

13: Modal And Distributional Approximations

Taylor

University of Virginia

We mention:

- ① a few ways to find the posterior mode
- ② how to approximate a posterior using a mode
- ③ slightly more involved ways to approximate your posterior

Newton's Method aka the Newton-Raphson algorithm

Based on a first-order approximation of the first derivative of the log-likelihood.

Approximate $L'(\theta) = (\log p(\theta | y))'$ as

$$\mathbf{0} \stackrel{\text{set}}{=} L'(\theta + \delta\theta) \approx L'(\theta) + L''(\theta)(\delta\theta)$$

rearranges to

$$\delta\theta = -[L''(\theta)]^{-1}L'(\theta)$$

Newton's Method

Repeat the following iteration until convergence:

$$\theta^t = \theta^{t-1} - [L''(\theta^{t-1})]^{-1}L'(\theta^{t-1})$$

Newton's Method aka the Newton-Raphson algorithm

Newton's Method

Repeat the following iteration until convergence:

$$\theta^t = \theta^{t-1} - [L''(\theta^{t-1})]^{-1} L'(\theta^{t-1})$$

Notes:

- 1 easily handles unnormalized densities
- 2 starting value is important because it is not guaranteed to converge from everywhere
- 3 The derivatives can be determined analytically or numerically

Quasi-Newton and conjugate gradient methods

Notes:

- 1 Quasi-Newton methods (approximate second derivatives) are available when second derivatives are too costly or unavailable
- 2 "Broyden-Fletcher-Goldfarb-Shanno" is a common example of a Quasi-Newton method
- 3 in R: `optim(2.9,F,method="BFGS")`
- 4 conjugate-gradient methods only use gradient information, but they are for models of the form $\|A\theta - b\|_2$ (also handled by `optim()`)
- 5 compared with the two above, they generally require more iterations, but use less computation per iteration and less storage

Numerical computation of derivatives

In `optim`, if you don't provide a function to calculate the gradient, then it uses a finite-difference approximation:

$$L'_i(\theta) = \frac{dL}{d\theta_i} \approx \frac{L(\theta + \delta_i e_i) - L(\theta - \delta_i e_i)}{2\delta_i}$$

and

$$\begin{aligned} L''_{ij}(\theta) &= \frac{d^2 L}{d\theta_i d\theta_j} \\ &\approx \frac{L'_i(\theta + \delta_j e_j) - L'_i(\theta - \delta_j e_j)}{2\delta_j} \end{aligned}$$

where e_j is the vector of all zeros except for a 1 in the j th spot, and δ_j is a small number (`optim`'s default is $1e-3$)

Gaussian approximations

Once the mode or modes have been found (perhaps after including a boundary-avoiding prior distribution as discussed in section 13.2, or after transforming the parameters appropriately), we can construct an approximation based on the multivariate normal distribution.

Let $\hat{\theta}$ be the mode, then

$$p(\theta | y) \approx N(\hat{\theta}, V_{\theta})$$

where

$$V_{\theta} = \left[- \frac{d^2 \log p(\theta | y)}{d\theta^2} \Big|_{\theta=\hat{\theta}} \right]^{-1}$$

is calculated exactly or approximated using the formula from a few slides ago.

Example

From chapter 3:

$$\textcircled{1} \quad p(y_i \mid \mu, \sigma^2) = \text{Normal}(\mu, \sigma^2)$$

$$\textcircled{2} \quad p(\mu, \sigma^2) \propto 1/\sigma^2$$

$$p(\mu, \sigma^2 \mid y) \propto (\sigma^2)^{-(n+2)/2} \exp \left[-\frac{1}{2\sigma^2} \{ (n-1)s^2 + n(\bar{y} - \mu)^2 \} \right]$$

Example

$$p(\mu, \sigma^2 \mid y) \propto (\sigma^2)^{-(n+2)/2} \exp \left[-\frac{1}{2\sigma^2} \{ (n-1)s^2 + n(\bar{y} - \mu)^2 \} \right]$$

Letting $\theta = \log \sigma$, $p(\mu, \theta \mid y)$ is proportional to

$$\exp[-n\theta] \exp \left[-\frac{1}{2 \exp[2\theta]} \{ (n-1)s^2 + n(\bar{y} - \mu)^2 \} \right]$$

So $\log p(\mu, \theta \mid y)$ is

$$\text{constant} - n\theta - .5 \exp(-2\theta) [(n-1)s^2 + n(\bar{y} - \mu)^2]$$

$$\text{and } L'(\theta) = \begin{bmatrix} \exp(-2\theta)(\bar{y} - \mu)n \\ -n + \exp(-2\theta) [(n-1)s^2 + n(\bar{y} - \mu)^2] \end{bmatrix}$$

Example

Warning: `optim` *minimizes*, so we use $-\log p(\mu, \theta \mid y)$

$$n\theta + .5 \exp(-2\theta) [(n-1)s^2 + n(\bar{y} - \mu)^2]$$

and

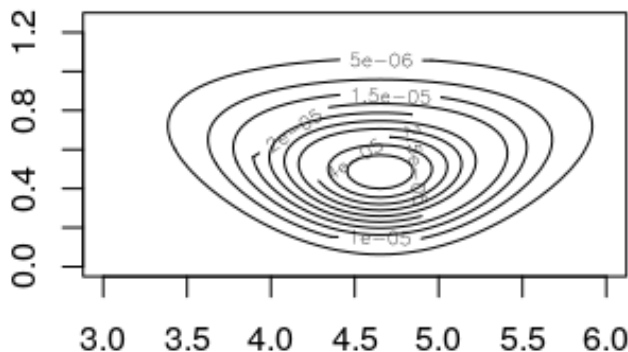
$$L'(\theta) = \begin{bmatrix} -\exp(-2\theta)(\bar{y} - \mu)n \\ n - \exp(-2\theta) [(n-1)s^2 + n(\bar{y} - \mu)^2] \end{bmatrix}$$

if `gr` left blank, finite difference approx. used

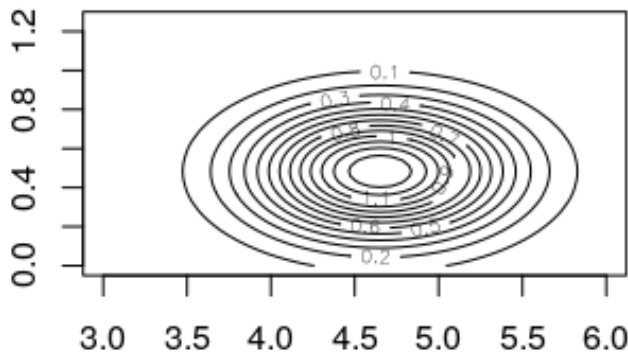
```
optim_results <- optim(par = c(5, 0),  
                      fn = neg_log_unnorm_post,  
                      gr = gradient,  
                      method = "BFGS",  
                      hessian = T)
```

See `mode_finding_examples.r`

Unnormalized true $p(\mu, \theta | y)$



Normal approx. $p(\mu, \theta \mid y)$



Gaussian approximations: Laplace's Method

If you want approximations to posterior *expectations* (say $E[h(\theta) | y]$), then you might consider Laplace's method, which is based on second-order Taylor approximations of the functions:

$$\textcircled{1} \quad u_1(\theta) = \log[h(\theta)q(\theta | y)]$$

$$\textcircled{2} \quad u_2(\theta) = \log q(\theta | y)$$

where $p(\theta | y) = q(\theta | y) / \int q(\theta | y) d\theta$.

Both are centered at maximizing values: θ_0^1, θ_0^2 , and this assumes h s are twice continuously differentiable.

Idea:

$$\frac{\int h(\theta)q(\theta | y)d\theta}{\int q(\theta | y)d\theta} = \frac{\int \exp[\log h(\theta) + \log q(\theta | y)] d\theta}{\int \exp[\log q(\theta | y)] d\theta}$$

Gaussian approximations: Laplace's Method

Exponentiating and integrating (typo on page 318?)

$$\begin{aligned}u(\theta) &\approx u(\theta_0) + (\theta - \theta_0)^T u'(\theta_0) + \frac{1}{2}(\theta - \theta_0)^T u''(\theta_0)(\theta - \theta_0) \\&= u(\theta_0) + \frac{1}{2}(\theta - \theta_0)^T u''(\theta_0)(\theta - \theta_0)\end{aligned}$$

gives us

$$\begin{aligned}\int \exp[u(\theta)] d\theta &\approx \int \exp[u(\theta_0) + \frac{1}{2}(\theta - \theta_0)^T u''(\theta_0)(\theta - \theta_0)] d\theta \\&= \exp[u(\theta_0)] \int \exp\left[\frac{1}{2}(\theta - \theta_0)^T u''(\theta_0)(\theta - \theta_0)\right] d\theta \\&= \exp[u(\theta_0)] \int \exp\left[-\frac{1}{2}(\theta - \theta_0)^T \{-u''(\theta_0)\}(\theta - \theta_0)\right] d\theta \\&= \exp[u(\theta_0)] (2\pi)^{d/2} \det[-u''(\theta_0)]^{-1/2}\end{aligned}$$

The book has a few more generalizations that we don't address:

- ① approximating multimodal distributions with normal mixtures
- ② approximating multimodal distributions with t mixtures

The EM Algorithm

The **expectation-maximization algorithm** finds the argument that maximizes the marginal posterior. It's useful in situations where there is missing data in a model (e.g. factor models, hidden markov models, state space models, etc.).

It follows the following steps

- 1 replace missing values by their expectations given the guessed parameters,
- 2 estimate parameters assuming the missing data are equal to their estimated values,
- 3 re-estimate the missing values assuming the new parameter estimates are correct,
- 4 re-estimate parameters,

and so forth, iterating until convergence.

The EM Algorithm

Call $\theta = (\gamma, \phi)$. You're interested in the mode of $p(\phi | y)$. Typically, γ is "hidden data."

$$\log p(\phi | y) = \log \frac{p(\gamma, \phi | y)}{p(\gamma | \phi, y)} = \log \underbrace{p(\gamma, \phi | y)}_{\text{joint posterior}} - \log \underbrace{p(\gamma | \phi, y)}_{\text{conditional posterior}}$$

The EM Algorithm

Call $\theta = (\gamma, \phi)$. You're interested in the mode of $p(\phi | y)$. Typically, γ is "hidden data."

$$\log p(\phi | y) = \log \frac{p(\gamma, \phi | y)}{p(\gamma | \phi, y)} = \log \underbrace{p(\gamma, \phi | y)}_{\text{joint posterior}} - \log \underbrace{p(\gamma | \phi, y)}_{\text{conditional posterior}}$$

taking expectations on both sides with respect to $p(\gamma | \phi^{\text{old}}, y)$ yields:

$$\log p(\phi | y) = E \left[\log p(\gamma, \phi | y) | \phi^{\text{old}}, y \right] - E \left[\log p(\gamma | \phi, y) | \phi^{\text{old}}, y \right]$$

The EM Algorithm

We iteratively use the middle term in

$$\log p(\phi | y) = E [\log p(\gamma, \phi | y) | \phi^{\text{old}}, y] - E [\log p(\gamma | \phi, y) | \phi^{\text{old}}, y].$$

The Q quantity in the "E" step

$$Q(\phi | \phi^{\text{old}}) = E [\log p(\gamma, \phi | y) | \phi^{\text{old}}, y]$$

The EM Algorithm

We iteratively use the middle term in

$$\log p(\phi | y) = E [\log p(\gamma, \phi | y) | \phi^{\text{old}}, y] - E [\log p(\gamma | \phi, y) | \phi^{\text{old}}, y].$$

The Q quantity in the "E" step

$$Q(\phi | \phi^{\text{old}}) = E [\log p(\gamma, \phi | y) | \phi^{\text{old}}, y]$$

The EM algorithm

Repeat the following until convergence:

- 1 E-step: calculate $Q(\phi | \phi^{\text{old}})$
- 2 M-step: replace ϕ^{old} with $\arg \max Q(\phi | \phi^{\text{old}})$

The EM Algorithm

If we follow this strategy, $\log p(\phi \mid y)$ increases at every iteration:

$$\begin{aligned}\log p(\phi \mid y) &= E \left[\log p(\gamma, \phi \mid y) \mid \phi^{\text{old}}, y \right] - E \left[\log p(\gamma \mid \phi, y) \mid \phi^{\text{old}}, y \right] \\ &= Q(\phi \mid \phi^{\text{old}}) - E \left[\log p(\gamma \mid \phi, y) \mid \phi^{\text{old}}, y \right] \quad (\text{defn. } Q) \\ &\geq Q(\phi \mid \phi^{\text{old}}) - E \left[\log p(\gamma \mid \phi^{\text{old}}, y) \mid \phi^{\text{old}}, y \right] \quad (\text{HW})\end{aligned}$$

The EM Algorithm

If we follow this strategy, $\log p(\phi | y)$ increases at every iteration:

$$\begin{aligned}\log p(\phi | y) &= E \left[\log p(\gamma, \phi | y) | \phi^{\text{old}}, y \right] - E \left[\log p(\gamma | \phi, y) | \phi^{\text{old}}, y \right] \\ &= Q(\phi | \phi^{\text{old}}) - E \left[\log p(\gamma | \phi, y) | \phi^{\text{old}}, y \right] \quad (\text{defn. } Q) \\ &\geq Q(\phi | \phi^{\text{old}}) - E \left[\log p(\gamma | \phi^{\text{old}}, y) | \phi^{\text{old}}, y \right] \quad (\text{HW})\end{aligned}$$

So

$$\begin{aligned}\log p(\phi^{\text{new}} | y) - \log p(\phi^{\text{old}} | y) &= \log p(\phi^{\text{new}} | y) - \left\{ Q(\phi^{\text{old}} | \phi^{\text{old}}) - E \left[\log p(\gamma | \phi^{\text{old}}, y) | \phi^{\text{old}}, y \right] \right\} \\ &\geq Q(\phi^{\text{new}} | \phi^{\text{old}}) - E \left[\log p(\gamma | \phi^{\text{old}}, y) | \phi^{\text{old}}, y \right] \\ &\quad - \left\{ Q(\phi^{\text{old}} | \phi^{\text{old}}) - E \left[\log p(\gamma | \phi^{\text{old}}, y) | \phi^{\text{old}}, y \right] \right\} \\ &= Q(\phi^{\text{new}} | \phi^{\text{old}}) - Q(\phi^{\text{old}} | \phi^{\text{old}})\end{aligned}$$

The EM Algorithm

Notes:

- 1 The EM algo isn't inherently Bayesian. It can also be used to accomplish maximum likelihood estimation.
- 2 The expectation of $\log p(\gamma, \phi \mid y)$ is usually easy to compute because it is a sum, and might only depend on sufficient statistics
- 3 The EM algorithm implicitly deals with parameter constraints in the M-step
- 4 The EM algorithm is parameterization independent
- 5 The *Generalized* EM (GEM) just increases Q instead of maximizing it.
- 6 The book describes many generalizations, in addition to this one
- 7 You might find multiple modes if you start from multiple starting points (using mixture approximations afterwards)
- 8 if you can, debug by printing $\log p(\phi^i \mid y)$ at every iteration and make sure it increases monotonically

Variational inference approximates an intractable posterior $p(\theta \mid y)$ with some chosen distribution $g(\theta \mid \phi)$ (e.g. multivariate normal).

Variational inference approximates an intractable posterior $p(\theta \mid y)$ with some chosen distribution $g(\theta \mid \phi)$ (e.g. multivariate normal).

We will assume this approximating distribution factors into J components:

$$g(\theta \mid \phi) = \prod_{j=1}^J g_j(\theta_j \mid \phi_j) = g_j(\theta_j \mid \phi_j) g_{-j}(\theta_{-j} \mid \phi_{-j}).$$

We will find ϕ using an EM-like algorithm that minimizes Kullback-Leibler divergence.

Kullback-Leibler divergence is “reversed” this time:

$$\begin{aligned}KL(g||p) &= - \int \log \left(\frac{p(\theta | y)}{g(\theta | \phi)} \right) g(\theta | \phi) d\theta \\&= - \int \log \left(\frac{p(\theta, y)}{g(\theta | \phi)} \right) g(\theta | \phi) d\theta + \int \log p(y) g(\theta | \phi) d\theta \\&= - \underbrace{\int \log \left(\frac{p(\theta, y)}{g(\theta | \phi)} \right) g(\theta | \phi) d\theta}_{\text{variational lower bound}} + \log p(y)\end{aligned}$$

The term that we maximize (minimize the negative) is called the **variational lower bound** aka the **evidence lower bound** (ELBO).

Variational Inference

Every iteration, we cycle through all the hyper-parameters ϕ_1, \dots, ϕ_J , and change them until convergence is reached.

Variational Inference

Every iteration, we cycle through all the hyper-parameters ϕ_1, \dots, ϕ_J , and change them until convergence is reached.

Looking at ϕ_j ...

$$\begin{aligned} & \int \log \left(\frac{p(\theta, y)}{g(\theta | \phi)} \right) g(\theta | \phi) d\theta \\ &= \iint [\log p(\theta, y) - \log g_j(\theta_j | \phi_j) - \log g_{-j}(\theta_{-j} | \phi_{-j})] \\ & \quad g_j(\theta_j | \phi_j) g_{-j}(\theta_{-j} | \phi_{-j}) d\theta_j d\theta_{-j} \\ &= \int \left[\int \log p(\theta, y) g_{-j}(\theta_{-j} | \phi_{-j}) d\theta_{-j} \right] g_j(\theta_j | \phi_j) d\theta_j \\ & \quad - \int \log g_j(\theta_j | \phi_j) g_j(\theta_j | \phi_j) d\theta_j - \int \log g_{-j}(\theta_{-j} | \phi_{-j}) g_{-j}(\theta_{-j} | \phi_{-j}) d\theta_{-j} \\ &= \int \log \left(\frac{\tilde{p}(\theta_j)}{g_j(\theta_j | \phi_j)} \right) g_j(\theta_j | \phi_j) d\theta_j + \text{constant} \end{aligned} \quad (*)$$

Variational Inference

We think of $\tilde{p}(\theta_j)$ as an unnormalized density

$$\log \tilde{p}(\theta_j) = \int \log p(\theta, y) g_{-j}(\theta_{-j} \mid \phi_{-j}) d\theta_{-j}$$

because usually

$$\begin{aligned} \int \tilde{p}(\theta_j) d\theta_j &= \int \exp \left[\int \log p(\theta, y) g_{-j}(\theta_{-j} \mid \phi_{-j}) d\theta_{-j} \right] d\theta_j \\ &\leq \int \exp \left[\log \int p(\theta, y) g_{-j}(\theta_{-j} \mid \phi_{-j}) d\theta_{-j} \right] d\theta_j \quad (\text{Jensen's}) \\ &= \iint p(\theta, y) g_{-j}(\theta_{-j} \mid \phi_{-j}) d\theta_{-j} d\theta_j \\ &< \infty \end{aligned}$$

VI algorithm

For $j = 1, \dots, J$ set ϕ_j so that $\log g_j(\theta_j \mid \phi_j)$ is equal to

$$\log \tilde{p}(\theta_j) = \int \log p(\theta, y) g_{-j}(\theta_{-j} \mid \phi_{-j}) d\theta_{-j}$$

Variational Inference: educational testing example

When the parameters are $\alpha_1, \dots, \alpha_8, \mu, \tau$, the log posterior is

$$\log p(\theta \mid y) = \text{constant} - \frac{1}{2} \sum_{j=1}^8 \frac{(y_j - \alpha_j)^2}{\sigma_j^2} - 8 \log \tau - \frac{1}{2} \frac{1}{\tau^2} \sum_{j=1}^8 (\alpha_j - \mu)^2$$

and we assume

$$g(\alpha_1, \dots, \alpha_8, \mu, \tau) = g(\alpha_1) \times \dots \times g(\alpha_8) g(\mu) g(\tau).$$

Let's reparameterize τ as τ^2 and assume $g(\alpha_1), \dots, g(\alpha_8) g(\mu)$ are all normal distributions, and $g(\tau^2)$ is an Inverse-Gamma.

Variational Inference: example

$$\begin{aligned} & \log g(\alpha_j) \\ & \stackrel{\text{set}}{=} \log \tilde{p}(\alpha_j) \\ & = \int \log p(\theta, y) g_{-j}(\theta_{-j}) d\theta_{-j} \\ & = -\frac{1}{2} \sum_{i=1}^8 \frac{E_{-j}[(y_i - \alpha_i)^2]}{\sigma_i^2} - 8E_{-j}[\log \tau] - \frac{1}{2} E_{-j} \left[\frac{1}{\tau^2} \right] \sum_{i=1}^8 E[(\alpha_i - \mu)^2] + c \\ & = -\frac{1}{2} \frac{(y_j - \alpha_j)^2}{\sigma_j^2} - \frac{1}{2} E_{-j} \left[\frac{1}{\tau^2} \right] E_{-j}[(\alpha_j - \mu)^2] + c' \\ & = -\frac{1}{2} \frac{(y_j - \alpha_j)^2}{\sigma_j^2} - \frac{1}{2} E_{-j} \left[\frac{1}{\tau^2} \right] (\alpha_j^2 - 2\alpha_j E_{-j}[\mu]) + c'' \end{aligned}$$

We are using linearity, independence, the data aren't random, and we're grouping all the terms that don't involve α_j into the constant.

Variational Inference: example

For μ :

$$\begin{aligned}\log \tilde{p}(\mu) &= \int \log p(\theta, y) g_{-j}(\theta_{-j} \mid \phi_{-j}) d\theta_{-j} \\ &= -\frac{1}{2} E_{-\mu} \left[\frac{1}{\tau^2} \sum_{j=1}^8 (\alpha_j - \mu)^2 \right] + \text{constant} \\ &= -\frac{1}{2} E_{-\mu} \left[\frac{1}{\tau^2} \right] \sum_{j=1}^8 (\mu^2 - 2\mu E_{-\mu}[\alpha_j]) + \text{constant} \\ &= -\frac{1}{2} E_{-\mu} \left[\frac{1}{\tau^2} \right] \left(8\mu^2 - 2\mu \sum_{j=1}^8 E_{-\mu}[\alpha_j] \right) + \text{constant}\end{aligned}$$

So $g(\mu) = \dots$

Variational Inference: example

For τ (not τ^2):

$$\begin{aligned}\log \tilde{p}(\tau) &= \int \log p(\theta, y) g_{-j}(\theta_{-j} \mid \phi_{-j}) d\theta_{-j} \\ &= -8 \log \tau - \frac{1}{2} \frac{1}{\tau^2} E_{-\tau} \left[\sum_{j=1}^8 (\alpha_j - \mu)^2 \right] + c\end{aligned}$$

So $g(\tau) \propto \tau^{-8} \exp \left[-\frac{\sum_j E_{-\tau}[(\alpha_j - \mu)^2]}{2\tau^2} \right]$ which means

$$g(\tau^2) = (\tau^2)^{-(\frac{7}{2}+1)} \exp \left[-\frac{\sum_j E_{-\tau}[(\alpha_j - \mu)^2]}{2\tau^2} \right]$$

which is an InverseGamma $\left(\frac{7}{2}, \frac{\sum_j E_{-\tau}[(\alpha_j - \mu)^2]}{2} \right)$

Variational Inference: example

To complete this example, we need to derive:

- for $g(\alpha_j)$:

- ① $E_{-j} \left[\frac{1}{\tau^2} \right] = E_{\tau^2} \left[\frac{1}{\tau^2} \right],$

- ② $E_{-j}[\mu] = E_{\mu}[\mu]$

- for $g(\mu)$:

- ① $E_{-\mu}[\alpha_j] = E_{\alpha_j}[\alpha_j],$

- ② $E_{-j} \left[\frac{1}{\tau^2} \right] = E_{\tau^2} \left[\frac{1}{\tau^2} \right]$

- for $g(\tau^2)$:

- ① $\sum_j E_{-\tau}[(\alpha_j - \mu)^2] = \sum_j E_{\alpha_j, \mu}[(\alpha_j - \mu)^2],$

Variational Inference: example

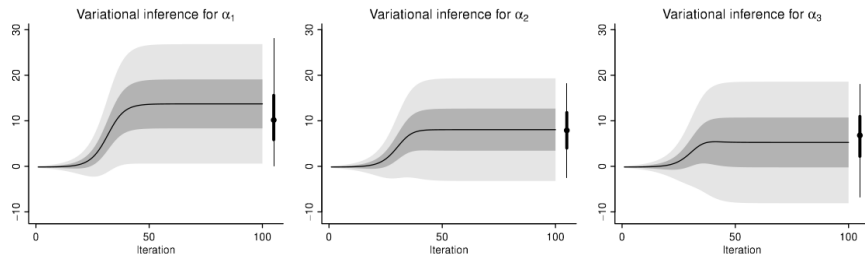


Figure 13.6 *Progress of inferences for the effects in schools A, B, and C, for 100 iterations of variational Bayes. The lines and shaded regions show the median, 50% interval, and 90% interval for the variational distribution. Shown to the right of each graph are the corresponding quantiles for the full Bayes inference as computed via simulation.*

Expectation Propagation: warmup

$p(x | \theta)$ is in the exponential family if it can be written as

$$h(x) \exp [\eta(\theta)' T(x) - A(\theta)]$$

Example:

$$N(\theta | \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp \left[-\frac{1}{2\sigma^2} \theta^2 + \frac{\mu}{\sigma^2} \theta - \frac{\mu^2}{2\sigma^2} \right]$$

sufficient statistic: (θ^2, θ)

canonical/natural parameters: $(-\frac{1}{2\sigma^2}, \frac{\mu}{\sigma^2})$

Expectation Propagation is another deterministic iterative technique that approximates the posterior with a distribution that is in the exponential family.

$$\textcircled{1} \quad p(\theta \mid y) = f(\theta) = \prod_{i=0}^n f_i(\theta)$$

$$\textcircled{2} \quad g(\theta) = \prod_{i=0}^n g_i(\theta)$$

$$f_0(\theta) = p(\theta), \quad f_1(\theta) = p(y_1 \mid \theta), \dots$$

For more info: <https://arxiv.org/abs/1412.4869>

Expectation Propagation

The **cavity distribution** is

$$g_{-i}(\theta) \propto g(\theta)/g_i(\theta),$$

and the **tilted distribution** is

$$g_{-i}(\theta)f_i(\theta).$$

At each stage, we update $g_i(\theta)$ so that we “target” $g_{-i}(\theta)f_i(\theta)$ with $g(\theta)$.

Expectation Propagation

At each stage, we update $g_i(\theta)$ so that we “target” $g_{-i}(\theta)f_i(\theta)$ with $g(\theta)$.

Notice that

$$\frac{\text{target}}{\text{“proposal”}} = \frac{g_{-i}(\theta)f_i(\theta)}{g(\theta)} = \frac{f_i(\theta)}{g_i(\theta)}.$$

However, we cannot ignore the cavity distribution in each “site” update. This is because we choose $g(\theta)$ so that its **moments match** those of $g_{-i}(\theta)f_i(\theta)$. This is like choosing $g_i(\theta)$ to approximate $f_i(\theta)$ **in the context of** $g_{-i}(\theta)$.

Expectation Propagation

If $g(\theta) = \text{Normal}(\mu, \Sigma)$, for each i we change μ and Σ by solving

$$\mu \stackrel{\text{set}}{=} E_{\text{tilted } i}[\theta]$$

and

$$\Sigma \stackrel{\text{set}}{=} \text{Var}_{\text{tilted } i}[\theta]$$

where $E_{\text{tilted } i}[\theta] = \int \theta g_{-i}(\theta) f_i(\theta) d\theta$ and
 $\text{Var}_{\text{tilted } i}[\theta] = \int (\theta - \mu)(\theta - \mu)' g_{-i}(\theta) f_i(\theta) d\theta$.

The hard part is integrating.

Expectation Propagation: example

Let θ be a vector of regression parameters for a logistic regression:

$$\begin{aligned} p(\theta \mid y) &\propto \prod_{i=1}^n p(y_i \mid \theta) p(\theta) \\ &= \prod_{i=0}^n f_i(\theta) \\ &= f_0(\theta) \prod_{i=1}^n [\text{invlogit}(X_i' \theta)]^{y_i} [1 - \text{invlogit}(X_i' \theta)]^{m_i - y_i} \end{aligned}$$

and choose $g(\theta)$ to be $\text{Normal}(\mu, \Sigma)$

Expectation Propagation: example

We choose $g(\theta)$ to be $\text{Normal}(\mu, \Sigma)$:

$$\begin{aligned} g(\theta) &\propto \prod_{i=0}^n \exp \left[-\frac{1}{2} (\theta - \mu_i)' \Sigma_i^{-1} (\theta - \mu_i) \right] \\ &\propto \exp \left[-\frac{1}{2} \sum_{i=0}^n (\theta' \Sigma_i^{-1} \theta - 2 \mu_i' \Sigma_i^{-1} \theta) \right] \\ &= \exp \left[-\frac{1}{2} \left(\theta' \underbrace{\left[\sum_{i=0}^n \Sigma_i^{-1} \right]}_{\Sigma^{-1}} \theta - 2 \underbrace{\left[\sum_{i=0}^n \mu_i' \Sigma_i^{-1} \right]}_{\text{natural param 2}} \theta \right) \right] \end{aligned}$$

Algorithmically, μ, Σ change at each iteration.

Expectation Propagation: example

$g(\theta)$ is Normal(μ, Σ):

$$g(\theta) \propto \exp \left[-\frac{1}{2} \left(\theta' \underbrace{\sum_{i=0}^n \Sigma_i^{-1}}_{\Sigma^{-1}} \theta - 2 \underbrace{\sum_{i=0}^n \mu_i' \Sigma_i^{-1}}_{\Sigma^{-1} \mu} \theta \right) \right]$$

Step 1: determine cavity distribution. $g_{-i}(\theta) = \text{Normal}(\mu_{-i}, \Sigma_{-i})$ where

$$\Sigma_{-i}^{-1} = \Sigma^{-1} - \Sigma_i^{-1}$$

and

$$\Sigma_{-i}^{-1} \mu_{-i} = \Sigma^{-1} \mu - \Sigma_i^{-1} \mu_i$$

Expectation Propagation: example

Step 2: find cavity distribution for $\eta = X_i' \theta$.

Because any linear transformation of normals is normal and because $g_{-i}(\theta) = \text{Normal}(\mu_{-i}, \Sigma_{-i})$:

$$g_{-i}(\eta) = \text{Normal}(M_{-i}, V_{-i})$$

where $M_{-i} = X_i' \mu_{-i}$ and $V_{-i} = X_i' \Sigma_{-i} X_i$.

Expectation Propagation: example

Step 3: define the unnormalized tilted distribution

$$g_{-i}(\eta)f_i(\eta) = g_{-i}(\eta)\text{Binomial}(m_i, \text{invlogit}(\eta)).$$

and find its expectations numerically with the Gauss-Kronrod quadrature method:

$$\begin{aligned} E_k &= \int_{-\infty}^{\infty} \eta^k g_{-i}(\eta) f_i(\eta) d\eta \\ &\approx \int_{M_{-i}-\delta\sqrt{V_{-i}}}^{M_{-i}+\delta\sqrt{V_{-i}}} \eta^k g_{-i}(\eta) f_i(\eta) d\eta \end{aligned}$$

for $k = 0, 1, 2$ and δ is some large number (e.g. 10). Finally compute $M = E_1/E_0$ and $V = E_2/E_0 - (E_1/E_0)^2$ and set $g(\eta) = \text{Normal}(M, V)$.

Expectation Propagation: example

In the previous steps we found $g(\eta) = \text{Normal}(M, V)$ and $g_{-i}(\eta) = \text{Normal}(M_{-i}, V_{-i})$.

Step 4: find $g_i(\eta) = \text{Normal}(M_i, V_i)$:

$$\begin{aligned} g_i(\eta) &= g(\eta) / g_{-i}(\eta) \\ &\propto \frac{\exp \left[-\frac{1}{2V} \eta^2 + \frac{M}{V} \eta \right]}{\exp \left[-\frac{1}{2V_{-i}} \eta^2 + \frac{M_{-i}}{V_{-i}} \eta \right]} \\ &= \exp \left[-\frac{1}{2} \underbrace{\left(\frac{1}{V} - \frac{1}{V_{-i}} \right)}_{\frac{1}{V_i}} \eta^2 + \underbrace{\left(\frac{M}{V} - \frac{M_{-i}}{V_{-i}} \right)}_{\frac{M_i}{V_i}} \eta \right] \end{aligned}$$

Expectation Propagation: example

Step 5: find $g_i(\theta)$.

$$g_i(\theta) = \text{Normal}(\mu_i, \Sigma_i)$$

where

$$\Sigma_i^{-1} \mu_i = X_i \frac{M_i}{V_i}$$

and

$$\Sigma_i^{-1} = X_i \frac{1}{V_i} X_i'$$

Expectation Propagation: example

Step 6: find $g(\theta) \propto g_i(\theta)g_{-i}(\theta)$.

$$g(\theta) \propto \exp \left[-\frac{1}{2} \left(\theta' \underbrace{\left[\sum_{i=0}^n \Sigma_i^{-1} \right]}_{\Sigma^{-1}} \theta - 2 \underbrace{\left[\sum_{i=0}^n \mu'_i \Sigma_i^{-1} \right]}_{\Sigma^{-1} \mu} \theta \right) \right]$$

$$\Sigma^{-1} \mu = \underbrace{\Sigma_{-i}^{-1} \mu_{-i}}_{\text{from step 1}} + \underbrace{\Sigma_i^{-1} \mu_i}_{\text{from step 5}}$$

and

$$\Sigma^{-1} = \underbrace{\Sigma_{-i}^{-1}}_{\text{from step 1}} + \underbrace{\Sigma_i^{-1}}_{\text{from step 5}}$$