definition of "similarity/distance"
- Representation for objects
- How many clusters?
- Clustering Algorithms
  - Partitional algorithms
  - Hierarchical algorithms
- Formal foundation and convergence

Clustering Algorithms

- Partitional algorithms
  - Usually start with a random (partial) partitioning
  - Refine it iteratively
    - K means clustering
    - Mixture-Model based clustering

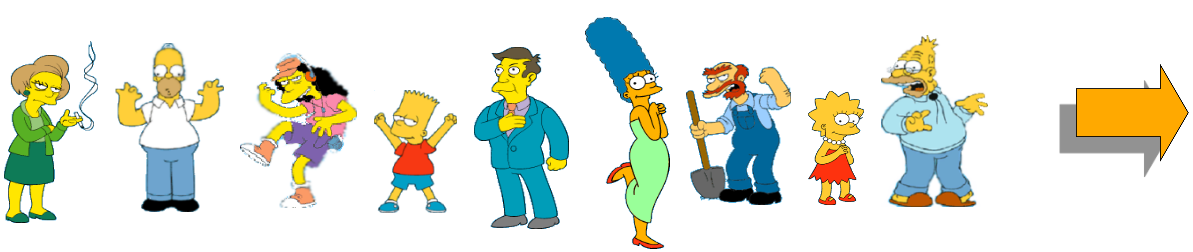


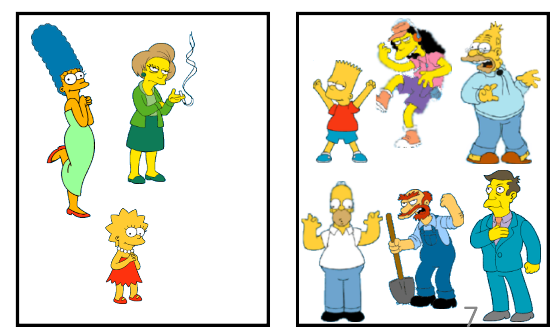
- Hierarchical algorithms
  - Bottom-up, agglomerative
  - Top-down, divisive



(2) Partitional Clustering

- Nonhierarchical
- Construct a partition of *n* objects into a set of *K* clusters
- User has to specify the desired number of clusters K.





Other partitioning Methods

- Partitioning around medoids (PAM): instead of averages, use multidim medians as centroids (cluster "prototypes"). Dudoit and Freedland (2002).

Other partitioning Methods

- Partitioning around medoids (PAM): instead of averages, use multidim medians as centroids (cluster "prototypes"). Dudoit and Freedland (2002).
- Self-organizing maps (SOM): add an underlying "topology" (neighboring structure on a lattice) that relates cluster centroids to one another. Kohonen (1997), Tamayo et al. (1999).

Other partitioning Methods

- Partitioning around medoids (PAM): instead of averages, use multidim medians as centroids (cluster "prototypes"). Dudoit and Freedland (2002).
- Self-organizing maps (SOM): add an underlying "topology" (neighboring structure on a lattice) that relates cluster centroids to one another. Kohonen (1997), Tamayo et al. (1999).
- Fuzzy k-means: allow for a "gradation" of points between clusters; soft partitions. Gash and Eisen (2002).

Other partitioning Methods

$C_j \in$ train Set

- Partitioning around medoids (PAM): instead of averages, use multidim medians as centroids (cluster "prototypes"). Dudoit and Freedland (2002).
- Self-organizing maps (SOM): add an underlying "topology" (neighboring structure on a lattice) that relates cluster centroids to one another. Kohonen (1997), Tamayo et al. (1999).
- Fuzzy k-means: allow for a "gradation" of points between clusters; soft partitions. Gash and Eisen (2002).
- Mixture-based clustering: implemented through an EM (Expectation-Maximization)algorithm. This provides soft partitioning, and allows for modeling of cluster centroids and shapes. (Yeung et al. (2001), McLachlan et al. (2002))

$m_{ij} \in \{1, 0\} \rightarrow [0, 1]$

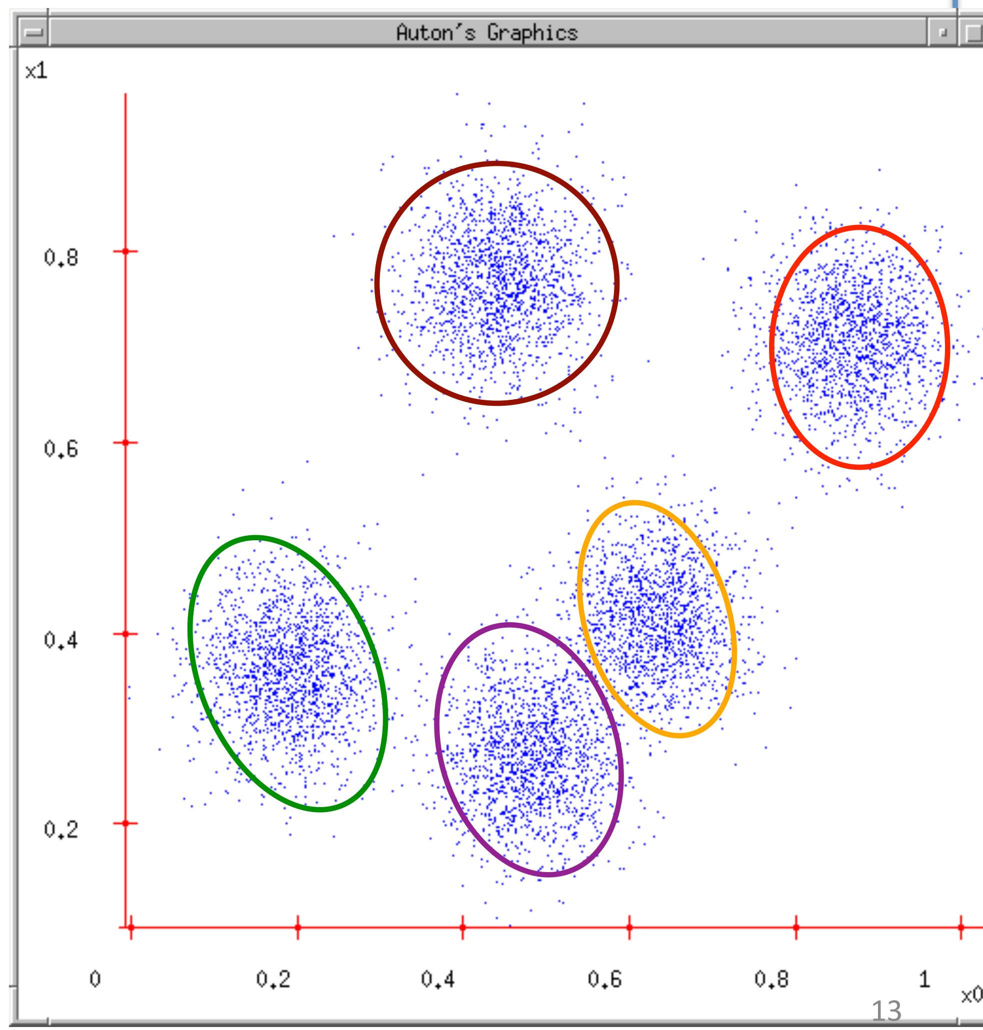
Partitional : Gaussian Mixture Model

- 1. Review of Gaussian Distribution
- 2. GMM for clustering : basic algorithm
- 3. GMM connecting to K-means
- 4. GMM examples
- 5. Problems of GMM and K-means

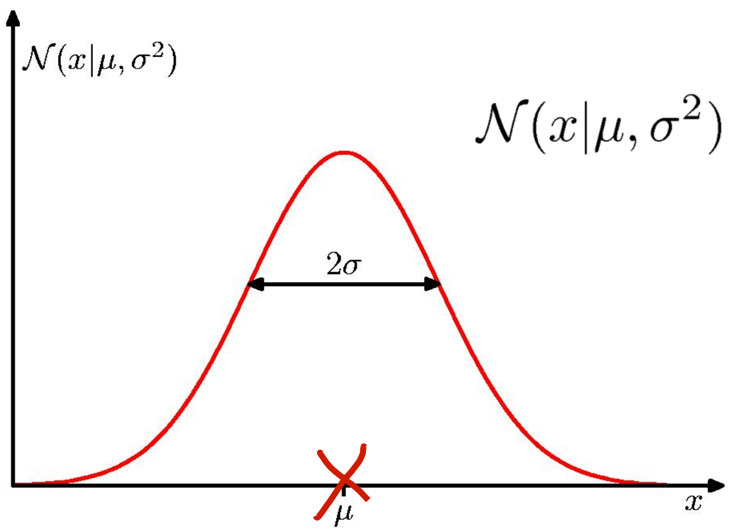
A Gaussian Mixture Model for Clustering

- Assume that data are generated from a mixture of Gaussian distributions
- For each Gaussian distribution
  - Center: μ_i
  - covariance: Σ_i
- For each data point
  - Determine membership

z_{ij} : if x_i belongs to j-th cluster

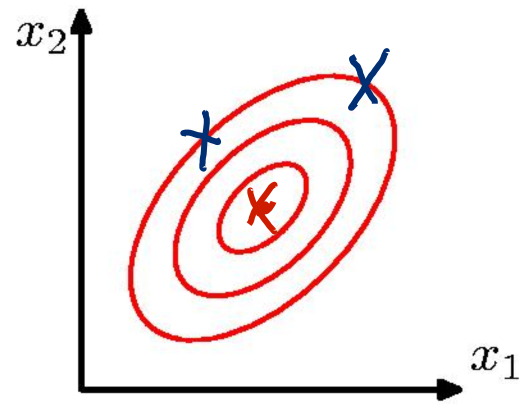


Gaussian Distribution



$$\mathcal{N}(x|\mu,\sigma^2) = \frac{1}{(2\pi\sigma^2)^{1/2}} \exp\left\{-\frac{1}{2\sigma^2}(x-\mu)^2\right\}$$

$$X \sim N(\mu, \sigma^2)$$



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

Mean

Covariance Matrix

Courtesy: http://research.microsoft.com/~cmbishop/PRML/index.htm

Example: the Bivariate Normal distribution

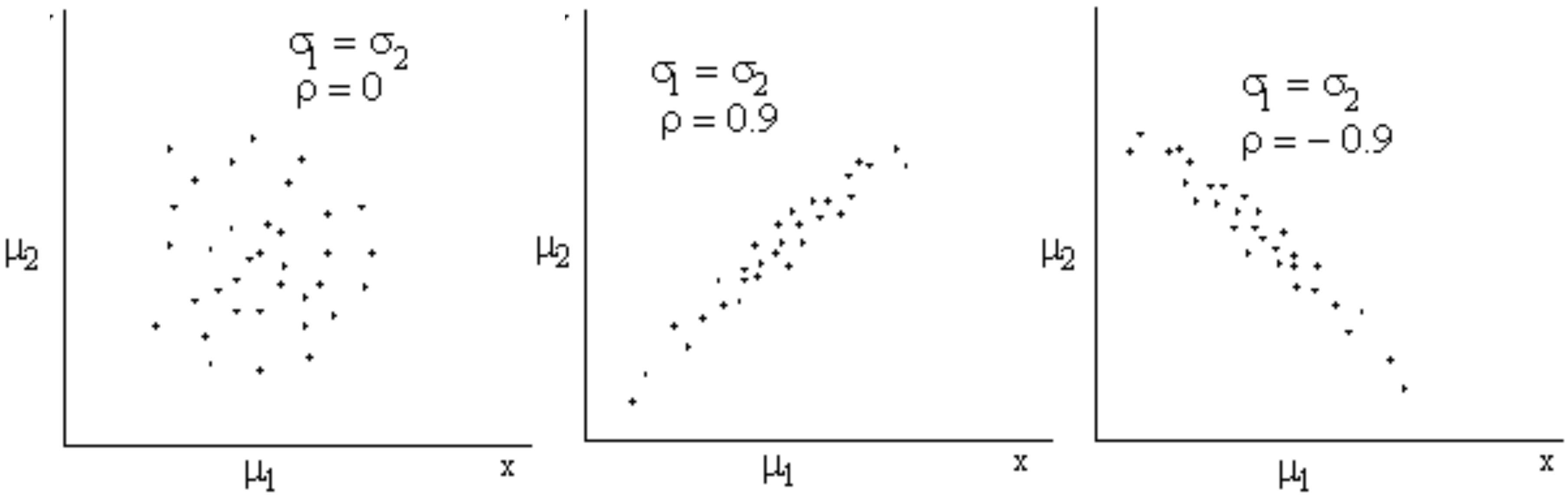
$$p(\vec{x}) = f\left(x_1, x_2\right) = \frac{1}{\left(2\pi\right)\left|\Sigma\right|^{1/2}} e^{-\frac{1}{2}\left(\vec{x}-\vec{\mu}\right)^T \Sigma^{-1}\left(\vec{x}-\vec{\mu}\right)}$$

with $\vec{\mu} = \begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix}$ 2x1 and

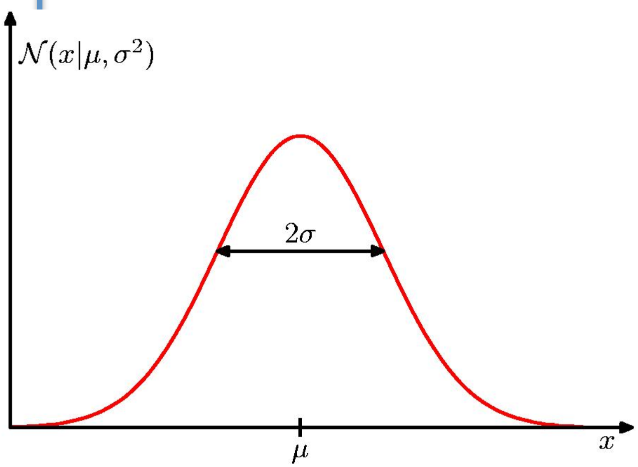
$$\underset{2\times 2}{\Sigma} = \begin{bmatrix} \sigma_{11} & \sigma_{12} \\ \sigma_{12} & \sigma_{22} \end{bmatrix} = \begin{bmatrix} \overset{V(X_1)}{\sigma_1^2} & \overset{Cov(X_1,X_2)}{\underline{\rho\sigma_1\sigma_2}} \\ \underline{\rho\sigma_1\sigma_2} & \underset{V(X_2)}{\sigma_2^2} \end{bmatrix} \; 2\times 2$$

$$\left|\Sigma\right| = \sigma_{11}\sigma_{22} - \sigma_{12}^2 = \sigma_1^2\sigma_2^2\left(1-\rho^2\right)$$

Scatter Plots of data from the bivariate Normal distribution



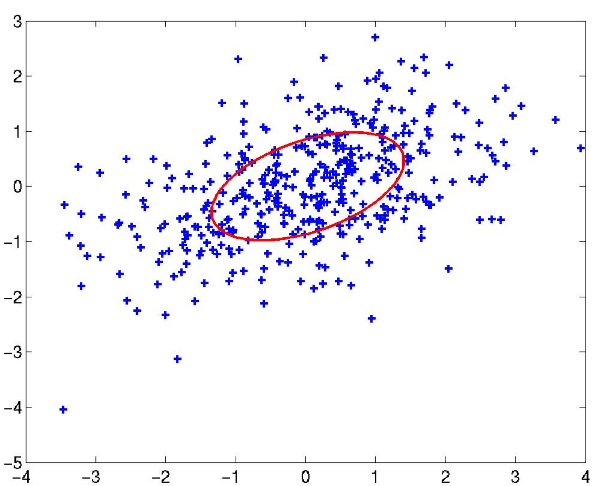
How to Estimate Gaussian: MLE



- In the 1D Gaussian case, we simply set the mean and the variance to the sample mean and the sample variance:

$$\overline{\mu} = \frac{1}{n}\sum_{i=1}^{n} x_i$$

$$\overline{\sigma}^2 = \frac{1}{n}\sum_{i=1}^{n} (x_i - \overline{\mu})^2$$



The p-multivariate Normal distribution

$$<X_1, X_2 \cdots, X_p> \sim N\left(\vec{\mu}, \Sigma\right)$$

$$\vec{\mu} = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_p \end{bmatrix}_{p \times 1}$$

$$\mu_i = \frac{1}{n} \sum_{j=1}^{N} X_j^{(i)}$$

$\in \{1, 2, \cdots, p\}$
i-th feature

j-th sample
$\in \{1, 2, \cdots, N\}$

$$\Sigma_{p \times p} = \begin{array}{cc} & \begin{array}{ccc} & & i \end{array} \\ & \begin{bmatrix} Var(X_1) & & Cov(X_i, X_j) \\ & \ddots & \\ & & Var(X_p) \end{bmatrix} \; j \end{array}$$

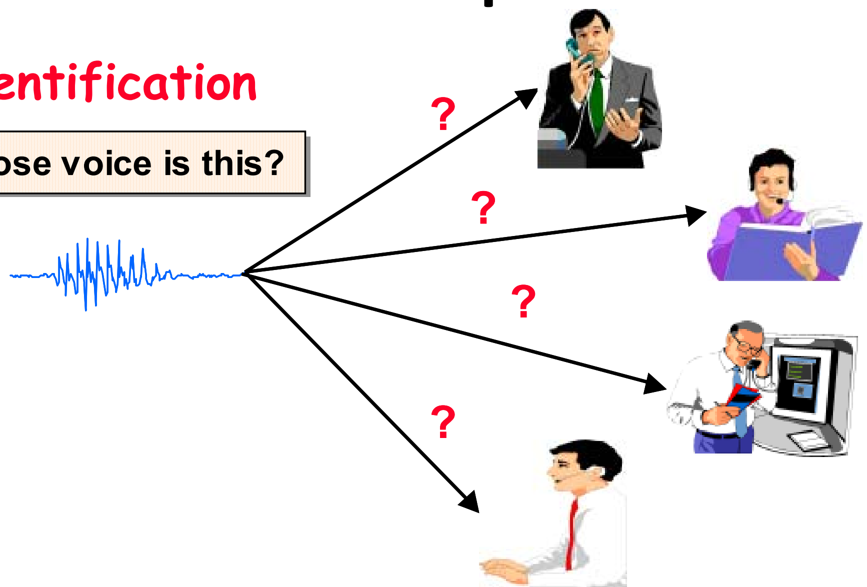
Partitional : Gaussian Mixture Model

- 1. Review of Gaussian Distribution
- 2. GMM for clustering : basic algorithm
- 3. GMM connecting to K-means
- 4. GMM examples
- 5. Problems of GMM and K-means

Application: Three Speaker Recognition Tasks

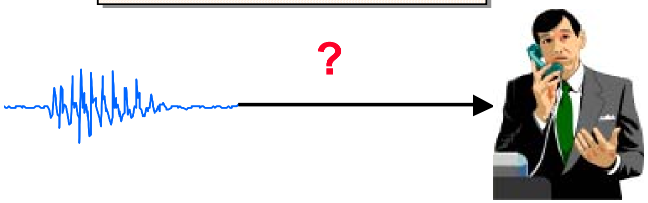
Identification

Whose voice is this?



Verification/Authentication/Detection

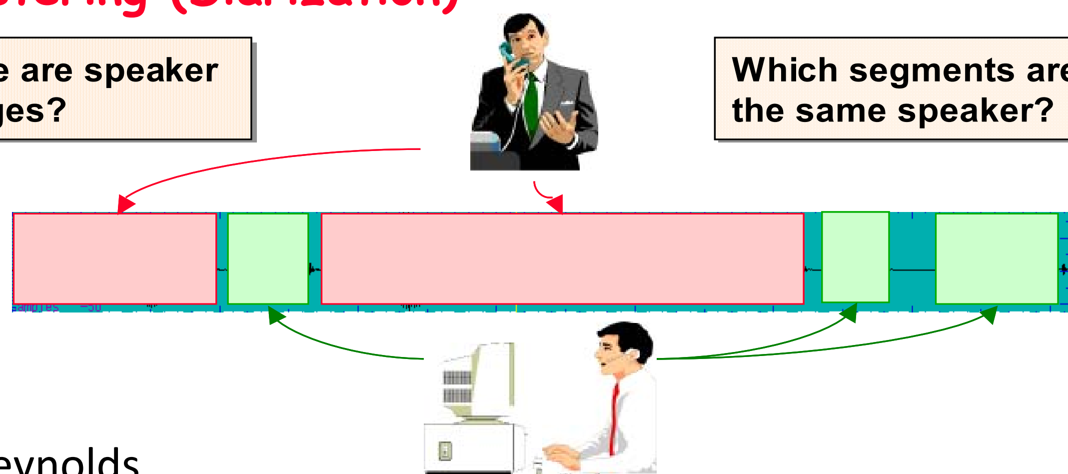
Is this Bob's voice?



Segmentation and Clustering (Diarization)

Where are speaker changes?

Which segments are from the same speaker?

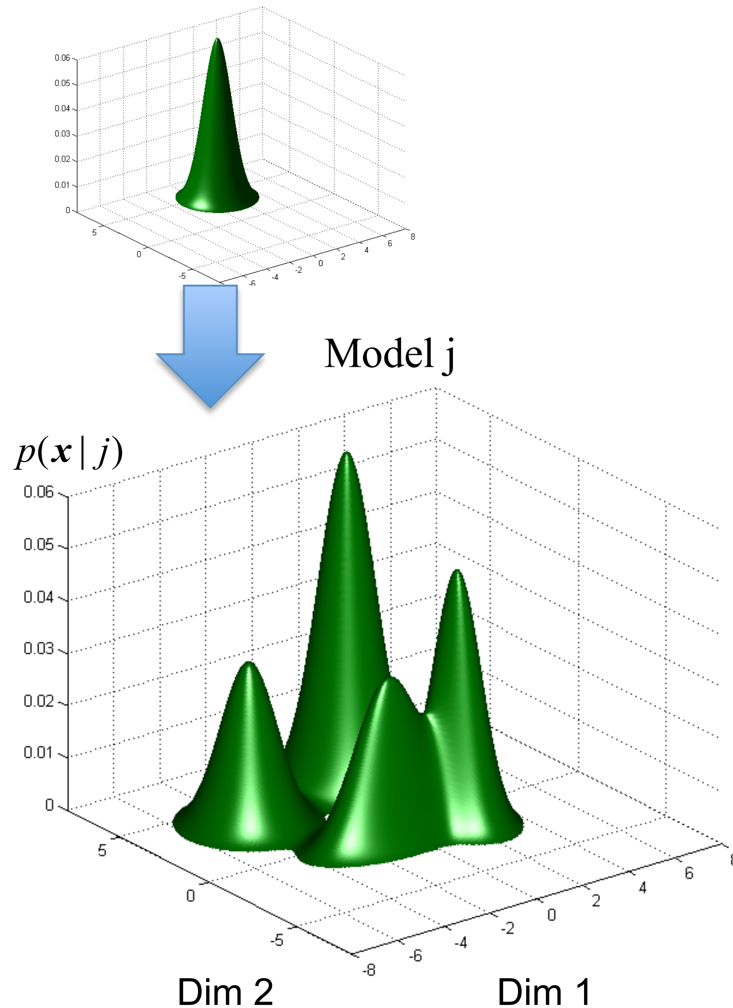


slide from Douglas Reynolds

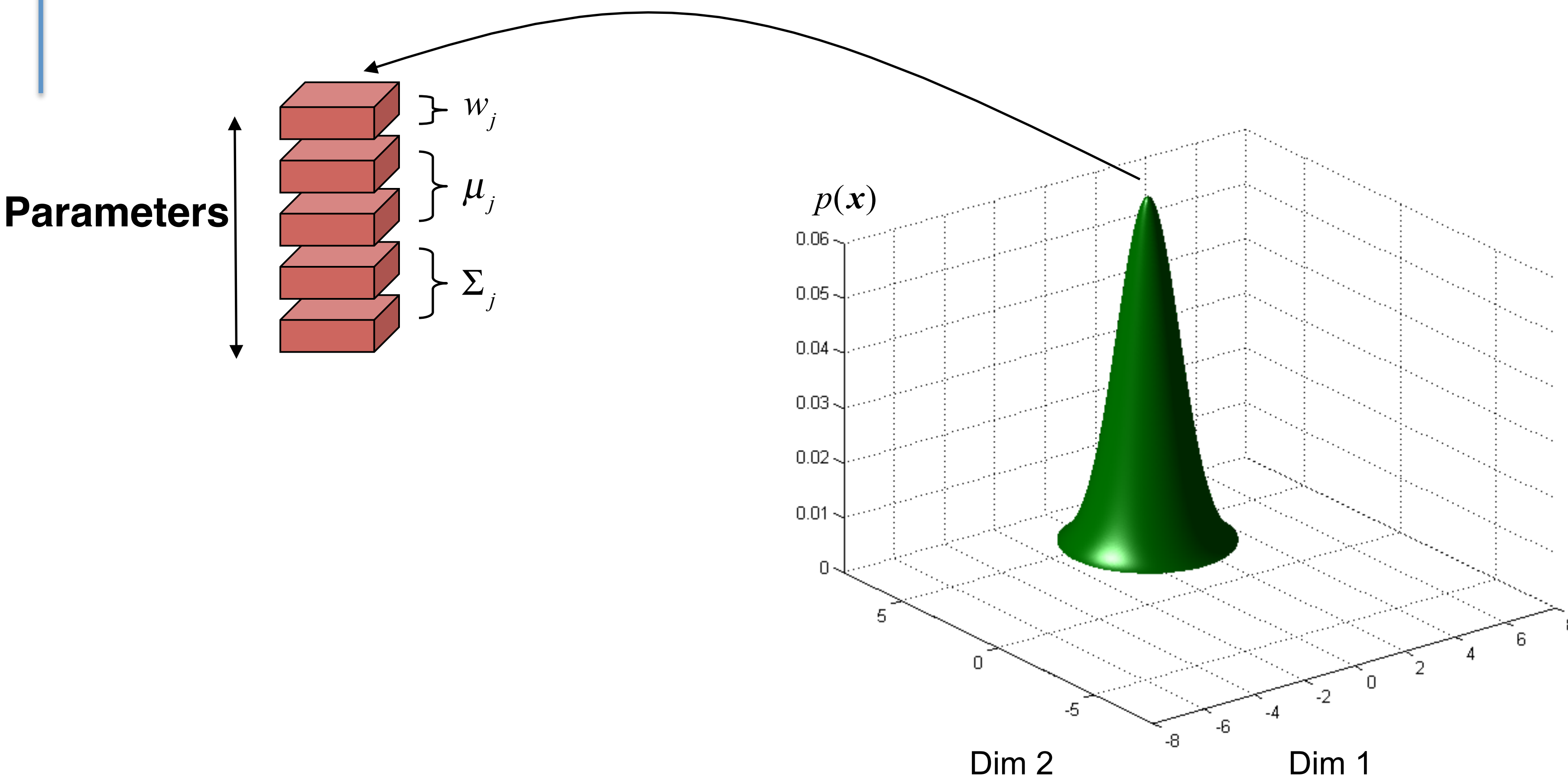
Application : GMMs for speaker recognition

- A Gaussian mixture model (GMM) represents as the weighted sum of multiple Gaussian distributions
- Each Gaussian state i has a
  - Mean μ_i
  - Covariance Σ_j
  - Weight

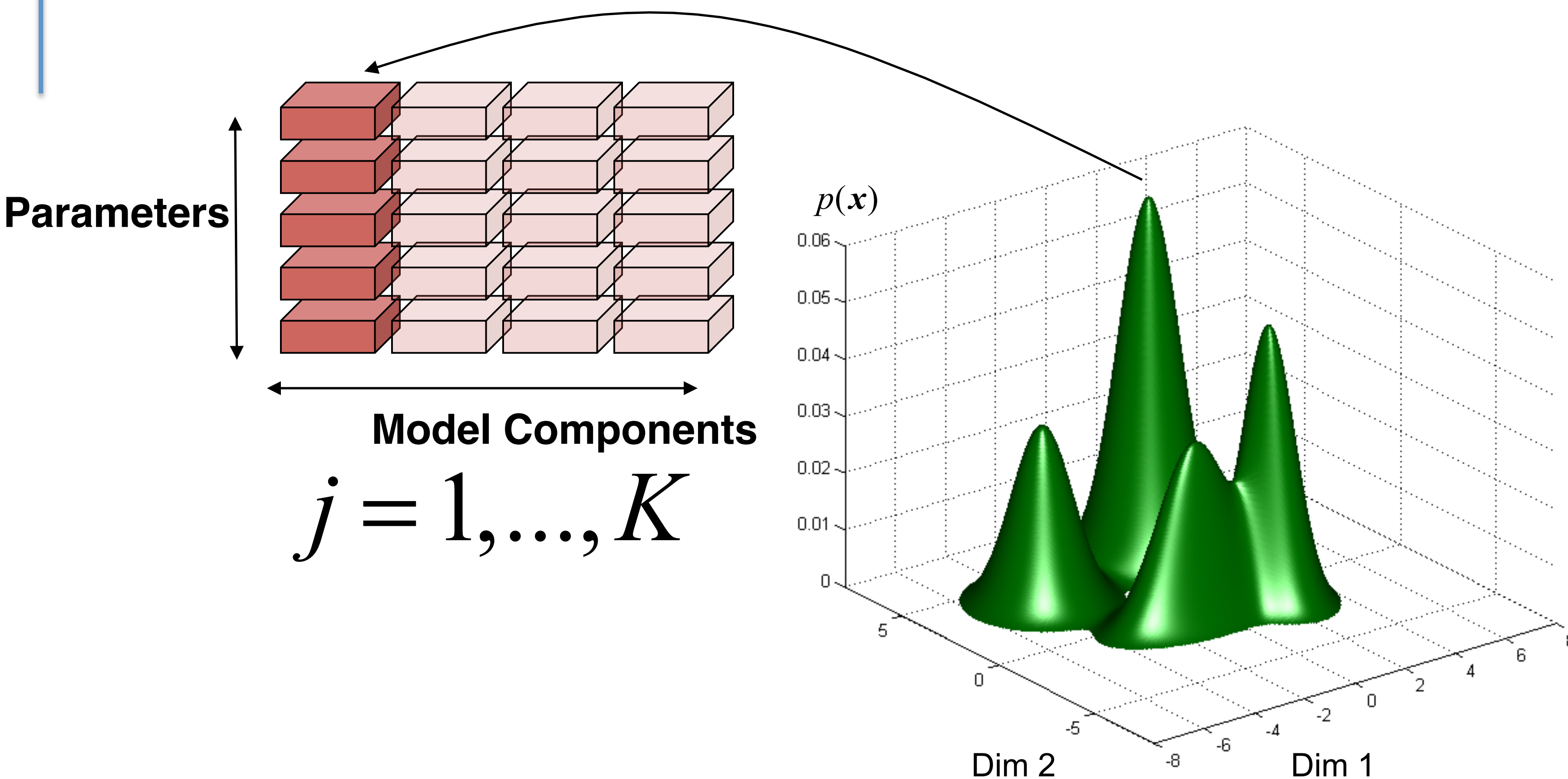
$$w_j \equiv p(\mu = \mu_j)$$



Recognition Systems
Gaussian Mixture Models



Recognition Systems
Gaussian Mixture Models



Learning a Gaussian Mixture

- Probability Model

A Gaussian mixture model (GMM) represents as the weighted sum of multiple Gaussian distributions

$$p(\vec{x} = \vec{x}_i)$$

$$= \sum_j p(\vec{x} = \vec{x}_i, \vec{\mu} = \vec{\mu}_j)$$

Total low of probability

$$= \sum_j p(\vec{\mu} = \vec{\mu}_j) p(\vec{x} = \vec{x}_i \mid \vec{\mu} = \vec{\mu}_j)$$

Chain rule

$$= \sum_j p(\vec{\mu} = \vec{\mu}_j) \frac{1}{(2\pi)^{p/2} \left|\Sigma_j\right|^{1/2}} \mathrm{e}^{-\frac{1}{2}\left(\vec{x} - \vec{\mu}_j\right)^T \Sigma_j^{-1} \left(\vec{x} - \vec{\mu}_j\right)}$$

Learning a Gaussian Mixture (when assuming with known shared covariance)

$$p(\vec{x}=\vec{x}_i)$$

$$=\sum_{\mu_j} p(\vec{x}=\vec{x}_i,\vec{\mu}=\vec{\mu}_j)$$

$$=\sum_j p(\vec{\mu}=\vec{\mu}_j)p(\vec{x}=\vec{x}_i\,|\,\vec{\mu}=\vec{\mu}_j)$$

$$=\sum_j p(\vec{\mu}=\vec{\mu}_j)\frac{1}{(2\pi)^{p/2}|\Sigma|^{1/2}}\mathrm{e}^{-\frac{1}{2}\left(\vec{x}_i-\vec{\mu}_j\right)^T\Sigma^{-1}\left(\vec{x}_i-\vec{\mu}_j\right)}$$

Assuming

Max Log-likelihood of Observed Data Samples

- Log-likelihood of data $logp(x_1, x_2, x_3, ..., x_n) =$

$$\log \prod_{i=1..n} \sum_{j=1..K} p(\vec{\mu} = \vec{\mu}_j) \frac{1}{(2\pi)^{p/2} \left|\Sigma_j\right|^{1/2}} e^{-\frac{1}{2}\left(\vec{x}_i - \vec{\mu}_j\right)^T \Sigma_j^{-1} \left(\vec{x}_i - \vec{\mu}_j\right)}$$

Apply MLE to find $\left\{\{p(\vec{\mu} = \mu_j)\}, j = 1...K\right\}$

optimal Gaussian parameters $\{\vec{\mu}_j, \Sigma_j, j = 1...K\}$

Expectation-Maximization for training GMM

- Start:
  - "Guess" the centroid and covariance for each of the K clusters
  - “Guess” the proportion of clusters, e.g., uniform prob 1/K

- Loop
  - For each point, revising its proportions belonging to each of the K clusters
  - For each cluster, revising both the mean (centroid position) and covariance (shape)

Learning a Gaussian Mixture

(when assuming with known shared covariance)

E-Step

[Bayes Rule]

$$E[z_{ij}] = p(\vec{\mu} = \mu_j \mid x = x_i)$$

assignment.

soft

$$= \frac{p(x = x_i \mid \mu = \mu_j)\, p(\mu = \mu_j)}{\sum_{s=1}^{k} p(x = x_i \mid \mu = \mu_s)\, p(\mu = \mu_s)}$$

$$= \frac{\frac{1}{(2\pi)^{p/2}|\Sigma|^{1/2}} e^{-\frac{1}{2}(\vec{x}_i - \vec{\mu}_j)^T \Sigma^{-1}(\vec{x}_i - \vec{\mu}_j)} \quad p(\mu = \mu_j)}{\sum_{s=1}^{k} \frac{1}{(2\pi)^{p/2}|\Sigma|^{1/2}} e^{-\frac{1}{2}(\vec{x}_i - \vec{\mu}_s)^T \Sigma^{-1}(\vec{x}_i - \vec{\mu}_s)} \quad p(\mu = \mu_s)}$$

Learning a Gaussian Mixture
(when assuming with known shared covariance)

$m_{ij} = \begin{cases} 0 \\ 1 \end{cases}$

E-Step

$$E[z_{ij}] = p(\mu = \mu_j \mid x = x_i)$$

Soft assignment
$p(u = \mu_j \mid x = x_i)$
How x_i belongs in proportion to cluster $\{1, 2, \dots, k\}$

vs. m_{ij} Hard assignment in k-means

$$= \frac{p(x = x_i \mid \mu = \mu_j)\, p(\mu = \mu_j)}{\sum_{s=1}^{k} p(x = x_i \mid \mu = \mu_s)\, p(\mu = \mu_s)}$$

$$= \frac{\cancel{\dfrac{1}{(2\pi)^{p/2}\left|\Sigma\right|^{1/2}}}\, e^{-\frac{1}{2}\left(\vec{x}-\vec{\mu}_j\right)^T \Sigma^{-1}\left(\vec{x}-\vec{\mu}_j\right)} \quad p(\mu = \mu_j)}{\sum_{s=1}^{k} \cancel{\dfrac{1}{(2\pi)^{p/2}\left|\Sigma\right|^{1/2}}}\, e^{-\frac{1}{2}\left(\vec{x}-\vec{\mu}_s\right)^T \Sigma^{-1}\left(\vec{x}-\vec{\mu}_s\right)} \quad p(\mu = \mu_s)}$$

Learning a Gaussian Mixture

when assuming with known shared covariance

M-Step

$$\mu_j^{(t+1)} \leftarrow \frac{1}{\sum_{i=1}^{n} E[z_{ij}]^{(t)}} \sum_{i=1}^{n} E[z_{ij}]^{(t)} x_i$$

$$p(\mu = \mu_j)^{(t+1)} \leftarrow \frac{1}{n} \sum_{i=1}^{n} E[z_{ij}]^{(t)}$$

Covariance: Σ_j (j: 1 to K) can also be derived in the M-step under a full setting

Learning a Gaussian Mixture

when assuming with known shared covariance

k mean => centroid $= \frac{1}{N_j} \sum_{i=1} m_{ij} X_i$

M-Step

$$\mu_j^{(t+1)} \leftarrow \frac{1}{\sum_{i=1}^{n} E[z_{ij}]} \sum_{i=1}^{n} \underbrace{E[z_{ij}]^{(t)}}_{\rightarrow [0,1]} x_i$$

$$\sum_{j=1}^{K} E[z_{ij}] = 1$$

$$p(\mu = \mu_j)^{(t+1)} \leftarrow \frac{1}{n} \sum_{i=1}^{n} E[z_{ij}]^{(t)}$$

Covariance: Σ_j (j: 1 to K) will also be derived in the M-step under a full setting

M-step for Estimating unknown Covariance Matrix (more general, details in EM-Extra lecture)

$$\Sigma_j^{(t+1)} = \frac{\sum_{i=1}^{n} E[z_{ij}]^{(t)} (x_i - \mu_j^{(t+1)})(x_i - \mu_j^{(t+1)})^T}{\sum_{i=1}^{n} E[z_{ij}]^{(t)}}$$

for small TrainSet
too many parameters
to estimate

$j = 1, \ldots, K$

$\Sigma_j \Rightarrow O(p^2/2)$

Σ_j $\Rightarrow O(Kp^2/2)$

μ_j $O(Kp + K)$ $p(\mu = \mu_j)$

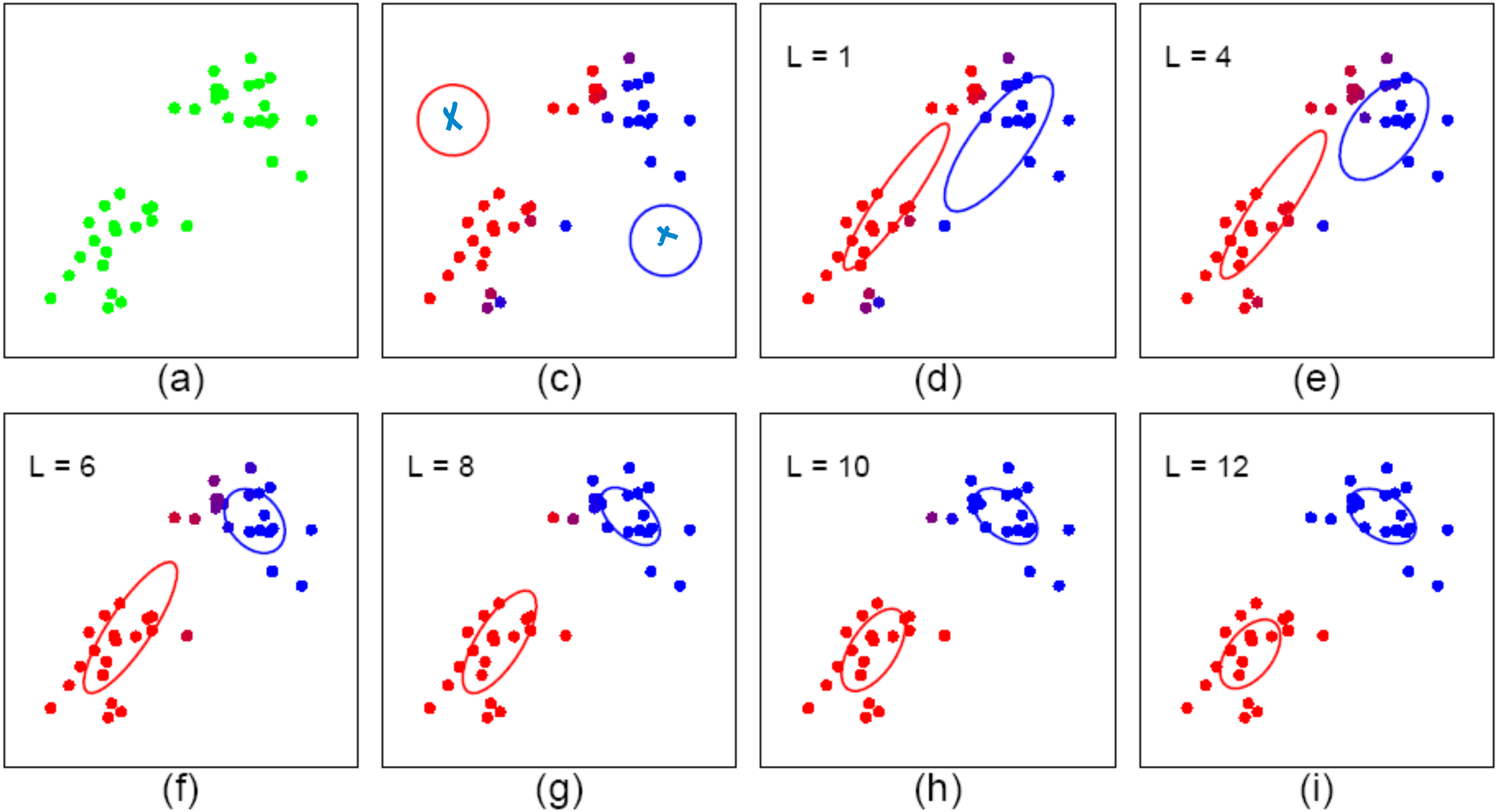
$E[z_{ij}]$ $O(Kn)$

Recap: Expectation-Maximization for training GMM

- Start:
  - "Guess" the centroid and covariance for each of the K clusters
  - "Guess" the proportion of clusters, e.g., uniform prob 1/K

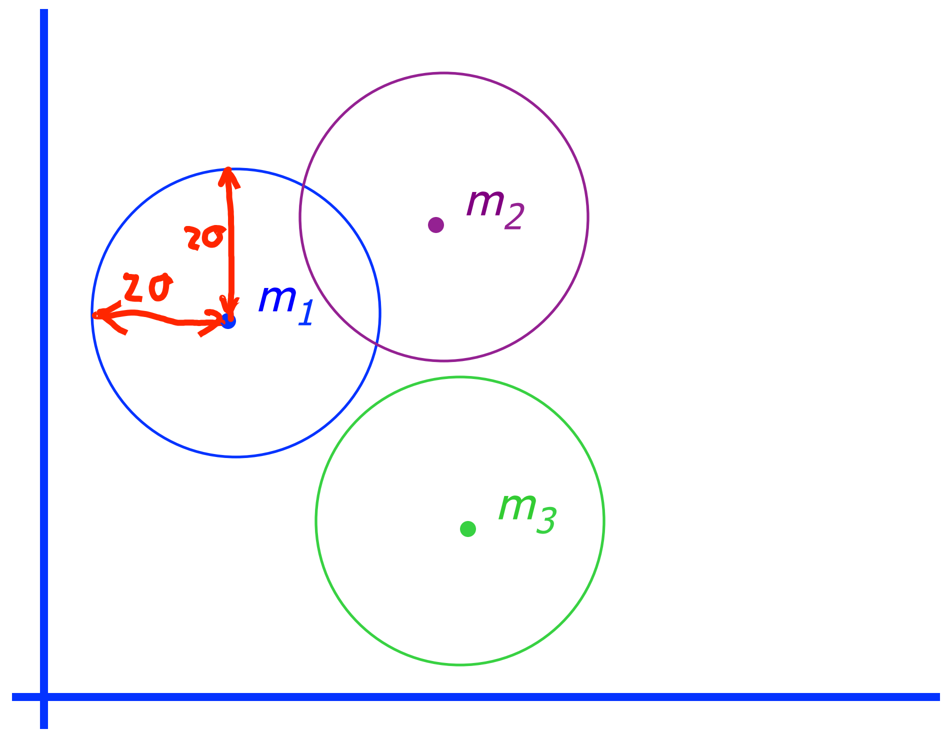
- Loop
  - For each point, revising its proportions belonging to each of the K clusters
  - For each cluster, revising both the mean (centroid position) and covariance (shape)

each cluster, revising both the mean (centroid position) and covariance (shape)



The Simplest GMM assumption

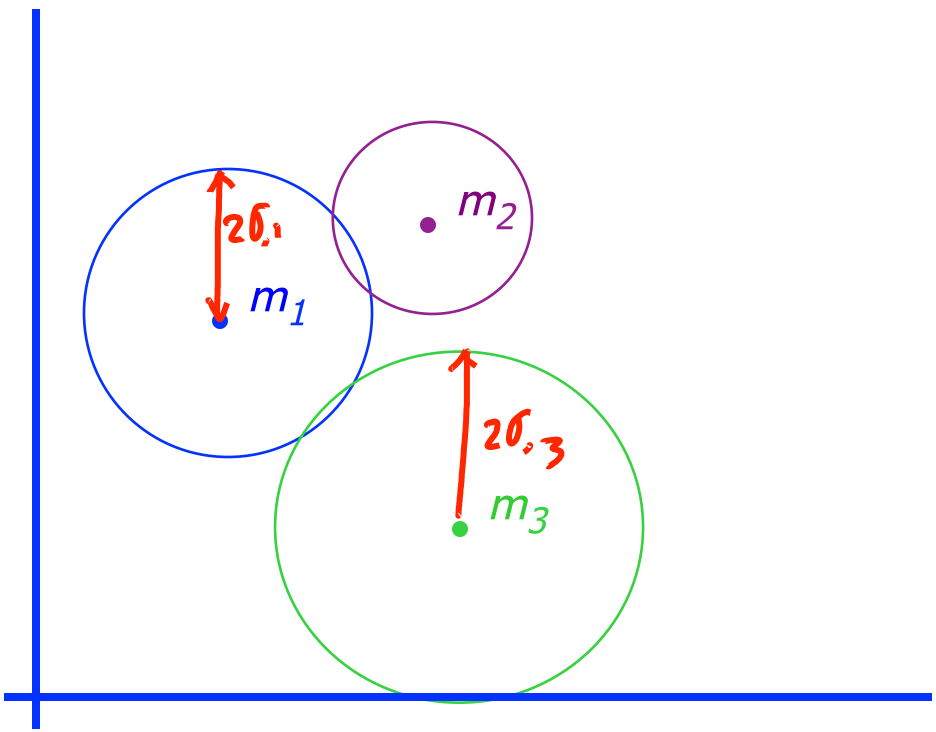
- Each component generates data from a Gaussian with
  - mean μ_i
  - Shared covariance matrix $\sigma^2 \boldsymbol{I}$



$$\Sigma_j = \Sigma = \begin{bmatrix} \sigma^2 & 0 \\ 0 & \sigma^2 \end{bmatrix}$$

A Simple GMM assumption

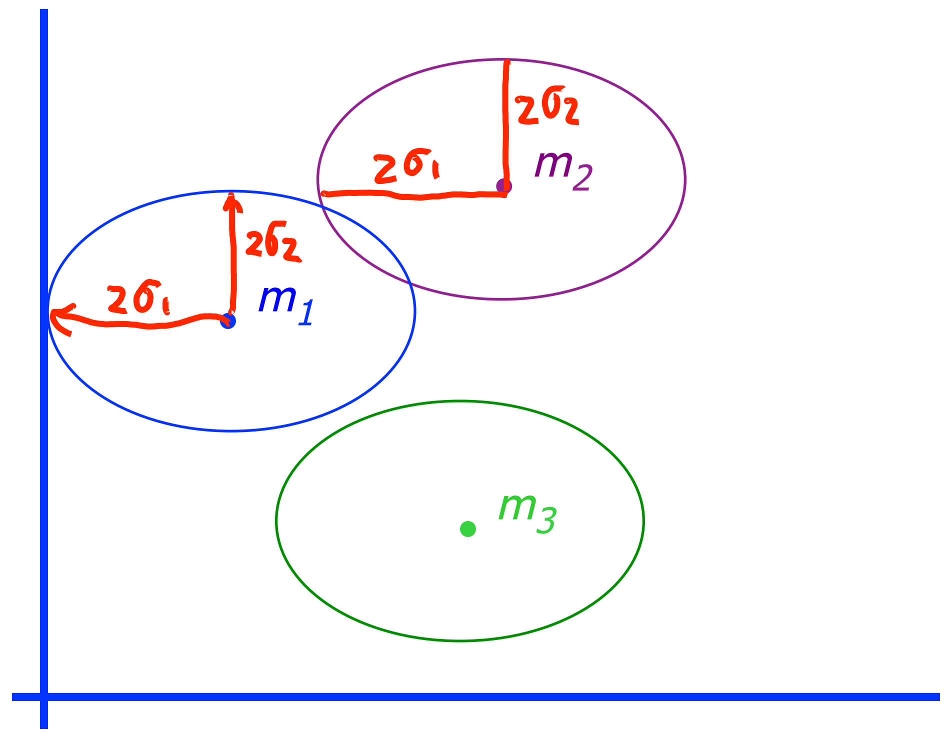
- Each component generates data from a Gaussian with
  - mean μ_i
  - Cluster-specific covariance matrix as $\sigma_j^2 \boldsymbol{I}$



$$\Sigma_j = \sigma_j^2 \boldsymbol{I} = \begin{bmatrix} \sigma_j^2 & 0 \\ 0 & \sigma_j^2 \end{bmatrix}$$

Another Simple GMM assumption

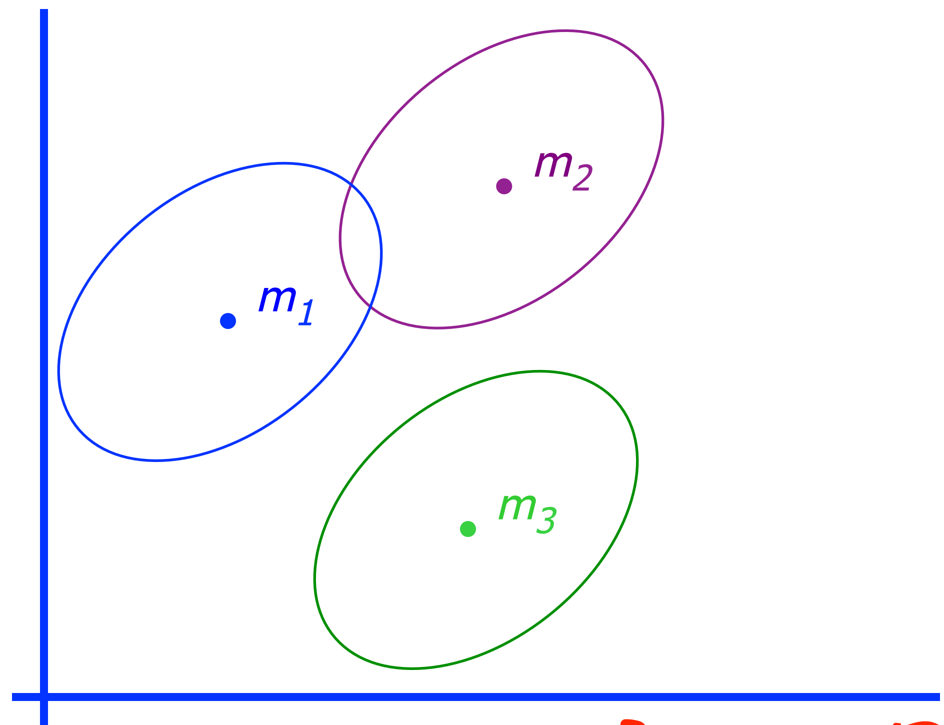
- Each component generates data from a Gaussian with
  - mean μ_i
  - Shared covariance matrix as diagonal matrix



$$\Sigma_j = \Sigma = \begin{bmatrix} \sigma_1^2 & 0 \\ 0 & \sigma_2^2 \end{bmatrix}$$

A bit More General GMM assumption

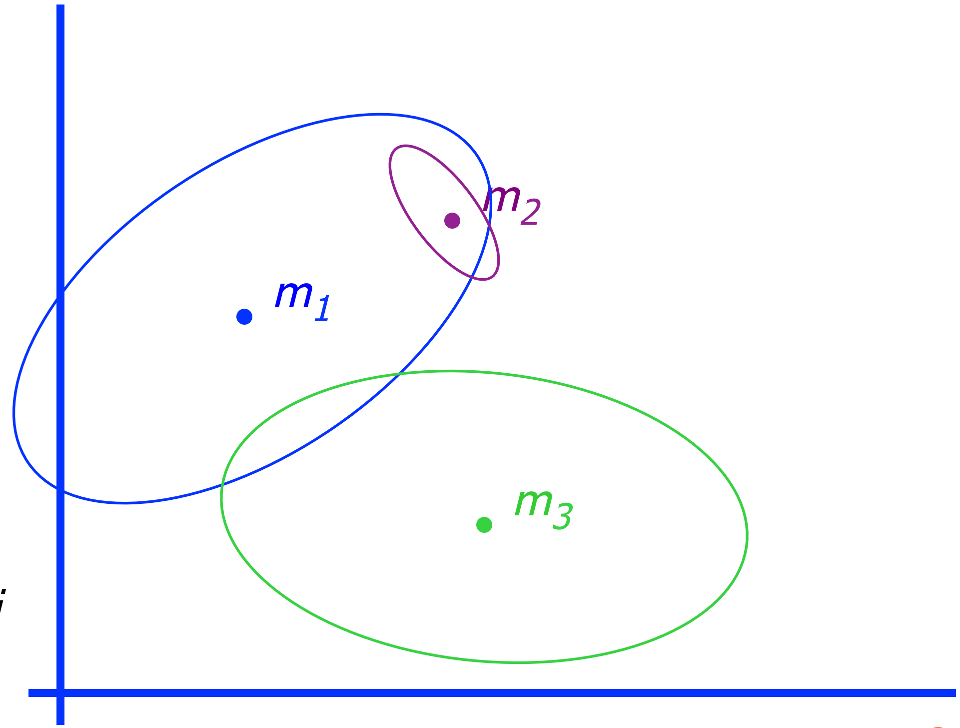
- Each component generates data from a Gaussian with
  - mean μ_i
  - Shared covariance matrix as full matrix



$$\Sigma_j = \Sigma = \begin{bmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{bmatrix}$$

The General GMM assumption

- Each component generates data from a Gaussian with
  - mean μ_i
  - covariance matrix Σ_i



$$\Sigma_j = \begin{bmatrix} \sigma_{1j} & cov_j(x_1, x_2) \\ cov_j(x_1, x_2) & \sigma_{2j} \end{bmatrix}$$

Partitional : Gaussian Mixture Model

- 1. Review of Gaussian Distribution
- 2. GMM for clustering : basic algorithm
- 3. GMM connecting to K-means
- 4. GMM examples
- 5. Problems of GMM and K-means

Recap: K-means iterative learning

$$\underset{\{\vec{C}_j, m_{i,j}\}}{\arg\min} \sum_{j=1}^{K} \sum_{i=1}^{n} m_{i,j} \left(\vec{x}_i - \vec{C}_j \right)^2$$

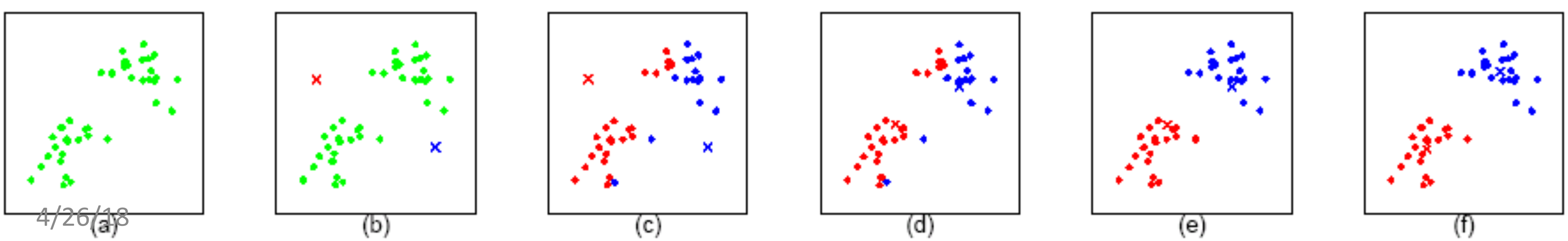
Memberships $\{m_{i,j}\}$ and centers $\{C_j\}$ are correlated.

E-Step Given centers $\{\vec{C}_j\}$, $m_{i,j} = \begin{cases} 1 & j = \arg\min_k (\vec{x}_i - \vec{C}_j)^2 \\ 0 & \text{otherwise} \end{cases}$

M-Step Given memberships $\{m_{i,j}\}$, $\vec{C}_j = \dfrac{\sum_{i=1}^{n} m_{i,j} \vec{x}_i}{\sum_{i=1}^{n} m_{i,j}}$

Compare: K-means

- The EM algorithm for mixtures of Gaussians is like a "soft version" of the K-means algorithm.
- In the K-means "E-step" we do hard assignment:
- In the K-means "M-step" we update the means as the weighted sum of the data, but now the weights are 0 or 1:



K-means: $$\arg\min_{\{\vec{C}_j, m_{i,j}\}} \sum_{j=1}^{K} \sum_{i=1}^{n} m_{i,j} \left(\vec{x}_i - \vec{C}_j \right)^2$$

$$m_{ij} = \begin{cases} 0 \\ 1 \end{cases}$$

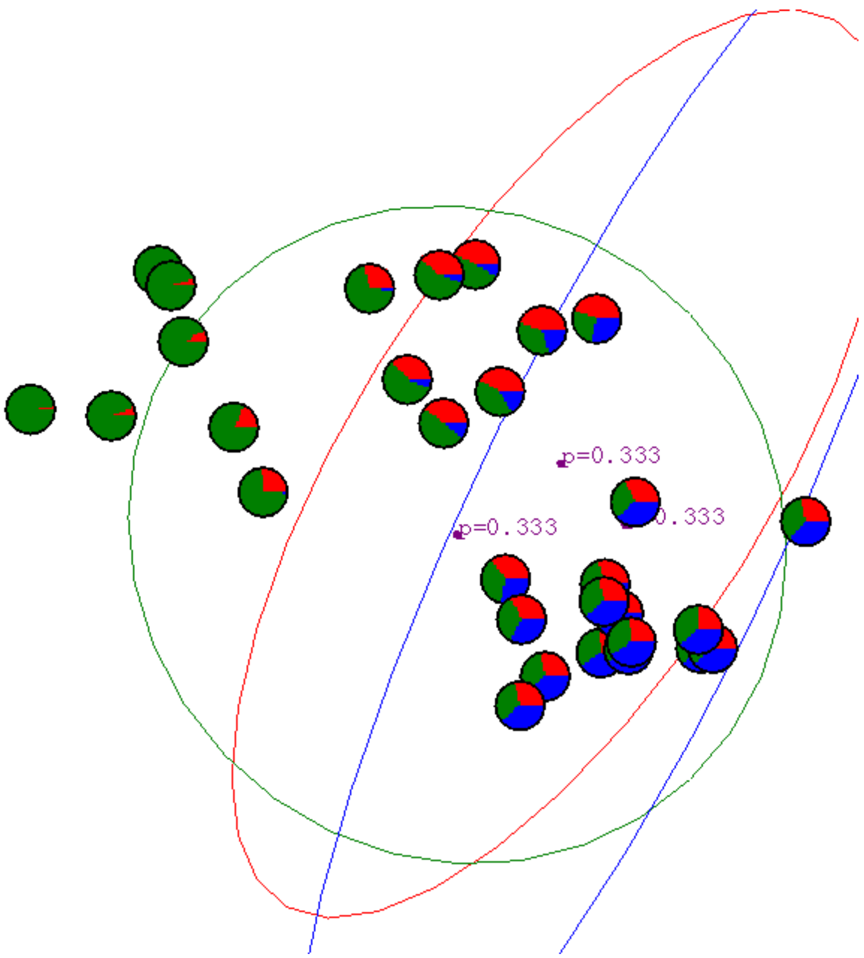
GMM: $$\sum_i \log \prod_{i=1}^{n} p(x = x_i) = \overbrace{\sum_i}^{i=1..n} \log \left[\overbrace{\sum_{\mu_j}}^{j=1,..k} p(\mu = \mu_j) \frac{1}{(2\pi)|\Sigma|^{1/2}} \mathrm{e}^{-\frac{1}{2}(\vec{x}-\vec{\mu}_j)^T \Sigma^{-1} (\vec{x}-\vec{\mu}_j)} \right]$$

- K-Mean only detect spherical clusters.
- GMM can adjust its self to elliptic shape clusters.

Partitional : Gaussian Mixture Model

- 1. Review of Gaussian Distribution
- 2. GMM for clustering : basic algorithm
- 3. GMM connecting to K-means
- 4. GMM examples
- 5. Problems of GMM and K-means

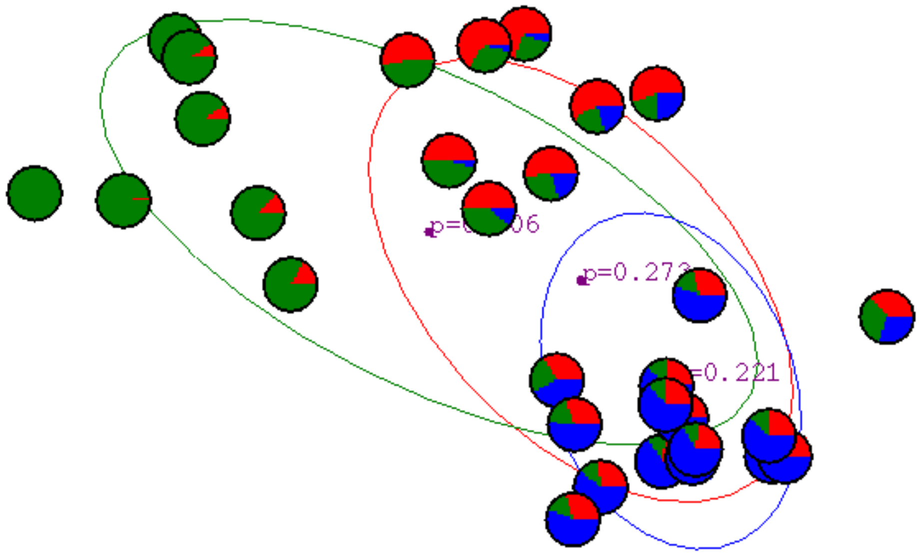
Gaussian Mixture Example: Start



$p(\mu_j | x_i)$

After First Iteration

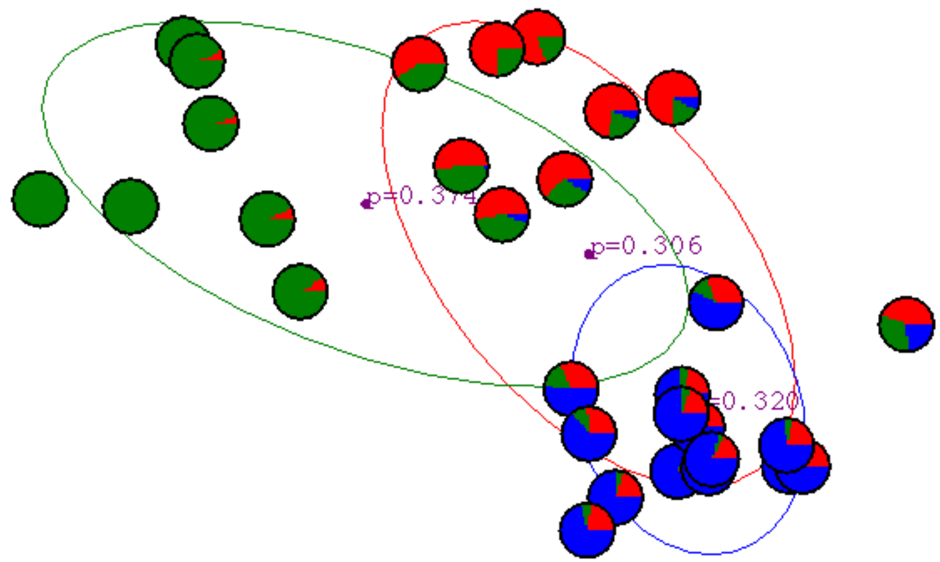
For each point, revising its proportions belonging to each of the K clusters



For each cluster, revising its mean (centroid position), covariance (shape) and proportion in the mixture

After 2nd Iteration

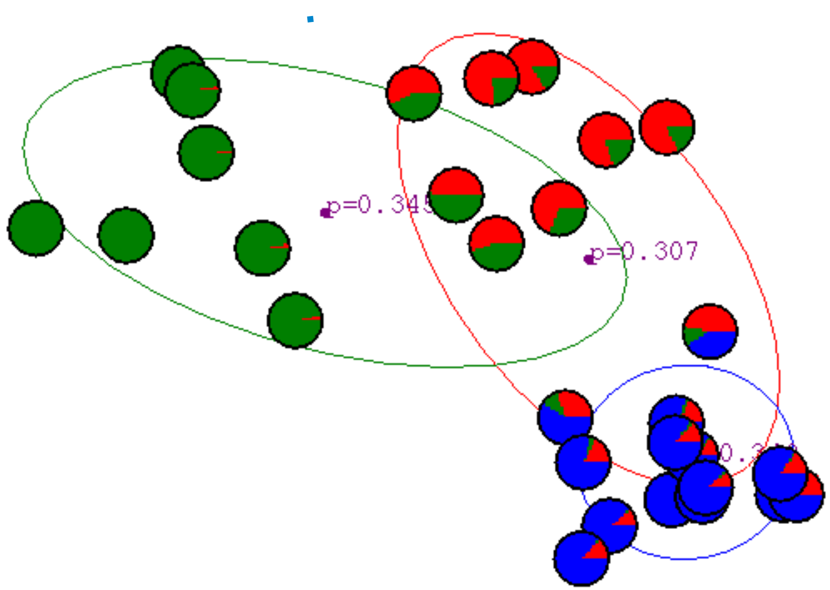
For each point, revising its proportions belonging to each of the K clusters



For each cluster, revising its mean (centroid position), covariance (shape) and proportion in the mixture

After 3rd Iteration

For each point, revising its proportions belonging to each of the K clusters

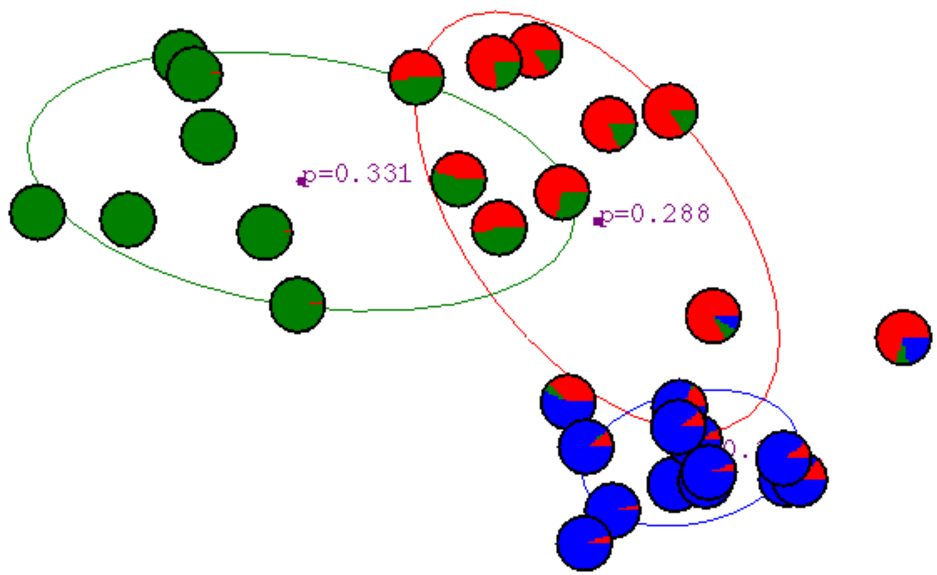


$p(u_j)$

For each cluster, revising its mean (centroid position), covariance (shape) and proportion in the mixture

After 4th Iteration

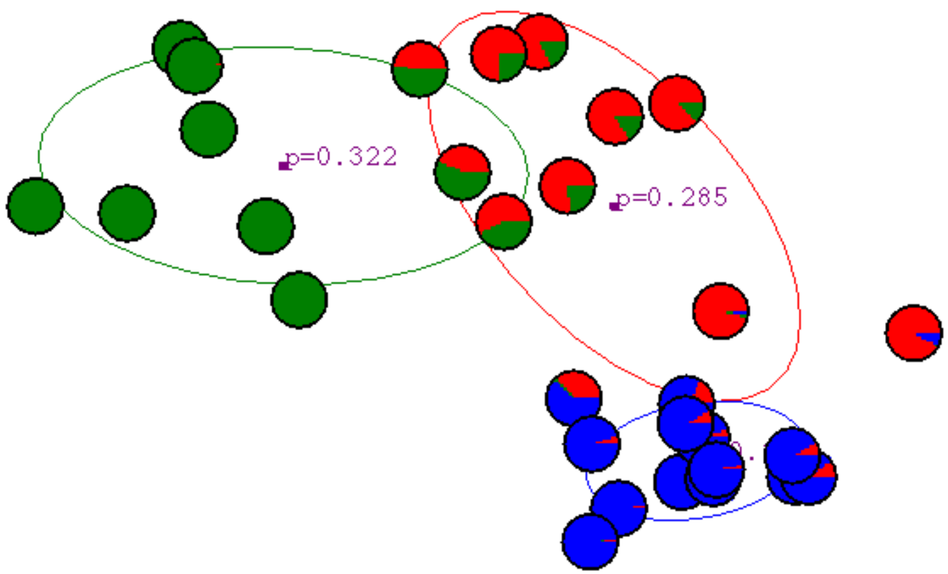
For each point, revising its proportions belonging to each of the K clusters



For each cluster, revising its mean (centroid position), covariance (shape) and proportion in the mixture

After 5th Iteration

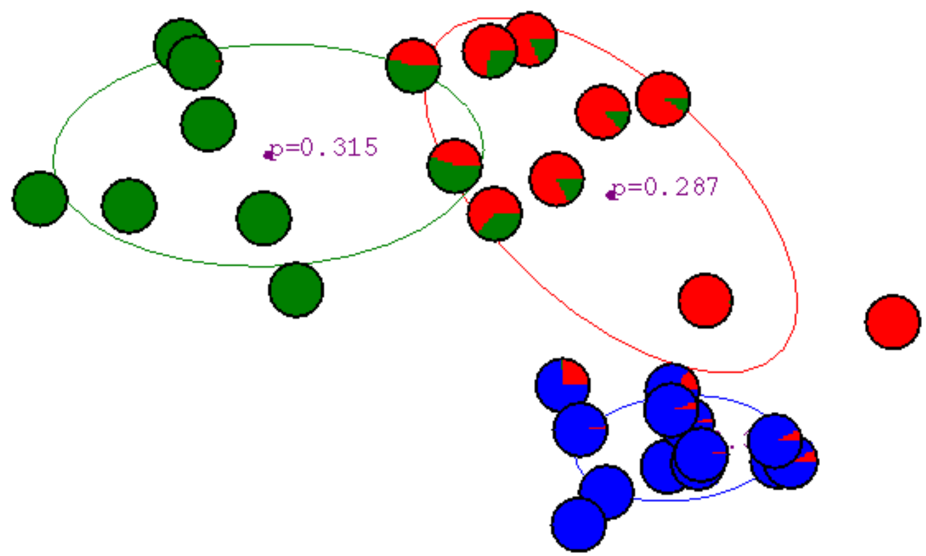
For each point, revising its proportions belonging to each of the K clusters



For each cluster, revising its mean (centroid position), covariance (shape) and proportion in the mixture

After 6th Iteration

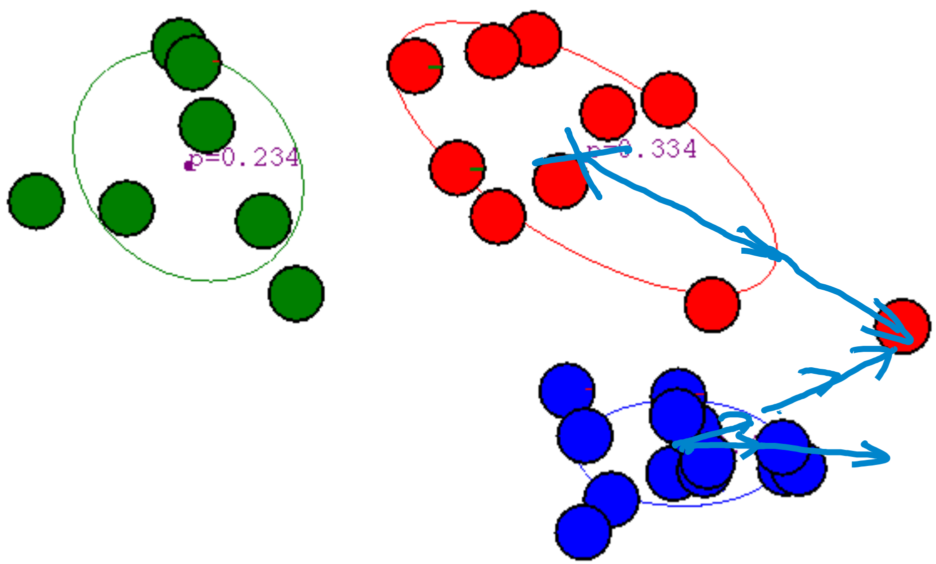
For each point, revising its proportions belonging to each of the K clusters



For each cluster, revising its mean (centroid position), covariance (shape) and proportion in the mixture

After 20th Iteration

For each point, revising its proportions belonging to each of the K clusters



For each cluster, revising its mean (centroid position), covariance (shape) and proportion in the mixture

(3) GMM Clustering

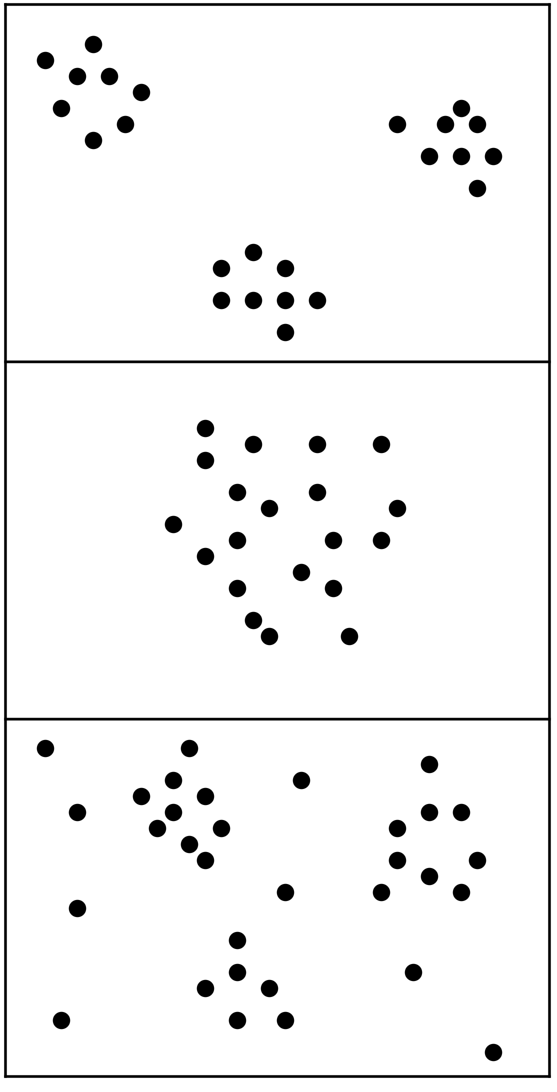
Task	Clustering
↓	↓
Representation	Mixture of Gaussian
↓	↓
Score Function	Likelihood
↓	↓
Search/Optimization	EM algorithm
↓	↓
Models, Parameters	Each point's soft membership & mean / covariance per cluster

$$\sum_i \log \prod_{i=1}^{n} p(x = x_i) = \sum_i \log \left[\sum_{\mu_j} p(\mu = \mu_j) \frac{1}{(2\pi)\left|\Sigma_j\right|^{1/2}} e^{-\frac{1}{2}\left(\vec{x}-\vec{\mu}_j\right)^T \Sigma_j^{-1}\left(\vec{x}-\vec{\mu}_j\right)} \right]$$

Partitional : Gaussian Mixture Model

- 1. Review of Gaussian Distribution
- 2. GMM for clustering : basic algorithm
- 3. GMM connecting to K-means
- 4. GMM examples
- 5. Problems of GMM and K-means

Unsupervised Learning: not as hard as it looks



Sometimes easy

Sometimes impossible

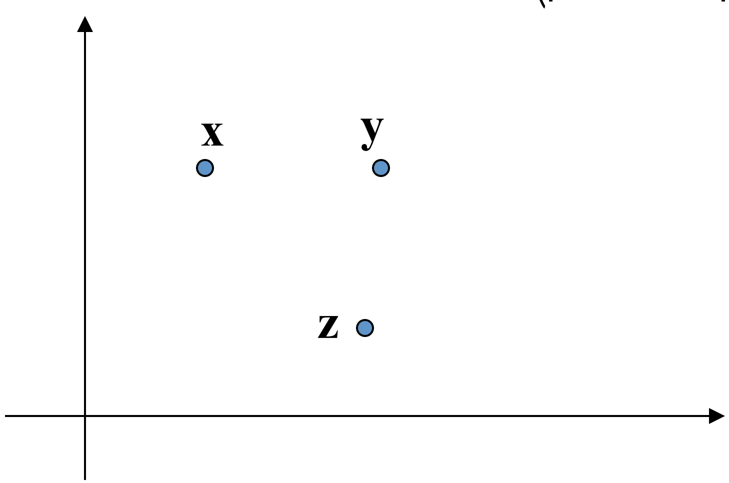
and sometimes in between

Problems (I)

- Both k-means and mixture models need to compute centers of clusters and explicit distance measurement
  - Given strange distance measurement, the center of clusters can be hard to compute

E.g.,

$$\left\|\vec{x} - \vec{x}'\right\|_\infty = \max\left(\left|x_1 - x_1'\right|, \left|x_2 - x_2'\right|, \ldots, \left|x_p - x_p'\right|\right)$$

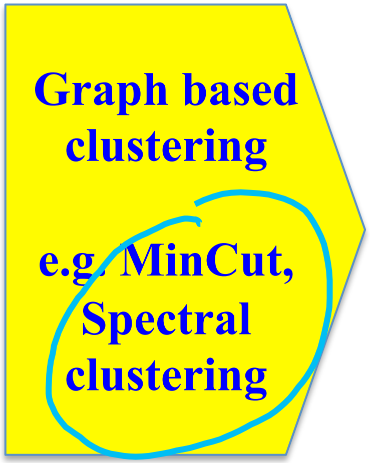


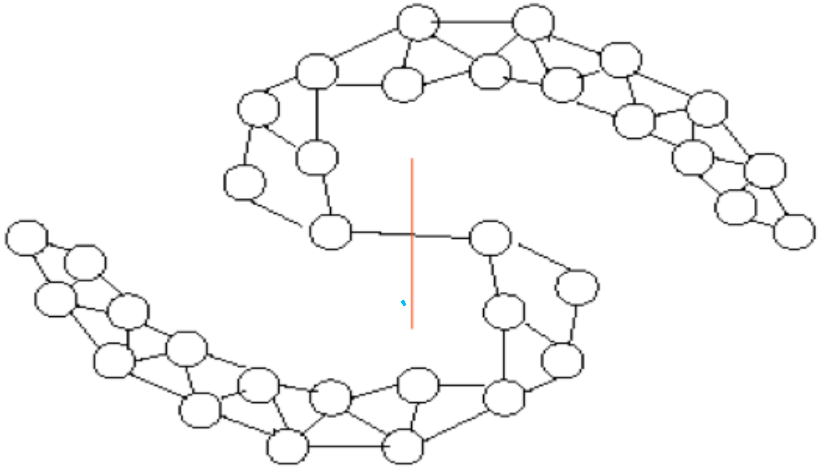
$$\|\mathbf{x} - \mathbf{y}\|_\infty = \|\mathbf{x} - \mathbf{z}\|_\infty$$

Problem (II)

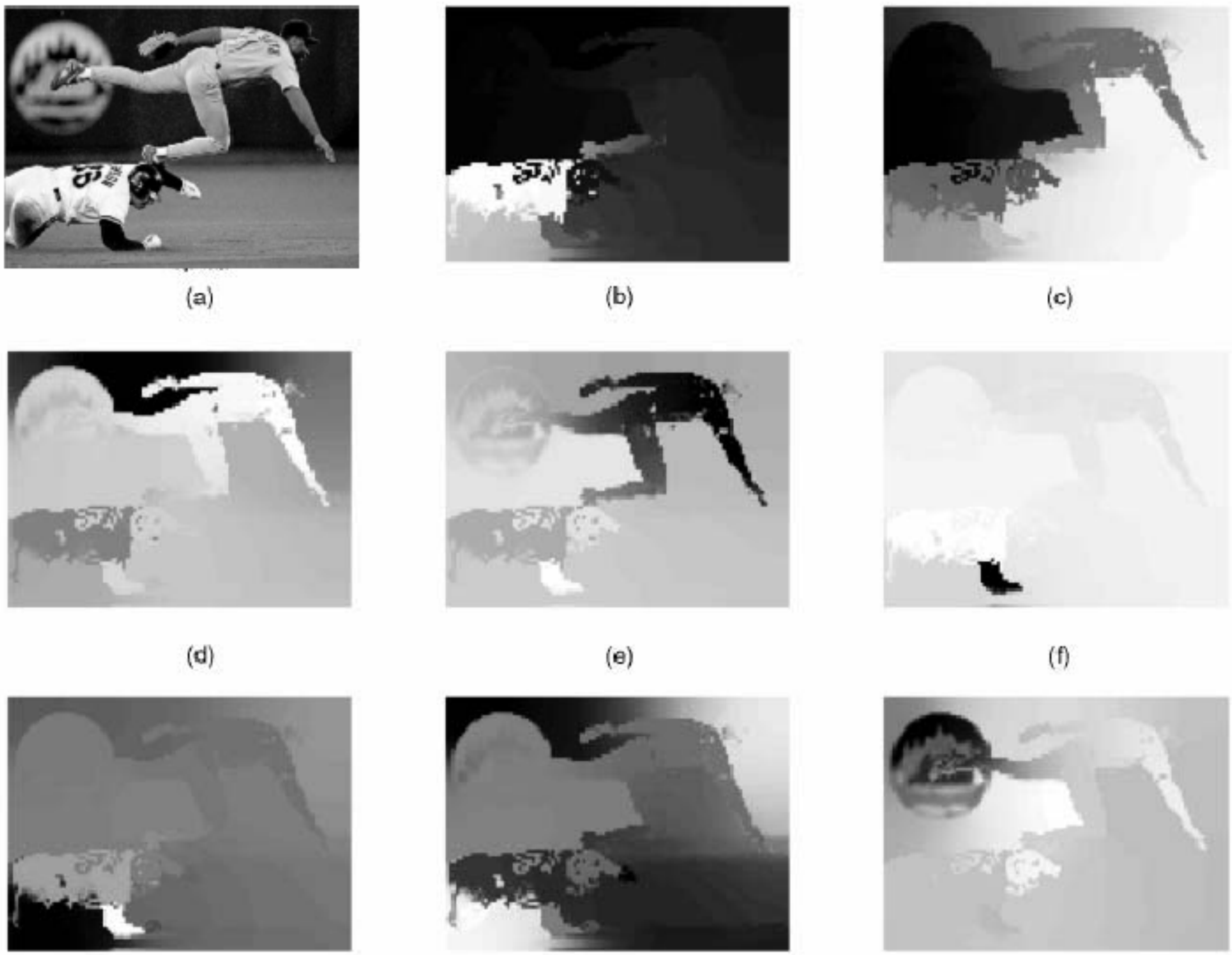
tight

- Both k-means and mixture models look for compact clustering structures
  - In some cases, connected clustering structures are more desirable





e.g. Image Segmentation through minCut



References

- Hastie, Trevor, et al. *The elements of statistical learning*. Vol. 2. No. 1. New York: Springer, 2009.

- Big thanks to Prof. Eric Xing @ CMU for allowing me to reuse some of his slides
- Big thanks to Prof. Ziv Bar-Joseph @ CMU for allowing me to reuse some of his slides
- clustering slides from Prof. Rong Jin @ MSU