CIS606 – Lecture 4

Woon Wei Lee, Jacob Crandall
Spring 2014, 10am-11:15am,
Mondays and Thursdays

For today:

- EM Algorithm

Expectation Maximization Algorithm

- **The EM algorithm is a method for finding maximum likelihood estimates to unknown parameters θ..**
  - In the presence of hidden or missing data.
  - Extremely broad applicability, and can be used to generalize a variety of algorithms, including:
    - K-Means
    - LSA
    - Markov Models
    - Etc...
  - Intuition is simple, and is based on a two step iterative procedure:
    1. Using current parameter set and observed data, estimate the unknown data (this is known as the “E-step”)
    2. Using estimate of unknown data, update the parameter values – this is the “M-step”
- **General form proposed in 1977 (yup, it's been around!), in seminal paper of [Dempster et al]**

Intuitive example: k-means intepretation

K-means algorithm:

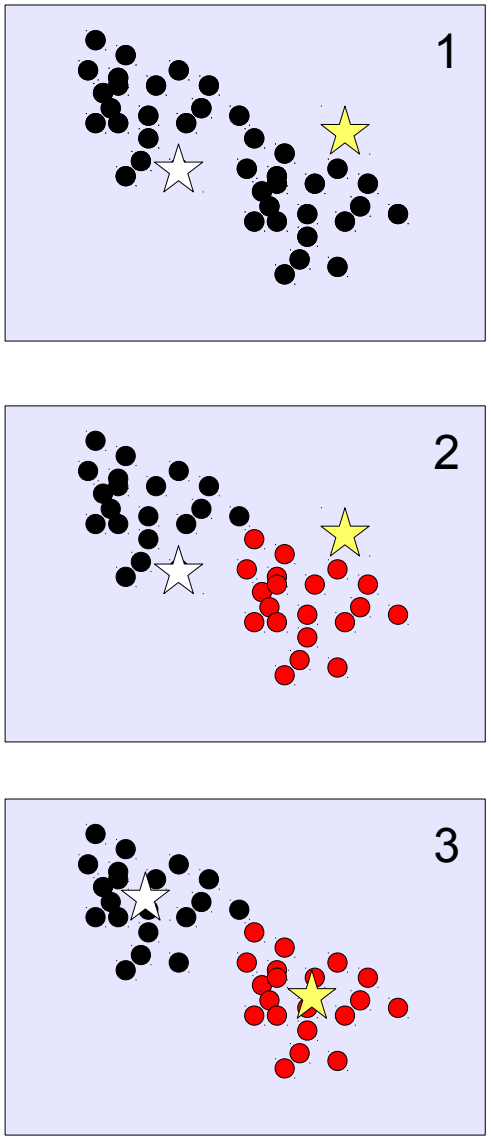
1. **Given set of data *x:***

- Task: partition data points into *k*-clusters
- Initiate *k*-centers randomly

2. **Based on distance to *k*-centers, assign each point to the closest of the centers**

3. **Update centers:**

- New locations set to means of assigned data points
- Repeat until convergence (eg: assignments are stable)



EM algorithm

1. **Given a set of data *x*:**

- Task: find parameters of model $\Sigma_z p(x,z;\theta)$ which best describes the data
- Initiate parameters randomly

Q: What are the parameters and the "missing" data?

2. **For each point, estimate "*z*"**

- Find the most likely cluster assignment

3. **Update θ:**

- Given the estimates of "z", update the parameters of each cluster.

EM Algorithm – basic setting and motivation

- **Define full/complete data as D={x,z},**
  - *x* is the *observed* data
  - z is the hidden or missing data
- **Also, denote as θ, the parameters of the model, also referred to as the complete data likelihood:**

$$p(x, z; \theta)$$

- **In practice, we would like to optimize the log likelihood of the parameters based on the *observed* data:**

$$\theta_{ML} = \underset{\theta}{argmax} \log p(x; \theta) \cdots\cdots (1)$$

$$= \underset{\theta}{argmax} \log \sum_{z} p(x, z; \theta) \cdots\cdots (2)$$

- **We can optimize this directly but in general it is very difficult to do so**
  - In some cases, (1) has a very complicated form (which is why hidden variables *z* are introduced in some cases!)
  - (2) has a log of sum terms – messy!

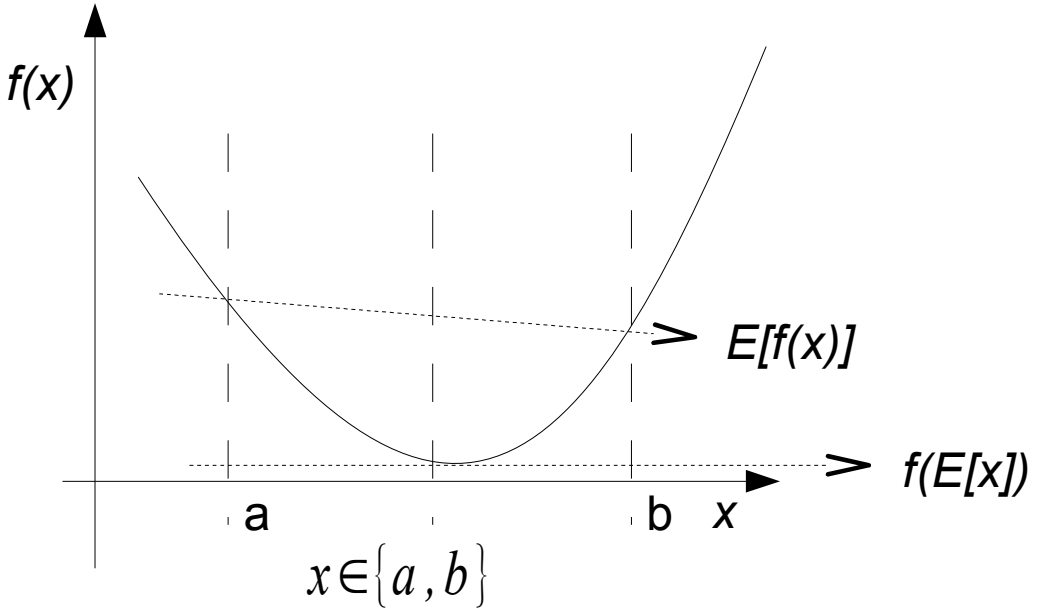
Jensen's inequality

- **General theorem:**

  For any convex function *f* and random variable X the following holds:

$$E[f(x)] \geqslant f(E[x])$$

- **To visualize/help remember:**



- **For a *concave* function *g*, the reverse holds:** $E[g(x)] \leqslant g(E[x])$

- **l.h.s = r.h.s *only* occurs in one situation:** $E(x) = x \rightarrow E[f(x)] = f(E[x])$

Optimization via lower bounding

- **Back to equation (2) from earlier:**

$$\theta_{ML} = \underset{\theta}{argmax} \log \sum_{z} p(x, z; \theta) \cdots\cdots (2)$$

- **Introduce a *variational* distribution Q(z):**

$$L(\theta) = \log \sum_{z} p(x, z; \theta)$$

$$= \log \sum_{z} Q(z) \frac{p(x, z; \theta)}{Q(z)}$$

- **Note that:**
  - The term in the logarithm on the r.h.s. Is an expectation
  - Logarithm is a concave function. Hence we can apply the “inverse” Jensen equality:

$$L(\theta) \geqslant \sum_{z} Q(z) \log \frac{p(x, z; \theta)}{Q(z)}$$

  - Hence, the r.h.s. Expression forms a *lower-bound* on the likelihood L(θ)

(Cont'd)

- **This expression holds for *any* Q(z)**
  - Question: How do we make the bound *tight*?

$$L(\theta) \geqslant \sum_z Q(z) \log \frac{p(x, z; \theta)}{Q(z)}$$

  - Remember earlier statement:

$$E(x) = x \rightarrow E[f(x)] = f(E[x])$$

- **i.e.: we can make this tight if the expression in the log is a *constant value***
  - This can be achieved by setting:

$$Q(z) = p(z|x; \hat{\theta})$$

$$\frac{p(x, z; \hat{\theta})}{Q(z)} = \frac{p(z|x; \hat{\theta})\, p(x; \hat{\theta})}{Q(z)}$$

$$Q(z) = p(z|x; \hat{\theta}) \rightarrow \frac{p(x, z; \hat{\theta})}{Q(z)} = p(x; \hat{\theta})$$

$$(\hat{\theta} \rightarrow \textit{current best guess of } \theta)$$

(Cont'd)

- **Hence, we can construct a tight lower bound on L(θ) by setting:**

$$Q^*(z) = p(z|x;\theta)$$

  - Therefore, the **E-Step**, consists of:

    i. Finding the distribution of *z*, given the data, and the current best guess of θ.

    ii. Calculating the expectation over the *incomplete data likelihood* over the current best guess of θ:

$$E_{\hat{\theta}}[p(x|\theta)] = \sum_z p(z|x;\hat{\theta}) \log \frac{p(x,z;\theta)}{p(z|x;\hat{\theta})}$$

  - In the **M-Step**, the aim is to optimize the above w.r.t. to θ:

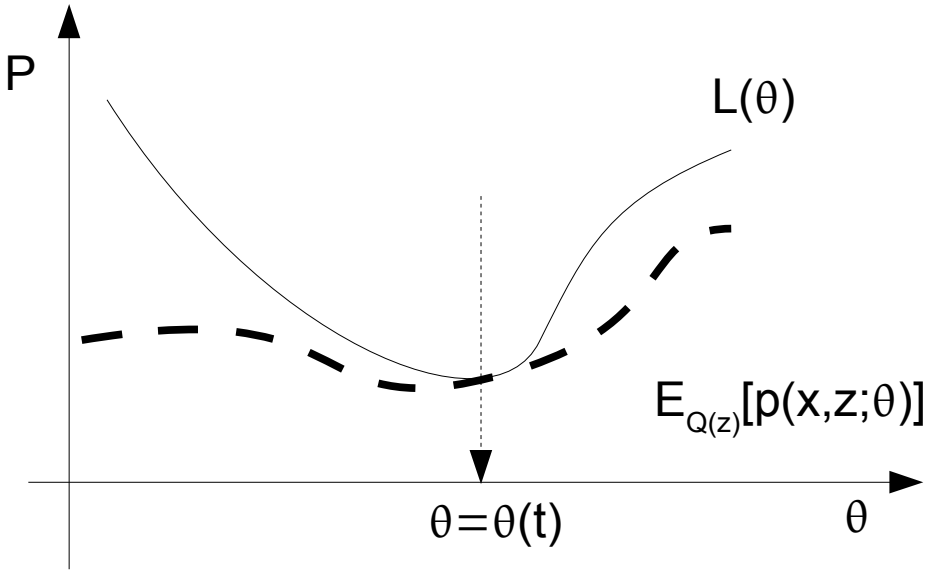
$$\theta^{t+1} = \underset{\theta}{argmax} \sum_z p(z|x;\hat{\theta}) \log \frac{p(x,z;\theta)}{p(z|x;\hat{\theta})}$$

    - Note that the value of p(z|x;θ) is based on the value calculated in the last iteration, and stays constant in this step

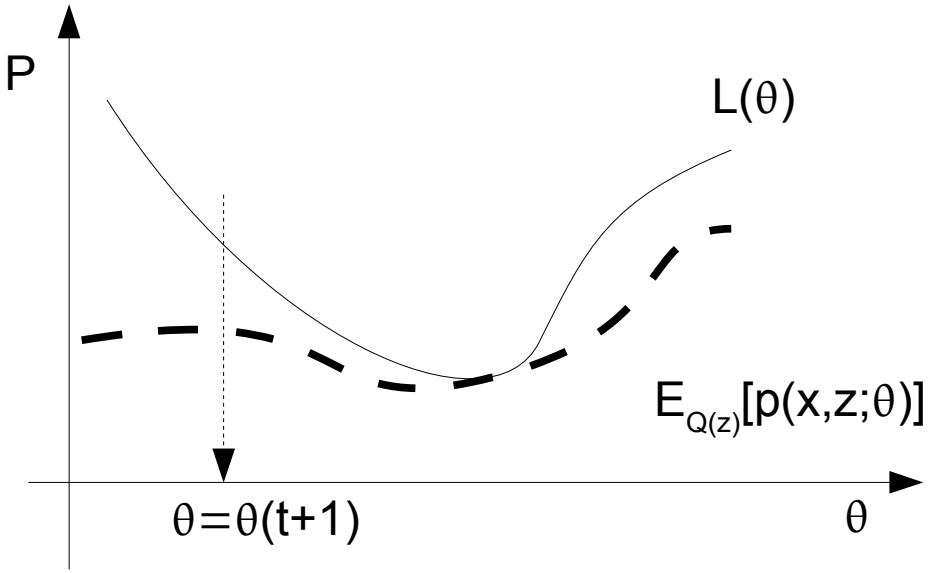
- **The *Expectation* and *Maximization* steps are repeated until convergence**

Graphical interpretation

- **In the E-Step,**
  - We set up lower bound on the likelihood
  - Take expectation w.r.t. Current best distribution of *z* – using current value of θ, creating a “tight” bound



- **In the M-Step:**
  - Fix the distribution by which the expectation is calculated
  - Optimize w.r.t. θ



Masdar INSTITUTE

Discussion

- **Applications**
  - pLSA
  - Factor Analysis
  - Mixture models
  - HMM
  - “Probabilistic Context Free Grammars”
  - Etc etc etc...
- **Advantages of EM algorithm**
  - Numerical stability → guaranteed to increase the likelihood with each iteration
  - Great for handling missing data and/or latent variables
  - Very elegant!
- **Disadvantages**
  - Convergence can be very slow depending on size of parameter space, etc.
  - Still subject to local minima problems
  - Setting up algorithm can be complex
  - Still a maximum-likelihood approach, not Bayesian!