Expectation-Maximization Algorithm

Timothy Hanson

Department of Statistics, University of South Carolina

Stat 740: Statistical Computing

Expectation-Maximization algorithm

Dempster, Laird, and Rubin (1977): groundbreaking paper with 100's (1000's?) of applications.

An iterative procedure (like Newton-Raphson) to obtain MLE of $L(\boldsymbol{\theta}|\mathbf{x})$ or posterior mode of $\pi(\boldsymbol{\theta}|\mathbf{x})$, i.e. algorithm creates a $\mathbf{g}(\cdot)$ for iterative procedure $\boldsymbol{\theta}^{t+1} = \mathbf{g}(\boldsymbol{\theta}^t)$, $\mathbf{g}(\cdot) : \mathbb{R}^k \to \mathbb{R}^k$.

In what follows we'll use $L(\boldsymbol{\theta}|\mathbf{x})$, but works the same for $\pi(\boldsymbol{\theta}|\mathbf{x}) = L(\boldsymbol{\theta}|\mathbf{x})\pi(\boldsymbol{\theta})$.

Idea & method

Introduce latent (sometimes called "missing") data (could be model parameters) $\mathbf{z}$ so that $L(\boldsymbol{\theta}|\mathbf{x},\mathbf{z})$ is easier to maximize than $L(\boldsymbol{\theta}|\mathbf{x})$. Hope that resulting $\mathbf{g}(\cdot)$ isn't too horrible (sometimes it is). Initialize $\boldsymbol{\theta}^0$ and $t = 0$.

1. E-step: $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^t) = E_{\mathbf{z}|\mathbf{x},\boldsymbol{\theta}^t}\{\log L(\boldsymbol{\theta}|\mathbf{z},\mathbf{x})\}$.
2. M-step: $\boldsymbol{\theta}^{t+1} = \text{argmax}_{\boldsymbol{\theta}\in\boldsymbol{\Theta}} Q(\boldsymbol{\theta}|\boldsymbol{\theta}^t) = \mathbf{g}(\boldsymbol{\theta}^t)$.

Repeat until $||\boldsymbol{\theta}^{t+1} - \boldsymbol{\theta}^t|| < \epsilon$ for some norm $||\cdot||$.

E-M success stories

- Linear mixed models (Laird & Ware, 1982); random effects "missing."
- Generalized linear mixed models; not as easy as LMM.
- Finite mixture models; component membership "missing."
- Various contingency tables arising from genetics (Tanner, 1996; Givens & Hoeting, 2013; Lange, 2010).
- Censored and/or truncated data models. Missing data are true observations.

Finite mixture models: unsupervised learning

Finite mixture of normals is often used for model-based clustering:

$$X_1, \ldots, X_n | \boldsymbol{\theta} \overset{iid}{\sim} f(x) = \sum_{j=1}^{J} \pi_j \phi(x|\mu_j, \sigma_j^2).$$

Parameters are
$\boldsymbol{\theta} = (\pi_1, \ldots, \pi_{J-1}, \mu_1, \ldots, \mu_J, \sigma_1^2, \ldots, \sigma_J^2)' \in \mathbb{R}^{3J-1}$. Direct maximization of

$$L(\boldsymbol{\theta}|\mathbf{x}) = \prod_{j=1}^{n} \sum_{j=1}^{J} \pi_j \phi(x_i|\mu_j, \sigma_j^2)$$

is *very challenging*.

Component membership

Recall method of composition $X_i|\boldsymbol{\theta}, z_i \sim N(\mu_{z_i}, \sigma^2_{z_i})$ conditionally, and $p(j|\boldsymbol{\theta}) = P(z_i = j|\boldsymbol{\theta}) = \pi_j$ marginally, gives same distribution $f(x)$ on previous slide. Add "missing" $\mathbf{z} = (z_1, \ldots, z_n)'$ to the model to get

$$
\begin{aligned}
L(\boldsymbol{\theta}|\mathbf{x}, \mathbf{z}) &= f(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta}) = f(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta}) f(\mathbf{z}|\boldsymbol{\theta}) \\
&= \left[\prod_{i=1}^{n} \phi(x_i|\mu_{z_i}, \sigma^2_{z_i})\right] \left[\prod_{i=1}^{n} p(z_i|\boldsymbol{\theta})\right] \\
&= \prod_{i=1}^{n} \phi(x_i|\mu_{z_i}, \sigma^2_{z_i}) \pi_{z_i}
\end{aligned}
$$

If we know $\mathbf{z}$, maximization is almost trivial. Let $n_j = \sum_{i=1}^{n} I\{z_i = j\}$.

$$
\hat{\mu}_j = \frac{1}{n_j} \Sigma_{i:z_i=j} x_i, \;\; \hat{\sigma}^2_j = \frac{1}{n_j} \sum_{i:z_i=j} (x_i - \hat{\mu}_j)^2, \;\; \hat{\pi}_j = n_j/n.
$$

Cross your fingers...

Need $E_{\mathbf{z}|\mathbf{x},\boldsymbol{\theta}^t}\{\log L(\boldsymbol{\theta}|\mathbf{z},\mathbf{x})\}$. Bayes' rule and conditional independence gives

$$
\begin{aligned}
P(z_i = j|\mathbf{x},\boldsymbol{\theta}) &= P(z_i = j|x_i,\boldsymbol{\theta}) \\
&= \frac{f(x_i|z_i = j,\boldsymbol{\theta})P(z_i = j|\boldsymbol{\theta})}{f(x_i|\boldsymbol{\theta})} \\
&= \frac{\phi(x_i|\mu_j,\sigma_j^2)\pi_j}{\sum_{k=1}^{J}\phi(x_i|\mu_k,\sigma_k^2)\pi_k} \equiv w_{ij}
\end{aligned}
$$

Note that $w_{\bullet j} = w_{i1} + \cdots + w_{iJ} = 1$ and $w_{\bullet\bullet} = n$. Ignoring $\frac{1}{\sqrt{2\pi}}$,

$$
E_{\mathbf{z}|\mathbf{x},\boldsymbol{\theta}^t}\{\log L(\boldsymbol{\theta}|\mathbf{z},\mathbf{x})\} = E_{\mathbf{z}|\mathbf{x},\boldsymbol{\theta}^t}\left\{\sum_{i=1}^{n} -\frac{1}{2}\log\sigma_{z_i}^2 - \frac{1}{2\sigma_{z_i}^2}(x_i - \mu_{z_i})^2 - \log\pi_{z_i}\right\},
$$

where $\pi_J = 1 - \sum_{j=1}^{J-1}\pi_j$.

$E_{\mathbf{z}|\mathbf{x},\theta^t}\{\log L(\theta|\mathbf{z},\mathbf{x})\}$...

This expectation is

$$\sum_{j=1}^{J}\sum_{i=1}^{n} w_{ij}[-\tfrac{1}{2}\log\sigma_j^2 - \tfrac{1}{2\sigma_j^2}(x_i-\mu_j)^2 - \log\pi_j].$$

Taking the first derivative and setting equal to zero (board) gives

$$\hat{\mu}_j = \frac{\sum_{i=1}^n w_{ij}x_i}{\sum_{i=1}^n w_{ij}},\ \hat{\sigma}_j^2 = \frac{\sum_{i=1}^n w_{ij}(x_i-\hat{\mu}_j)^2}{\sum_{i=1}^n w_{ij}},\ \hat{\pi}_j = \tfrac{1}{n}\sum_{i=1} w_{ij}.$$

We got lucky!

Note: in the actual algorithm $w_{ij} = w_{ij}^t$ (depend on the last $\boldsymbol{\theta}^t$) and these solutions represent $\boldsymbol{\theta}^{t+1}$...

So algorithm is...

Initialize $\boldsymbol{\theta}^0$ (how?) and $t = 0$, then

1. Compute
$$w_{ij} = \frac{\phi(x_i|\mu_j^t, \sigma_j^{2t})\pi_j^t}{\sum_{k=1}^J \phi(x_i|\mu_k^t, \sigma_k^{2t})\pi_k^t},$$
2. Set $\hat{\mu}_j^{t+1} = \frac{\sum_{i=1}^n w_{ij} x_i}{\sum_{i=1}^n w_{ij}}$, $(\hat{\sigma}_j^2)^{t+1} = \frac{\sum_{i=1}^n w_{ij}(x_i - \hat{\mu}_j^{t+1})^2}{\sum_{i=1}^n w_{ij}}$, and $\hat{\pi}_j^{t+1} = \frac{1}{n}\sum_{i=1}^n w_{ij}$.

Repeat until $||\boldsymbol{\theta}^{t+1} - \boldsymbol{\theta}^t|| < \epsilon$.

Note this defines a $\mathbf{g}(\cdot)$ so that $\boldsymbol{\theta}^{t+1} = \mathbf{g}(\boldsymbol{\theta}^j)$.

Multivariate version is almost the same!

$$\mathbf{x}_1, \ldots, \mathbf{x}_n | \boldsymbol{\theta} \overset{iid}{\sim} f(\mathbf{x}) = \sum_{j=1}^{J} \pi_j \phi_p(\mathbf{x}|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j).$$

Initialize $\boldsymbol{\theta}^0$ and $t = 0$, then

1. Compute
$$w_{ij} = \frac{\phi_p(\mathbf{x}_i|\boldsymbol{\mu}_j^t, \boldsymbol{\Sigma}_j^t)\pi_j^t}{\sum_{k=1}^J \phi_p(\mathbf{x}_i|\boldsymbol{\mu}_k^t, \boldsymbol{\Sigma}_k^t)\pi_k^t},$$
2. Set $\hat{\boldsymbol{\mu}}_j^{t+1} = \frac{\sum_{i=1}^n w_{ij}\mathbf{x}_i}{\sum_{i=1}^n w_{ij}}$, $\hat{\boldsymbol{\Sigma}}_j^{t+1} = \frac{\sum_{i=1}^n w_{ij}(\mathbf{x}_i - \hat{\boldsymbol{\mu}}_j^{t+1})(\mathbf{x}_i - \hat{\boldsymbol{\mu}}_j^{t+1})'}{\sum_{i=1}^n w_{ij}}$, and $\hat{\pi}_j^{t+1} = \frac{1}{n}\sum_{i=1}^n w_{ij}$.

Repeat until $||\boldsymbol{\theta}^{t+1} - \boldsymbol{\theta}^t|| < \epsilon$. How many parameters?

Comments

- Need to choose starting values for $\boldsymbol{\theta}$...any thoughts? Lange (2010) sugggests k-means clustering to start the μ_j^0.
- How to pick J? Many people use AIC or BIC. $AIC = -\log L(\hat{\boldsymbol{\theta}}|\mathbf{x}) + 2(3J - 1)$ for univariate data.
- MLE not unique & multiple modes...be careful!
- I would recommend bootstrap to get SE's and/or CI's here.

Bootstrap in one slide

Here's the process; explanation of why it works will come later.
Repeat $t = 1, \ldots, T$ times:

1. Sample from a uniform distribution on the integers $\{1, \ldots, n\}$ with replacement to get indices $(i_1, \ldots, i_n)$.
2. Compute the parameter of interest, maybe just $\hat{\boldsymbol{\theta}}^{(t)}$, from bootstrap sample $\mathbf{x}_{i_1}, \ldots, \mathbf{x}_{i_n}$.

Treat $\hat{\theta}_k^{(1)}, \ldots, \hat{\theta}_k^{(T)}$ as a Monte Carlo sample from the sampling distribution of $\hat{\boldsymbol{\theta}}(\mathbf{x})$.

SE of, e.g. $\hat{\theta}_k$, is simply sample standard deviation of $\hat{\theta}_k^{(1)}, \ldots, \hat{\theta}_k^{(T)}$. Can get CI from percentiles of $\hat{\theta}_k^{(1)}, \ldots, \hat{\theta}_k^{(T)}$.

Idea is same for any function $\mathbf{g}(\boldsymbol{\theta})$.

Louis' method

Recall to estimate variability we need the inverse of $-\nabla^2 \log L(\boldsymbol{\theta}|\mathbf{x})$. A result due to Louis (1982) leads to

$$-\nabla^2 \log L(\boldsymbol{\theta}|\mathbf{x}) = -E_{\mathbf{z}|\mathbf{x},\boldsymbol{\theta}}\{\nabla^2 \log L(\boldsymbol{\theta}|\mathbf{x},\mathbf{z})\} - cov_{\mathbf{z}|\mathbf{x},\boldsymbol{\theta}}\{\nabla \log L(\boldsymbol{\theta}|\mathbf{x},\mathbf{z})\}.$$

This may be easier and/or more efficient than direct numerical differentiation of $\log L(\boldsymbol{\theta}|\mathbf{x})$ or the bootstrap. It also might not.

Another method is the "supplemental EM," or SEM algorithm. See Givens & Hoeting (2013).

Censored exponential data

Censored exponential data follow

$$t_1, \ldots, t_n \overset{iid}{\sim} \exp(\lambda) \text{ indep. } c_1, \ldots, c_n \overset{iid}{\sim} h(\cdot).$$

We see $y_i = \min\{t_i, c_i\}$ and $\delta_i = I\{t_i < c_i\}$. The observed data is $\mathbf{x} = \{(y_i, \delta_i)\}_{i=1}^n$. Missing data are $\mathbf{z} = \{t_i : \delta_i = 0\}$.

Missing data are the true survival times t_i for $\delta_i = 0$. When $\delta_i = 0$ all we know is that $t_i \sim \exp(\lambda)$ and $t_i > y_i$. Thus, for $\delta_i = 0$

$$t_i \sim f(t|t_i > y_i, \lambda) = \frac{\lambda e^{-\lambda t}}{e^{-\lambda y_i}} I\{t > y_i\}.$$

Augmented likelihood is

$$L(\lambda|\mathbf{x}, \mathbf{z}) = \lambda^n \exp\left(-\lambda \sum_{i:\delta_i=1} y_i + \sum_{i:\delta_i=0} t_i\right).$$

Expected log-likelihood...

Taking expectation w.r.t. $[\{t_i : \delta_i = 0\} | \{y_i : \delta_i = 1\}, \lambda^j]$ gives

$$n \log \lambda - \lambda \left[\sum_{i:\delta_i=1} y_i + \sum_{i:\delta_i=0} E(t_i | t_i > y_i, \lambda^t) \right].$$

Note that

$$E(t_i | t_i > y_i, \lambda^j) = \int_{y_i}^{\infty} t \frac{\lambda^j e^{-\lambda^j t}}{e^{-\lambda^j y_i}} = y_i + \frac{1}{\lambda^j}.$$

So expected log-likelihood is

$$n \log \lambda - \lambda \left[\sum_{i:\delta_i=1} t_i + \sum_{i:\delta_i=0} (y_i + \tfrac{1}{\lambda^j}) \right].$$

Thus

$$\lambda^{j+1} = n \left[\sum_{i:\delta_i=1} y_i + \sum_{i:\delta_i=0} (y_i + \tfrac{1}{\lambda^j}) \right]^{-1} = \left[\frac{1}{n} \sum_{i=1}^{n} \{ y_i + (1 - \delta_i)/\lambda^j \} \right]^{-1}.$$

Louis' method...

Need

$$\nabla^2 \log L(\boldsymbol{\theta}|\mathbf{x},\mathbf{z}) = -\frac{n}{\lambda^2}, \ \ \nabla \log L(\boldsymbol{\theta}|\mathbf{x},\mathbf{z}) = \frac{n}{\lambda} - \left(\sum_{i:\delta_i=1} y_i + \sum_{i:\delta_i=0} t_i \right)$$

Need to take expectation of first and variance of second w.r.t. $[\{t_i : \delta_i = 0\}|\{y_i : \delta_i = 1\}, \lambda] = [\{t_i : \delta_i = 0\}|\lambda]$. Since for $\delta_i = 1$ we have $var(y_i|y_i, \lambda) = 0$, Louis method gives

$$-\nabla^2 \log L(\lambda|\mathbf{x}) = -(-\tfrac{n}{\lambda^2}) - \left(\sum_{i:\delta_i=0} \frac{1}{\lambda^2} \right) = \tfrac{u}{\lambda^2},$$

where $u = \sum_{i=1}^{n} I\{\delta_i = 1\}$ is the number of uncensored observations.

Example: V.A. data in R.

Homework...

In your homework, you will derive the EM algorithm for censored normal data.

If $x \sim N(\mu, \sigma)$ restricted to $x > c$, what is $E(x)$ and $E(x^2)$? Start with $N(0,1)$:

$$\int_c^\infty x \tfrac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2} dx = \tfrac{1}{\sqrt{2\pi}} \int_c^\infty \frac{d}{dx}[-e^{-\frac{1}{2}x^2}] dx = \tfrac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}c^2} = \phi(c).$$

Where $\phi(\cdot)$ is the pdf and $\Phi(\cdot)$ is the cdf of a standard normal r.v. Note that the density of $x|x > c$ is

$$f(x|x > c) = \tfrac{\phi(x)}{P(x>c)} = \tfrac{\phi(x)}{1-\Phi(c)},$$

so $E(x|x > c) = \frac{\phi(c)}{1-\Phi(c)}$.

General normal

For $x \sim N(\mu, \sigma^2)$ make the change of variables $y = \frac{x-\mu}{\sigma}$, so $x = \sigma y + \mu$ and $dx = \sigma dy$.

$$\begin{aligned} \int_c^\infty x \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2\sigma^2}(x-\mu)^2} dx &= \int_{\frac{c-\mu}{\sigma}}^\infty (\sigma y + \mu) \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}y^2} dy \\ &= \sigma\phi(\tfrac{c-\mu}{\sigma}) + \mu[1 - \Phi(\tfrac{c-\mu}{\sigma})] \end{aligned}$$

So

$$E(x|x > c) = \mu + \sigma \frac{\phi(\frac{c-\mu}{\sigma})}{1 - \Phi(\frac{c-\mu}{\sigma})}.$$

In homework 3 you will show...

$$E(x^2|x > c) = \mu^2 + \sigma^2 + \sigma(c + \mu) \frac{\phi(\frac{c-\mu}{\sigma})}{1 - \Phi(\frac{c-\mu}{\sigma})}.$$