Mixture Language Models and EM Algorithm

Pu-Jen Cheng

Outline

- Mixture Model & its Applications
- EM Algorithm

Outline

- Mixture Model & its Applications
- EM Algorithm

General Formula: Mixture Model

- **Unigram Mixture Model**
  - **Document LM + Background LM**

    model discriminative and common words

    - **Single document**

$$P(w_i \mid \theta_D) = \lambda P(w_i \mid \theta_D) + (1-\lambda) P(w_i \mid \theta_C)$$

    - **Multiple feedback documents**

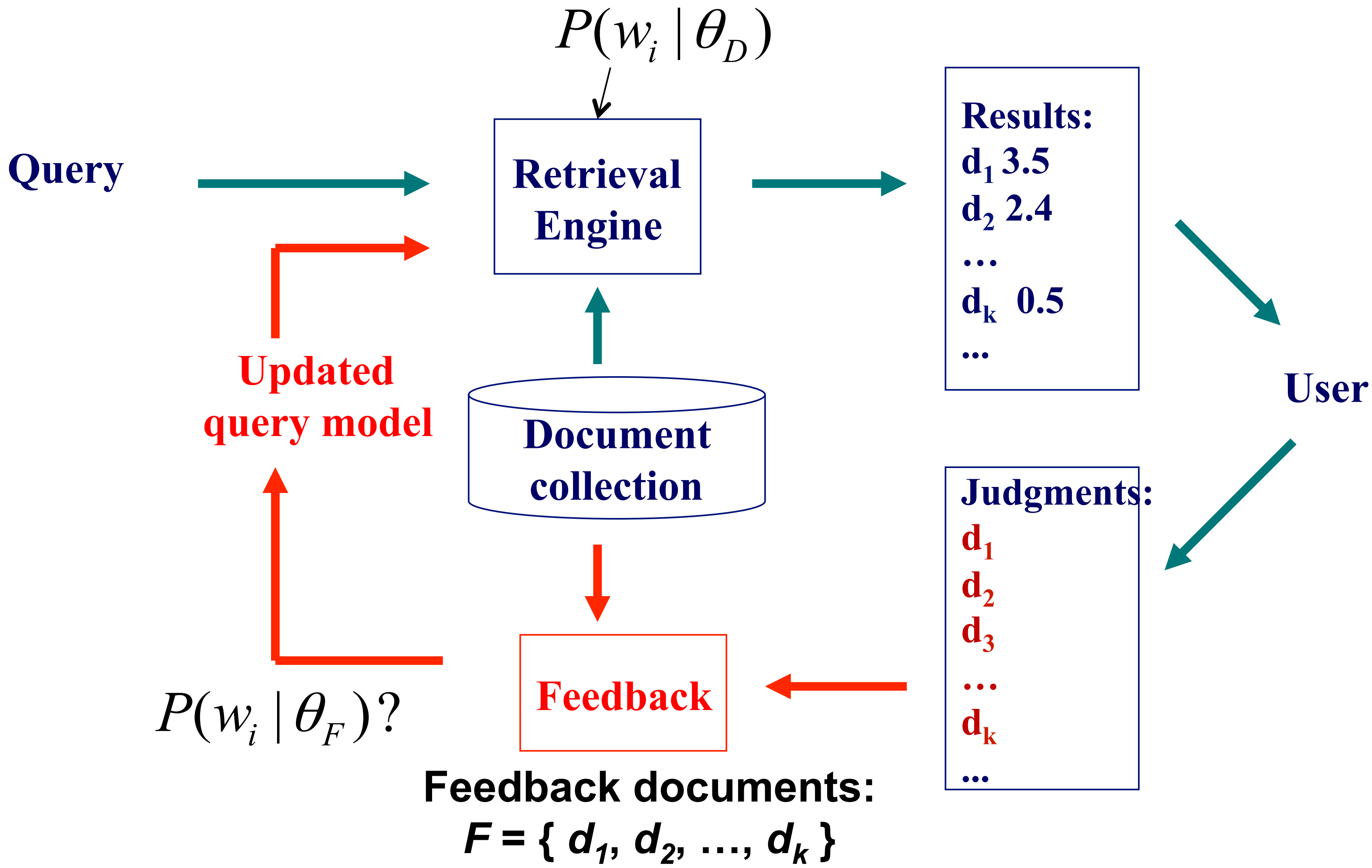
    Feedback documents: *F* = { d_1, d_2, …, d_k }

$$P(w_i \mid \theta_F) = \lambda P(w_i \mid \theta_F) + (1-\lambda) P(w_i \mid \theta_C)$$

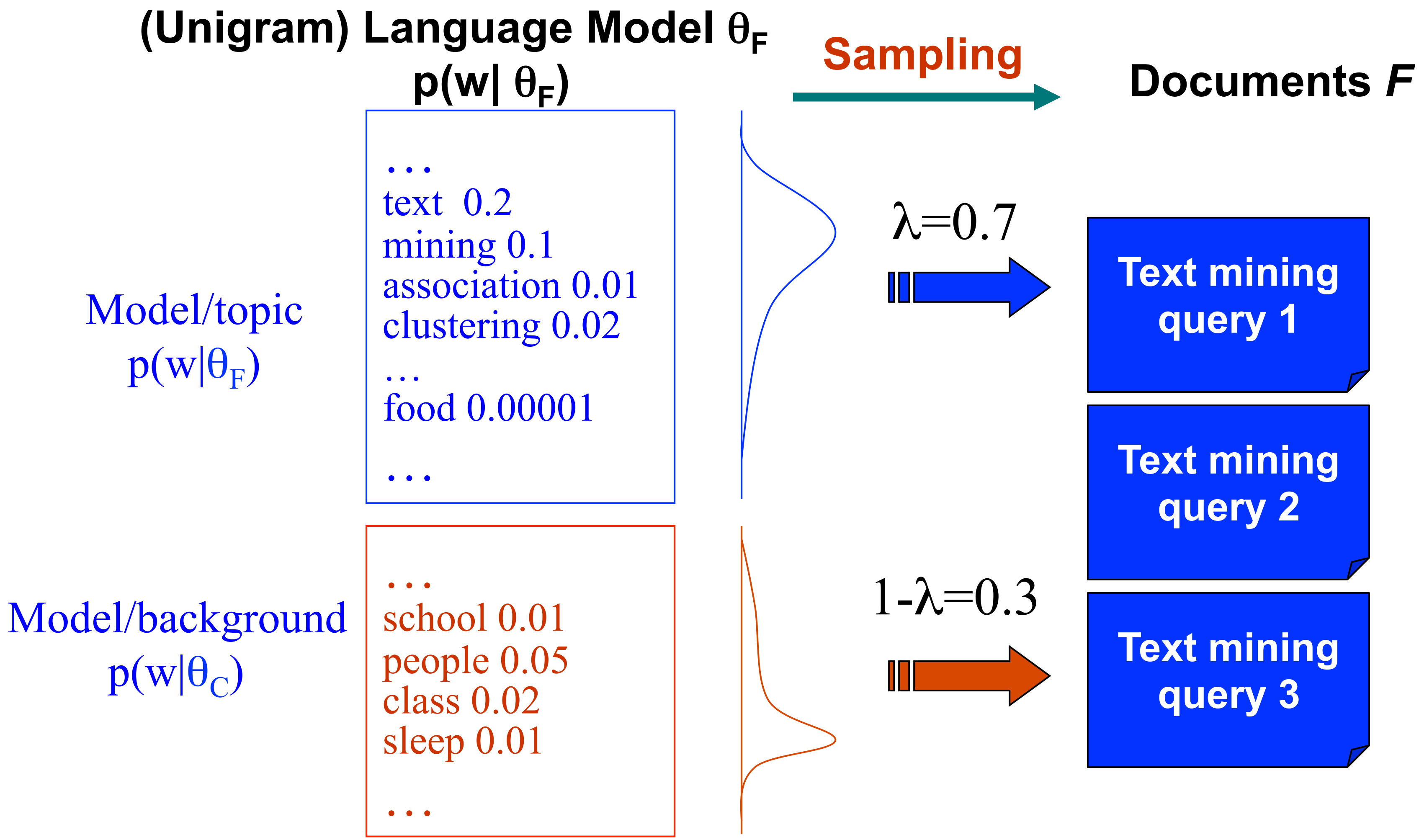
  - **Multi-topic Documents**

$$P(w_i \mid \theta_1 \oplus \theta_2) = \lambda P(w_i \mid \theta_1) + (1-\lambda) P(w_i \mid \theta_2)$$

Relevance Feedback



Model Multiple Feedback Documents



$$P(w_i \mid \theta_F) = \lambda P(w_i \mid \theta_F) + (1-\lambda) P(w_i \mid \theta_C)$$

Modeling a Multi-topic Document

A document with 2 types of vocabulary

…
text mining passage

food nutrition passage

text mining passage

text mining passage

food nutrition passage

…

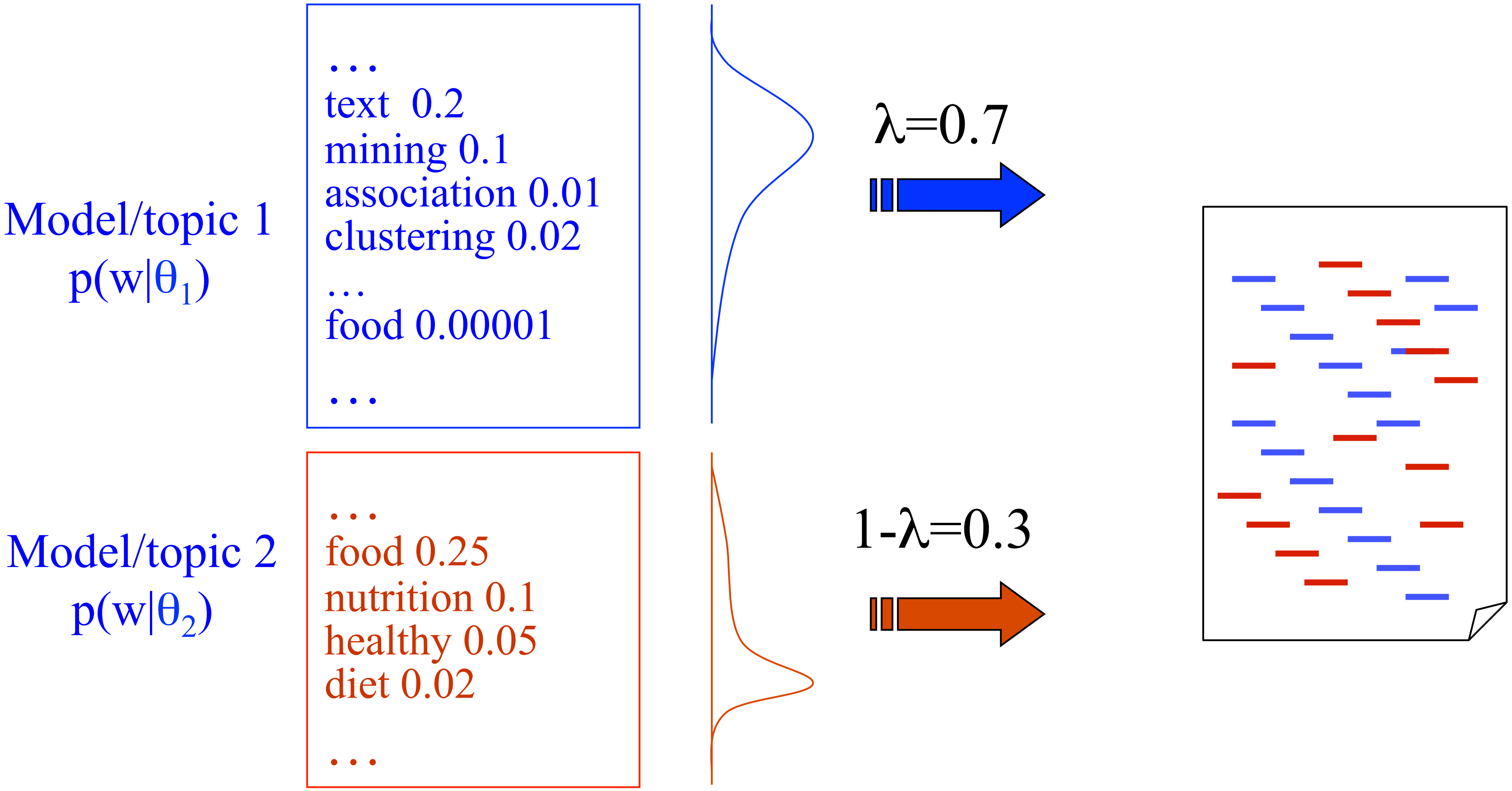
How do we model such a document?

How do we "generate" such a document?

How do we estimate our model?

Solution:
A mixture model + EM

$$p(w|\theta_1 \oplus \theta_2) = \lambda p(w|\theta_1) + (1 - \lambda) p(w|\theta_2)$$



Parameter Estimation

Likelihood:

$$p(d \mid \theta_1 \oplus \theta_2) = \prod_{w \in V} [p(w|\theta_1) + (1-\lambda) p(w \mid \theta_2)]^{c(w,d)}$$

$$\log p(d \mid \theta_1 \oplus \theta_2) = \sum_{w \in V} c(w,d) \log[\lambda p(w|\theta_1) + (1-\lambda) p(w \mid \theta_2)]$$

Estimation scenarios:

-p(w|θ_1) & p(w|θ_2) are known; estimate λ	The doc is about text mining and food nutrition, how much percent is about text mining?
-p(w|θ_1) & λ are known; estimate p(w|θ_2)	30% of the doc is about text mining, what's the rest about?
-p(w|θ_1) is known; estimate λ & p(w|θ_2)	The doc is about text mining, is it also about some other topic, and if so to what extent?
-λ is known; estimate p(w|θ_1)& p(w|θ_2)	30% of the doc is about one topic and 70% is about another, what are these two topics?
-Estimate λ, p(w|θ_1), p(w|θ_2) =clustering	The doc is about two subtopics, find out what these Two subtopics are and to what extent the doc covers Each.

Will talk about EM algorithm to estimate solve these parameters

Outline

- Mixture Model & its Applications
- EM Algorithm

Unigram Mixture Model & EM

- **Unigram Mixture Models**
  - Slightly more sophisticated unigram LMs
  - Related to smoothing
- **EM Algorithm**
  - VERY useful for estimating parameters of a mixture model or when latent/hidden variables are involved

A General Introduction to EM

The Expectation-Maximization (EM) algorithm is a general algorithm for maximum-likelihood estimation, where

the data are "incomplete" or

the likelihood function involves "latent variables"

(i.e., we can associate some latent variables with the missing data)

Data: X (observed) + H(hidden) Parameter: θ

"Incomplete" likelihood: $L(\theta) = \log p(X|\theta)$

"Complete" likelihood: $L_c(\theta) = \log p(X,H|\theta)$

For LM, EM is often used to estimate parameters of a mixture model, in which the exact component model (from which a data point is "generated") is hidden from us

Parameter Estimation Example:

Given λ and p(w|C), estimate p(w|θ_F)

- **Log-likelihood of the feedback documents *F* for θ_F**

  $F = \{ d_1, d_2, \ldots, d_k \}$

$$\log L(\theta_F) = \log p(\mathcal{F} \mid \theta_F) \quad = \quad \sum_{i=1}^{k} \sum_{j=1}^{|d_i|} \log((1-\lambda)p(d_{ij} \mid \theta_F) + \lambda p(d_{ij} \mid \mathcal{C}))$$

d_{ij}: *j*-th word in d_i

- **ML estimation**

$$\begin{aligned} \hat{\theta}_F &= \arg\max_{\theta_F} L(\theta_F) \\ &= \arg\max_{\theta_F} \sum_{i=1}^{k} \sum_{j=1}^{|d_i|} \log((1-\lambda)p(d_{ij} \mid \theta_F) + \lambda p(d_{ij} \mid \mathcal{C})) \end{aligned}$$

Any simple solution?
Think about p(d_{ij}|θ_F) as a variable

Basic Idea of EM

- **"Augment" the observed data X with some latent/hidden variables H so that the complete data has a much simpler likelihood function for finding a maxima**

Data: X (observed) + H(hidden) Parameter: θ

- **Maximizing the incomplete data likelihood through maximizing the expected completed data likelihood**

"Incomplete" likelihood: $L(\theta) = \log p(X|\theta)$

"Complete" likelihood: $L_c(\theta) = \log p(X,H|\theta)$

- **Expectation is taken over all possible values of the hidden variables**

Hidden Variable

- **Introduce a hidden variable *z* to indicate whether a word is generated from model *C* or model θ_F**

$$z_{ij} = \begin{cases} 1 & \text{if word } d_{ij} \text{ is from background} \\ 0 & \text{otherwise} \end{cases}$$

- **Complete data log-likelihood**

$$\begin{aligned} L_c(\theta_F) &= \log p(\mathcal{F}, \mathbf{z} \,|\, \theta_F) \\ &= \sum_{i=1}^{k} \sum_{j=1}^{|d_i|} [(1 - z_{ij}) \log((1-\lambda) p(d_{ij} \,|\, \theta_F)) + z_{ij} \log(\lambda p(d_{ij} \,|\, \mathcal{C}))] \end{aligned}$$

Assume that we know which model generates d_{ij}

Relation between $L_c(\theta_F)$ and $L(\theta_F)$

- **Assume our parameter is θ**

$$\underline{L_c(\theta)} = \log p(X, H|\theta) = \log p(X|\theta) + \log p(H|X,\theta) = \underline{L(\theta)} + \underline{\log p(H|X,\theta)}$$

"Incomplete" likelihood: $L(\theta) = \log p(X|\theta)$
"Complete" likelihood: $L_c(\theta) = \log p(X,H|\theta)$

Lower Bound of Likelihood

- **More specifically, the idea of EM is to**
  - **start with some initial guess of the parameter values $\theta^{(0)}$, and then**
  - **iteratively search for better values for the parameters**
  - **$L(\theta^{(n+1)})$ is better than $L(\theta^{(n)})$**
- **A potentially better parameter value θ**

$$L(\theta) - L(\theta^{(n)}) = L_c(\theta) - L_c(\theta^{(n)}) + \log \frac{p(H|X,\theta^{(n)})}{p(H|X,\theta)}$$

- **The expectation**

$$L(\theta) - L(\theta^{(n)}) = \sum_H L_c(\theta) p(H|X,\theta^{(n)}) - \sum_H L_c(\theta^{(n)}) p(H|X,\theta^{(n)})$$

$$+ \sum_H p(H|X,\theta^{(n)}) \log \frac{p(H|X,\theta^{(n)})}{p(H|X,\theta)}$$

Lower Bound of Likelihood (cont.)

$$L(\theta) - L(\theta^{(n)}) = \sum_H L_c(\theta) p(H|X,\theta^{(n)}) - \sum_H L_c(\theta^{(n)}) p(H|X,\theta^{(n)}) \cdot$$

$$+ \underbrace{\sum_H p(H|X,\theta^{(n)}) \log \frac{p(H|X,\theta^{(n)})}{p(H|X,\theta)}}$$

KL-divergence of $p(H|X,\theta^{(n)})$ and $p(H|X,\theta)$

$$D(p\,\|\,q) = \sum_x p(x) \log \frac{p(x)}{q(x)}$$

$$L(\theta) - L(\theta^{(n)}) \geq \sum L_c(\theta) p(H|X,\theta^{(n)}) - \sum L_c(\theta^{(n)}) p(H|X,\theta^{(n)})$$

$$\underline{L(\theta)} \geq \underline{\sum_H L_c(\theta) p(H|X,\theta^{(n)}) + L(\theta^{(n)}) - \sum_H L_c(\theta^{(n)}) p(H|X,\theta^{(n)})}$$

original incomplete data likelihood **lower bound**

Lower Bound of Likelihood (cont.)

- $$L(\theta) \geq \underbrace{\sum_H L_c(\theta) p(H|X, \theta^{(n)})}_{\text{The expectation of the complete likelihood } L_c(\theta)} + \underbrace{L(\theta^{(n)}) - \sum_H L_c(\theta^{(n)}) p(H|X, \theta^{(n)})}_{\text{Ignore because of no } \theta}$$

  The expectation of the complete likelihood $L_c(\theta)$

  Ignore because of no θ

  Maximizing this lower bound is to maximize the original (incomplete) likelihood

- **Q-function**

$$Q(\theta; \theta^{(n)}) = E_{p(H|X, \theta^{(n)})}[L_c(\theta)] = \sum_H L_c(\theta) p(H|X, \theta^{(n)})$$

Lower Bound of Likelihood (cont.)

- **Q-function of our mixture model L(θ_F)**

$$Q(\theta_F;\theta_F^{(n)}) = \sum_{\mathbf{z}} L_c(\theta_F)p(\mathbf{z}|\mathcal{F},\theta_F^{(n)})$$

$$= \sum_{i=1}^{k}\sum_{j=1}^{|d_i|}[p(z_{ij}=0|\mathcal{F},\theta_F^{(n)})\log((1-\lambda)p(d_{ij}\,|\,\theta_F)) + p(z_{ij}=1|\mathcal{F},\theta_F^{(n)})\log(\lambda p(d_{ij}\,|\,\mathcal{C}))]$$

$$\begin{aligned} L_c(\theta_F) &= \log p(\mathcal{F},\mathbf{z}\,|\,\theta_F) \\ &= \sum_{i=1}^{k}\sum_{j=1}^{|d_i|}[(1-z_{ij})\log((1-\lambda)p(d_{ij}\,|\,\theta_F)) + z_{ij}\log(\lambda p(d_{ij}\,|\,\mathcal{C}))] \end{aligned}$$

A General Introduction to EM

"Incomplete" likelihood: $L(\theta) = \log p(X|\theta)$
"Complete" likelihood: $L_c(\theta) = \log p(X,H|\theta)$

EM tries to iteratively maximize the complete likelihood:
Starting with an initial guess $\theta^{(0)}$,

1. E-step: compute the expectation of the complete likelihood

$$Q(\theta;\theta^{(n)}) = E_H[L_c(\theta) \mid X, \theta^{(n)}]$$
$$= \sum_{h_i} p(H = h_i \mid X, \theta^{(n)}) \log p(X, H = h_i \mid \theta)$$

2. M-step: compute $\theta^{(n)}$ by maximizing the Q-function

$$\theta^{(n+1)} = \arg\max_{\theta} Q(\theta;\theta^{(n)})$$
$$= \arg\max_{\theta} \sum_{h_i} p(H = h_i \mid X, \theta^{(n)}) \log p(X, H = h_i \mid \theta)$$

E-step for L(θ_F)

- **Compute** p(H| X, $\theta^{(n)}$)

$$p(z_{ij} = 1|\mathcal{F}, \theta_F^{(n)}) = \frac{\lambda p(d_{ij}|C)}{\lambda p(d_{ij}|C) + (1-\lambda)p(d_{ij}|\theta_F^{(n)})}$$

$$p(z_{ij} = 0|\mathcal{F}, \theta_F^{(n)}) = 1 - p(z_{ij} = 1|\mathcal{F}, \theta_F^{(n)})$$

Replace d_{ij} with z_w

$$p(z_w = 1|\mathcal{F}, \theta_F^{(n)}) = \frac{\lambda p(w|C)}{\lambda p(w|C) + (1-\lambda)p(w|\theta_F^{(n)})}$$

M-step for L(θ_F)

- **Maximize the Q-function**
- **Apply Lagrange multiplier** $\sum_{w \in V} p(w|\theta_F) = 1$

$$g(\theta_F) = Q(\theta_F; \theta_F^{(n)}) + \mu(1 - \sum_{w \in V} p(w|\theta_F))$$

$$\frac{\partial g(\theta_F)}{\partial p(w|\theta_F)} = [\sum_{i=1}^{k} \sum_{j=1, d_{ij}=w}^{|d_i|} \frac{p(z_{ij}=0|\mathcal{F}, \theta_F^{(n)})}{p(w \mid \theta_F)}] - \mu$$

$$\begin{aligned} p(w|\theta_F) &= \frac{\sum_{i=1}^{k} \sum_{j=1, d_{ij}=w}^{|d_i|} p(z_{ij}=0|\mathcal{F}, \theta_F^{(n)})}{\sum_{i=1}^{k} \sum_{j=1}^{|d_i|} p(z_{ij}=0|\mathcal{F}, \theta_F^{(n)})} \\ &= \frac{\sum_{i=1}^{k} p(z_w=0|\mathcal{F}, \theta_F^{(n)}) c(w, d_i)}{\sum_{i=1}^{k} \sum_{w \in V} p(z_w=0|\mathcal{F}, \theta_F^{(n)}) c(w, d_i)} \end{aligned}$$

EM for L(θ_F)

- **E-step:**

$$p(z_w = 1|\mathcal{F}, \theta_F^{(n)}) \quad = \quad \frac{\lambda p(w|C)}{\lambda p(w|C) + (1-\lambda)p(w|\theta_F^{(n)})}$$

- **M-step:**

$$p(w|\theta_F^{(n+1)}) \quad = \quad \frac{\sum_{i=1}^{k}(1 - p(z_w = 1|\mathcal{F}, \theta_F^{(n)}))c(w, d_i)}{\sum_{i=1}^{k}\sum_{w \in V}(1 - p(z_w = 1|\mathcal{F}, \theta_F^{(n)})c(w, d_i))}$$

EM Iteration

EM repeats the E and M steps until convergence

$$\theta^{i+1} = \underset{\theta}{\operatorname{argmax}} E_H \left[\log P(X, H \mid \theta) \mid X, \theta^i \right]$$

1. **E (expectation) Step:**
   - **To expect the value distribution of *H* according to current hypothesis (θ^i) → H^i**
2. **M (maximization) Step:**
   - **To compute the optimal hypothesis according to current data distribution (H^i, X) → θ^{i+1}**

(X are fixed values while H^i are value distribution and change in each iteration)

Parameter Estimation Example:

Given $p(w|\theta_1)$ and $p(w|\theta_2)$, estimate λ

Maximum Likelihood:

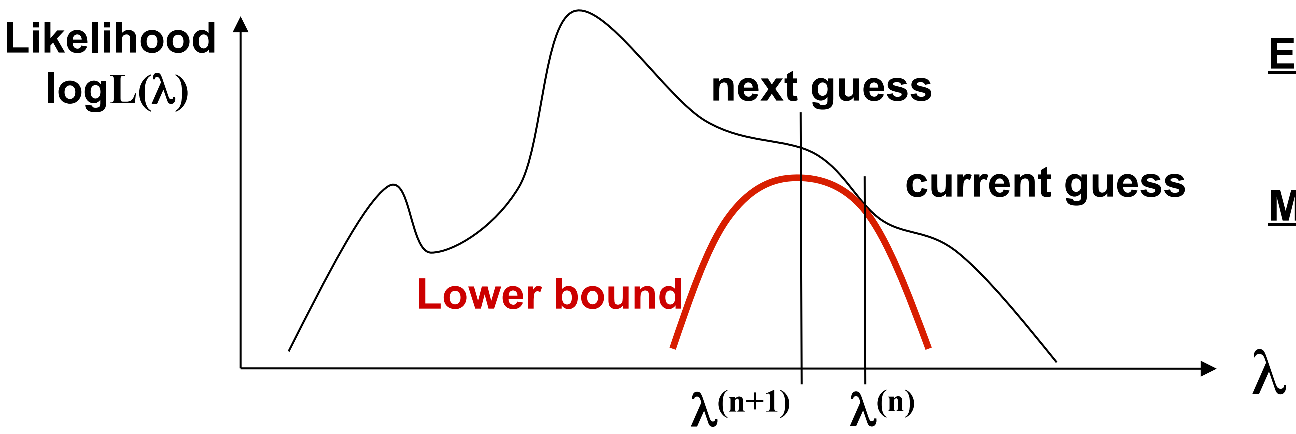
$$L(\lambda) = \prod_{w \in V} [\lambda p(w|\theta_1) + (1-\lambda) p(w|\theta_2)]^{c(w,d)}$$

$$\log L(\lambda) = \sum_{w \in V} c(w,d) \log[\lambda p(w|\theta_1) + (1-\lambda) p(w|\theta_2)]$$

$$\lambda^* = \arg\max_{\lambda} \log L(\lambda)$$

Expectation-Maximization (EM) Algorithm is a commonly used method

Basic idea: Start from some random guess of parameter values, and then Iteratively improve our estimates ("hill climbing")



E-step: compute the lower bound

M-step: find a new λ that maximizes the lower bound

EM Algorithm: Intuition

$p(w|\theta_1)$

…
text 0.2
mining 0.1
assocation 0.01
clustering 0.02
…
food 0.00001

…

λ=?

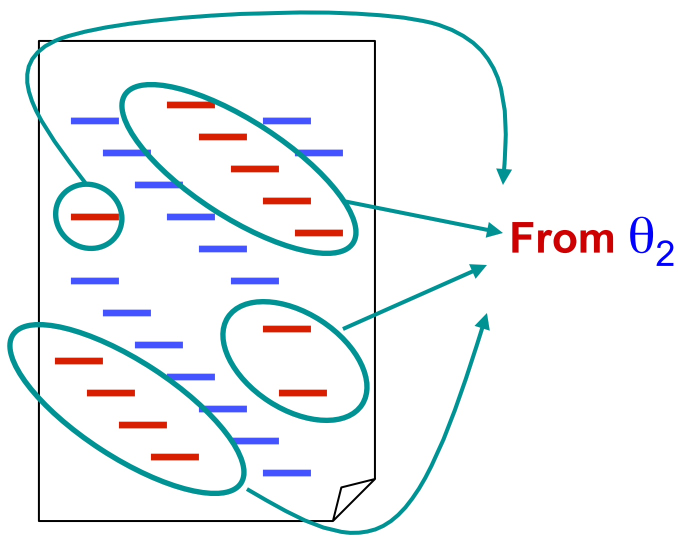
$p(w|\theta_2)$

…
food 0.25
nutrition 0.1
healthy 0.05
diet 0.02

…

1-λ=?

Observed
Doc d



$$p(w|\theta_1 \oplus \theta_2) = \lambda p(w|\theta_1) + (1-\lambda) p(w|\theta_2)$$

Suppose we know the identity of each word…

$$\lambda^* = \frac{\sum_{w \in V} c(w,d)\delta(\text{"}w \ from \ \theta_1\text{"})}{\sum_{w \in V} c(w,d)}$$

Can We "Guess" the Identity?

Identity ("hidden") variable: $z_w \in \{1$ (w from θ_1), 0(w from θ_2)$\}$

	z_w
the	1
paper	1
presents	1
a	1
text	0
mining	0
algorithm	0
the	1
paper	0
...	...

$$p(z_w=1|w)=\frac{p(z_w=1)p(w|z_w=1)}{p(z_w=1)p(w|z_w=1)+p(z_w=0)p(w|z_w=0)}$$

$$=\frac{\lambda p(w|\theta_1)}{\lambda p(w|\theta_1)+(1-\lambda)p(w|\theta_2)}$$

E-step

$$\lambda^{new}=\frac{\sum_{w\in V} c(w,d)p(z_w=1)}{\sum_{w} c(w,d)}$$

M-step

Initially, set λ to some random value, then iterate …

An Example of EM Computation

$$Log-Likelihood: \log L(\lambda) = \sum_{w \in V} c(w,d) \log[\lambda p(w|\theta_1) + (1-\lambda) p(w|\theta_2)]$$

$$E-step: \quad p(z_w = 1|w) = \frac{\lambda p(w|\theta_1)}{\lambda p(w|\theta_1) + (1-\lambda) p(w|\theta_2)}$$

$$M-step: \quad \lambda^{new} = \frac{\sum_{w \in V} c(w,d) p(z_w = 1|w)}{\sum_{w \in V} c(w,d)}$$

Word	#	$P(w\|\theta_1)$	$P(w\|\theta_2)$	Init	Iteration 1		Iteration 2	
				$\lambda^{(0)}$	$P(z=1\|w)$	$\lambda^{(1)}$	$P(z=1\|w)$	$\lambda^{(2)}$
The	4	0.5	0.2	0.5	0.71	0.46	0.68	0.43
Paper	2	0.3	0.1		0.75		0.72	
Text	4	0.1	0.5		0.17		0.14	
Mining	2	0.1	0.3		0.25		0.22	
Log-Likelihood				-15.45	-15.39		-15.35	

Any Theoretical Guarantee?

- EM is guaranteed to reach a LOCAL maximum
- When “local maxima” = “global maxima”, EM can find the global maximum
- But, when there are multiple local maximas, “special techniques” are needed (e.g., try different initial values)

Convergence Guarantee

Goal: maximizing "Incomplete" likelihood: $L(\theta) = \log p(X|\theta)$

I.e., choosing $\theta^{(n+1)}$, **so that** $L(\theta^{(n+1)}) - L(\theta^{(n)}) \geq 0$

Note that, since $p(X,H|\theta) = p(H|X,\theta)\, P(X|\theta)$, $L(\theta) = L_c(\theta) - \log p(H|X,\theta)$

$$L(\theta^{(n+1)}) - L(\theta^{(n)}) = L_c(\theta^{(n+1)}) - L_c(\theta^{(n)}) + \log [p(H|X,\theta^{(n)}) / p(H|X,\theta^{(n+1)})]$$

Taking expectation w.r.t. $p(H|X,\theta^{(n)})$,

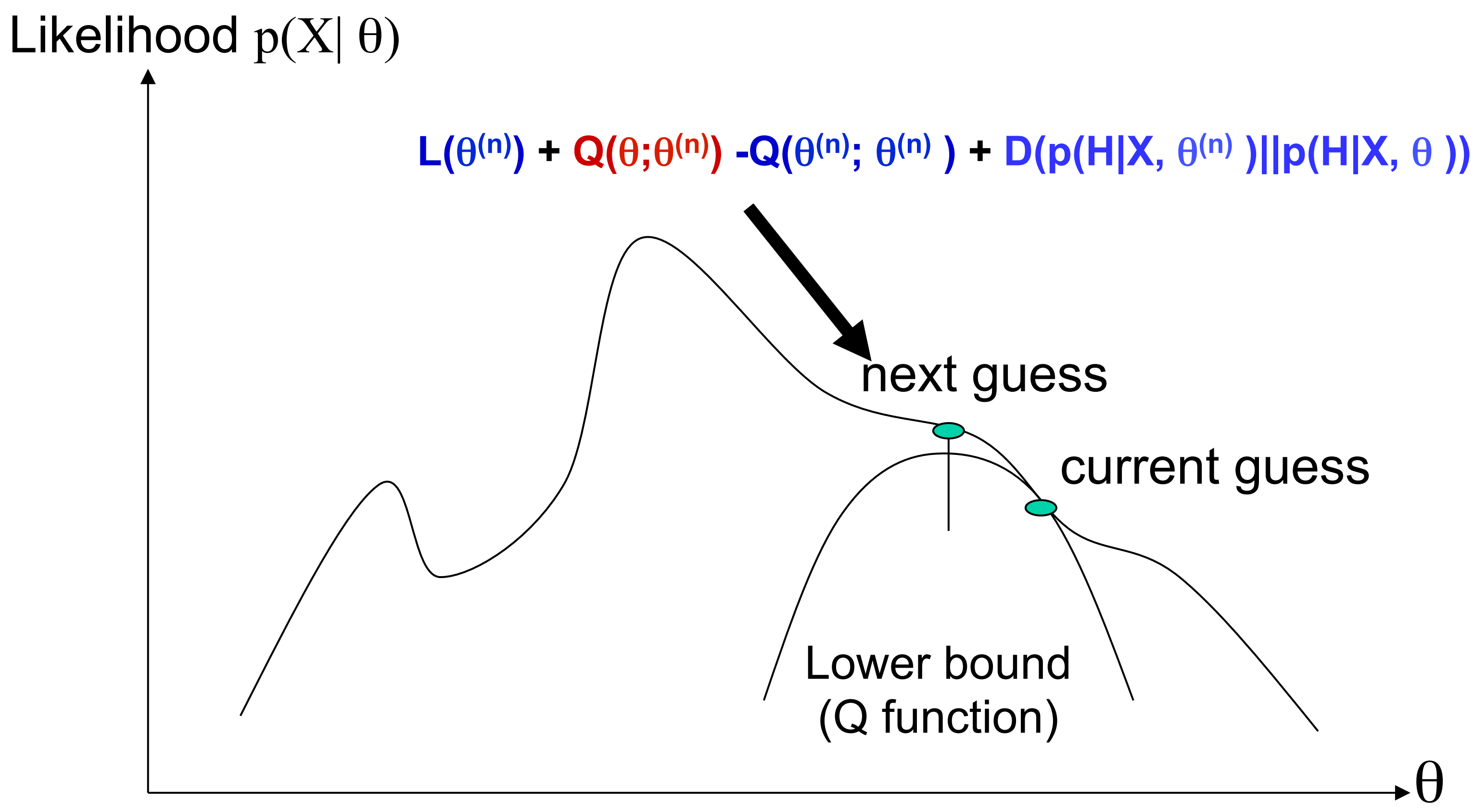
$$L(\theta^{(n+1)}) - L(\theta^{(n)}) = \underbrace{Q(\theta^{(n+1)};\theta^{(n)}) - Q(\theta^{(n)};\theta^{(n)})}_{\text{EM chooses } \theta^{(n+1)} \text{ to maximize } Q} + \underbrace{D(p(H|X,\theta^{(n)})||p(H|X,\theta^{(n+1)}))}_{\text{KL-divergence, always non-negative}}$$

Since we have maximized the *Q* function for model θ,

$$Q(\theta^{(n+1)};\theta^{(n)}) - Q(\theta^{(n)};\theta^{(n)}) \geq 0$$

$$\boxed{\textbf{Therefore,} \quad L(\theta^{(n+1)}) \geq L(\theta^{(n)})!}$$

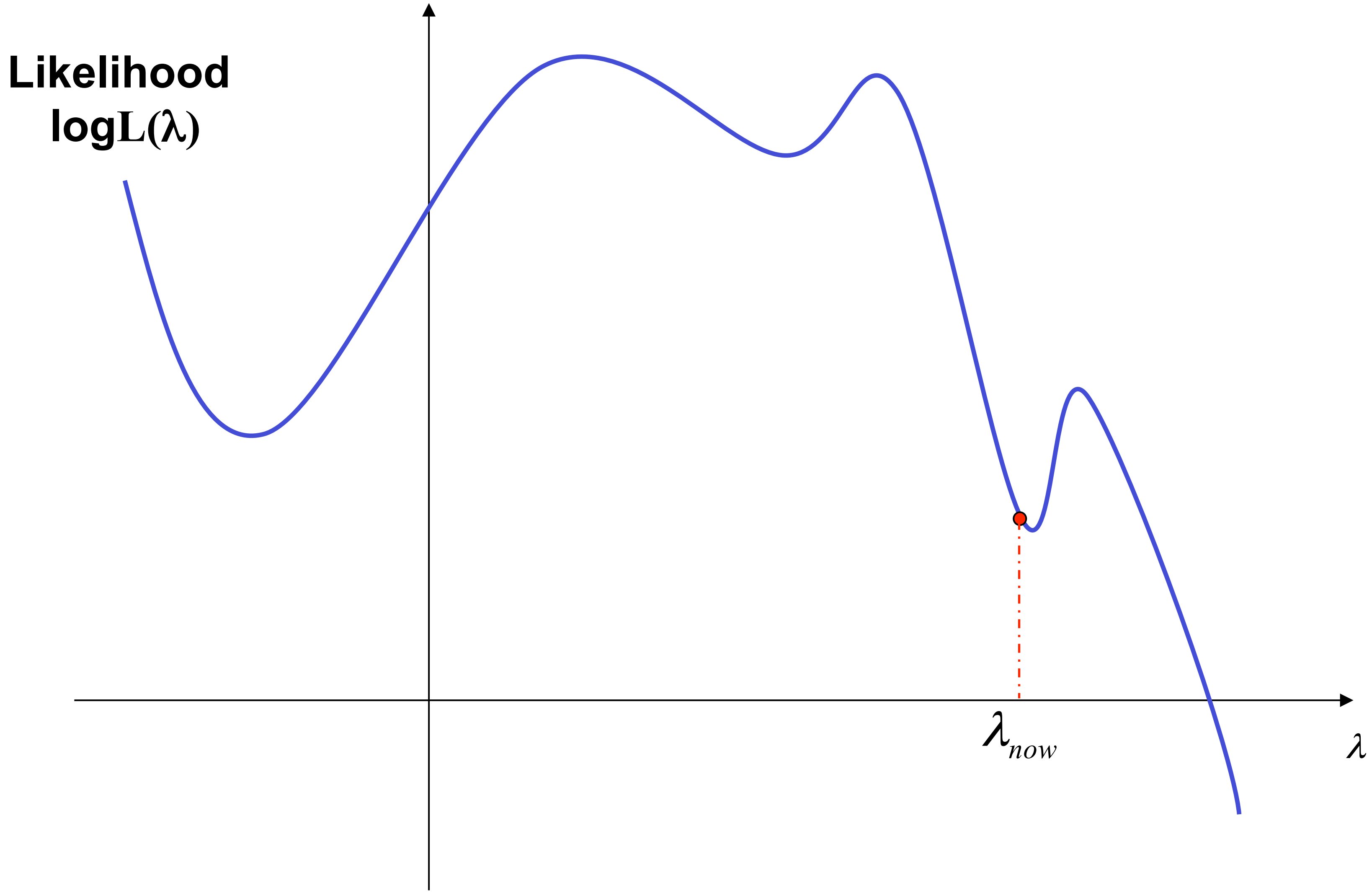
Another way of looking at EM



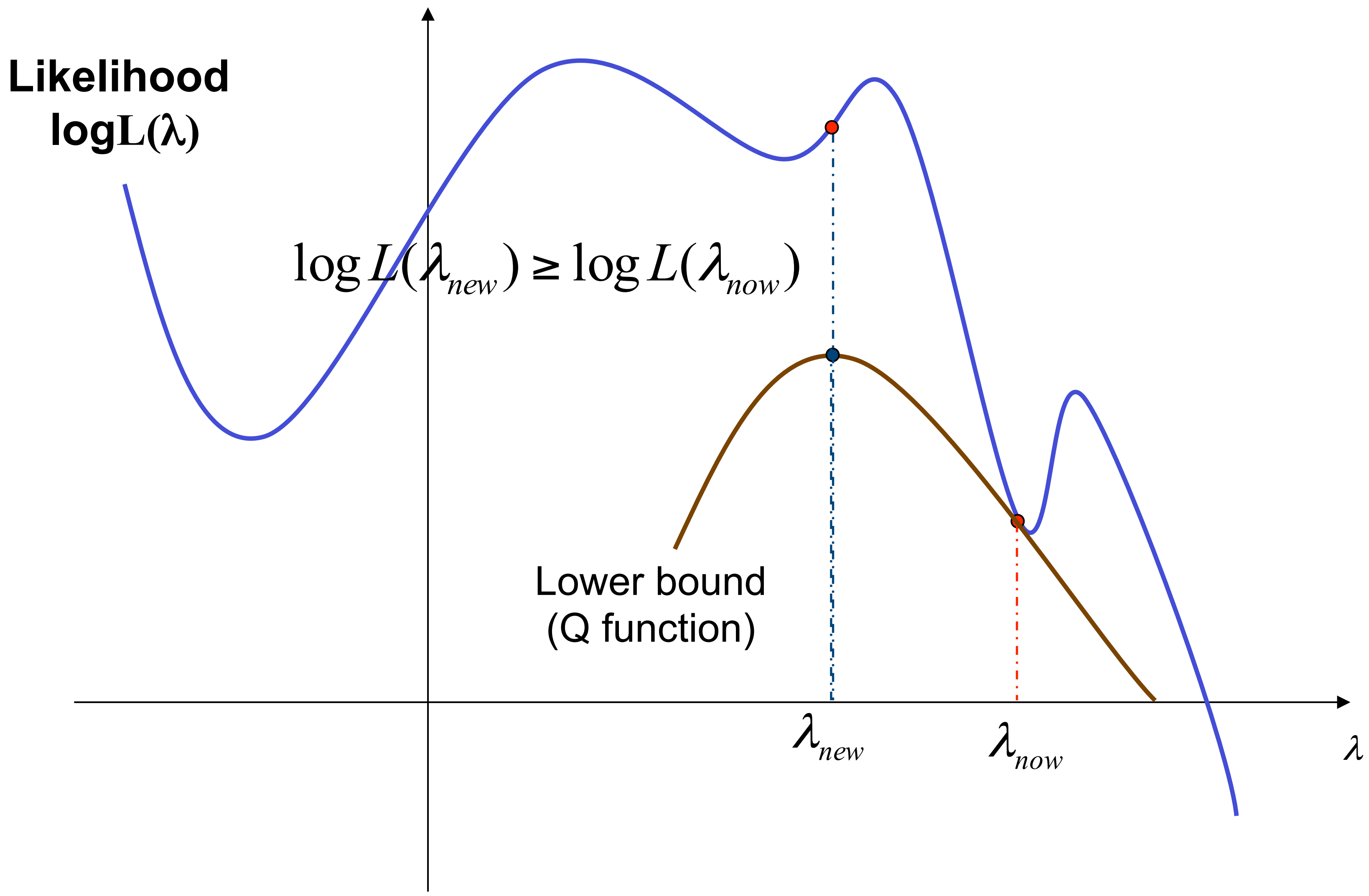
E-step = computing the lower bound

M-step = maximizing the lower bound

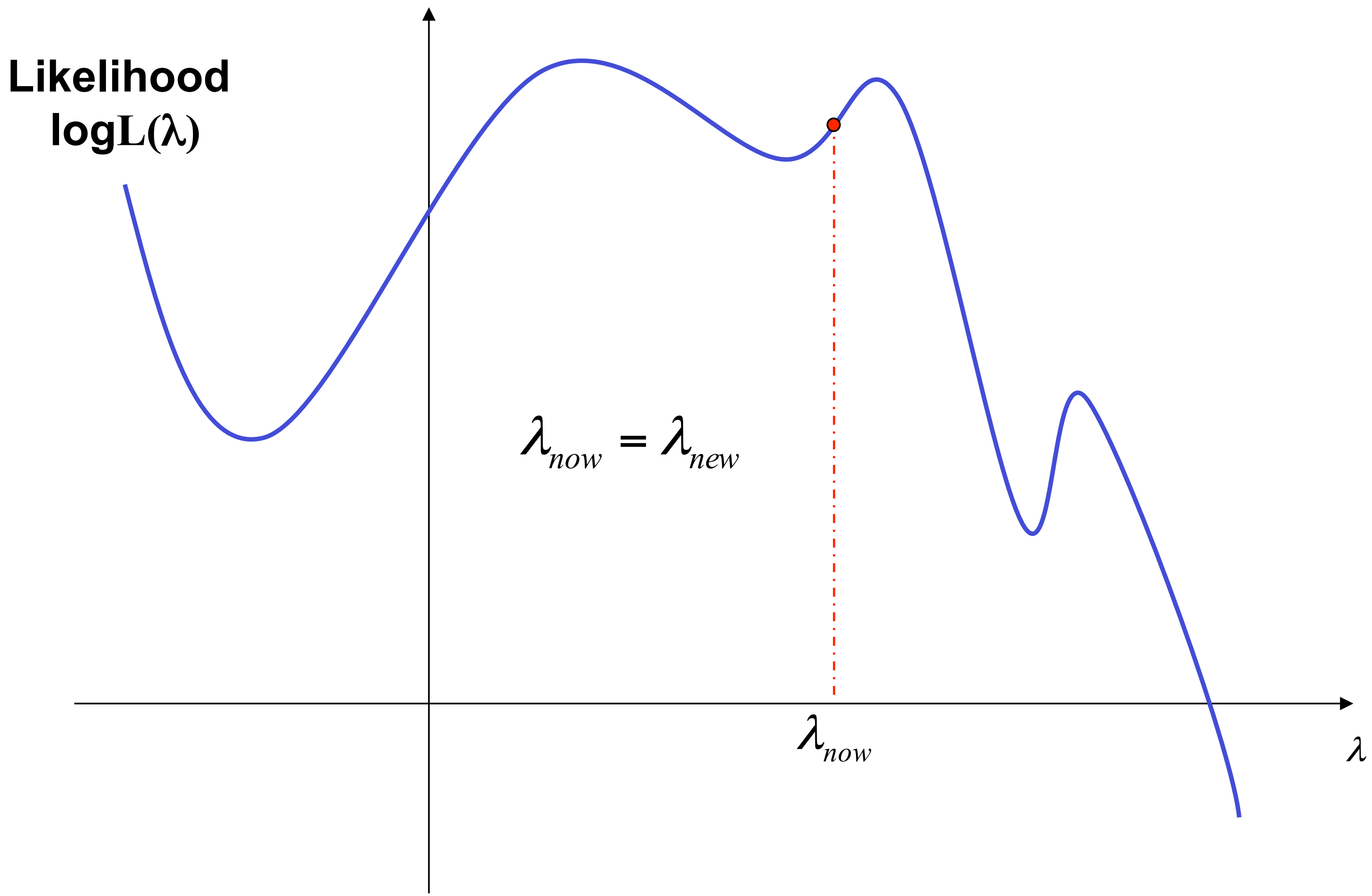
Parameter λ Estimation



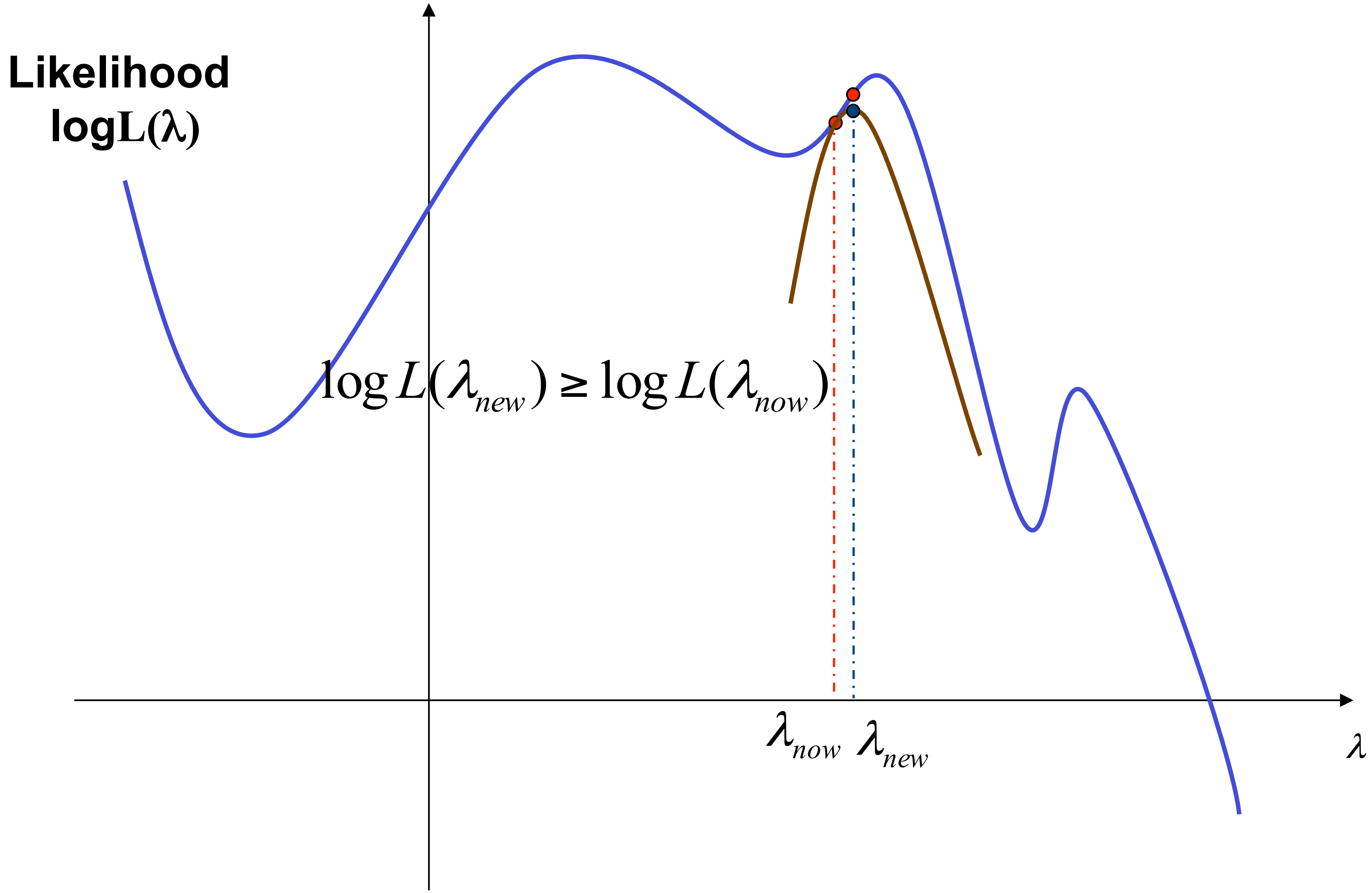
Parameter λ Estimation (cont.)



Parameter λ Estimation (cont.)



Parameter λ Estimation (cont.)



What You Should Know

- What is mixture model?
- How to estimate parameters of simple unigram mixture models using EM
- Know the general idea of EM