

# Computational Data Analysis

## Machine Learning

**Yao Xie, Ph.D.**

*Associate Professor*

Harold R. and Mary Anne Nash Early Career Professor  
H. Milton Stewart School of Industrial and Systems  
Engineering

Gaussian Mixture Model and  
EM Algorithm



# Gaussian mixture model

A density model  $p(X)$  may be multi-modal: model it as a mixture of uni-modal distributions (e.g. Gaussians) for  $n$ -dimensional observations

$$\mathcal{N}(X|\mu, \Sigma) := \frac{1}{|\Sigma|^{\frac{1}{2}}(2\pi)^{\frac{n}{2}}} \exp\left(-\frac{1}{2}(X - \mu)^\top \Sigma^{-1}(X - \mu)\right)$$

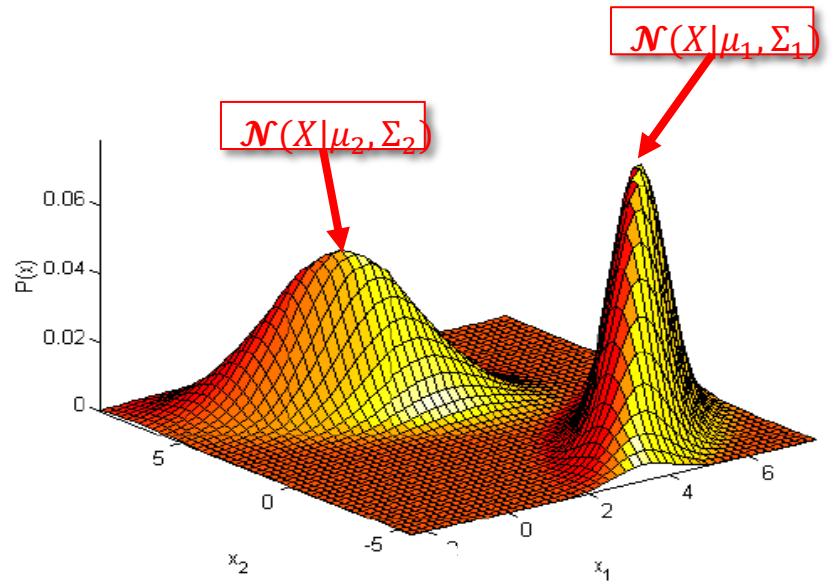
Consider a mixture of  $K$  Gaussians

$$p(X) = \sum_{k=1}^K \pi_k \mathcal{N}(X|\mu_k, \Sigma_k)$$

Parametric or nonparametric?

Learn  $\pi_k \in (0,1), \mu_k, \Sigma_k$ ;

Constraint  $\sum_{k=1}^K \pi_k$



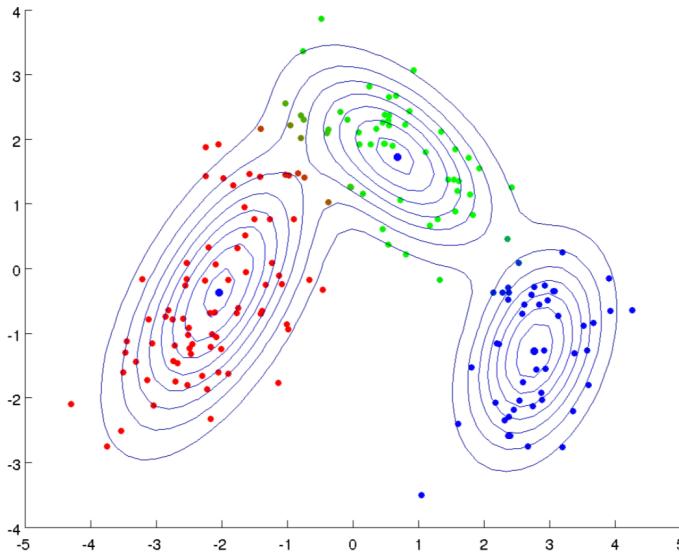
# Motivation: Wine data

- The wine data set was introduced by Forina et al. (1986).
- It originally included the results of 27 chemical measurements on 178 wines made in the same region in Italy but derived from three different cultivars: Barolo, Grignolino and Barbera.
- We extract the first two principle components of the data, and aim to fit a density distribution



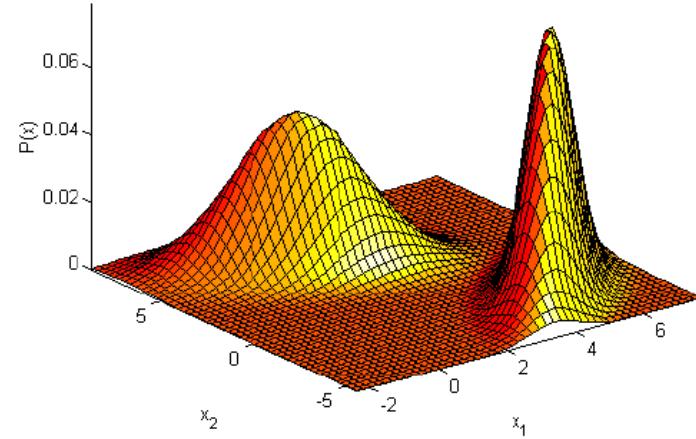
# Mixture of 3 Gaussians

- First run PCA to reduce the dimension to 2
- $k = 1 \text{ or } 2 \text{ or } 3$
- Use  $\tau_1^i$  as the proportion of red,  $\tau_2^i$  proportion of green, and  $\tau_3^i$  proportion of green



# Understanding GMM

- For each data point  $x^i$ :
  - Randomly choose a mixture component,  $z^i = \{1, 2, \dots, K\}$ , with probability  $\pi_{z^i}$
  - Then sample the actual value of  $x^i$  from a Gaussian distribution  $\mathcal{N}(x| \mu_{z^i}, \Sigma_{z^i})$
- Joint distribution over  $p(x, z)$
- $p(x, z) = \pi_z \mathcal{N}(x| \mu_z, \Sigma_z)$
- Marginal distribution  $p(x)$
- $p(x) = \sum_{z=1}^K p(x, z) = \sum_{z=1}^K p(x|z)p(z)$



# Learning Parameters

- How to learn?
- Maximum likelihood learning (let  $\theta = (\pi_k, \mu_k, \Sigma_k)$ ,  $k = 1 \dots K$ )
- $\theta^* = \operatorname{argmax} l(\theta; D) = \log \prod_{i=1}^m p(x^i)$
- Write down the log-likelihood function

$$l(\theta; D) = \log \prod_{i=1}^m \left( \sum_{k=1}^K p(x^i, z^i = k | \theta) \right)$$

- However, we do not know latent factors  $z^i$  thus cannot evaluate  $l(\theta; D)$  directly
- Idea: imputing missing information: taking “expectation” with respect to unknown latent factors

# Details of EM

- We intend to learn the parameters that maximizes the log-likelihood function

$$l(\theta; D) = \log \prod_{i=1}^m \left( \sum_{z^i=1}^K p(x^i, z^i | \theta) \right)$$

- Expectation step (E-step): take expectation over **posterior** distribution conditioning on data: it can be shown this forms a lower bound (in the  $t$ -th iteration)

$$l(\theta; D) \geq f(\theta) = E_{q(z^1, z^2, \dots, z^m | \theta^t)} [\log \prod_{i=1}^m \left( \sum_{z^i=1}^K p(x^i, z^i | \theta) \right)]$$

- Maximization step (M-step): how to maximize?

$$\theta^{t+1} = \operatorname{argmax}_{\theta} f(\theta)$$

Bayes rule

$$P(z|x) = \frac{P(x|z)P(z)}{P(x)} = \frac{P(x,z)}{\sum_{z'} P(x,z')}$$

likelihood      Prior  
posterior      normalization constant

Prior:  $p(z) = \pi_z$

Likelihood:  $p(x|z) = \mathcal{N}(x|\mu_z, \Sigma_z)$

Posterior:  $p(z|x) = \frac{\pi_z \mathcal{N}(x|\mu_z, \Sigma_z)}{\sum_{z'} \pi_{z'} \mathcal{N}(x|\mu_{z'}, \Sigma_{z'})}$

# E-step: find the posterior distribution

$q(z^1, z^2, \dots, z^m)$ : posterior distribution of the latent variables in  $t$ -th iteration

$$q(z^1, z^2, \dots, z^m) = \prod_{i=1}^m p(z^i | x^i, \theta^t)$$

For each data point  $x^i$ , compute  $p(z^i = k | x^i)$  for each  $k$

$$\tau_k^i = p(z^i = k | x^i, \theta^t) = \frac{p(x^i | z^i = k)p(z^i = k)}{\sum_{k'=1..K} p(z^i = k', x^i)}$$

$$= \frac{\pi_k \mathcal{N}(x^i | \mu_k, \Sigma_k)}{\sum_{k'=1..K} \pi_{k'} \mathcal{N}(x^i | \mu_{k'}, \Sigma_{k'})}$$

# E-step: compute the expectation

$$\begin{aligned} f(\theta) &\coloneqq E_{q(z^1, z^2, \dots, z^m)} \left[ \log \prod_{i=1}^m p(x^i, z^i | \theta) \right] = \sum_{i=1}^m E_{p(z^i | x^i, \theta^t)} [\log p(x^i, z^i | \theta)] \\ &= \sum_{i=1}^m E_{p(z^i | x^i, \theta^t)} [\log \pi_{z^i} \mathcal{N}(x^i | \mu_{z^i}, \Sigma_{z^i})] \end{aligned}$$

Expand log of Gaussian density  $\log \mathcal{N}(x^i | \mu_{z^i}, \Sigma_{z^i})$

$$f(\theta) = \sum_{i=1}^m E_{p(z^i | x^i, \theta^t)} \left[ \log \pi_{z^i} - \frac{1}{2} (x^i - \mu_{z^i})^\top \Sigma_{z^i}^{-1} (x^i - \mu_{z^i}) - \frac{1}{2} \log |\Sigma_{z^i}| - \frac{n}{2} \log(2\pi) \right]$$

$$= \sum_{i=1}^m \sum_{k=1}^K \tau_k^i \left[ \log \pi_k - \frac{1}{2} (x^i - \mu_k)^\top \Sigma_k^{-1} (x^i - \mu_k) - \frac{1}{2} \log |\Sigma_k| - \frac{n}{2} \log(2\pi) \right]$$

# M-step: maximize $f(\theta)$

- $f(\theta) = \sum_{i=1}^m \sum_{k=1}^K \tau_k^i \left[ \log \pi_k - \frac{1}{2} (\mathbf{x}^i - \boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}^i - \boldsymbol{\mu}_k) - \frac{1}{2} \log |\boldsymbol{\Sigma}_k| - \frac{n}{2} \log(2\pi) \right]$

For instance, we want to find  $\pi_k$ , and  $\sum_{k=1}^K \pi_k = 1$

- Form Lagrangian

$$L = \sum_{i=1}^m \sum_{k=1}^K \tau_k^i [\log \pi_k + \text{other terms}] + \lambda (1 - \sum_{i=1}^K \pi_k)$$

- Take partial derivative and set to 0

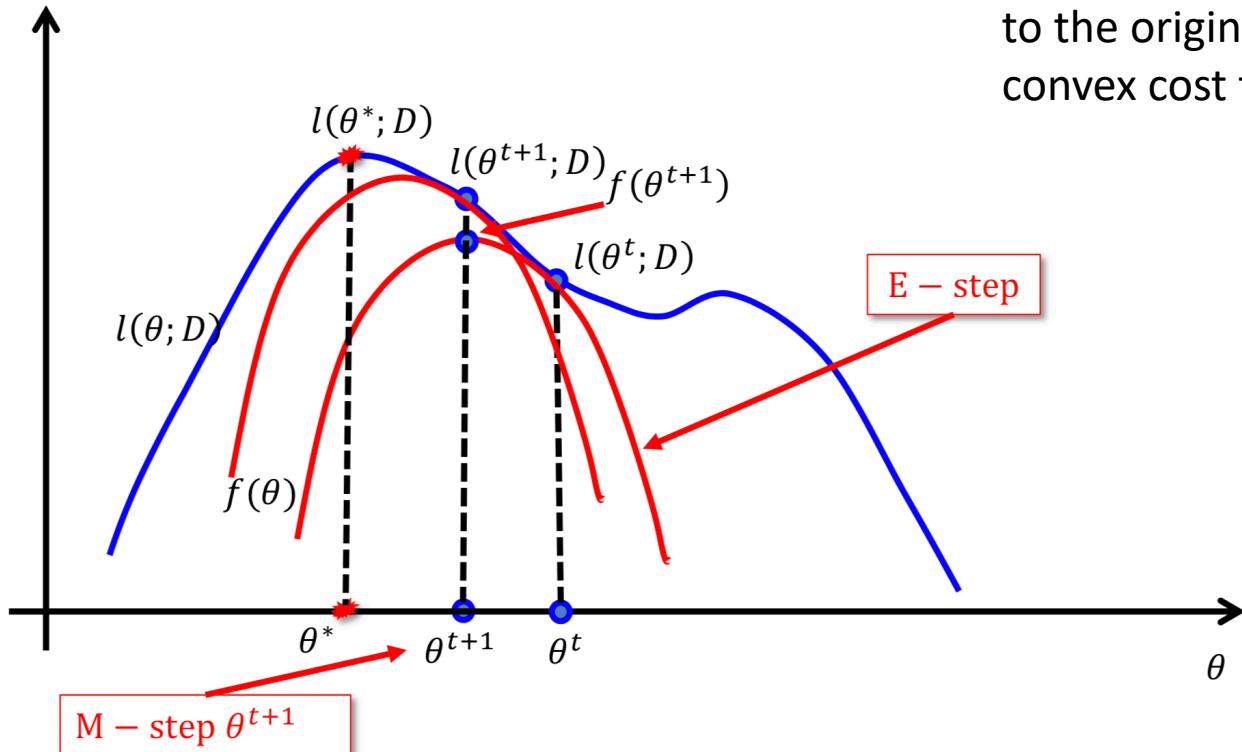
$$\frac{\partial L}{\partial \pi_k} = \sum_{i=1}^m \frac{\tau_k^i}{\pi_k} - \lambda = 0$$

$$\Rightarrow \pi_k = \frac{1}{\lambda} \sum_{i=1}^m \tau_k^i$$

Since  $\sum_{k=1}^K \pi_k = 1$ ,  $\frac{1}{\lambda} \sum_{k=1}^K \sum_{i=1}^m \tau_k^i = 1 \Rightarrow \lambda = m \Rightarrow \pi_k = \frac{1}{m} \sum_{i=1}^m \tau_k^i$

# EM graphically

Maximizing a sequence of quadratic lower bound to the original non-convex cost function.



# EM algorithm

Associate the  $i$ th data and each component with a  $\tau_k^i$

Initialize  $(\pi_k, \mu_k, \Sigma_k)$ ,  $k = 1 \dots K$

Iterate the following two steps till convergence:

- **Expectation step (E-step):** update  $\tau_k^i$  given current  $(\pi_k, \mu_k, \Sigma_k)$

$$\tau_k^i = p(z_k^i = 1 | D, \mu, \Sigma) = \frac{\pi_k \mathcal{N}(x^i | \mu_k, \Sigma_k)}{\sum_{k'=1}^K \pi_{k'} \mathcal{N}(x^i | \mu_{k'}, \Sigma_{k'})}$$

$$(k = 1 \dots K, i = 1 \dots m)$$

- **Maximization step (M-step):** update  $(\pi_k, \mu_k, \Sigma_k)$  given  $\tau_k^i$

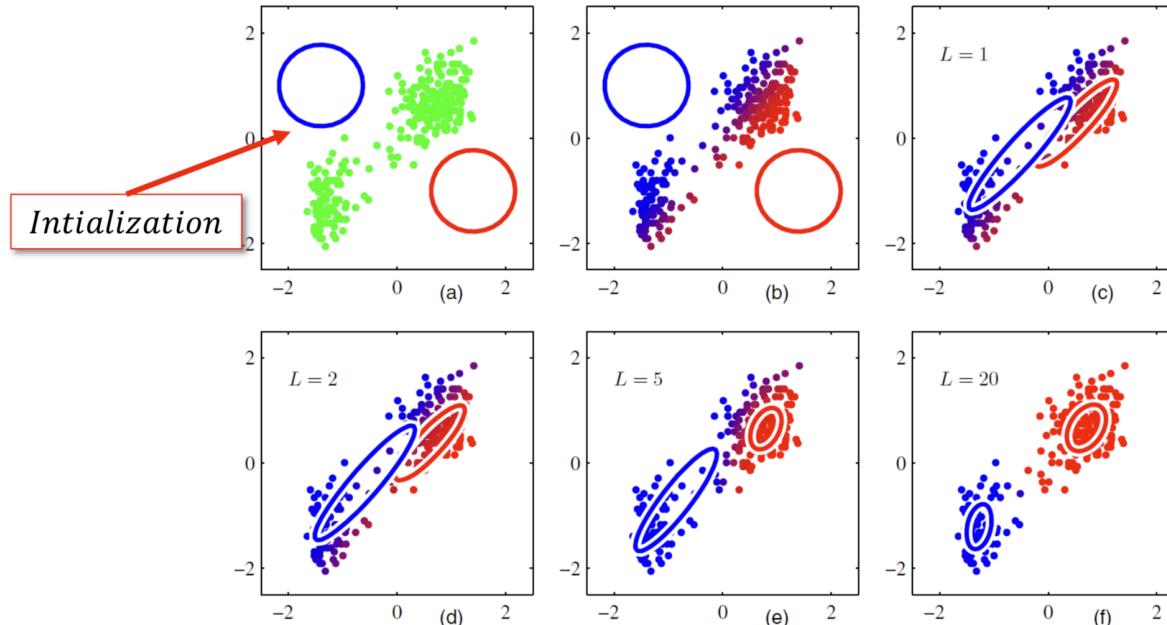
$$\pi_k = \frac{\sum_i \tau_k^i}{m}, \quad \mu_k = \frac{\sum_i \tau_k^i x^i}{\sum_i \tau_k^i}$$

$$\Sigma_k = \frac{\sum_i \tau_k^i (x^i - \mu_k)(x^i - \mu_k)^T}{\sum_i \tau_k^i}$$

$$(k = 1 \dots K, i = 1, \dots, m)$$

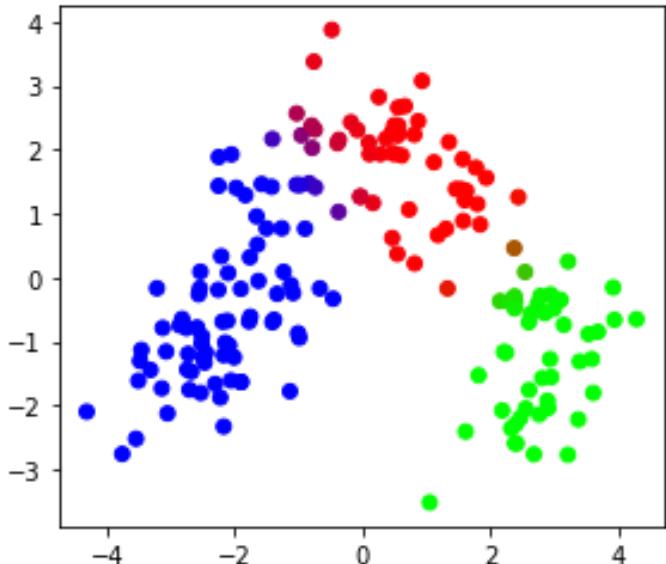
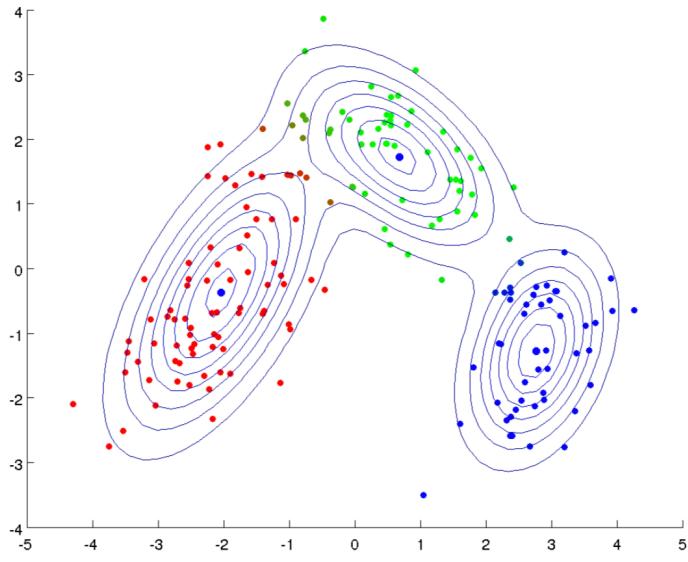
# Expectation-Maximization Iterations

- $k = 1$  or  $2$
- Use  $\tau_1^i$  as the proportion of red, and  $\tau_2^i$  proportion of blue
- Draw only one contour for each Gaussian component



# Demo: Wine data

- First run PCA to reduce the dimension to 2
- $k = 1 \text{ or } 2 \text{ or } 3$
- Use  $\tau_1^i$  as the proportion of red,  $\tau_2^i$  proportion of green, and  $\tau_3^i$  proportion of green



# EM vs. K-means

- EM algorithm for GMM can be viewed as a “soft” clustering algorithm  
Assignment is in probability sense  $\{\pi_k^i, k = 1, \dots, m\}$  for each data point  $i$

- K-means:

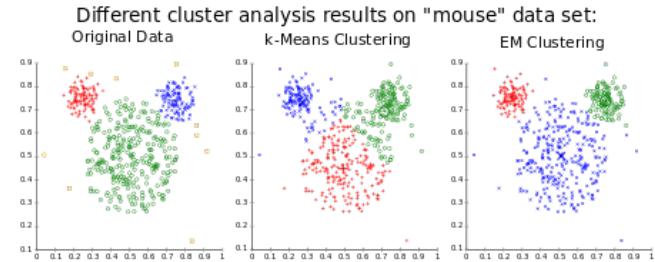
- “E-step”: hard assignment:

$$\bullet z^i = \operatorname{argmax}_k (x^i - \mu_k) \Sigma_k^{-1} (x^i - \mu_k)$$

- “M-step”, we update the means and covariance of cluster using maximum likelihood estimate:

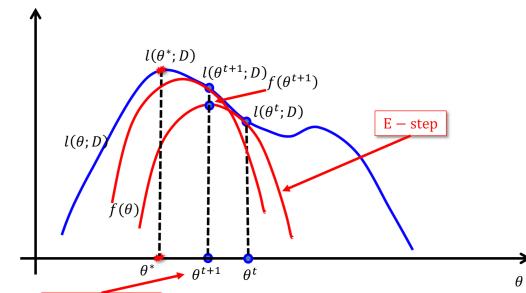
$$\bullet \mu_k = \frac{\sum_i \delta(z^{i,k}) x^i}{\sum_i \delta(z^{i,k})}$$

$$\bullet \Sigma_k = \frac{\sum_i \delta(z^{i,k}) (x^i - \mu_k) (x^i - \mu_k)^T}{\sum_i \delta(z^{i,k})}$$



# History and Theory of EM

- Dempster, Laird, Rubin 1977
- Convergence (to local solution) proof: Jeff Wu 1983
- Useful for finding **local** maximum likelihood parameters of statistical models when the equation **cannot** be solved directly (involving latent factors, missing data, mixture model)
- No guarantee to converge to true maximum likelihood estimator



## Maximum Likelihood from Incomplete Data via the *EM* Algorithm

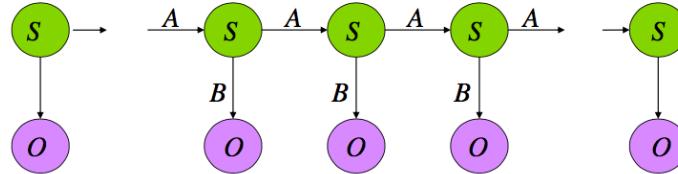
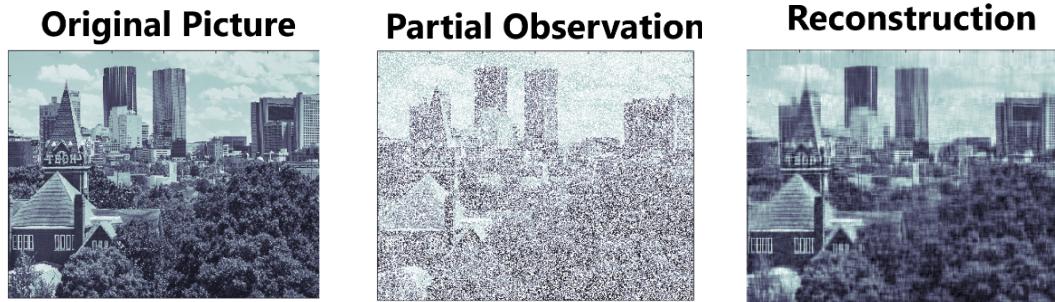
By A. P. DEMPSTER, N. M. LAIRD and D. B. RUBIN

*Harvard University and Educational Testing Service*

[Read before the ROYAL STATISTICAL SOCIETY at a meeting organized by the RESEARCH SECTION on Wednesday, December 8th, 1976, Professor S. D. SILVEY in the Chair]

# Applications

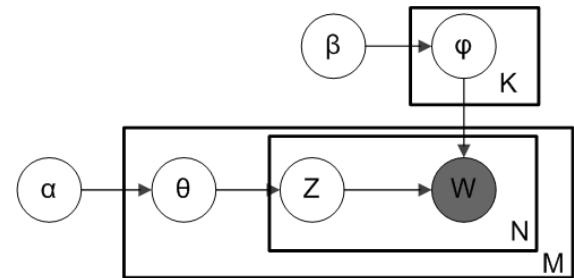
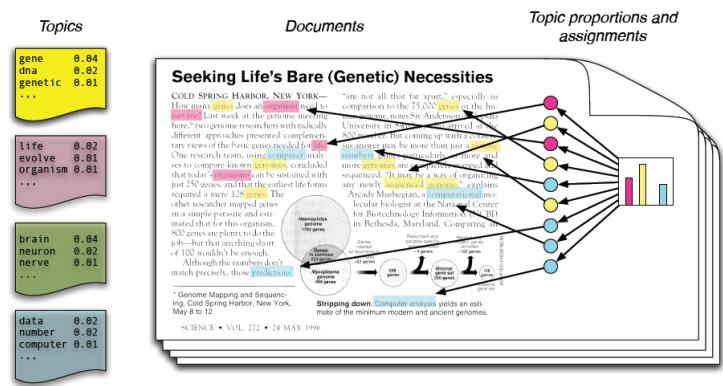
- Data clustering
- Missing data
- Estimating hidden Markov models (for dependent sequence data)
- Estimating latent factor models (in psychometrics, genetics, medical imaging)



- General types of mixture model: Topic modeling

# Topic models

- Mixture models over words provide an alternative to latent semantic indexing
- Instead of finding the principal components of the bag-of-words vectors, assume there are a certain number of topics which documents in the corpus can be about; each topic corresponds to a distribution over words.
- Hofmann (1999) estimated everything by EM
- Latent Dirichlet allocation (Blei *et al.*, 2003): a type of probabilistic graphical model / Bayesian model
- Estimation: finding posterior can be hard, using MCMC



- (a) Choose a topic  $z_{i,j} \sim \text{Multinomial}(\theta_i)$ .
- (b) Choose a word  $w_{i,j} \sim \text{Multinomial}(\varphi_{z_{i,j}})$ .

