The EM Algorithm: An Optimization View

Chase Geigle
University of Illinois at Urbana-Champaign
Department of Computer Science
geigle1@illinois.edu

September 30, 2016

Suppose you have a probability model for some data you've obtained, but you want to model its generative process using values that are unobserved. For example, you might assume that you have words being generated from a two-component mixture model

$$\lambda p(w_i \mid C) + (1-\lambda) p(w_i \mid \theta)$$

but you only observe the individual words w_i and do not know from which component they were generated. Thus, you would have a log-likelihood function

$$\log p(D \mid \theta) = \sum_{d \in D} \sum_{i=1}^{|d|} \log \left\{ \lambda p(w_i \mid C) + (1-\lambda) p(w_i \mid \theta) \right\}.$$

Finding the maximum likelihood estimate for θ analytically is difficult because of the summation inside of the logarithm, so we are forced to use some sort of numerical algorithm.

The EM algorithm [1] is one such optimization algorithm for solving for maximum-likelihood estimates. This note will briefly introduce *one particular way* of deriving and thinking about the EM algorithm by looking at it from an coordinate-ascent optimization (maximization-maximization) perspective. We first will give a general derivation of the EM algorithm, and then will investigate a few specific examples where we might leverage the EM algorithm for parameter learning.

1 Maximizing the Marginal Likelihood

Suppose you have a probability model $p(X, Z \mid \Theta)$ where X are observed data and Z are latent variables, and Θ are the model's parameters. (For the above case, the latent variables Z would be the indicator variables z_i that tell us from which component word w_i was drawn.) We wish to solve the problem

$$\Theta^* = \arg\max_{\Theta} p(X \mid \Theta) = \arg\max_{\Theta} \log p(X \mid \Theta) = \arg\max_{\Theta} \log \left\{ \sum_{Z} p(X, Z \mid \Theta) \right\}.$$

This is problematic due to the summation inside the logarithm (which results from having to marginalize out the hidden[1] variables Z), which makes finding an analytical solution for Θ^* either too difficult or flat-out impossible.

[1] Also: "unobserved" or "latent"

Let's derive a way to cause the summation to be *outside* the logarithm rather than within it. We start with

$$\log p(X \mid \Theta) = \sum_Z q(Z) \log p(X \mid \Theta)$$

where we suppose that $q(Z)$ is some distribution over the latent variables Z. This may seem strange, but we can now rewrite this as

$$\log p(X \mid \Theta) = \sum_Z q(Z) \log \frac{p(X, Z \mid \Theta)}{p(Z \mid X, \Theta)}$$

by using the fact that $p(X, Z \mid \Theta) = p(X \mid \Theta) p(Z \mid X, \Theta)$. This rewriting allows us to avoid having to sum over all of the latent variable values within the logarithm. Furthermore, we can see from this rewriting that

$$\log p(X \mid \Theta) = \sum_Z q(Z) \log p(X, Z \mid \Theta) + H(q, p)$$

where $H(q,p)$ is the cross-entropy between $q(Z)$ and $p(Z \mid X, \Theta)$. Things may become even more clear if we add and subtract $H(q)$, the entropy of $q(Z)$:

$$\begin{aligned}\log p(X \mid \Theta) &= \sum_Z q(Z) \log p(X, Z \mid \Theta) + H(q,p) - H(q) + H(q) \\ &= \sum_Z q(Z) \log \frac{p(X, Z \mid \Theta)}{q(Z)} + KL(q \mid\mid p) \\ &= F(q, \Theta) + KL(q \mid\mid p)\end{aligned}$$

where $KL(q \mid\mid p)$ is the Kullback-Leibler divergence from $p(Z \mid X, \Theta)$ to $q(Z)$. Since the KL-divergence is non-negative[2], we have thus established $F(q, \Theta)$ to be a lower bound for the marginal log-likelihood $\log p(X \mid \Theta)$. We can now view the EM algorithm as being a coordinate ascent method for maximizing $F(q, \Theta)$ directly, which will in turn maximize $\log p(X \mid \Theta)$[3]. Specifically, we can view its steps as:

E-Step: Fix $\Theta^{(n)}$ and maximize $F(q, \Theta^{(n)})$ with respect to q. Since we treat $\Theta^{(n)}$ as fixed, the only term that varies in $F(q, \Theta^{(n)})$ is the KL-divergence term, which is minimized when $q(z) = p(Z \mid X, \Theta^{(n)})$, which can be solved analytically in most cases. Algorithmically, this would be computing the probability of the individual latent variable assignments conditioned on the observed data and the current parameter settings.

In some cases, however, it can be difficult to compute $p(Z \mid X, \Theta^{(n)})$ directly, so we can instead choose $q(z)$ to be from some simpler family of distributions and solve an optimization subproblem to minimize $KL(q \mid\mid p)$. In these cases, the algorithm may be referred to as "variational inference" or "variational EM", as $q(z)$ is a variational distribution[4].

M-Step: Fix q and maximize $F(q, \Theta^{(n+1)})$ with respect to $\Theta^{(n+1)}$. Algorithmically, this amounts to re-estimating the model parameters using the $q(Z)$ distribution and our observations X (for example, by re-normalizing expected counts computed using $q(Z)$). This is much easier to do than solving the original optimization problem directly, as now we can pretend that we have the "complete" data, including (soft, fractional) latent variable assignments.

[2] This can be proved using Jensen's inequality, among other ways.

[3] In the case that the function has only one maximum, this will guarantee that you arrive at the global maximum. In general, however, this optimization only guarantees that you arrive at a local maximum, which means that your starting point becomes quite important.

[4] Variational inference is a commonly used technique for finding parameters in graphical models, such as LDA.

2 Deriving the EM Algorithm for Feedback Modeling

Recall that in the mixture model feedback approach from Zhai and Lafferty [3] we assumed that we had a set of feedback documents $D = \{d_1, \ldots, d_N\}$ which were generated from a two-component multinomial mixture model. The parameter λ, which is assumed to be provided in advance and chosen empirically, represents the probability of a word w_i in a document d being generated by the background language model $p(w \mid C)$ which is also assumed to be provided in advance and is learned using MLE on the entire collection of documents C. With probability $1 - \lambda$, a word is generated from the distribution $p(w \mid \theta_F)$. We wish to learn the parameters for this distribution.

Our log likelihood function given the above generative process is

$$\log p(D \mid \theta_F) = \sum_{i=1}^{N} \sum_{j=1}^{|d_i|} \log \left\{ \lambda p(w_{i,j} \mid C) + (1 - \lambda) p(w_{i,j} \mid \theta_F) \right\}$$

and we can immediately see the troublesome summation occurring within the logarithm, making finding a simple solution using Lagrange multipliers difficult. We can, however, view our data as being "incomplete" in that it is missing binary latent variables $z_{i,j}$ that each indicate which distribution the word $w_{i,j}$ was drawn from. Let's say $z_{i,j} = 0$ indicates that the word was drawn from the background $p(w_{i,j} \mid C)$ and $z_{i,j} = 1$ indicates that it was drawn instead from the feedback distribution $p(w_{i,j} \mid \theta_F)$.

Let's write the complete data likelihood, $p(D, Z \mid \theta_F)$.

$$\begin{aligned} p(D, Z \mid \theta_F) &= p(D \mid Z, \theta_F) P(Z \mid \theta_F) \\ &= \prod_{i=1}^{N} \prod_{j=1}^{|d_i|} p(w_{i,j} \mid z_{i,j}, \theta_F) p(z_{i,j} \mid \theta_F) \end{aligned}$$

where we have

$$p(w_{i,j} \mid z_{i,j}, \theta_F) = \begin{cases} p(w_{i,j} \mid \theta_F) & \text{if } z_{i,j} = 1 \\ p(w_{i,j} \mid C) & \text{if } z_{i,j} = 0 \end{cases}$$

and

$$p(z_{i,j} \mid \theta_F) = \begin{cases} 1 - \lambda & \text{if } z_{i,j} = 1 \\ \lambda & \text{if } z_{i,j} = 0. \end{cases}$$

This can then be written as

$$p(D, Z \mid \theta_F) = \prod_{i=1}^{N} \prod_{j=1}^{|d_i|} \left(\lambda p(w_{i,j} \mid C) \right)^{1 - z_{i,j}} \left((1 - \lambda) p(w_{i,j} \mid \theta_F) \right)^{z_{i,j}}$$

and thus our complete data log likelihood is

$$\log p(D, Z \mid \theta_F) = \sum_{i=1}^{N} \sum_{j=1}^{|d_i|} \left((1 - z_{i,j}) \log(\lambda p(w_{i,j} \mid C)) + z_{ij} \log((1 - \lambda) p(w_{i,j} \mid \theta_F)) \right).$$

We can now write the lower bound function $F(q, \theta_F)$ as

$$\begin{aligned} F(q, \theta_F) &= \sum_{Z} q(Z) \log \frac{p(D, Z \mid \theta_F)}{q(Z)} \\ &= \sum_{i=1}^{N} \sum_{j=1}^{|d_i|} \left(q(z_{i,j} = 0) \log(\lambda p(w_{i,j} \mid C)) + q(z_{i,j} = 1) \log((1 - \lambda) p(w_{i,j} \mid \theta_F)) \right) + H(q) \end{aligned}$$

With this, we have everything we need to derive an EM algorithm for solving for the maximum likelihood estimate for θ_F.

E-step: Since we established earlier that maximizing $F(q, \theta_F^{(n)})$ with $\theta_F^{(n)}$ fixed means minimizing the KL-divergence between $q(Z)$ and $p(Z \mid D, \theta_F^{(n)})$, the main computation here is to compute

$$p(z_{i,j} = 0 \mid D, \theta_F^{(n)}) = \frac{\lambda p(w_{i,j} \mid C)}{\lambda p(w_{i,j} \mid C) + (1 - \lambda) p(w_{i,j} \mid \theta_F^{(n)})}$$

which can be rationalized using Bayes' rule. We can then simply let

$$q(z_{i,j} = 0) = p(z_{i,j} = 0 \mid D, \theta_F^{(n)})$$

and

$$q(z_{i,j} = 1) = 1 - q(z_{i,j} = 0)$$

to complete the setting of q to maximize $L(q, \theta_F^{(n)})$.

M-step: Now we need to maximize $L(q, \theta_F^{(n+1)})$, holding q fixed to the value we computed in the E-step. We note that the entropy term $H(q)$ is a constant and thus we can ignore it. We can then set about maximizing the first term in the expression. We can do this analytically by introducing Lagrange multipliers and taking derivatives with respect to each parameter $p(w \mid \theta_F)$.

We could also, however, recognize that the term we are trying to maximize is simply the expectation of the complete data log likelihood with respect to the distribution q from the E-step. This observation allows us to consider instead computing "expected counts" of events from our observed data, using q to distribute each actual count among the uncertainty in Z.

In other words, we can begin by collecting all of the counts in the data for observing a particular word w. We then distribute these counts, fractionally, into the cases where w was drawn from the background and into the cases where w was drawn from the feedback distribution $\theta_F^{(n)}$ by weighting them by the probability of those two cases based on q.

Let's let $n_{w,F}$ be the number of times we expect to see word w drawn from the feedback distribution given our data D and our latent variable distribution q. We can see that

$$n_{w,F} = \sum_{i=1}^{N} \sum_{j=1, d_{i,j}=w}^{|d_i|} q(z_{i,j} = 1) = \sum_{d \in D} q(z_w = 1) c(w, d).$$

where we've relabeled z to be indexed by w by noting that $z_{i,j}$ depends only on the specific word type $d_{i,j} = w$.

Since we know how to estimate a multinomial distribution given count data, we can use these numbers directly to re-estimate our parameters for $p(w \mid \theta_F^{(n+1)})$. Specifically, we have

$$p(w \mid \theta_F^{(n+1)}) = \frac{n_{w,F}}{\sum_{w' \in V} n_{w',F}} = \frac{\sum_{i=1}^{N} q(z_w = 1) c(w, d_i)}{\sum_{i=1}^{N} \sum_{w' \in V} q(z_{w'} = 1) c(w', d_i)}$$

which is exactly the same estimate we would arrive at using the Lagrange multiplier approach[5].

[5] You can see this approach worked out in detail here: `http://sifaka.cs.uiuc.edu/czhai/pub/em-note.pdf`

3 Deriving the EM Algorithm for PLSA

In the original formulation for PLSA from Hofmann [2], we assume that we have a collection of documents $D = \{d_1, d_2, \ldots d_N\}$ that are generated by drawing individual words from a set of K topics, which are multinomial distributions over words in a fixed vocabulary V. Each document d_i has some distribution π_i that is used to first pick one of the K topics, and then a word $w_{i,j}$ is drawn from the topic distribution θ_k. With such a generative process, we would have a log likelihood of

$$\log p(D \mid \Pi, \Theta) = \sum_{i=1}^{N} \sum_{j=1}^{|d_i|} \log \left\{ \sum_{k=1}^{K} p(z_{i,j} = k \mid \pi_i) p(w_{i,j} \mid \theta_k) \right\}.$$

We can immediately recognize the same problematic form of the log likelihood function like before—we have a summation over the different topics occurring *within* the logarithm. Fortunately, we can turn to the EM algorithm to help us solve for maximum likelihood estimates for the parameters Π and Θ.

E-step: Again, the main computational challenge here is to compute the distribution over the latent variables given the current model parameters $\Pi^{(n)}$ and $\Theta^{(n)}$ and the observations. Again by Bayes' rule, we have

$$p(z_{i,j} = k \mid D, \Pi^{(n)}, \Theta^{(n)}) = \frac{p(z_{i,j} = k \mid \pi_i^{(n)}) p(w_{i,j} \mid \theta_k^{(n)})}{\sum_{k'=1}^{K} p(z_{i,j} = k' \mid \pi_i^{(n)}) p(w_{i,j} \mid \theta_{k'}^{(n)})}$$

and we simply let $q(z_{i,j} = k) = p(z_{i,j} = k \mid D, \Pi^{(n)}, \Theta^{(n)})$.

M-step: Given the distribution q over the latent variable assignments, we can now re-estimate the parameters $\Pi^{(n+1)}$ and $\Theta^{(n+1)}$. We will again take an "expected counts" view. Let $n_{w,k}$ indicate the number of times we expect to see a word type w assigned to topic k, and let $n_{d,k}$ indicate the number of times we expect to see a word in d assigned to topic k.

We have

$$n_{w,k} = \sum_{i=1}^{N} \sum_{j=1, d_{i,j}=w} q(z_{i,j} = k) = \sum_{d \in D} c(w, d) q(z_{d,w} = k)$$

and

$$n_{d,k} = \sum_{j=1}^{|d|} q(z_{d,j} = k) = \sum_{w \in d} c(w, d) q(z_{d,w} = k).$$

We can then re-estimate our parameters by normalizing these counts. Specifically,

$$p(w \mid \theta_k^{(n+1)}) = \frac{n_{w,k}}{\sum_{w' \in V} n_{w',k}} = \frac{\sum_{d \in D} c(w, d) q(z_{d,w} = k)}{\sum_{w' \in V} \sum_{d \in D} c(w', d) q(z_{d,w'} = k)}$$

and

$$p(z_{d,w} = k \mid \pi_d^{(n+1)}) = \frac{n_{d,k}}{\sum_{k'=1}^{K} n_{d,k'}} = \frac{\sum_{w \in d} c(w, d) q(z_{d,w} = k)}{\sum_{k'=1}^{K} \sum_{w \in d} c(w, d) q(z_{d,w} = k')}.$$

4 Deriving the EM Algorithm for a PLSA Mixture Model

In Zhai et al. [4], a modification of the original PLSA model is proposed by suggesting that we view the generative process for a word as consisting of two main steps. First, with probability λ we generate a word from the background language model $p(w \mid D)$ (which is estimated with MLE and is fixed) and with probability $1-\lambda$ we generate the word from the standard PLSA model, where λ is a parameter set heuristically in advance.

This can be thought of as combining the feedback model with the PLSA model, and we can thus define two different latent variables in play[6]. Let $y_{i,j}$ be a binary random variable that, mirroring the feedback model, indicates whether word occurrence $d_{i,j}$ was drawn from the background ($y_{i,j} = 0$) or not ($y_{i,j} = 1$). Then we can let $z_{i,j}$ be an indicator variable that denotes which of the K topics in the PLSA mixture the word $d_{i,j}$ was drawn from, or 0 in the case it was drawn from the background. We can then define the likelihood of the data D as

$$\begin{aligned}
p(D \mid \Theta, \Pi) &= \prod_{i=1}^{N}\prod_{j=1}^{|d_i|}\sum_{\ell=0}^{1}\sum_{k=0}^{K} p(d_{i,j} = w, y_{i,j} = \ell, z_{i,j} = k \mid \Theta, \Pi) \\
&= \prod_{i=1}^{N}\prod_{j=1}^{|d_i|}\sum_{\ell=0}^{1}\sum_{k=0}^{K} p(y_{i,j} = \ell \mid \Theta, \Pi) p(z_{i,j} = k \mid y_{i,j} = \ell, \Theta, \Pi) p(d_{i,j} = w \mid z_{i,j} = k, y_{i,j} = \ell, \Theta, \Pi) \\
&= \prod_{i=1}^{N}\prod_{j=1}^{|d_i|}\Bigg(\lambda \sum_{k=0}^{K} p(z_{i,j} = k \mid y_{i,j} = 0, \Theta, \Pi) p(d_{i,j} = w \mid z_{i,j} = k \mid \Theta, \Pi) \\
&\qquad + (1-\lambda)\sum_{k=0}^{K} p(z_{i,j} = k \mid y_{i,j} = 1, \Theta, \Pi) p(d_{i,j} = w \mid z_{i,j} = k \mid \Theta, \Pi)\Bigg)
\end{aligned}$$

where in the third line we've simply expanded out the sum over the uncertainty in $y_{i,j}$ and observed that $p(y_{i,j} \mid \Theta, \Pi)$ is conditionally independent of our model parameters (since we are assuming it to be a parameter specified in advance). We can now simplify this expression by making the observation that $p(z_{i,j} = k \mid y_{i,j} = 0, \Theta, \Pi) = 0$ for $k > 0$ (since if $y_{i,j} = 0$ we are guaranteed to draw from the background). Similarly, $p(z_{i,j} = 0 \mid y_{i,j} = 1, \Theta, \Pi) = 0$ since if $y_{i,j} = 1$ we are forced to draw from one of the K topics from the PLSA mixture. Thus, we have

$$\begin{aligned}
p(D \mid \Theta, \Pi) &= \prod_{i=1}^{N}\prod_{j=1}^{|d_i|}\left(\lambda p(d_{i,j} = w \mid D) + (1-\lambda)\sum_{k=1}^{K} p(z_{i,j} = k \mid y_{i,j} = 1, \Theta, \Pi) p(d_{i,j} = w \mid \Pi, \Theta)\right) \\
&= \prod_{i=1}^{N}\prod_{j=1}^{|d_i|}\left(\lambda p(d_{i,j} = w \mid D) + (1-\lambda)\sum_{k=1}^{K} p(z_{i,j} = k \mid \pi_i) p(d_{i,j} = w \mid \theta_k)\right)
\end{aligned}$$

when we substitute in our model estimates.

It is important to note here that $p(z_{i,j} = k \mid \pi_i)$ is modeling the probability that $z_{i,j} = k$ *given that we are generating the word from the PLSA mixture*, and thus it sums to 1 when summing from $k = 1$ up to K rather than from $k = 0$ up to K.

[6] This isn't strictly necessary, but it is done here to show the connection with the previous two models we discussed. For an alternative derivation that uses only one set of latent variables Z, see this note: `http://times.cs.uiuc.edu/course/598f16/plsa-note.pdf`.

We thus have a log likelihood of

$$\log p(D \mid \Theta, \Pi) = \sum_{i=1}^{N} \sum_{j=1}^{|d_i|} \log \left\{ \lambda p(d_{i,j} = w \mid D) + (1-\lambda) \sum_{k=1}^{K} p(z_{i,j} = k \mid \pi_i) p(d_{i,j} = w \mid \theta_k) \right\}$$

where we can observe again the problematic summation occurring within the logarithm. We thus turn to EM again for finding the maximum likelihood estimates for the model parameters Θ and Π.

E-step: Our main computation in the E-step is to estimate the joint distribution over the latent variables Y and Z given the observations D and our current model parameters $\Theta^{(n)}$ and $\Pi^{(n)}$.

First, we can observe that

$$p(y_{i,j} = \ell, z_{i,j} = k \mid D, \Theta^{(n)}, \Pi^{(n)}) = p(y_{i,j} = \ell \mid D, \Theta^{(n)}, \Pi^{(n)}) p(z_{i,j} = k \mid y_{i,j} = \ell, D, \Theta^{n}, \Pi^{(n)})$$

and thus we can break this problem down into estimating two distributions: q_y and $q_{z|y}$ for the first and second term, respectively.

Focusing on the first term, and noting that since $y_{i,j}$ is binary random variable we can focus on only one specific case, we have

$$p(y_{i,j} = 1 \mid D, \Theta^{(n)}, \Pi^{(n)}) = p(y_{i,j} = 1 \mid d_{i,j} = w, \Theta^{(n)}, \pi_i)$$

based on our independence assumptions, and

$$= \frac{p(d_{i,j} = w \mid y_{i,j} = 1, \Theta^{(n)}, \pi_i^{(n)}) p(y_{i,j} = 1 \mid \Theta^{(n)}, \pi_i^{(n)})}{p(d_{i,j} = w \mid \Theta^{(n)}, \pi_i^{(n)})}$$

by Bayes' rule. Substituting in our model distributions, we have

$$= \frac{(1-\lambda) \sum_{k=1}^{K} p(z_{i,j} = k \mid \pi_i^{(n)}) p(w \mid \theta_k)}{\lambda p(w \mid D) + (1-\lambda) \sum_{k=1}^{K} p(z_{i,j} = k \mid \pi_i^{(n)}) p(w \mid \theta_k)}$$

and we can then set $q_y(y_{i,j} = 1) = p(y_{i,j} = 1 \mid D, \Theta^{(n)}, \Pi^{(n)})$.[7]

Let's now focus on the second term. We know that $p(z_{i,j} = 0 \mid y_{i,j} = 0, \Theta^{(n)}, \Pi^{(n)}) = 1$ by our model definition, so we only need to concern ourselves with estimating $p(z_{i,j} = k \mid y_{i,j} = 1, \Theta^{(n)}, \Pi^{(n)})$. Notice, however, that if $y_{i,j} = 1$ then we know for certain that we are sampling from the PLSA mixture (and thus $p(z_{i,j} = 0 \mid y_{i,j} = 1, \Theta^{(n)}, \Pi^{(n)}) = 0$), so we will end up with the exact same estimate for $q_{z|y}$ as we had for q in the PLSA derivation. Specifically, we have, for $k > 0$,

$$p(z_{i,j} = k \mid y_{i,j} = 1, \Theta^{(n)}, \Pi^{(n)}) = \frac{p(z_{i,j} = k \mid \pi_i^{(n)}) p(w_{i,j} \mid \theta_k^{(n)})}{\sum_{k'=1}^{K} p(z_{i,j} = k' \mid \pi_i^{(n)}) p(w_{i,j} \mid \theta_{k'}^{(n)})}$$

and we simply let $q_{z|y}(z_{i,j} = k) = p(z_{i,j} = k \mid y_{i,j} = 1, \Theta^{(n)}, \Pi^{(n)})$.

[7] Notice the similarity of this formula to that we discovered in the E-step for the feedback model. We've simply replaced $p(w \mid \theta_F)$ with the marginal probability of generating a word w from the PLSA mixture model.

M-step: We now need to re-estimate the parameters for our model $\Theta^{(n+1)}$ and $\Pi^{(n+1)}$ using the distributions q_y and $q_{z|y}$ that we estimated in the E-step. We again will take an "expected counts" view. Let $n_{d,k}$ be the number of times we expect to see a word in document d assigned to topic k from the PLSA mixture model, and let $n_{w,k}$ be the number of times we expect to see a specific word type w assigned to topic k from the PLSA mixture model.

We have

$$n_{d,k} = \sum_{w \in d} c(w,d) q_y(y_{d,w} = 1) q_{z|y}(z_{d,w} = k)$$

and

$$n_{w,k} = \sum_{d \in D} c(w,d) q_y(y_{d,w} = 1) q_{z|y}(z_{d,w} = k).$$

Let's look at the product $q_y(y_{d,w} = 1)q_{z|y}(z_{d,w} = k)$. We see that

$$\begin{aligned} q_y(y_{d,w} = 1) q_{z|y}(z_{d,w} = k) &= \left(\frac{(1-\lambda) \sum_{k'=1}^{K} p(z_{d,w} = k' \mid \pi_d^{(n)}) p(w \mid \theta_{k'}^{(n)})}{\lambda p(w \mid D) + (1-\lambda) \sum_{k'=1}^{K} p(z_{d,w} = k' \mid \pi_d^{(n)}) p(w \mid \theta_{k'}^{(n)})} \right) \\ &\quad \times \left(\frac{p(z_{d,w} = k \mid \pi_d^{(n)}) p(w \mid \theta_k^{(n)})}{\sum_{k'=1}^{K} p(z_{d,w} = k' \mid \pi_d^{(n)}) p(w \mid \theta_{k'}^{(n)})} \right) \\ &= \frac{(1-\lambda) p(z_{d,w} = k \mid \pi_d^{(n)}) p(w \mid \theta_k^{(n)})}{\lambda p(w \mid D) + (1-\lambda) \sum_{k'=1}^{K} p(z_{d,w} = k' \mid \pi_d^{(n)}) p(w \mid \theta_{k'}^{(n)})} \end{aligned}$$

which can be used to simplify the computation of the expected counts[8].

Finally, we can normalize the expected counts to come up with the new estimates of our model's parameters. Specifically,

$$p(z = k \mid \pi_d^{(n+1)}) = \frac{n_{d,k}}{\sum_{k'=1}^{K} n_{d,k'}} = \frac{\sum_{w \in d} c(w,d) q_y(y_{d,w} = 1) q_{z|y}(z_{d,w} = k)}{\sum_{k'=1}^{K} \sum_{w \in d} c(w,d) q_y(y_{d,w} = 1) q_{z|y}(z_{d,w} = k')}$$

and

$$p(w \mid \theta_k^{(n+1)}) = \frac{n_{w,k}}{\sum_{w' \in V} n_{w',k}} = \frac{\sum_{d \in D} c(w,d) q_y(y_{d,w} = 1) q_{z|y}(z_{d,w} = k)}{\sum_{w' \in V} \sum_{d \in D} c(w',d) q_y(y_{d,w'} = 1) q_{z|y}(z_{d,w'} = k)}.$$

References

[1] A. P. Dempster, N. M. Laird, and D. B. Rudin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B*, 39(1):1–38, 1977.

[2] Thomas Hofmann. Probabilistic latent semantic analysis. In *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence*, UAI'99, pages 289–296, 1999.

[3] Chengxiang Zhai and John Lafferty. Model-based feedback in the language modeling approach to information retrieval. In *Proceedings of the Tenth International Conference on Information and Knowledge Management*, pages 403–410. ACM, 2001.

[8] This equation is exactly $p(y_{d,w} = 1, z_{d,w} = k \mid D, \Theta^{(n)}, \Pi^{(n)})$ (for $k > 0$), which we can compute during the E-step instead of explicitly representing this probability with two different distributions q_y and $q_{z|y}$, which was done here mostly for notational clarity.

[4] ChengXiang Zhai, Atulya Velivelli, and Bei Yu. A cross-collection mixture model for comparative text mining. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 743–748. ACM, 2004.