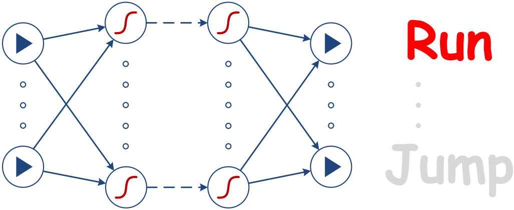
Outline

Model-based method

- Machine learning provides a wide range of model-based approaches for speaker recognition
- Model-based approach aims to incorporate the physical phenomena, measurements, uncertainties and noises in the form of mathematical models
- This approach is developed in a unified manner through different algorithms, examples, applications, and case studies
- Main-stream methods are based on the statistical models
- Latent variable models in speaker recognition include
  - joint factor analysis (JFA)
  - probabilistic linear discriminant analysis (PLDA)
  - Gaussian mixture model (GMM)
  - mixture of PLDA

Neural network

- Deep structured/hierarchical learning
- Rapidly developed and widely applied for many applications
- Multiple layers of nonlinear processing units
- High-level abstraction



Model-based method vs. neural network

	Model-based method	Neural network
Structure	Top-down	Bottom-up
Representation	Intuitive	Distributed
Interpretation	**Easy**	**Harder**

Model-based method vs. neural network

	Model-based method	Neural network
Semi/unsupervised	**Easier**	**Harder**
Incorp. domain knowl.	**Easy**	**Hard**
Incorp. constraint	**Easy**	**Hard**
Incorp. uncertainty	**Easy**	**Hard**

Model-based method vs. neural network

	Model-based method	Neural network
Learning	Many algorithms	Back-propagation
Inference/decode	**Harder**	**Easier**
Evaluation on	ELBO	**End performance**

Modern machine learning

	Model-based method	Neural network
Structure	Top-down	Bottom-up
Representation	Intuitive	Distributed
Interpretation	**Easy**	Harder
Semi/unsupervised	**Easier**	Harder
Incorp. domain knowl.	**Easy**	Hard
Incorp. constraint	**Easy**	Hard
Incorp. uncertainty	**Easy**	Hard
Learning	Many algorithms	Back-propagation
Inference/decode	Harder	**Easier**
Evaluation on	ELBO	**End performance**

Outline

Parameter estimation

- Assume we have a collection of acoustic frames $X = \{\mathbf{x}_t\}_{t=1}^{T}$ for estimation of model parameters $\boldsymbol{\theta}$
- Maximum likelihood (ML) estimation

$$\boldsymbol{\theta}_{\mathrm{ML}} = \arg\max_{\boldsymbol{\theta}} p(X|\boldsymbol{\theta})$$

- Maximum *a posteriori* (MAP) estimation

$$\boldsymbol{\theta}_{\mathrm{MAP}} = \arg\max_{\boldsymbol{\theta}} p(\boldsymbol{\theta}|X) = \arg\max_{\boldsymbol{\theta}} p(X|\boldsymbol{\theta})p(\boldsymbol{\theta})$$

where $p(\boldsymbol{\theta})$ denotes the prior distribution of $\boldsymbol{\theta}$

Expectation-maximization algorithm

- Likelihood function for observations **x** in latent variable model with latent variable **z**

$$p(\boldsymbol{x}|\boldsymbol{\theta}) = \sum_{\boldsymbol{z}} p(\boldsymbol{x}, \boldsymbol{z}|\boldsymbol{\theta})$$

- Expectation (E) step: calculate an auxiliary function

$$Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{\text{old}}) = \mathbb{E}_{\mathbf{z}}[\log p(\boldsymbol{x}, \boldsymbol{z}|\boldsymbol{\theta})|\boldsymbol{x}, \boldsymbol{\theta}^{\text{old}}]$$

- Maximization (M) step: find a new estimate $\boldsymbol{\theta}^{\text{new}}$ via

$$\boldsymbol{\theta}^{\text{new}} = \arg\max_{\boldsymbol{\lambda}} Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{\text{old}})$$

- EM algorithm [Dempster et al., 1977] for ML can be extended for MAP

Lower bound & KL divergence

- Introduce an approximate or variational distribution $q(\mathbf{z})$ and adopt the Jensen's inequality for convex function $-\log(\cdot)$ to obtain

$$\log p(\boldsymbol{x}|\boldsymbol{\theta}) = \log \sum_{\mathbf{z}} \frac{p(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta})}{q(\mathbf{z})} q(\mathbf{z}) = \log \mathbb{E}_q \left[\frac{p(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta})}{q(\mathbf{z})} \right]$$
$$\geq \mathbb{E}_q \left[\log \frac{p(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta})}{q(\mathbf{z})} \right] \triangleq \mathcal{L}(q, \boldsymbol{\theta})$$

$$\sum_{\mathbf{z}} q(\mathbf{z}) \log p(\mathbf{x}|\boldsymbol{\theta}) - \mathcal{L}(q, \boldsymbol{\theta}) = -\sum_{\mathbf{z}} q(\mathbf{z}) \log \left\{ \frac{p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta})}{q(\mathbf{z})} \right\} \triangleq \mathrm{KL}(q \| p)$$

Evidence Decomposition

$$\log p(\boldsymbol{x}|\boldsymbol{\theta}) = \mathrm{KL}(q\|p) + \mathcal{L}(q, \boldsymbol{\theta})$$

Maximum Likelihood

$$\mathrm{KL}(q\|p) = -\mathbb{E}_q[\log p(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{\theta})] - \mathbb{H}_q[\boldsymbol{z}]$$

$$\mathcal{L}(q,\boldsymbol{\theta}) = \mathbb{E}_q[\log p(\boldsymbol{x},\boldsymbol{z}|\boldsymbol{\theta})] + \mathbb{H}_q[\boldsymbol{z}]$$

- Maximizing $p(\mathbf{x}|\boldsymbol{\theta})$ is equivalent to first setting $\mathrm{KL}(q\|p) = 0$ or approximating (E-step)

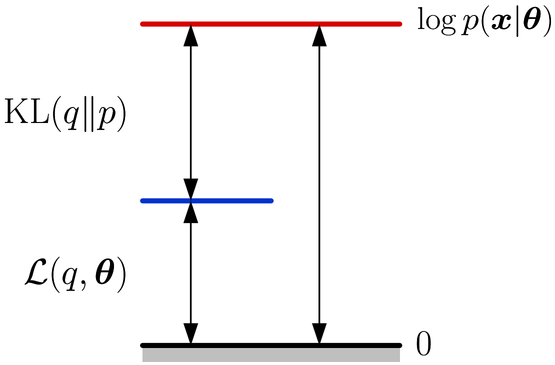
$$q(\boldsymbol{z}) = p(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{\theta}^{old})$$

then maximizing the resulting lower bound (M-step)

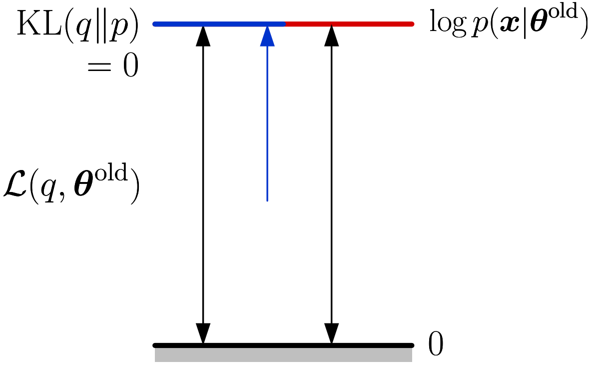
$$\mathcal{L}(q,\boldsymbol{\theta}) \triangleq Q(\boldsymbol{\theta},\boldsymbol{\theta}^{\text{old}}) + \text{const}$$

where $Q(\boldsymbol{\theta},\boldsymbol{\theta}^{\text{old}}) \triangleq \mathbb{E}_q[\log p(\boldsymbol{x},\boldsymbol{z}|\boldsymbol{\theta})|\boldsymbol{x},\boldsymbol{\theta}^{\text{old}}]$ is a concave function

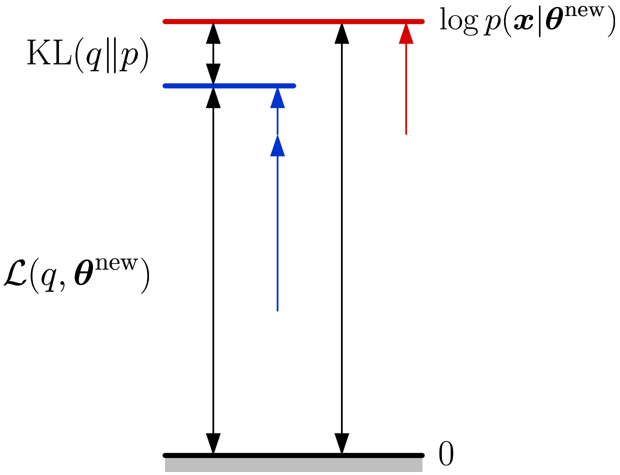
EM algorithm



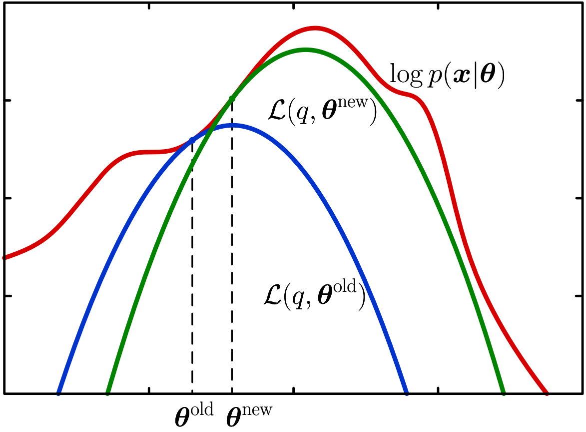
EM algorithm: E-step



EM algorithm: M-step



EM algorithm: lower bound



Outline

Why approximate inference?

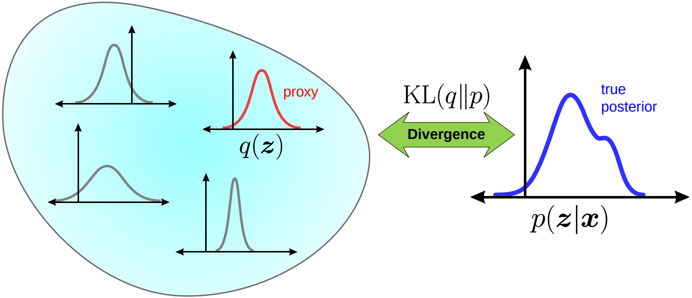
- There are a number of latent variables in model-based speaker recognition
  - i-vectors
  - common factors
  - variability matrix
  - mixture labels
  - channel, speaker and noise information
- Posterior distribution of latent variables should be analytical and factorizable
- Evolution of inference algorithms
  - maximum likelihood
  - maximum *a posteriori*
  - variational Bayesian
  - Gibbs sampling

Posterior distribution

$$\underbrace{p(\boldsymbol{z}|\boldsymbol{x})}_{\text{Posterior}} = \frac{\overbrace{p(\boldsymbol{x}|\boldsymbol{z})}^{\text{Likelihood}}\ \overbrace{p(\boldsymbol{z})}^{\text{Prior}}}{\underbrace{\int p(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z})d\boldsymbol{z}}_{\text{Marginal Likelihood } p(\boldsymbol{x}) \text{ (model evidence)}}}$$

- Latent variables and parameters $\boldsymbol{z} = \{z_1, \ldots, z_m\}$ are coupled

Approximate posterior



- Find an approximate distribution $q(\mathbf{z})$ that is *factorizable* and maximally similar to the true posterior $p(\mathbf{z}|\mathbf{x})$

Variational Bayesian inference

$$q(z_{1:m}|\nu_{1:m}) = \prod_{j=1}^{m} q(z_j|\nu_j)$$

Variational calculus

functional

$\mathcal{L}(q) : q \mapsto \mathcal{L}(q)$

Optimization problem

$\max_q \quad \mathcal{L}(q)$

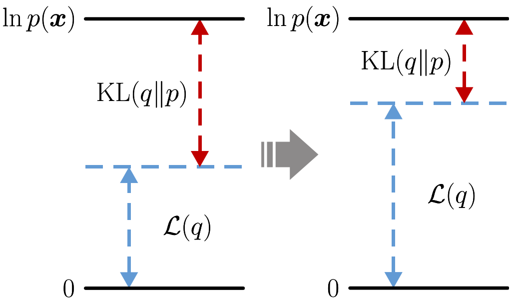
s.t. $\int_{\boldsymbol{z}} q(d\boldsymbol{z}) = 1$

$$p(\boldsymbol{x}) = \mathrm{KL}(q\|p) + \mathcal{L}(q)$$

$$\text{where } \mathrm{KL}(q\|p) = -\mathbb{E}_q[\ln p(\boldsymbol{z}|\boldsymbol{x})] - \mathbb{H}_q[\boldsymbol{z}]$$

$$\boxed{\mathcal{L}(q) = \mathbb{E}_q[\ln p(\boldsymbol{x}, \boldsymbol{z})] + \mathbb{H}_q[\boldsymbol{z}]}$$

(Evidence Lower BOund, ELBO)



Estimation for variational distribution

$$\max_{q(\boldsymbol{z})} \quad \mathbb{E}_q[\log p(\boldsymbol{x}, \boldsymbol{z})] + \mathbb{H}_q[\boldsymbol{z}]$$

$$\text{s.t.} \quad \int_{\boldsymbol{z}} q(d\boldsymbol{z}) = 1$$

$$\hat{q}(z_j|\nu_j) = \frac{\exp(\mathbb{E}_{i\neq j}[\log p(\boldsymbol{x}, \boldsymbol{z}|\boldsymbol{\nu})])}{\int \exp(\mathbb{E}_{i\neq j}[\log p(\boldsymbol{x}, \boldsymbol{z}|\boldsymbol{\nu})]) dz_j}$$

- Variational Bayesian (VB) inference is implemented via a doubly-looped algorithm

VB-EM algorithm

- VB-E step: calculate the variational distribution $q(\mathbf{z})$ in inner loop

$$\hat{q}(\mathbf{z}) = \arg\max_{q(\mathbf{z})} \mathcal{L}(q, \boldsymbol{\theta})$$

- VB-M step: calculate the model parameter $\boldsymbol{\theta}$ in outer loop

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} \mathcal{L}(\hat{q}, \boldsymbol{\theta})$$

- Convex optimization is performed
- VB-EM steps converge by a number of iterations

Gibbs sampling algorithm

Initialize $\boldsymbol{z}^{(1)}$, where $\boldsymbol{z} = z_{1:m}$

for $\tau \leftarrow 1$ **to** $T - 1$ **do**

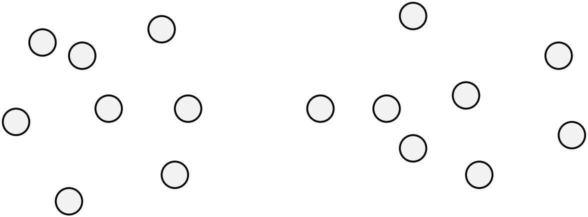
for $j \leftarrow 1$ **to** m **do**

Sample $z_j^{(\tau+1)} \sim p(z_j | z_{1:(j-1)}^{(\tau+1)}, z_{j+1:m}^{(\tau)})$

end for

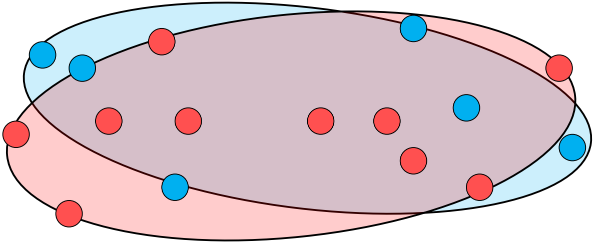
end for

Gibbs sampling



Two dimensional Gaussian mixture model with two mixture components

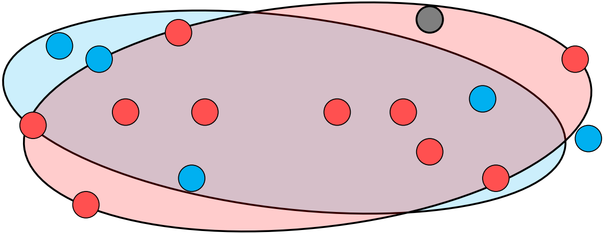
Gibbs sampling



$$z_j \sim p(\,\cdot \mid \boldsymbol{z}_{-j}, \boldsymbol{x})$$

Randomly assign mixture component for each sample j

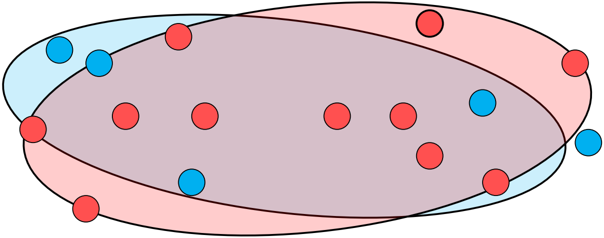
Gibbs sampling



$$z_j \sim p(\,\cdot \mid \boldsymbol{z}_{-j}, \boldsymbol{x})$$

Extract one sample and compute the conditional distribution

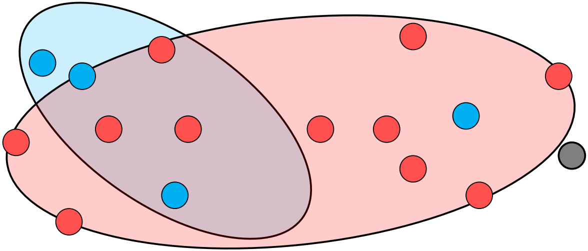
Gibbs sampling



$$z_j \sim p(\,\cdot \mid \boldsymbol{z}_{-j}, \boldsymbol{x})$$

Sample a mixture component from the conditional distribution

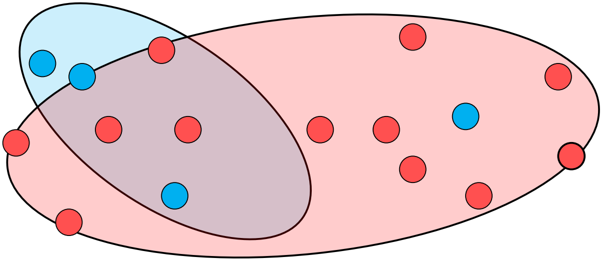
Gibbs sampling



$$z_j \sim p(\,\cdot \mid \boldsymbol{z}_{-j}, \boldsymbol{x})$$

Extract one sample and compute the conditional distribution

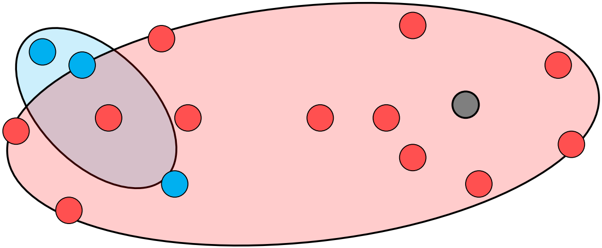
Gibbs sampling



$$z_j \sim p(\,\cdot \mid \boldsymbol{z}_{-j}, \boldsymbol{x})$$

Sample a mixture component from the conditional distribution

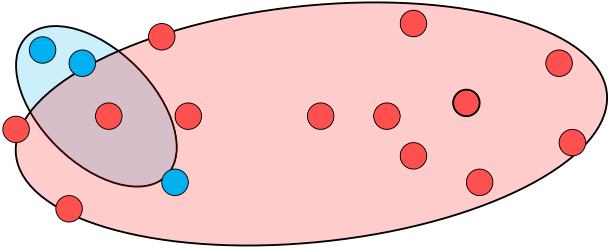
Gibbs sampling



$$z_j \sim p(\,\cdot \mid \boldsymbol{z}_{-j}, \boldsymbol{x})$$

Extract one sample and compute the conditional distribution

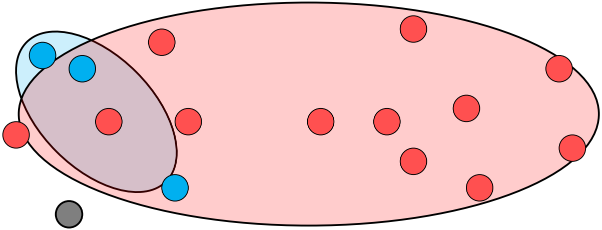
Gibbs sampling



$$z_j \sim p(\,\cdot \mid \boldsymbol{z}_{-j}, \boldsymbol{x})$$

Sample a mixture component from the conditional distribution

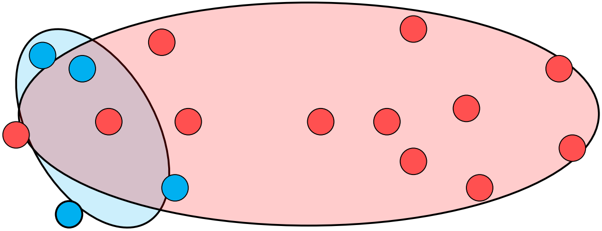
Gibbs sampling



$$z_j \sim p(\,\cdot \mid \boldsymbol{z}_{-j}, \boldsymbol{x})$$

Extract one sample and compute the conditional distribution

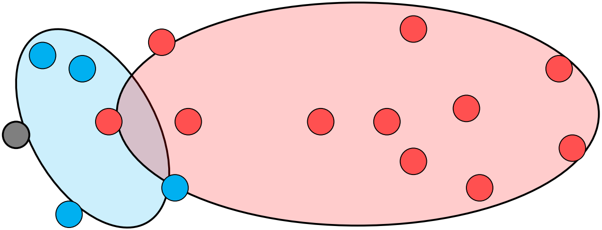
Gibbs sampling



$$z_j \sim p(\,\cdot \mid \boldsymbol{z}_{-j}, \boldsymbol{x})$$

Sample a mixture component from the conditional distribution

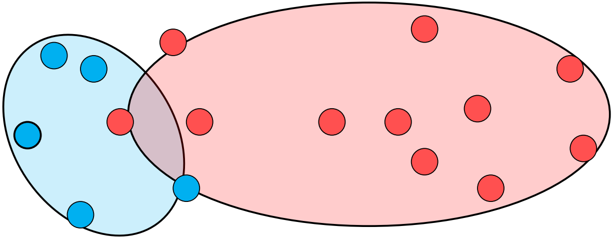
Gibbs sampling



$$z_j \sim p(\,\cdot \mid \boldsymbol{z}_{-j}, \boldsymbol{x})$$

Extract one sample and compute the conditional distribution

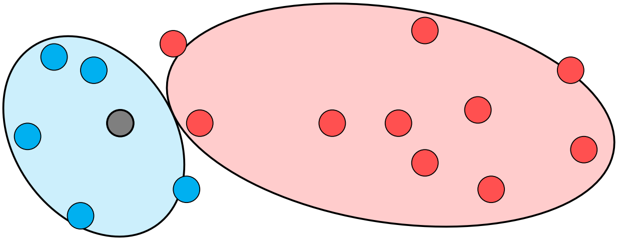
Gibbs sampling



$$z_j \sim p(\,\cdot \mid \boldsymbol{z}_{-j}, \boldsymbol{x})$$

Sample a mixture component from the conditional distribution

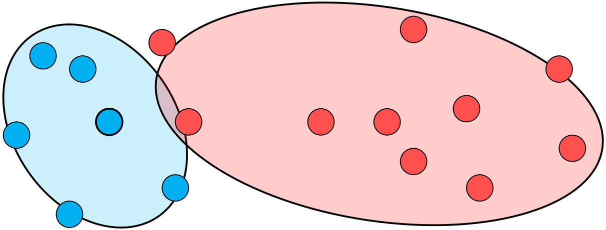
Gibbs sampling



$$z_j \sim p(\,\cdot \mid \boldsymbol{z}_{-j}, \boldsymbol{x})$$

Extract one sample and compute the conditional distribution

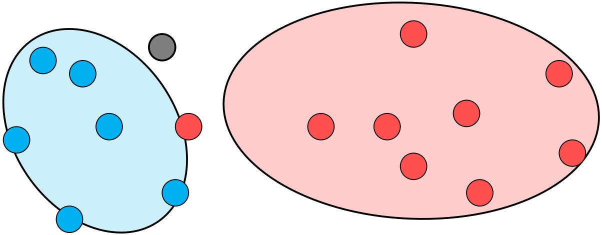
Gibbs sampling



$$z_j \sim p(\,\cdot \mid \boldsymbol{z}_{-j}, \boldsymbol{x})$$

Sample a mixture component from the conditional distribution

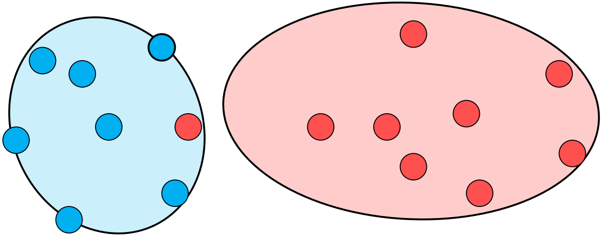
Gibbs sampling



$$z_j \sim p(\,\cdot \mid \boldsymbol{z}_{-j}, \boldsymbol{x})$$

Extract one sample and compute the conditional distribution

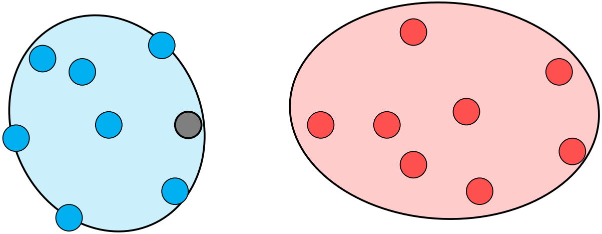
Gibbs sampling



$$z_j \sim p(\,\cdot \mid \boldsymbol{z}_{-j}, \boldsymbol{x})$$

Sample a mixture component from the conditional distribution

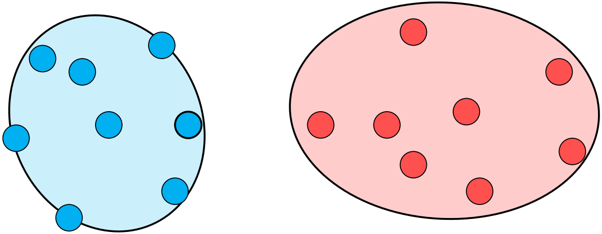
Gibbs sampling



$$z_j \sim p(\,\cdot \mid \boldsymbol{z}_{-j}, \boldsymbol{x})$$

Extract one sample and compute the conditional distribution

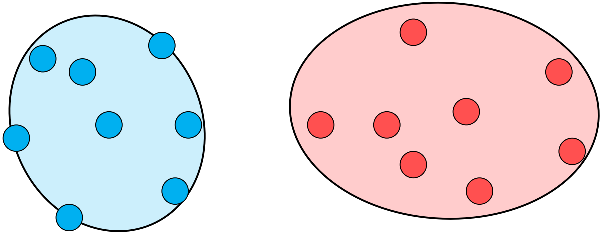
Gibbs sampling



$$z_j \sim p(\,\cdot \mid \boldsymbol{z}_{-j}, \boldsymbol{x})$$

Sample a mixture component from the conditional distribution

Gibbs sampling



$$z_j \sim p(\,\cdot\mid \boldsymbol{z}_{-j}, \boldsymbol{x})$$

Finally obtain an appropriate clustering result

Variational Bayes

- deterministic approximation
- find an analytical proxy $q(\boldsymbol{z})$ that is maximally similar to $p(\boldsymbol{z}|\boldsymbol{x})$
- inspect distribution statistics
- never generate exact results
- fast
- often hard work to derive
- convergence guarantees
- need a specific parametric form

Gibbs sampling

- stochastic approximation
- design an algorithm that draws samples $\boldsymbol{z}^{(1)}, \ldots, \boldsymbol{z}^{(\tau)}$ from $p(\boldsymbol{z}|\boldsymbol{x})$
- inspect sample statistics
- asymptotically exact
- computationally expensive
- tricky engineering concerns
- no convergence guarantees
- no need parametric form

Outline

Challenges in model-based approach

Thomas Bayes (1701-1761)

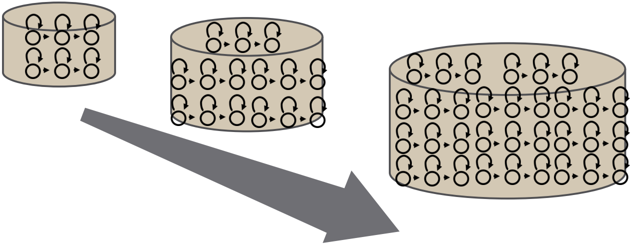
- We are facing the challenges of big data
- An enormous amount of multimedia data is available in internet which contains speech, text, image, music, video, social networks and any specialized technical data
- The collected data are usually noisy, non-labeled, non-aligned, mismatched, and ill-posed
- Probabilistic models may be improperly-assumed, over-estimated, or under-estimated

Uncertainty modeling

- We need tools for modeling, analyzing, searching, recognizing and understanding real-world data
- Our modeling tools should
  - faithfully represent uncertainty in model structure and its parameters
  - reflect noise condition in observed data
  - be automated and adaptive
  - assure robustness
  - scalable for large data sets
- Uncertainty can be properly expressed by prior distribution or process

Model regularization

- Regularization refers to a process of introducing additional information in order to solve the ill-posed problem or to prevent overfitting
- Occam's razor is imposed to deal with the issue of model selection
- Scalable modeling



Bayesian speaker recognition

- Real-world speaker recognition
  - unsupervised learning
  - number of factors is unknown
  - very short enrollment utterance
  - high inter/intra speaker variabilities
  - variabilities from channel and noise
- Why Bayesian? [Watanabe and Chien, 2015]
  - exploration for latent variables
  - model regularization
  - uncertainty modeling
  - approximate Bayesian inference
  - better prediction