14 Lecture 14, Feb 18

Announcements

- HW4 due next Tue @ 11:59PM.
- HW5 (Newton's method, handwritten digit recognition) posted. Due Tue Mar 1 @ 11:59PM. `http://hua-zhou.github.io/teaching/biostatm280-2016winter/biostat_m280_2016_hw5.pdf`

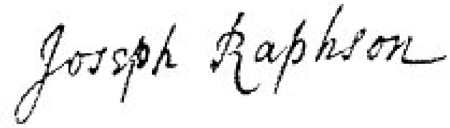
Last time

- Optimality conditions for unconstrained and constrained problems.
- Convexity.
- Newton-Raphson: introduction.

Today

- Newton-Raphson and Fisher scoring method.
- Fitting GLMs.
- Non-linear regression and Gauss-Newton algorithm.
- EM algorithm: introduction.

Newton's method and Fisher's scoring (KL Chapter 14)



Consider maximizing log-likelihood $L(\boldsymbol{\theta})$, $\boldsymbol{\theta} \in \boldsymbol{\Theta} \subset \mathbb{R}^p$.

- Newton's method was originally developed for finding roots of nonlinear equations $f(\boldsymbol{x}) = \mathbf{0}$ (KL 5.4).
- Newton's method (aka *Newton-Raphson method*) is considered the gold standard for its fast (quadratic) convergence

$$\frac{\|\boldsymbol{\theta}^{(t+1)} - \boldsymbol{\theta}^*\|}{\|\boldsymbol{\theta}^{(t)} - \boldsymbol{\theta}^*\|^2} \to \text{constant}.$$

- Idea: iterative quadratic approximation.
- Taylor expansion around the current iterate $\boldsymbol{\theta}^{(t)}$

$$L(\boldsymbol{\theta}) \approx L(\boldsymbol{\theta}^{(t)}) + dL(\boldsymbol{\theta}^{(t)})(\boldsymbol{\theta} - \boldsymbol{\theta}^{(t)}) + \frac{1}{2}(\boldsymbol{\theta} - \boldsymbol{\theta}^{(t)})^\intercal d^2L(\boldsymbol{\theta}^{(t)})(\boldsymbol{\theta} - \boldsymbol{\theta}^{(t)})$$

  and then maximize the quadratic approximation.

- To maximize the quadratic function, we equate its gradient to zero

$$\nabla L(\boldsymbol{\theta}^{(t)}) + [d^2L(\boldsymbol{\theta}^{(t)})](\boldsymbol{\theta} - \boldsymbol{\theta}^{(t)}) = \mathbf{0}_p,$$

  which suggests the next iterate

$$\begin{aligned}\boldsymbol{\theta}^{(t+1)} &= \boldsymbol{\theta}^{(t)} - [d^2L(\boldsymbol{\theta}^{(t)})]^{-1}\nabla L(\boldsymbol{\theta}^{(t)}) \\ &= \boldsymbol{\theta}^{(t)} + [-d^2L(\boldsymbol{\theta}^{(t)})]^{-1}\nabla L(\boldsymbol{\theta}^{(t)}).\end{aligned}$$

- Some issues with the Newton's iteration
  - Need to derive, evaluate, and "invert" the observed information matrix. In statistical problems, often evaluating Hessian costs $O(np^2)$ flops and inverting it costs $O(p^3)$ flops. Remedies:
    1. exploit structure in Hessian whenever possible,
    2. numerical differentiation (works for small problems), or
    3. quasi-Newton method (to be discussed later)
  - Stability: Newton's iterate is not guaranteed to be an ascent algorithm. It's equally happy to head uphill or downhill. Remedies:

1. approximate $-d^2L(\boldsymbol{\theta}^{(t)})$ by a positive definite $\boldsymbol{A}$ (if it's not), *and*
2. line search (backtracking).

Why insist on a *positive definite* approximation of Hessian? By first-order Taylor expansion,

$$
\begin{aligned}
& L(\boldsymbol{\theta}^{(t)} + s\Delta\boldsymbol{\theta}^{(t)}) - L(\boldsymbol{\theta}^{(t)}) \\
= \; & dL(\boldsymbol{\theta}^{(t)})s\Delta\boldsymbol{\theta}^{(t)} + o(s) \\
= \; & sdL(\boldsymbol{\theta}^{(t)})\boldsymbol{A}^{-(t)}\nabla L(\boldsymbol{\theta}^{(t)}) + o(s).
\end{aligned}
$$

For s sufficiently small, right hand side is strictly positive.

- In summary, *Newton type algorithm* iterates according to

$$
\boxed{\boldsymbol{\theta}^{(t+1)} = \boldsymbol{\theta}^{(t)} + s[\boldsymbol{A}^{(t)}]^{-1}\nabla L(\boldsymbol{\theta}^{(t)}) = \boldsymbol{\theta}^{(t)} + s\Delta\boldsymbol{\theta}^{(t)}}
$$

where $\boldsymbol{A}^{(t)}$ is a pd approximation of $-d^2L(\boldsymbol{\theta}^{(t)})$ and s is a step length.

- Line search strategy: step-halving ($s = 1, 1/2, \ldots$), golden section search, cubic interpolation, Amijo rule, ...

- How to approximating $-d^2L(\boldsymbol{\theta})$? More of an art than science. Often requires problem specific analysis.

- Taking $\boldsymbol{A} = \boldsymbol{I}$ leads to the method of *steepest ascent*, aka *gradient ascent*.

- *Fisher's scoring method*: replace $-d^2L(\boldsymbol{\theta})$ by the expected Fisher information matrix

$$
\boldsymbol{I}(\boldsymbol{\theta}) = \mathbf{E}[-d^2L(\boldsymbol{\theta})] = \mathbf{E}[\nabla L(\boldsymbol{\theta})dL(\boldsymbol{\theta})] \succeq \mathbf{0}_{p\times p},
$$

which is psd under exchangeability of expectation and differentiation.

Therefore the Fisher's scoring algorithm iterates according to

$$
\boxed{\boldsymbol{\theta}^{(t+1)} = \boldsymbol{\theta}^{(t)} + s[\boldsymbol{I}(\boldsymbol{\theta}^{(t)})]^{-1}\nabla L(\boldsymbol{\theta}^{(t)})}
$$

Generalized linear model (GLM) (KL 14.7)

Let's consider a concrete example: logistic regression.

- The goal is to predict whether a credit card transaction is fraud ($y_i = 1$) or not ($y_i = 0$). Predictors ($\boldsymbol{x}_i$) include: time of transaction, last location, merchant, ...

- $y_i \in \{0, 1\}$, $\boldsymbol{x}_i \in \mathbb{R}^p$. Model $y_i \sim$Bernoulli(p_i).

- Logistic regression. Density

$$\begin{aligned} f(y_i|p_i) &= p_i^{y_i}(1-p_i)^{1-y_i} \\ &= e^{y_i \ln p_i + (1-y_i)\ln(1-p_i)} \\ &= e^{y_i \ln \frac{p_i}{1-p_i} + \ln(1-p_i)}, \end{aligned}$$

where

$$\begin{aligned} E(y_i) = p_i &= \frac{e^{\boldsymbol{x}_i^\mathsf{T}\boldsymbol{\beta}}}{1+e^{\boldsymbol{x}_i^\mathsf{T}\boldsymbol{\beta}}} \quad \text{(mean function, inverse link function)} \\ \boldsymbol{x}_i^T\boldsymbol{\beta} &= \ln\left(\frac{p_i}{1-p_i}\right) \quad \text{(logit link function).} \end{aligned}$$

- Given data $(y_i, \boldsymbol{x}_i)$, $i = 1, \ldots, n$,

$$\begin{aligned} L_n(\boldsymbol{\beta}) &= \sum_{i=1}^n [y_i \ln p_i + (1-y_i)\ln(1-p_i)] \\ &= \sum_{i=1}^n \left[y_i\boldsymbol{x}_i^\mathsf{T}\boldsymbol{\beta} - \ln(1+e^{\boldsymbol{x}_i^\mathsf{T}\boldsymbol{\beta}})\right] \\ \nabla L_n(\boldsymbol{\beta}) &= \sum_{i=1}^n \left(y_i\boldsymbol{x}_i - \frac{e^{\boldsymbol{x}_i^\mathsf{T}\boldsymbol{\beta}}}{1+e^{\boldsymbol{x}_i^\mathsf{T}\boldsymbol{\beta}}}\boldsymbol{x}_i\right) \\ &= \sum_{i=1}^n (y_i - p_i)\boldsymbol{x}_i = \boldsymbol{X}^\mathsf{T}(\boldsymbol{y}-\boldsymbol{p}) \\ -d^2L_n(\boldsymbol{\beta}) &= \sum_{i=1}^n p_i(1-p_i)\boldsymbol{x}_i\boldsymbol{x}_i^\mathsf{T} = \boldsymbol{X}^\mathsf{T}\boldsymbol{W}\boldsymbol{X}, \\ \text{where} \quad & \boldsymbol{W} = \text{diag}(w_1, \ldots, w_n), w_i = p_i(1-p_i) \\ \boldsymbol{I}_n(\boldsymbol{\beta}) &= \mathbf{E}[-d^2L_n(\boldsymbol{\beta})] = -d^2L_n(\boldsymbol{\beta}). \end{aligned}$$

- Newton's method = Fisher's scoring iteration:

$$
\begin{aligned}
\boldsymbol{\beta}^{(t+1)} &= \boldsymbol{\beta}^{(t)} + s[-d^2 L(\boldsymbol{\beta}^{(t)})]^{-1} \nabla L(\boldsymbol{\beta}^{(t)}) \\
&= \boldsymbol{\beta}^{(t)} + s(\boldsymbol{X}^{\mathsf{T}} \boldsymbol{W}^{(t)} \boldsymbol{X})^{-1} \boldsymbol{X}^{\mathsf{T}} (\boldsymbol{y} - \boldsymbol{p}^{(t)}) \\
&= (\boldsymbol{X}^{\mathsf{T}} \boldsymbol{W}^{(t)} \boldsymbol{X})^{-1} \boldsymbol{X}^{\mathsf{T}} \boldsymbol{W}^{(t)} \left[\boldsymbol{X}\boldsymbol{\beta}^{(t)} + s(\boldsymbol{W}^{(t)})^{-1} (\boldsymbol{y} - \boldsymbol{p}^{(t)}) \right] \\
&= (\boldsymbol{X}^{\mathsf{T}} \boldsymbol{W}^{(t)} \boldsymbol{X})^{-1} \boldsymbol{X}^{\mathsf{T}} \boldsymbol{W}^{(t)} \boldsymbol{z}^{(t)},
\end{aligned}
$$

where

$$
\boldsymbol{z}^{(t)} = \boldsymbol{X}\boldsymbol{\beta}^{(t)} + s(\boldsymbol{W}^{(t)})^{-1}(\boldsymbol{y} - \boldsymbol{p}^{(t)})
$$

are the working responses. A Newton's iteration is equivalent to solving a weighed least squares problem $\sum_{i=1}^n w_i(z_i - \boldsymbol{x}_i^{\mathsf{T}}\boldsymbol{\beta})^2$. Thus the name IRWLS (iteratively re-weighted least squares).

Common distributions with typical uses and canonical link functions

Distribution	Support of distribution	Typical uses	Link name	Link function	Mean function
Normal	real: $(-\infty, +\infty)$	Linear-response data	Identity	$\mathbf{X}\boldsymbol{\beta} = \mu$	$\mu = \mathbf{X}\boldsymbol{\beta}$
Exponential	real: $(0, +\infty)$	Exponential-response data, scale parameters	Inverse	$\mathbf{X}\boldsymbol{\beta} = \mu^{-1}$	$\mu = (\mathbf{X}\boldsymbol{\beta})^{-1}$
Gamma					
Inverse Gaussian	real: $(0, +\infty)$		Inverse squared	$\mathbf{X}\boldsymbol{\beta} = \mu^{-2}$	$\mu = (\mathbf{X}\boldsymbol{\beta})^{-1/2}$
Poisson	integer: $[0, +\infty)$	count of occurrences in fixed amount of time/space	Log	$\mathbf{X}\boldsymbol{\beta} = \ln(\mu)$	$\mu = \exp(\mathbf{X}\boldsymbol{\beta})$
Bernoulli	integer: $[0, 1]$	outcome of single yes/no occurrence	Logit	$\mathbf{X}\boldsymbol{\beta} = \ln\left(\dfrac{\mu}{1-\mu}\right)$	$\mu = \dfrac{\exp(\mathbf{X}\boldsymbol{\beta})}{1+\exp(\mathbf{X}\boldsymbol{\beta})} = \dfrac{1}{1+\exp(-\mathbf{X}\boldsymbol{\beta})}$
Binomial	integer: $[0, N]$	count of # of "yes" occurrences out of N yes/no occurrences			
Categorical	integer: $[0, K)$	outcome of single K-way occurrence			
	K-vector of integer: $[0, 1]$, where exactly one element in the vector has the value 1				
Multinomial	K-vector of integer: $[0, N]$	count of occurrences of different types (1 .. K) out of N total K-way occurrences			

Let's consider the more general class of generalized linear models (GLM).

- Y belongs to an exponential family with density

$$
p(y|\theta, \phi) = \exp\left\{ \frac{y\theta - b(\theta)}{a(\phi)} + c(y, \phi) \right\}.
$$

θ: natural parameter. $\phi > 0$: dispersion parameter. GLM relates the mean $\mu = \mathbf{E}(Y|\boldsymbol{x})$ via a strictly increasing link function

$$g(\mu) = \eta = \boldsymbol{x}^{\mathsf{T}}\boldsymbol{\beta}, \quad \mu = g^{-1}(\eta)$$

- Score, Hessian, information

$$\begin{aligned}
\nabla L_n(\boldsymbol{\beta}) &= \sum_{i=1}^{n} \frac{(y_i - \mu_i)\mu_i'(\eta_i)}{\sigma_i^2}\boldsymbol{x}_i \\
-d^2 L_n(\boldsymbol{\beta}) &= \sum_{i=1}^{n} \frac{[\mu_i'(\eta_i)]^2}{\sigma_i^2}\boldsymbol{x}_i\boldsymbol{x}_i^{\mathsf{T}} - \sum_{i=1}^{n} \frac{(y_i - \mu_i)\theta''(\eta_i)}{\sigma_i^2}\boldsymbol{x}_i\boldsymbol{x}_i^{\mathsf{T}} \\
\boldsymbol{I}_n(\boldsymbol{\beta}) &= \mathbf{E}[-d^2 L_n(\boldsymbol{\beta})] = \sum_{i=1}^{n} \frac{[\mu_i'(\eta_i)]^2}{\sigma_i^2}\boldsymbol{x}_i\boldsymbol{x}_i^{\mathsf{T}} = \boldsymbol{X}^T\boldsymbol{W}\boldsymbol{X}.
\end{aligned}$$

- Fisher scoring method

$$\boldsymbol{\beta}^{(t+1)} = \boldsymbol{\beta}^{(t)} + s[\boldsymbol{I}(\boldsymbol{\beta}^{(t)})]^{-1}\nabla L_n(\boldsymbol{\beta}^{(t)})$$

IRWLS with weights $w_i = [\mu_i(\eta_i)]^2/\sigma_i^2$ and some working responses z_i.

- For *canonical link*, $\theta = \eta$, the second term of Hessian vanishes and Hessian coincides with Fisher information matrix. Convex problem ☺

$$\text{Fisher's scoring} = \text{Newton's method.}$$

- Non-canonical link, non-convex problem ☹

$$\text{Fisher's scoring algorithm} \neq \text{Newton's method.}$$

Example: Probit regression (binary response with probit link). $y_i \sim \text{Bernoulli}(p_i)$ and

$$p_i = \Phi(\boldsymbol{x}_i^T\boldsymbol{\beta}), \quad \eta_i = \boldsymbol{x}_i^T\boldsymbol{\beta} = \Phi^{-1}(p_i),$$

where $\Phi(\cdot)$ is the cdf of a standard normal.

- `glmfit()` in R and MATLAB implements the Fisher scoring method, aka IRWLS, for GLMs.

Nonlinear regression – Gauss-Newton method (KL 14.4-14.6)

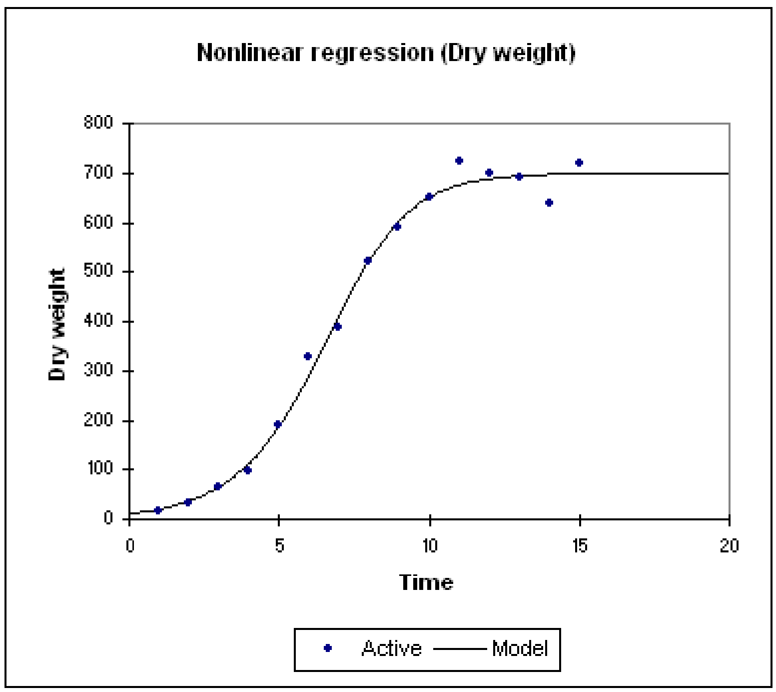
- Now we finally get to the problem Gauss faced in 1800!
  Relocate Ceres by fitting 41 observations to a 6-parameter (nonlinear) orbit.

- Nonlinear least squares (curve fitting):

$$\text{minimize } f(\boldsymbol{\beta}) = \frac{1}{2}\sum_{i=1}^{n}[y_i - \mu_i(\boldsymbol{x}_i, \boldsymbol{\beta})]^2$$

  For example, y_i = dry weight of onion and x_i = growth time, and we want to fit a 3-parameter growth curve

$$\mu(x, \beta_1, \beta_2, \beta_3) = \frac{\beta_3}{1 + e^{-\beta_1 - \beta_2 x}}.$$



- "Score" and "information matrices"

$$\begin{aligned}
\nabla f(\boldsymbol{\beta}) &= -\sum_{i=1}^{n}[y_i - \mu_i(\boldsymbol{\beta})]\nabla\mu_i(\boldsymbol{\beta}) \\
d^2 f(\boldsymbol{\beta}) &= \sum_{i=1}^{n}\nabla\mu_i(\boldsymbol{\beta})d\mu_i(\boldsymbol{\beta}) - \sum_{i=1}^{n}[y_i - \mu_i(\boldsymbol{\beta})]d^2\mu_i(\boldsymbol{\beta}) \\
\boldsymbol{I}(\boldsymbol{\beta}) &= \sum_{i=1}^{n}\nabla\mu_i(\boldsymbol{\beta})d\mu_i(\boldsymbol{\beta}) = \boldsymbol{J}(\boldsymbol{\beta})^{\mathsf{T}}\boldsymbol{J}(\boldsymbol{\beta}),
\end{aligned}$$

where $\boldsymbol{J}(\boldsymbol{\beta})^{\intercal} = [\nabla \mu_1(\boldsymbol{\beta}), \ldots, \nabla \mu_n(\boldsymbol{\beta})] \in \mathbb{R}^{p \times n}$.

- *Gauss-Newton* (= "Fisher's scoring algorithm") uses $\boldsymbol{I}(\boldsymbol{\beta})$, which is always psd.

$$\boxed{\boldsymbol{\beta}^{(t+1)} = \boldsymbol{\beta}^{(t)} + s\boldsymbol{I}(\boldsymbol{\beta}^{(t)})^{-1} \nabla L(\boldsymbol{\beta}^{(t)})}$$

- *Levenberg-Marquardt* method, aka *damped least squares algorithm* (DLS), adds a ridge term to the approximate Hessian

$$\boxed{\boldsymbol{\beta}^{(t+1)} = \boldsymbol{\beta}^{(t)} + s[\boldsymbol{I}(\boldsymbol{\beta}^{(t)}) + \tau \boldsymbol{I}_p]^{-1} \nabla L(\boldsymbol{\beta}^{(t)})}$$

  bridging between Gauss-Newton and steepest descent.

- Other approximation to Hessians: nonlinear GLMs.
  See KL 14.4 for examples.

Which statistical papers are most cited?

Paper	Citations	Per Year
Kaplan-Meier (Kaplan and Meier, 1958)	46886	808
EM (Dempster et al., 1977a)	44050	**1129**
Cox model (Cox, 1972)	40920	930
Metropolis (Metropolis et al., 1953)	31284	497
FDR (Benjamini and Hochberg, 1995)	30975	**1450**
Unit root test (Dickey and Fuller, 1979)	18259	493
Lasso (Tibshirani, 1996)	15306	765
bootstrap (Efron, 1979)	12992	351
FFT (Cooley and Tukey, 1965)	11319	222
Gibbs sampler (Gelfand and Smith, 1990)	6531	251

- Citation counts from Google Scholar on Feb 17, 2016.

- EM is one of the most influential statistical ideas, finding applications in various branches of science.

EM algorithm

- History: Dempster et al. (1977b).

  [PDF] Maximum likelihood from incomplete data via the EM algorithm
  AP Dempster, NM Laird, DB Rubin - Journal of the Royal Statistical Society. ..., 1977 - JSTOR
  A broadly applicable **algorithm** for computing **maximum likelihood** estimates from **incomplete data** is presented at various levels of generality. Theory showing the monotone behaviour of the **likelihood** and convergence of the **algorithm** is derived. Many examples are sketched, **...**
  Cited by 39167 Related articles All 76 versions Web of Science: 16067 Cite Save More

  Same idea appears in parameter estimation in HMM (Baum-Welch algorithm) (Baum et al., 1970).

  A maximization technique occurring in the **statistical** analysis of probabilistic functions of Markov chains
  LE Baum, T Petrie, G Soules, N Weiss - The annals of mathematical **statistics**, 1970 - JSTOR
  PYI... YT (**A**, **a**, f) and the difficult analysis of **maximizing** this function of **A** for very special choices of f presented in [2],[8] that **a** simple explicit procedure for **maximization** for **a** general f would be quite difficult; however, this is not the case.
  Cited by 3102 Related articles All 4 versions Cite

- Notations
  - $\boldsymbol{Y}$: observed data
  - $\boldsymbol{Z}$: missing data
  - $\boldsymbol{X} = (\boldsymbol{Y}, \boldsymbol{Z})$: complete data
- Goal: maximize the log-likelihood of the observed data $\ln g(\boldsymbol{y}|\boldsymbol{\theta})$ (optimization!)
- Idea: choose $\boldsymbol{Z}$ such that MLE for the complete data is trivial.
- Let $f(\boldsymbol{x}|\boldsymbol{\theta}) = f(\boldsymbol{y}, \boldsymbol{z}|\boldsymbol{\theta})$ be the density of complete data
- Iterative two step procedure
  - E step: calculate the conditional expectation

$$Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) = \mathbf{E}_{\boldsymbol{Z}|\boldsymbol{Y}=\boldsymbol{y},\boldsymbol{\theta}^{(t)}}[\ln f(\boldsymbol{Y}, \boldsymbol{Z}|\boldsymbol{\theta}) \mid \boldsymbol{Y} = \boldsymbol{y}, \boldsymbol{\theta}^{(t)}]$$

  - M step: maximize $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})$ to generate the next iterate

$$\boldsymbol{\theta}^{(t+1)} = \text{argmax}_{\boldsymbol{\theta}}\, Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})$$

- (Ascent property of EM algorithm) By the information inequality,

$$\begin{aligned}
& Q(\boldsymbol{\theta} \mid \boldsymbol{\theta}^{(t)}) - \ln g(\boldsymbol{y}|\boldsymbol{\theta}) \\
= \; & \mathbf{E}[\ln f(\boldsymbol{Y}, \boldsymbol{Z}|\boldsymbol{\theta})|\boldsymbol{Y} = \boldsymbol{y}, \boldsymbol{\theta}^{(t)}] - \ln g(\boldsymbol{y}|\boldsymbol{\theta}) \\
= \; & \mathbf{E}\left\{ \ln \left[\frac{f(\boldsymbol{Y}, \boldsymbol{Z} \mid \boldsymbol{\theta})}{g(\boldsymbol{Y} \mid \boldsymbol{\theta})} \right] \mid \boldsymbol{Y} = \boldsymbol{y}, \boldsymbol{\theta}^{(t)} \right\} \\
\le \; & \mathbf{E}\left\{ \ln \left[\frac{f(\boldsymbol{Y}, \boldsymbol{Z} \mid \boldsymbol{\theta}^{(t)})}{g(\boldsymbol{Y} \mid \boldsymbol{\theta}^{(t)})} \right] \mid \boldsymbol{Y} = \boldsymbol{y}, \boldsymbol{\theta}^{(t)} \right\} \\
= \; & Q(\boldsymbol{\theta}^{(t)} \mid \boldsymbol{\theta}^{(t)}) - \ln g(\boldsymbol{y}|\boldsymbol{\theta}^{(t)}).
\end{aligned}$$

  Rearranging shows that (fundamental inequality of EM)

$$\ln g(\boldsymbol{y} \mid \boldsymbol{\theta}) \ge Q(\boldsymbol{\theta} \mid \boldsymbol{\theta}^{(t)}) - Q(\boldsymbol{\theta}^{(t)} \mid \boldsymbol{\theta}^{(t)}) + \ln g(\boldsymbol{y} \mid \boldsymbol{\theta}^{(t)})$$

  for all $\boldsymbol{\theta}$ and in particular

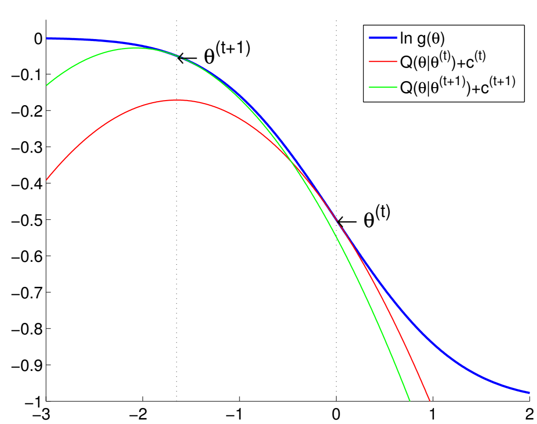
$$\begin{aligned}
\ln g(\boldsymbol{y} \mid \boldsymbol{\theta}^{(t+1)}) & \ge Q(\boldsymbol{\theta}^{(t+1)} \mid \boldsymbol{\theta}^{(t)}) - Q(\boldsymbol{\theta}^{(t)} \mid \boldsymbol{\theta}^{(t)}) + \ln g(\boldsymbol{y} \mid \boldsymbol{\theta}^{(t)}) \\
& \ge \ln g(\boldsymbol{y} \mid \boldsymbol{\theta}^{(t)}).
\end{aligned}$$

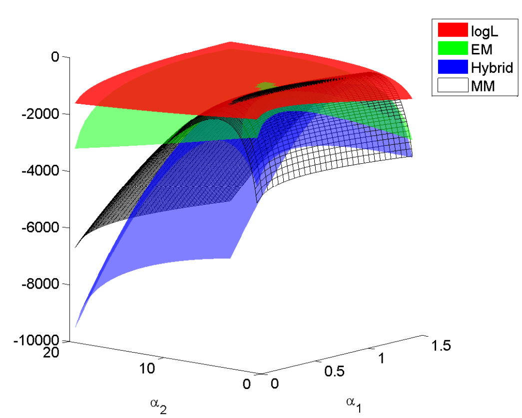
  Obviously we only need

$$Q(\boldsymbol{\theta}^{(t+1)} \mid \boldsymbol{\theta}^{(t)}) - Q(\boldsymbol{\theta}^{(t)} \mid \boldsymbol{\theta}^{(t)}) \ge 0$$

  for this ascent property to hold (*generalized EM*).

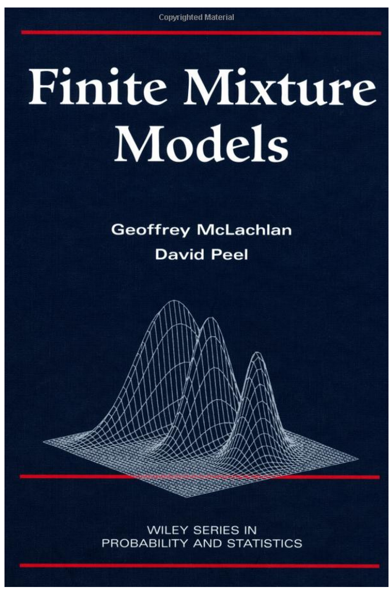
- Intuition? Keep these pictures in mind

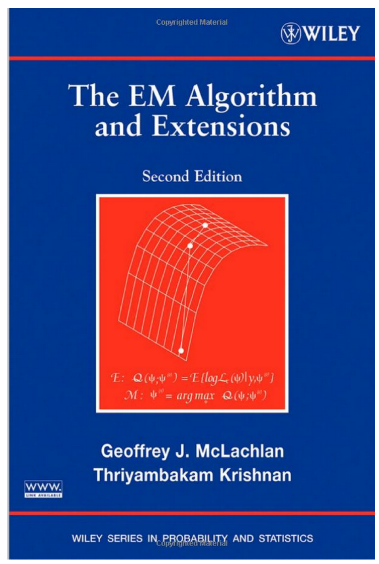




- Under mild regularity conditions, $\boldsymbol{\theta}^{(t)}$ converges to a stationary point of $\ln g(\boldsymbol{y}|\boldsymbol{\theta})$.
- Numerous applications of EM:
  finite mixture model, HMM (Baum-Welch algorithm), factor analysis, variance components model aka linear mixed model (LMM), hyper-parameter estimation in empirical Bayes procedure $\max_{\boldsymbol{\alpha}} \int f(\boldsymbol{y}|\boldsymbol{\theta})\pi(\boldsymbol{\theta}|\boldsymbol{\alpha})\, d\boldsymbol{\theta}$ (e.g., HW6/7), missing data, group/censorized/truncated model, ...

A canonical example: finite mixture models





- Gaussian finite mixture models: mixture density

$$h(\boldsymbol{y}) = \sum_{j=1}^{k} \pi_j h_j(\boldsymbol{y} \mid \boldsymbol{\mu}_j, \boldsymbol{\Omega}_j), \quad \boldsymbol{y} \in \mathbb{R}^d,$$

  where

$$h_j(\boldsymbol{y} \mid \boldsymbol{\mu}_j, \boldsymbol{\Omega}_j) = \left(\frac{1}{2\pi}\right)^{d/2} |\det(\boldsymbol{\Omega}_j)|^{-1/2} e^{-\frac{1}{2}(\boldsymbol{y}-\boldsymbol{\mu}_j)^\intercal \boldsymbol{\Omega}_j^{-1}(\boldsymbol{y}-\boldsymbol{\mu}_j)}$$

  are multivariate normals $N_d(\boldsymbol{\mu}_j, \boldsymbol{\Omega}_j)$.

- Given data $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n$, want to estimate parameters

$$\boldsymbol{\theta} = (\pi_1, \ldots, \pi_k, \boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_k, \boldsymbol{\Omega}_1, \ldots, \boldsymbol{\Omega}_k).$$

(Incomplete) data log-likelihood is

$$\ln g(\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n|\boldsymbol{\theta}) = \sum_{i=1}^n \ln h(\boldsymbol{y}_i) = \sum_{i=1}^n \ln \sum_{j=1}^k \pi_j h_j(\boldsymbol{y}_i \mid \boldsymbol{\mu}_j, \boldsymbol{\Omega}_j).$$

- Let $z_{ij} = I\{\boldsymbol{y}_i \text{ comes from group } j\}$. Complete data likelihood is

$$f(\boldsymbol{y}, \boldsymbol{z}|\boldsymbol{\theta}) = \prod_{i=1}^n \prod_{j=1}^k [\pi_j h_j(\boldsymbol{y}_i|\boldsymbol{\mu}_j, \boldsymbol{\Omega}_j)]^{z_{ij}}$$

and thus complete log-likelihood is

$$\ln f(\boldsymbol{y}, \boldsymbol{z}|\boldsymbol{\theta}) = \sum_{i=1}^n \sum_{j=1}^k z_{ij}[\ln \pi_j + \ln h_j(\boldsymbol{y}_i|\boldsymbol{\mu}_j, \boldsymbol{\Omega}_j)].$$

- E step: need to evaluate conditional expectation

$$\begin{aligned} & Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) \\ = \;& \mathbf{E}\left\{\sum_{i=1}^n \sum_{j=1}^k z_{ij}[\ln \pi_j + \ln h_j(\boldsymbol{y}_i|\boldsymbol{\mu}_j, \boldsymbol{\Omega}_j) \mid \boldsymbol{Y} = \boldsymbol{y}, \boldsymbol{\pi}^{(t)}, \boldsymbol{\mu}_1^{(t)}, \ldots, \boldsymbol{\mu}_k^{(t)}, \boldsymbol{\Omega}_1^{(t)}, \ldots, \boldsymbol{\Omega}_k^{(t)}]\right\}. \end{aligned}$$

By Bayes rule, we have

$$\begin{aligned} w_{ij}^{(t)} &:= \mathbf{E}[z_{ij} \mid \boldsymbol{y}, \boldsymbol{\pi}^{(t)}, \boldsymbol{\mu}_1^{(t)}, \ldots, \boldsymbol{\mu}_k^{(t)}, \boldsymbol{\Omega}_1^{(t)}, \ldots, \boldsymbol{\Omega}_k^{(t)}] \\ &= \frac{\pi_j^{(t)} h_j(\boldsymbol{y}_i|\boldsymbol{\mu}_j^{(t)}, \boldsymbol{\Omega}_j^{(t)})}{\sum_{j'=1}^k \pi_{j'}^{(t)} h_{j'}(\boldsymbol{y}_i|\boldsymbol{\mu}_{j'}^{(t)}, \boldsymbol{\Omega}_{j'}^{(t)})}. \end{aligned}$$

So the Q function becomes

$$\begin{aligned} & Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) \\ = \;& \sum_{i=1}^n \sum_{j=1}^k w_{ij}^{(t)} \ln \pi_j + \sum_{i=1}^n \sum_{j=1}^k w_{ij}^{(t)} \left[-\frac{1}{2} \ln \det \boldsymbol{\Omega}_j - \frac{1}{2}(\boldsymbol{y}_i - \boldsymbol{\mu}_j)^\intercal \boldsymbol{\Omega}_j^{-1}(\boldsymbol{y}_i - \boldsymbol{\mu}_j)\right]. \end{aligned}$$

- M step: maximizer of the Q function gives the next iterate

$$
\begin{aligned}
\pi_j^{(t+1)} &= \frac{\sum_i w_{ij}^{(t)}}{n} \\
\boldsymbol{\mu}_j^{(t+1)} &= \frac{\sum_{i=1}^n w_{ij}^{(t)} \boldsymbol{y}_i}{\sum_{i=1}^n w_{ij}^{(t)}} \\
\boldsymbol{\Omega}_j^{(t+1)} &= \frac{\sum_{i=1}^n w_{ij}^{(t)} (\boldsymbol{y}_i - \boldsymbol{\mu}_j^{(t+1)})(\boldsymbol{y}_i - \boldsymbol{\mu}_j^{(t+1)})^\mathsf{T}}{\sum_i w_{ij}^{(t)}}.
\end{aligned}
$$

  See KL Example 11.3.1 for multinomial MLE. See KL Example 11.2.3 for multivariate normal MLE.

- Compare these extremely simple updates to Newton type algorithms!

- Also note the ease of parallel computing with this EM algorithm. See, e.g., Suchard, M. A.; Wang, Q.; Chan, C.; Frelinger, J.; Cron, A. & West, M. Understanding GPU programming for statistical computation: studies in massively parallel massive mixtures. *Journal of Computational and Graphical Statistics*, 2010, 19, 419-438.

- In general, EM/MM algorithms are particularly attractive for parallel computing. See, e.g.,
  H Zhou, K Lange, & M Suchard. (2010) Graphical processing units and high-dimensional optimization, *Statistical Science*, 25:311-324.