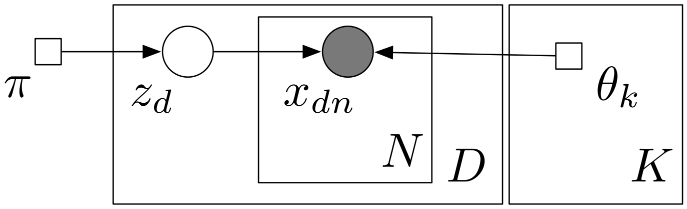
# Mixture Models and Expectation-Maximization

David M. Blei

March 9, 2012

## EM for mixtures of multinomials

- The graphical model for a mixture of multinomials



- How should we fit the parameters? Maximum likelihood.

- There is an issue

$$\begin{aligned}\log p(\mathbf{x}_{1:D} \,|\, \pi, \theta_{1:K}) &= \sum_{d=1}^{D} \log p(\mathbf{x}_d \,|\, \pi, \theta_{1:K}) && (1)\\ &= \sum_{d=1}^{D} \log \sum_{z_d=1}^{K} p(z_d \,|\, \pi) p(\mathbf{x}_d \,|\, \pi, \theta_{1:K}) && (2)\end{aligned}$$

- In the second line, the log cannot help simplify the summation.

- Why is that summation there? Because $z_d$ is a *hidden variable*.

- If $z_d$ were observed, what happens? Recall classification. The likelihood would decompose.

- In general, fitting hidden variable models with maximum likelihood is hard. THe EM algorithm can help. (But there are exceptions in both ways: some hidden variable models don't require EM and for others EM is not enough.)

- The expectation-maximimization algorithm is a general-purpose technique for fitting parameters in models with hidden variables.

- Here is the algorithm for mixtures (in English)

- REPEAT:

  1. For each document $d$, compute the conditional distribution of its cluster assignment $z_d$ given the current setting of the parameters $\pi^{(t)}$ and $\theta_{1:K}^{(t)}$.
  2. Obtain $\pi^{(t+1)}$ and $\theta_{1:K}^{(t+1)}$ by computing "MLE"s for each cluster, but weighting each data point by its posterior probability of being in that cluster.

- Notice that this resembles $k$-means. What are the differences?

- This strategy simply works. You can apply it in all the settings that we saw at the end of the slides. But we need to be precise about what we mean by "weighting."

- REPEAT:

  1. E-step: For each document $d$, compute $\lambda_d = p(z_d \mid \pi^{(t)}, \theta_{1:K}^{(t)})$
  2. M-step: For each cluster label's distribution over terms

  $$\theta_{k,v}^{(t+1)} = \frac{\sum_{d=1}^{D} \lambda_{d,k} n_v(\mathbf{x}_d)}{\sum_{d=1}^{D} \lambda_{d,k} n(\mathbf{x}_d)} \tag{3}$$

  and for the cluster proportions

  $$\pi_k^{(t+1)} = \frac{\sum_{d=1}^{D} \lambda_{d,k}}{D} \tag{4}$$

- Do we know how to do the E-step?

  $$p(z_d = k \mid \pi^{(t)}, \theta_{1:K}^{(t)}) \propto \pi_k \prod_{v=1}^{V} \theta_{k,v}^{n_v(\mathbf{x}_d)}. \tag{5}$$

  This is precisely the prediction computation from classification (e.g., from "is this email Spam or Ham?" to "is this email class 1,2,...,10?") We emphasize that the parameters (in the E-step) are fixed.

- How about the M-step for cluster-conditional distributions $\theta_k$?

  - The numerator is the expected number of times I saw word $v$ in class $k$. What is the expectation taken with respect to? The posterior distribution of the hidden variable, the variable that determines which cluster parameter each word was generated from.

    - The denominator is the expected number of words (total) that came from class $k$.

- How about the M-step for cluster proportions $\pi$?
    - The numerator is the expected number of times I saw a document from cluster $k$.
    - The denominator is the number of documents

- These are like traditional MLEs, but with expected counts. This is how EM works.

- It is actually maximizing the likelihood. Let's see why.

## Mixtures of Gaussians

- The mixture of Gaussians model is equivalent, except $x_d$ is now a Gaussian observation.

- The mixture components are means $\mu_{1:K}$ and variances $\sigma^2_{1:K}$. The proportions are $\pi$.

- Mixtures of Gaussians can express different kinds of distributions
    - Multimodal
    - Heavy-tailed

- As a density estimator, arbitrarily complex densities can be approximated with a mixture of Gaussians (given enough components).

- We fit these mixtures with EM.

- The E-step is conceptually the same. With the current setting of the parameters, compute the conditional distribution of the hidden variable $z_d$ for each data point $x_d$,

$$p(z_d \,|\, x_d, \pi, \mu_{1:K}, \sigma^2_{1:K}) \propto p(z_d \,|\, \pi) p(x_d \,|\, z_d, \mu_{1:K}, \sigma^2_{1:K}). \tag{6}$$

  This is just like for the mixtures of multinomials, only the data are now Gaussian.

- The M-step computes the empirical mean and variance for each component.
    - In $\mu_k$, each point is weighted by its posterior probability of being in the cluster.
    - In $\sigma^2_k$, weight $(x_d - \hat{\mu}_k)^2$ by the same posterior probability.

- The complex density estimate comes out of the predictive distribution

$$\begin{aligned} p(x \mid \pi, \mu_{1:K}, \sigma^2_{1:K}) &= \sum_z p(z \mid \pi) p(x \mid z, \mu_{1:K}, \sigma^2_{1:K}) && (7) \\ &= \sum_{k=1}^{K} \pi_k p(x \mid \mu_k, \sigma^2_k) && (8) \end{aligned}$$

  This is $K$ Gaussian bumps, each weighted by $\pi_k$.

## Why does this work?

- EM maximizes the likelihood.

- Generic model
  - The hidden variables are $z$.
  - The observations are $x$.
  - Tha parameters are $\theta$.
  - The joint distribution is $p(z, x \mid \theta)$.

- The *complete log likelihood* is

$$\ell_c(\theta; x, z) = \log p(z, x \mid \theta) \tag{9}$$

- For a distribution $q(z)$, the *expected complete log likelihood* is $\mathrm{E}[\log p(Z, x \mid \theta)]$.

- *Jensen's inequality*: In for a concave function and $\lambda \in (0,1)$,

$$f(\lambda x + (1-\lambda) y) \geq \lambda f(x) + (1-\lambda) f(y) \tag{10}$$

- Draw a picture with
  - $x$
  - $y$
  - $\lambda x + (1-\lambda) y$
  - $f(\lambda x + (1-\lambda) y)$
  - $\lambda f(x) + (1-\lambda) f(y)$

- This generalizes to probability distributions. For concave $f$,

$$f(\mathrm{E}[X]) \geq \mathrm{E}[f(X)]. \tag{11}$$

- Jensen's bound and $\mathscr{L}(\theta, q)$

$$\begin{aligned} \ell(\theta; x) &= \log p(x \mid \theta) && (12) \\ &= \log \sum_z p(x, z \mid \theta) && (13) \\ &= \log \sum_z q(z) \frac{p(x, z \mid \theta)}{q(z)} && (14) \\ &\geq \sum_z q(z) \log \frac{p(x, z \mid \theta)}{q(z)} && (15) \\ &= \mathscr{L}(q, \theta). && (16) \end{aligned}$$

  This bound on the log likelihood depends on a distribution over the hidden variables $q(z)$ and on the parameters $\theta$.

- EM is a coordinate ascent algorithm. We optimize $\mathscr{L}$ with respect to each argument, holding the other parameter fixed.

$$\begin{aligned} q^{(t+1)}(z) &= \arg\max_q \quad \mathscr{L}(q, \theta^(t)) && (17) \\ \theta^{(t+1)} &= \arg\max_\theta \quad \mathscr{L}(q^{(t+1)}, \theta) && (18) \end{aligned}$$

- In the E-step, we find $q^*(z)$. This equals the posterior $p(z \mid x, \theta)$,

$$\begin{aligned} \mathscr{L}(p(z \mid x, \theta), \theta) &= \sum_z p(z \mid x, \theta) \log \frac{p(x, z \mid \theta)}{p(z \mid x, \theta)} && (19) \\ &= \sum_z p(z \mid x, \theta) \log p(x \mid \theta) && (20) \\ &= \log p(x \mid \theta) && (21) \\ &= \ell(\theta; x) && (22) \end{aligned}$$

- In the M-step, we only need to worry about the expected complete log likelihood because

$$\mathscr{L}(q, \theta) = \sum_z q(z) \log p(x, z \mid \theta) - \sum_z q(z) \log q(z). \qquad (23)$$

  The second term does not depend on the parameter $\theta$. The first term is the expected complete log likelihood.

- Perspective of optimize bound; tighten bound

- This finds a local optimum & is sensitive to starting points

- Test for convergence: After the E-step, we can compute the likelihood at the point $\theta^{(t)}$.

## Big picture

- With EM, we can fit all kinds of hidden variable models with maximum likelihood. EM has had a huge impact on statistics, machine learning, signal processing, computer vision, natural language processing, speech recognition, genomics and other fields.

- Missing data was no longer a block. It opened up the idea of
  - Imagine I can observe everything. Then I can fit the MLE.
  - But I cannot observe everything. With EM, I can still try to fit the MLE.

- For example
  - Used to "fill in" missing pixels in tomography.
  - Modeling nonresponses in sample surveys.
  - Many applications in genetics—treating ancentral variables and various unknown rates as unobserved random variables.
  - Time series models (we'll see these)
  - Financial models
  - EM application at Amazon: Users and items are clustered; a user's rating on an item has to do with their cluster combination.

- Closely related to
  - Variational methods for approximate posterior inference
  - Gibbs sampling

- Both let us have the same flexibility in modeling, but in the set up where we cannot compute the real posterior. (This often occurs in Bayesian models.)

- History of the EM algorithm
  - Dempster et al. (1977): "Maximum likelihood from incomplete data via the EM algorithm"
  - Hartley: "I feel like the old minstrel who has been singing his song for 18 years and now finds, with considerable satisfaction, that his folklore is the theme of an overpowering symphony."
  - Hartley (1958) "Maximum likelihood procedures for incomplete data"
  - McKendrik (1926) "Applications of mathematics to medical problems"