Probabilistic Graphical Models

David Sontag

New York University

Lecture 11, April 12, 2012
Today: learning with partially observed data

- Overview of EM (expectation maximization) algorithm
- Application to mixture models
- Derivation of EM algorithm
- Variational EM
- Application to learning parameters of LDA
Maximum likelihood

- Recall from last week, that the *density estimation* approach to learning leads to *maximizing* the **empirical log-likelihood**

  \[
  \max_{\theta} \frac{1}{|\mathcal{D}|} \sum_{x \in \mathcal{D}} \log p(x; \theta)
  \]

- Suppose that our joint distribution is

  \[ p(X, Z; \theta) \]

  where our samples \( X \) are observed and the variables \( Z \) are never observed in \( \mathcal{D} \)

- That is, \( \mathcal{D} = \{(0,1,0,?,?,?), (1,1,1,?,?,?), (1,1,0,?,?,?), \ldots\} \)

- Assume that the hidden variables are *missing completely at random* (otherwise, we should explicitly model *why* the values are missing)
Maximum likelihood

- We can still use the same maximum likelihood approach. The objective that we are maximizing is

\[ \ell(\theta) = \frac{1}{|D|} \sum_{x \in D} \log \sum_{z} p(x, z; \theta) \]

- Because of the sum over \( z \), there is no longer a closed-form solution for \( \theta^* \) in the case of Bayesian networks.

- Furthermore, the objective is no longer convex, and potentially can have a different mode for every possible assignment \( z \).

- One option is to apply (projected) gradient ascent to reach a local maxima of \( \ell(\theta) \).
The expectation maximization (EM) algorithm is an alternative approach to reach a local maximum of $\ell(\theta)$.

Particularly useful in settings where there exists a closed form solution for $\theta^{\text{ML}}$ if we had fully observed data.

For example, in Bayesian networks, we have the closed form ML solution

$$
\theta_{x_i|x_{pa(i)}}^{\text{ML}} = \frac{N_{x_i,x_{pa(i)}}}{\sum_{\hat{x}_i} N_{\hat{x}_i,x_{pa(i)}}}
$$

where $N_{x_i,x_{pa(i)}}$ is the number of times that the (partial) assignment $x_i, x_{pa(i)}$ is observed in the training data.
Expectation maximization

Algorithm is as follows:

1. Write down the **complete log-likelihood** $\log p(x, z; \theta)$ in such a way that it is linear in $z$
2. Initialize $\theta_0$, e.g. at random or using a good first guess
3. Repeat until convergence:

   $$\theta_{t+1} = \arg \max_{\theta} \sum_{m=1}^{M} E_p(z_m|x_m; \theta_t) [\log p(x_m, Z; \theta)]$$

- Notice that $\log p(x_m, Z; \theta)$ is a random function because $Z$ is unknown
- By linearity of expectation, objective decomposes into expectation terms and data terms
- “E” step corresponds to computing the objective (i.e., the expectations)
- “M” step corresponds to **maximizing** the objective
Model on left is a **mixture model**
- Document is generated from a **single topic**

Model on right (latent Dirichlet Allocation) is an **admixture model**
- Document is generated from a **distribution** over topics
EM for mixture models

The complete likelihood is 

$$p(w, Z; \theta, \beta) = \prod_{d=1}^{D} p(w_d, Z_d; \theta, \beta),$$

where

$$p(w_d, Z_d; \theta, \beta) = \theta_{Z_d} \prod_{i=1}^{N} \beta_{Z_d,w_{id}}$$

Trick #1: re-write this as

$$p(w_d, Z_d; \theta, \beta) = \prod_{k=1}^{K} \theta_{k}^{1[Z_d=k]} \prod_{i=1}^{N} \prod_{k=1}^{K} \beta_{k,w_{id}}^{1[Z_d=k]}$$
EM for mixture models

- Thus, the complete log-likelihood is:

\[
\log p(w, Z; \theta, \beta) = \sum_{d=1}^{D} \left( \sum_{k=1}^{K} 1[Z_d = k] \log \theta_k + \sum_{i=1}^{N} \sum_{k=1}^{K} 1[Z_d = k] \log \beta_{k,wid} \right)
\]

- In the “E” step, we take the expectation of the complete log-likelihood with respect to \( p(z | w; \theta^t, \beta^t) \), applying linearity of expectation, i.e.

\[
E_{p(z|w;\theta^t,\beta^t)}[\log p(w, z; \theta, \beta)] =
\]

\[
\sum_{d=1}^{D} \left( \sum_{k=1}^{K} p(Z_d = k | w; \theta^t, \beta^t) \log \theta_k + \sum_{i=1}^{N} \sum_{k=1}^{K} p(Z_d = k | w; \theta^t, \beta^t) \log \beta_{k,wid} \right)
\]

- In the “M” step, we maximize this with respect to \( \theta \) and \( \beta \)
EM for mixture models

- Just as with complete data, this maximization can be done in closed form.
- First, re-write expected complete log-likelihood from

\[
\sum_{d=1}^{D} \left( \sum_{k=1}^{K} p(Z_d = k \mid w; \theta^t, \beta^t) \log \theta_k + \sum_{i=1}^{N} \sum_{k=1}^{K} p(Z_d = k \mid w; \theta^t, \beta^t) \log \beta_{k,w,d} \right)
\]

to

\[
\sum_{k=1}^{K} \log \theta_k \sum_{d=1}^{D} p(Z_d = k \mid w_d; \theta^t, \beta^t) + \sum_{k=1}^{K} \sum_{w=1}^{W} \log \beta_{k,w} \sum_{d=1}^{D} N_{dw} p(Z_d = k \mid w_d; \theta^t, \beta^t)
\]

- We then have that

\[
\theta_{k}^{t+1} = \frac{\sum_{d=1}^{D} p(Z_d = k \mid w_d; \theta^t, \beta^t)}{\sum_{\hat{k}=1}^{K} \sum_{d=1}^{D} p(Z_d = \hat{k} \mid w_d; \theta^t, \beta^t)}
\]
Derivation of EM algorithm

Derivation of EM algorithm

\[
L(\theta) = l(\theta | \theta_n)
\]

\[
\theta_n \rightarrow \theta_{n+1}
\]

\[
L(\theta) = l(\theta_n | \theta_n)
\]

\[
L(\theta_{n+1}) = l(\theta_{n+1} | \theta_n)
\]

Figure 2: Graphical interpretation of a single iteration of the EM algorithm:
The function \( l(\theta_n | \theta_n) \) is bounded above by the likelihood function \( L(\theta_n) \). The functions are equal at \( \theta_n \). The EM algorithm chooses \( \theta_{n+1} \) as the value of \( \theta \) for which \( l(\theta_n | \theta_n) \) is maximized. Since \( L(\theta_n) \geq l(\theta_n | \theta_n) \) increasing \( l(\theta_n | \theta_n) \) ensures that the value of the likelihood function \( L(\theta) \) is increased at each step.

We have now a function, \( l(\theta_n | \theta_n) \), which is bounded above by the likelihood function \( L(\theta_n) \). Additionally, observe that,

\[
l(\theta_n | \theta_n) = L(\theta_n) + \Delta(\theta_n | \theta_n)
\]

\[
= L(\theta_n) + \sum_z P(z | X, \theta_n) \ln P(X | z, \theta_n) P(z | \theta_n) P(z | X, \theta_n) P(X | \theta_n)
\]

\[
= L(\theta_n) + \sum_z P(z | X, \theta_n) \ln P(X, z | \theta_n)
\]

\[
= L(\theta_n).
\]

so for \( \theta = \theta_n \) the functions \( l(\theta | \theta_n) \) and \( L(\theta) \) are equal.

Our objective is to choose a values of \( \theta \) so that \( L(\theta) \) is maximized. We have shown that the function \( l(\theta | \theta_n) \) is bounded above by the likelihood function \( L(\theta_n) \) and that the value of the functions \( l(\theta | \theta_n) \) and \( L(\theta) \) are equal at the current estimate for \( \theta = \theta_n \). Therefore, any \( \theta \) which increases \( l(\theta | \theta_n) \) will also increase \( L(\theta) \). In order to achieve the greatest possible increase in the value of \( L(\theta) \), the EM algorithm calls for selecting \( \theta \) such that \( l(\theta | \theta_n) \) is maximized. We denote this updated value as \( \theta_{n+1} \). This process is illustrated in Figure (2).

(Figure from tutorial by Sean Borman)
Parameters are $\alpha$ and $\beta$

Both $\theta_d$ and $z_d$ are unobserved

The difficulty here is that inference is intractable

Could use Monte carlo methods to approximate the expectations
Variational EM

- Mean-field is ideally suited for this type of approximate inference together with learning

- Use the variational distribution

\[
q(\theta_d, z_d | \gamma_d, \phi_d) = q(\theta_d | \gamma_d) \prod_{n=1}^{N} q(z_n | \phi_{dn})
\]

- We then lower bound the log-likelihood using Jensen’s inequality:

\[
\log p(\mathbf{w} | \alpha, \beta) = \sum_d \log \int \sum_{z_d} p(\theta_d, \mathbf{z}_d, \mathbf{w}_d | \alpha, \beta) d\theta_d
\]

\[
= \sum_d \log \int \sum_{z_d} \frac{p(\theta_d, \mathbf{z}_d, \mathbf{w}_d | \alpha, \beta)q(\theta, \mathbf{z})}{q(\theta, \mathbf{z})} d\theta_d
\]

\[
\geq \sum_d E_q[\log p(\theta_d, \mathbf{z}_d, \mathbf{w}_d | \alpha, \beta)] - E_q[\log q(\theta, \mathbf{z})].
\]

- Finally, we maximize the lower bound with respect to \(\alpha, \beta,\) and \(q\).