MATH 680 Computation Intensive Statistics

December 3, 2018

MM & EM Algorithms

Contents

1 MM Algorithm

The MM algorithm is an iterative algorithm to minimize an objective function $f : \Theta \to \mathbb{R}$

$$\arg\min_{\theta \in \Theta} f(\theta)$$

over its open domain $\Theta \subset \mathbb{R}^p$.

Definition 1 (Majorization function)**.** The majorization function $g(\theta|\theta^{(k)})$ is said to majorize the function $f(\theta)$ at $\theta^{(k)}$ provided

$$\begin{aligned} f(\theta^{(k)}) &= g(\theta^{(k)}|\theta^{(k)}) \\ f(\theta) &\leq g(\theta|\theta^{(k)}) \qquad \text{for all } \theta. \end{aligned}$$

Remark:

1. The majorization relation between functions is closed under

- Formation of sums,

- Nonnegative products,
- Limits,
- Composition with an increasing function.

1. If $g(\theta|\theta^{(k)})$ minorizes the function $f(\theta)$ at $\theta^{(k)}$ then $-g(\theta|\theta^{(k)})$ majorizes $-f(\theta)$ at $\theta^{(k)}$.
2. In minimization, choose majorization $g(\theta|\theta^{(k)})$ and minimize it. This produces the next point $\theta^{(k+1)}$ in the algorithm.

The MM algorithm is summarized below:

Algorithmus 1 : MM algorithm.

---

1. Initialize $\theta^{(0)}$.
2. **Majorization:** at the k-th iteration, for $f(\theta)$ we find its majorization function $g(\theta|\theta^{(k)})$ at $\theta^{(k)}$ and
3. **Minimization:** compute
$$\theta^{(k+1)} = \arg\min_{\theta} g(\theta|\theta^{(k)}).$$
4. Check for convergence of either θ or the objective function. If the convergence criterion is not satisfied then set $k := k + 1$ and return to step 2.

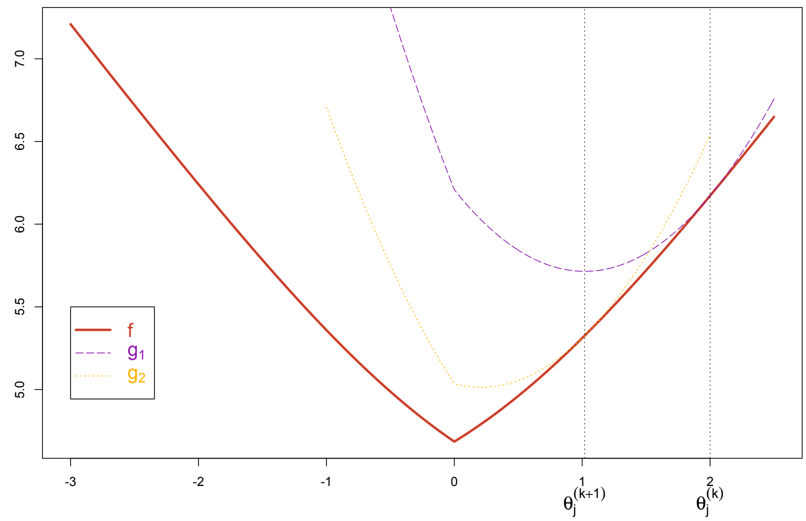


Figure 1: MM algorithm.

1.1 Descent property of MM algorithm

An MM minimization algorithm satisfies the descent property $f(\theta^{(k+1)}) \leq f(\theta^{(k)})$ with strict inequality unless both

$$\begin{aligned} g(\theta^{(k+1)}|\theta^{(k)}) &= g(\theta^{(k)}|\theta^{(k)}) \\ f(\theta^{(k+1)}) &= g(\theta^{(k+1)}|\theta^{(k)}). \end{aligned}$$

The descent property follows from the definitions and

$$f(\theta^{(k+1)}) \leq g(\theta^{(k+1)}|\theta^{(k)}) \leq g(\theta^{(k)}|\theta^{(k)}) = f(\theta^{(k)}).$$

The descent property makes the MM algorithm very stable.

1.2 Convergence of an MM Algorithm

- If an objective function is strictly convex or concave, then the MM algorithm will converge to the unique optimal point, assuming it exists.
- If convexity or concavity fail, but all stationary points are isolated, then the MM algorithm will converge to one of them.
- This point can be a local optimum, or in unusual circumstances, even a saddle point.

Proposition 1. *Let $\Theta \subset \mathbb{R}^p$ be open and suppose that the following conditions hold:*

1. The objective function $f : \Theta \to \mathbb{R}$ is differentiable;
2. $S(\theta^{(0)}) = \{\theta \in \Theta : f(\theta) \le f(\theta^{(0)})\}$ is closed and bounded;
3. the majorizing function $g(\theta|\bar{\theta}) : \Theta \to \mathbb{R}$ is differentiable with a unique minimizer for each $\bar{\theta} \in \Theta$;
4. $g(\cdot|\cdot) : \Theta \times \Theta \to \mathbb{R}$ is jointly continuous.

Let $\{\theta^{(k)}\}$ be the sequence of iterates generated by the MM algorithm. Then every limit point of $\{\theta^{(k)}\}$ is a stationary point of f.

Proof. Define $A : \Theta \to \Theta$, where $A(\theta^{(k)}) = \arg\min_{\theta \in \Theta} g(\theta|\theta^{(k)})$. Then $\theta^{(k+1)} = A(\theta^{(k)})$. From the MM property $f(\theta^{(k+1)}) \le f(\theta^{(k)})$ so $\{\theta^{(k)}\} \subset S(\theta^{(0)})$, which is closed and bounded. So there exists a subsequence $\{\theta^{j(k)}\} \to \bar{\theta}$. Similarly, $\{A(\theta^{j(k)})\} \subset S(\theta^{(0)})$ so there exists a subsubseqence $\{A(\theta^{m(j(k))})\} \to \bar{A}$ and

$$g(A(\theta^{m(j(k))})|\theta^{m(j(k))}) \le g(\theta|\theta^{m(j(k))}), \tag{1}$$

for all $\theta \in \Theta$. Since $g(\cdot|\cdot)$ is jointly continuous we take limits in (1) and see that

$$g(\bar{A}|\bar{\theta}) = \lim_{k\to\infty} g(A(\theta^{m(j(k))})|\theta^{m(j(k))}) \leq \lim_{k\to\infty} g(\theta|\theta^{m(j(k))}) = g(\theta|\bar{\theta}),$$

for all $\theta \in \Theta$. This implies that $\bar{A} = A(\bar{\theta})$. We assumed that the minimizer of $g(\cdot|\bar{\theta})$ is unique, so only one limit point of $\{A(\theta^{j(k)})\}$ exists, thus $\{A(\theta^{j(k)})\} \to A(\bar{\theta})$. Using the MM property,

$$f(\theta^{j(k+1)}) \leq f(\theta^{j(k)+1}) = f(A(\theta^{j(k)})) \leq f(\theta^{j(k)}). \tag{2}$$

Since f is continuous, we take limits in (2) and see that $f(\bar{\theta}) = f(A(\bar{\theta}))$. Also

$$f(A(\bar{\theta})) \leq g(A(\bar{\theta})|\bar{\theta}) \leq g(\bar{\theta}|\bar{\theta}) = f(\bar{\theta}) = f(A(\bar{\theta})).$$

So $g(A(\bar{\theta})|\bar{\theta}) = g(\bar{\theta}|\bar{\theta})$ and since $A(\bar{\theta})$ is the unique minimizer of $g(\cdot|\bar{\theta})$, we have that $A(\bar{\theta}) = \bar{\theta}$. Since $g(\cdot|\bar{\theta})$ is differentiable and $\bar{\theta}$ is its unique minimizer, $\nabla g(\bar{\theta}|\bar{\theta}) = 0$ (meaning $\nabla g(\cdot|\bar{\theta})$ evaluated at $\bar{\theta}$ equals zero). It remains to show that $\nabla g(\bar{\theta}|\bar{\theta}) = 0$ implies $\nabla g(\bar{\theta}) = 0$

Suppose that $\nabla g(\bar{\theta}|\bar{\theta}) = 0$ and $\nabla f(\bar{\theta}) \neq 0$, then $d = \nabla f(\bar{\theta})$ is an ascent direction and $\|d\| > 0$. Form the definition of differentiability and the definition of a majorization,

$$\begin{aligned}
\|d\|^2 + \|d\| a_2(\lambda d|\bar{\theta}) = \nabla f(\bar{\theta})' d + \|d\| a_2(\lambda d|\bar{\theta}) &= \frac{f(\bar{\theta} + \lambda d) - f(\bar{\theta})}{\lambda} \\
&\leq \frac{g(\bar{\theta} + \lambda d|\bar{\theta}) - g(\bar{\theta}|\bar{\theta})}{\lambda} \\
&= \nabla g(\bar{\theta}|\bar{\theta})' d + \|d\| a_1(\lambda d|\bar{\theta}) \\
&= \|d\| a_1(\lambda d|\bar{\theta}),
\end{aligned} \tag{3}$$

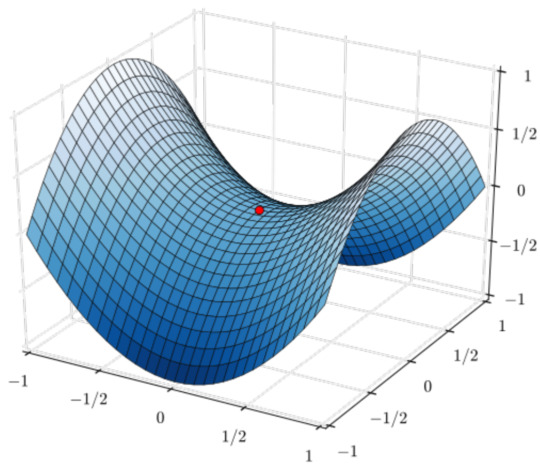


Figure 2: A saddle point on the graph of $z = x^2 - y^2$ (in red).

for all $\lambda > 0$ sufficiently small, where $\lim_{\lambda \to 0} a_2(\lambda d|\bar{\theta}) = 0$ and $\lim_{\lambda \to 0} a_1(\lambda d|\bar{\theta}) = 0$. The inequality in (3) implies that

$$\|d\| + a_2(\lambda d|\bar{\theta}) - a_1(\lambda d|\bar{\theta}) \leq 0, \tag{4}$$

for all $\lambda > 0$ sufficiently small, but this is impossible because $\|d\| > 0$ and choosing $\lambda > 0$ sufficiently close to zero will make the left side of (4) positive. $\square$

1.3 Case study: bridge penalty

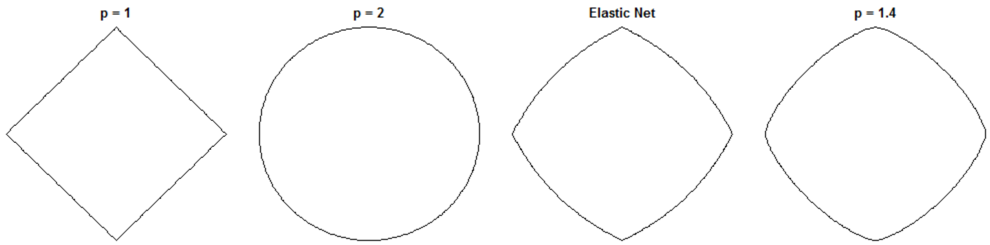


Figure 3: Compare unit circles of the different penalties. $p = 1$ corresponds to the LASSO, $p = 2$ to the Ridge, and $p = 1.4$ to one possible Bridge. The Elastic Net was generated with equal weighting on ℓ_1 and ℓ_2 penalties.

1.3.1 Model and properties

Suppose we want to compute the bridge penalized least-squares estimate of β_* defined by

$$\begin{aligned} \hat{\beta}_{-1}^{(\lambda,\gamma)} &\in \arg\min_{\tilde{\beta}\in\mathbb{R}^{p-1}} \left\{ \frac{1}{2}\|\tilde{y} - \tilde{X}\tilde{\beta}\|^2 + \frac{\lambda}{\gamma}\sum_{j=1}^{p-1} |\tilde{\beta}_j|^\gamma \right\} \\ \hat{\beta}_1^{(\lambda,\gamma)} &= \bar{y} - \bar{x}'\hat{\beta}_{-1}^{(\lambda,\gamma)}, \end{aligned} \tag{5}$$

where $\lambda \geq 0$ and $\gamma \in [1, 2]$ are tuning parameters. When $\gamma = 2$, (5) becomes ridges-penalized least-squares. When $\gamma = 1$, (5) becomes lasso-penalized least squares. Please see **Homework 3** for definitions of X, y, $\tilde{X}$, $\tilde{y}$ and $\tilde{\beta}$.

From Figure 3 we see that Bridge clearly lacks sparsity while Elastic Net preserves sparsity from its LASSO component.

- Lasso ($\gamma = 1$) shrinks small OLS estimates to zero and large by a constant;
- Ridge regression ($\gamma = 2$) shrinks the OLS estimates proportionally;
- Bridge regression ($1 < \gamma < 2$) shrinks small OLS estimates by a large rate and large

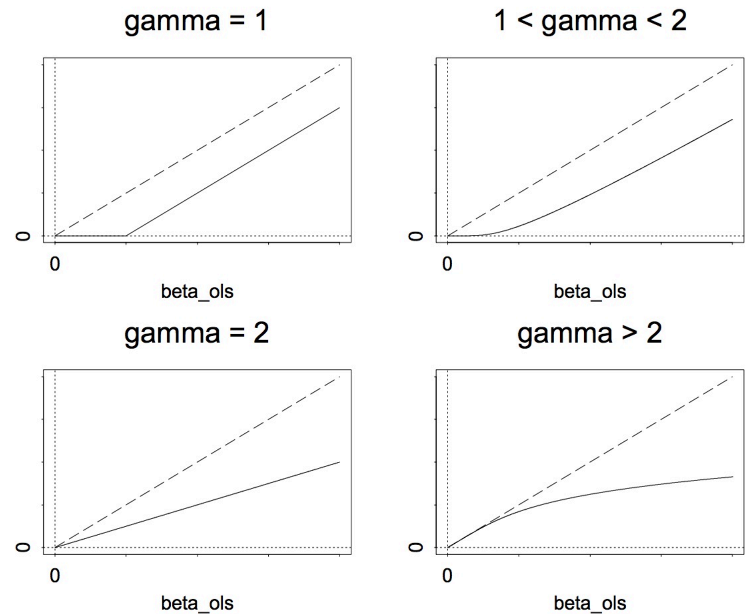


Figure 4: Shrinkage Effect of Bridge Regressions for Fixed $\lambda > 0$. Solid—the bridge estimator; dashed—the OLS estimator.

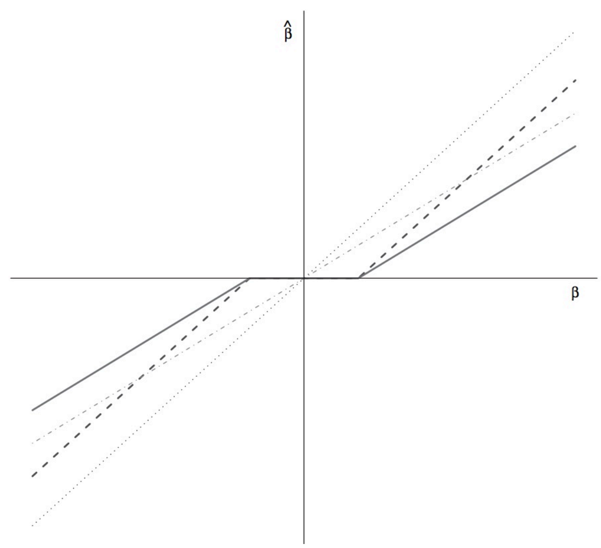


Figure 5: Exact solutions for the lasso ($- - --$), ridge regression ($- \cdot - \cdot -\cdot$) and the naive elastic net in an orthogonal design (______): the shrinkage parameters are $\lambda_1 = 2$ and $\lambda_2 = 1$.

by a small rate;

- Bridge regression ($\gamma > 2$) shrinks small OLS estimates by a small rate and large by a large rate.

In summary, bridge regression of large value of $\gamma(\gamma \geq 2)$ tends to retain small parameters while small value of γ ($\gamma < 2$) tends to shrink small parameters to zero.

1.3.2 Computation

First we majorize the penalty function $p(\theta)$: $\mathbb{R} \to \mathbb{R}$ defined by:

$$p(\theta) = |\theta|^\gamma, \qquad \gamma \in [1, 2].$$

Note that p is differentiable for $\gamma > 1$, but not twice differentiable when $\gamma < 2$ at 0.

To construct a majorization of $p(\theta)$ at $\bar{\theta}$, consider the function $h : \mathbb{R}_+ \to \mathbb{R}$, where $h(u) = u^{\gamma/2}$, a first-order Taylor series approximation of h at $\bar{u}$ is

$$h(u) \approx h(\bar{u}) + \nabla h(\bar{u})(u - \bar{u}) = g(u|\bar{u}). \tag{6}$$

where

$$\nabla h(u) = (\gamma/2)u^{\gamma/2-1}.$$

We see that $\nabla^2 h(u) = \frac{\gamma}{2}(\frac{\gamma}{2} - 1)u^{\gamma/2-1} < 0$, when $\gamma \in [1, 2]$, for all $u \in \mathbb{R}$. So h is concave, meaning that $g(u|\bar{u})$ majorizes h at $\bar{u}$. Plug in $u = |\theta|^2$ and $\bar{u} = |\bar{\theta}|^2$ in (6), we the majorization function for $|\theta|^\gamma$

$$|\theta|^\gamma \leq |\bar{\theta}|^\gamma + \frac{\gamma}{2}|\bar{\theta}|^{\gamma-2}(\theta^2 - \bar{\theta}^2),$$

for all $\theta \in \mathbb{R}$ with equality holding when $\theta = \bar{\theta}$. So the right hand side defines the majorization to p at $\bar{\theta}$.

Let f be the objective function in (5), and let $\beta^{(k)} \in \mathbb{R}^{p-1}$ be our current iterate. Then $g(\cdot|\beta^{(k)}) : \mathbb{R}^{p-1} \to \mathbb{R}$ defined by

$$g(\beta|\beta^{(k)}) = \frac{1}{2}\|\tilde{y} - \tilde{X}\tilde{\beta}\|^2 + \frac{\lambda}{\gamma}\sum_{j=1}^{p-1}\left[\left|\beta_j^{(k)}\right|^\gamma + \frac{\gamma}{2}\left|\beta_j^{(k)}\right|^{\gamma-2}\left\{\beta_j^2 - \left(\beta_j^{(k)}\right)^2\right\}\right]$$

majorizes f at $\beta^{(k)}$. We can write

$$g(\beta|\beta^{(k)}) = \text{constants} + \frac{1}{2}\|\tilde{y} - \tilde{X}\tilde{\beta}\|^2 + \frac{\lambda}{2}\sum_{j=1}^{p-1} m_j\beta_j^2,$$

where $m_j = \left|\beta_j^{(k)}\right|^{\gamma-2}$ for $j = 1, \ldots, p-1$ are the non-negative ridge penalty weights. We write

$$\nabla g(\beta^{(k+1)}|\beta^{(k)}) = 0$$

as

$$-\tilde{X}'\tilde{y} + \tilde{X}'\tilde{X}\beta^{(k+1)} + \lambda M\beta^{(k+1)} = 0.$$

So

$$\left(\tilde{X}'\tilde{X} + \lambda M\right)\beta^{(k+1)} = \tilde{X}'\tilde{y}$$

where $M = \text{diag}(m_1, \ldots, m_{p-1})$. We solve this system of equations using our favorite solver.

1.4 Example: Concave Penalized Linear Regression

$$\min ||\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}||^2 + \lambda \cdot \sum_{j=1}^{p} p_\lambda\Big(|\beta_j|\Big)$$

$\text{Pen}(t)$ is a concave increasing function.

e.g.

$$\begin{aligned}
& p_\lambda(t) = \lambda^2 - (t-\lambda)^2 I(t < \lambda) && \cdots\cdots \text{Hard thresholding} \\
\text{or} \quad & p'_\lambda(t) = (\lambda - \frac{t}{a})_+ && \cdots\cdots \text{MCP} \\
& p'_\lambda(t) = \lambda I(t \le \lambda) + \frac{(a\lambda - t)_+}{a-1} I(t > \lambda) && \cdots\cdots \text{SCAD}
\end{aligned}$$

Given $\boldsymbol{\beta}^0$, majorization can be

$$\begin{aligned}
\sum_{j=1}^{p} p_\lambda\Big(|\beta_j|\Big) &\le \sum_{j=1}^{p} \Big\{ p_\lambda\Big(|\beta_j^0|\Big) + p'_\lambda\Big(|\beta_j^0|\Big)\Big(|\beta_j| - |\beta_j^0|\Big)\Big\} \\
&\le \sum_{j=1}^{p} p_\lambda\Big(|\beta_j^0|\Big) + \sum_{j=1}^{p} p'_\lambda\Big(|\beta_j^0|\Big)|\beta_j| - \sum_{j=1}^{p} p'_\lambda\Big(|\beta_j^0|\Big)|\beta_j^0|
\end{aligned}$$

$$\begin{aligned}
Q(\boldsymbol{\beta}|\boldsymbol{\beta}^0) =& ||\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}||^2 + \sum_{j=1}^{p} p'_\lambda\Big(|\beta_j^0|\Big)|\beta_j| \\
& + \sum_{j=1}^{p} p_\lambda\Big(|\beta_j^0|\Big) - \sum_{j=1}^{p} p'_\lambda\Big(|\beta_j^0|\Big)|\beta_j^0|
\end{aligned}$$

$$\begin{aligned}\boldsymbol{\beta}^1 &= \arg\min_{\boldsymbol{\beta}} Q(\boldsymbol{\beta}|\boldsymbol{\beta}^0) \\ &= \arg\min_{\boldsymbol{\beta}} ||\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}||^2 + \sum_{j=1}^{p} \widehat{w}_j \cdot |\beta_j|\end{aligned}$$

$$\widehat{w}_j = p'_\lambda\Big(|\beta_j^0|\Big)$$

This is the <u>LLA</u> for concave penalization.

1.5 Example: Quadratic majorization for logistic regression

$$-\ell(\boldsymbol{\beta}) = \sum_{i=1}^{n} -y_i \mathbf{x}_i^T \boldsymbol{\beta} + \log\Big(1 + e^{\mathbf{x}_i^T \boldsymbol{\beta}}\Big) \quad \Big(= f(\boldsymbol{\beta})\Big)$$

$$\widehat{\boldsymbol{\beta}}^{\text{mle}} = \arg\min_{\boldsymbol{\beta}} -\ell(\boldsymbol{\beta})$$

$$\begin{aligned}\nabla^2 f(\boldsymbol{\beta}) = -\nabla^2 \ell(\boldsymbol{\beta}) &= \mathbf{X}^T \cdot \text{Diag}\Big(p(\mathbf{x}_i)(1 - p(\mathbf{x}_i))\Big)\mathbf{X} \\ &\leq \mathbf{X}^T \cdot \frac{1}{4} \cdot \mathbf{I} \cdot \mathbf{X} = \frac{1}{4}\mathbf{X}^T\mathbf{X} = \mathbf{H}\end{aligned}$$

Then

$$\begin{aligned}f(\boldsymbol{\beta}) &\leq f(\boldsymbol{\beta}^0) + \nabla f(\boldsymbol{\beta}^0) \cdot (\boldsymbol{\beta} - \boldsymbol{\beta}^0) + \frac{1}{2} \cdot (\boldsymbol{\beta} - \boldsymbol{\beta}^0)^T \mathbf{H} (\boldsymbol{\beta} - \boldsymbol{\beta}^0) \\ &= Q(\boldsymbol{\beta}|\boldsymbol{\beta}^0)\end{aligned}$$

$$\begin{aligned}\boldsymbol{\beta}^1 &= \arg\min_{\boldsymbol{\beta}} Q(\boldsymbol{\beta}|\boldsymbol{\beta}^0)\\ &= \arg\min_{\boldsymbol{\beta}} \frac{1}{2}(\boldsymbol{\beta}-\boldsymbol{\beta}^0)^T\mathbf{H}(\boldsymbol{\beta}-\boldsymbol{\beta}^0)+\nabla f(\boldsymbol{\beta}^0)\cdot(\boldsymbol{\beta}-\boldsymbol{\beta}^0)\end{aligned}$$

thus

$$\boldsymbol{\beta}^1 = \boldsymbol{\beta}^0 - \mathbf{H}^{-1}\cdot\nabla f(\boldsymbol{\beta}^0)$$

Compare it to Newton's method

$$\boldsymbol{\beta}^1 = \boldsymbol{\beta}^0 - \left(\nabla^2 f(\boldsymbol{\beta}^0)\right)^{-1}\cdot\nabla f(\boldsymbol{\beta}^0)$$

Newton's may fail to converge, the fixed Hessian method always converges, but at a slower rate.

1.6 Example: Lasso Penalized Median Regression

$$f(\theta) = \sum_{i=1}^{n} |y_i - \theta|$$

$$\widehat{\theta} = \min_{\theta} f(\theta) = \text{sample median of } \{y_i\}$$

$$|y_i - \theta| \le \frac{1}{2}\frac{(y_i-\theta)^2}{|y_i-\theta^0|} + \frac{1}{2}|y_i-\theta^0|$$

$$Q(\theta|\theta^0) = \sum_{i=1}^{n} \frac{1}{2}\frac{(y_i-\theta)^2}{|y_i-\theta^0|} + \frac{1}{2}|y_i-\theta^0|$$

$$
\begin{aligned}
\theta^1 &= \arg\min_{\theta} Q(\theta|\theta^0) \\
&= \arg\min_{\theta} \sum_{i=1}^{n} w_i^0 (y_i - \theta)^2 \qquad\qquad w_i^0 = \frac{1}{|y_i - \theta^0|} \\
&= \frac{\sum_{i=1}^{n} w_i^0 y_i}{\sum_{i=1}^{n} w_i^0} \qquad\quad \text{(weighted average)}
\end{aligned}
$$

Warning: $w_i^0 = \infty$ if $\theta^0 = y_i$

Now for the case with covariates.

$$
\widehat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \sum_{i=1}^{n} |y_i - \mathbf{x}_i^T \boldsymbol{\beta}|
$$

The same procedure applies.

$$
|y_i - \mathbf{x}_i^T \boldsymbol{\beta}| \leq \frac{1}{2} \frac{(y_i - \mathbf{x}_i^T \boldsymbol{\beta})^2}{|y_i - \mathbf{x}_i^T \boldsymbol{\beta}^0|} + \frac{1}{2} |y_i - \mathbf{x}_i^T \boldsymbol{\beta}^0|
$$

Thus

$$
Q(\boldsymbol{\beta}|\boldsymbol{\beta}^0) = \sum_{i=1}^{n} \frac{1}{2} \frac{(y_i - \mathbf{x}_i^T \boldsymbol{\beta})^2}{|y_i - \mathbf{x}_i^T \boldsymbol{\beta}^0|} + \sum_{i=1}^{n} \frac{1}{2} |y_i - \mathbf{x}_i^T \boldsymbol{\beta}^0|
$$

$$
\boldsymbol{\beta}^1 = \arg\min_{\boldsymbol{\beta}} \sum_{i=1}^{n} w_i^0 \cdot (y_i - \mathbf{x}_i^T \boldsymbol{\beta})^2
$$

$$
w_i^0 = \frac{1}{|y_i - \mathbf{x}_i^T \boldsymbol{\beta}^0|}
$$

Now consider Lasso Penalized Median Regression.

$$
\widehat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \sum_{i=1}^{n} |y_i - \mathbf{x}_i^T \boldsymbol{\beta}| + \lambda ||\boldsymbol{\beta}||_1
$$

$$|y_i - \mathbf{x}_i^T\boldsymbol{\beta}| \le \frac{1}{2}\frac{(y_i - \mathbf{x}_i^T\boldsymbol{\beta})^2}{|y_i - \mathbf{x}_i^T\boldsymbol{\beta}^0|} + \frac{1}{2}|y_i - \mathbf{x}_i^T\boldsymbol{\beta}^0|$$

$$Q(\boldsymbol{\beta}|\boldsymbol{\beta}^0) = \sum_{i=1}^n \frac{1}{2}\frac{(y_i - \mathbf{x}_i^T\boldsymbol{\beta})^2}{|y_i - \mathbf{x}_i^T\boldsymbol{\beta}^0|} + \lambda||\boldsymbol{\beta}||_1 + \sum_{i=1}^n \frac{1}{2}|y_i - \mathbf{x}_i^T\boldsymbol{\beta}^0|$$

$$\begin{aligned}\boldsymbol{\beta}^1 &= \arg\min_{\boldsymbol{\beta}} Q(\boldsymbol{\beta}|\boldsymbol{\beta}^0)\\ &= \arg\min_{\boldsymbol{\beta}} \sum_{i=1}^n w_i^0 \cdot (y_i - \mathbf{x}_i^T\boldsymbol{\beta})^2 + 2\lambda \cdot ||\boldsymbol{\beta}||_1\end{aligned}$$

$$w_i^0 = \frac{1}{|y_i - \mathbf{x}_i^T\boldsymbol{\beta}^0|}$$

So do not use $|y_i - \mathbf{x}_i^T\boldsymbol{\beta}| \approx \frac{|y_i - \mathbf{x}_i^T\boldsymbol{\beta}|^2}{|y_i - \mathbf{x}_i^T\boldsymbol{\beta}^0|}$ to derive the algorithm.

1.7 Example: Bradley-Terry Ranking Model

A sports league has m teams. Team i has skill level θ_i. Then

$$\text{Prob}\Big(\text{team } i \text{ beats team } j\Big) = \frac{\theta_i}{\theta_i + \theta_j}, \quad \theta_1 = 1$$

Observe b_{ij} = the number of times team i beats team j. Find $\widehat{\boldsymbol{\theta}}^{\text{mle}}$.

The likelihood is

$$L(\boldsymbol{\theta}) = \prod_{i,j}\left(\frac{\theta_i}{\theta_i + \theta_s j}\right)^{b_{ij}}$$

$$\ell(\boldsymbol{\theta}) = \log L(\boldsymbol{\theta}) = \sum_{i,j} b_{ij}\Big(\log\theta_i - \log(\theta_i + \theta_j)\Big)$$

$$
\begin{aligned}
\widehat{\boldsymbol{\theta}}^{\text{mle}} &= \arg\min_{\boldsymbol{\theta}} -\ell(\boldsymbol{\theta}) \\
&= \arg\min_{\boldsymbol{\theta}} \sum_{i,j} b_{ij}\Big(-\log\theta_i + \log(\theta_i+\theta_j)\Big)
\end{aligned}
$$

$\log(t)$ is concave.

$$
\begin{aligned}
\log(t) &\leq \log(t_0) + \log'(t_0)(t-t_0) \\
&= \log(t_0) + \frac{1}{t_0}(t-t_0)
\end{aligned}
$$

$$
\log(\theta_i+\theta_j) \leq \log(\theta_i^0+\theta_j^0) + \frac{\theta_i+\theta_j}{\theta_i^0+\theta_j^0} - 1
$$

$$
Q(\boldsymbol{\theta}|\boldsymbol{\theta}^0) = \sum_{i,j} b_{ij}\Big(-\log\theta_i + \log(\theta_i^0+\theta_j^0) + \frac{\theta_i+\theta_j}{\theta_i^0+\theta_j^0} - 1\Big)
$$

$$
\begin{aligned}
\boldsymbol{\theta}^1 &= \arg\min_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}|\boldsymbol{\theta}^0) \\
&= \arg\min_{\boldsymbol{\theta}} \sum_{i,j} b_{ij}\Big(-\log\theta_i + \frac{\theta_i+\theta_j}{\theta_i^0+\theta_j^0}\Big)
\end{aligned}
$$

$$
\begin{aligned}
&\frac{\partial}{\partial \theta_k}\Big(\sum_{i,j} b_{ij}(-\log\theta_i + \frac{\theta_i+\theta_j}{\theta_i^0+\theta_j^0})\Big)\\
=&\sum_{i,j} b_{ij}\Big(-\frac{1}{\theta_k}I(i=k) + \frac{1}{\theta_i^0+\theta_j^0}I(i=k) + \frac{1}{\theta_i^0+\theta_j^0}I(j=k)\Big)\\
=&-\frac{1}{\theta_k}\sum_{j\neq k} b_{kj} + \sum_{j\neq k}\frac{b_{kj}}{\theta_k^0+\theta_j^0} + \sum_{i\neq k}\frac{b_{ik}}{\theta_i^0+\theta_k^0} = 0\\
\Longrightarrow&\widehat{\theta}_k^1 = \frac{\sum\limits_{j\neq k} b_{kj}}{\sum\limits_{j\neq k}\frac{b_{kj}}{\theta_k^0+\theta_j^0} + \sum\limits_{i\neq k}\frac{b_{ik}}{\theta_i^0+\theta_k^0}}
\end{aligned}
$$

2 EM Algorithm

In a statistical model, we have observed variable $\mathbf{y}$ and parameter $\boldsymbol{\theta}$. Then the likelihood is

$$L(\boldsymbol{\theta}|\mathbf{x}) = p(\mathbf{x}|\boldsymbol{\theta}).$$

Let $\mathbf{Z}$ be another random vector. If we consider the joint distribution of $(\mathbf{X},\mathbf{Z})|\boldsymbol{\theta}$ as $p(\mathbf{x},\mathbf{z}|\boldsymbol{\theta})$, then

$$p(\mathbf{x}|\boldsymbol{\theta}) = \int p(\mathbf{x},\mathbf{z}|\boldsymbol{\theta})\,\mathrm{d}\mathbf{z}.$$

The MLE of $\boldsymbol{\theta}$ is

$$\widehat{\boldsymbol{\theta}}^{\text{mle}} = \arg\max_{\boldsymbol{\theta}} \log L(\boldsymbol{\theta}|\mathbf{x}).$$

The E-M algorithm can be used to find $\widehat{\boldsymbol{\theta}}^{\text{mle}}$ by including $\mathbf{Z}$ variables, although $\mathbf{Z}$ variables are not part of the data. $\mathbf{Z}$ is called "missing data" or "latent variables".

The "complete" likelihood is

$$L(\boldsymbol{\theta}|\mathbf{x},\mathbf{z}) = p(\mathbf{x},\mathbf{z}|\boldsymbol{\theta}).$$

- **E-step.** Compute the Q function

$$Q(\boldsymbol{\theta}|\boldsymbol{\theta}_k) = \mathbb{E}_{\mathbf{Z}|\mathbf{X},\boldsymbol{\theta}_k}\{\log L(\boldsymbol{\theta}|\mathbf{x},\mathbf{z})\}.$$

- **M-step.**

$$\boldsymbol{\theta}_{k+1} = \arg\max_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}|\boldsymbol{\theta}_k).$$

2.1 Why EM works? – the EM algorithm is an MM algorithm

Ascent property: $L(\boldsymbol{\theta}_{k+1}|\mathbf{x}) \geq L(\boldsymbol{\theta}_k|\mathbf{x})$.

Proof.

$$\begin{aligned}
& p(\mathbf{x},\mathbf{z}|\boldsymbol{\theta}) = p(\mathbf{z}|\mathbf{x},\boldsymbol{\theta})p(\mathbf{x}|\boldsymbol{\theta}) \\
\Longleftrightarrow\ & \log L(\boldsymbol{\theta}|\mathbf{x},\mathbf{z}) = \log p(\mathbf{z}|\mathbf{x},\boldsymbol{\theta}) + \log L(\boldsymbol{\theta}|\mathbf{x}) \\
\Longrightarrow\ & Q(\boldsymbol{\theta}|\boldsymbol{\theta}_k) = \mathbb{E}_{\mathbf{z}|\mathbf{x},\boldsymbol{\theta}_k} \log L(\boldsymbol{\theta}|\mathbf{x},\mathbf{z}) \\
& \quad = \int \log p(\mathbf{z}|\mathbf{x},\boldsymbol{\theta}) \cdot p(\mathbf{z}|\mathbf{x},\boldsymbol{\theta}_k)\,\mathrm{d}\mathbf{z} + \log L(\boldsymbol{\theta}|\mathbf{x}).
\end{aligned}$$

By the M-step, $Q(\boldsymbol{\theta} = \boldsymbol{\theta}_{k+1}|\boldsymbol{\theta}_k) \geq Q(\boldsymbol{\theta}_k|\boldsymbol{\theta}_k)$, thus

$$\begin{aligned}
&\int \log p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta}_{k+1}) \cdot p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta}_k)\, \mathrm{d}\mathbf{z} + \log L(\boldsymbol{\theta}_{k+1}|\mathbf{x}) \\
&\geq \int \log p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta}_k) \cdot p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta}_k)\, \mathrm{d}\mathbf{z} + \log L(\boldsymbol{\theta}_k|\mathbf{x}) \\
\iff &\log L(\boldsymbol{\theta}_{k+1}|\mathbf{x}) - \log L(\boldsymbol{\theta}_k|\mathbf{x}) \\
&\geq -\int \log \frac{p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta}_{k+1})}{p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta}_k)} \cdot p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta}_k)\, \mathrm{d}\mathbf{z} \\
&\geq 0,
\end{aligned}$$

where the last step is from

$$\mathbb{E}(-\log X) \geq -\log(\mathbb{E}X)$$

since $-\log(t)$ is convex, so use Jensen's inequality,

$$\begin{aligned}
&-\int \log \frac{p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta}_{k+1})}{p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta}_k)} \cdot p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta}_k)\, \mathrm{d}\mathbf{z} \\
&= \mathbb{E}\left(-\log \tilde{\mathbf{X}}\right) & \tilde{\mathbf{X}} = \frac{p(\mathbf{Z}|\mathbf{X}, \boldsymbol{\theta}_{k+1})}{p(\mathbf{Z}|\mathbf{X}, \boldsymbol{\theta}_k)} \\
&\geq -\log\left(\mathbb{E}\tilde{\mathbf{X}}\right) & \text{where } \ \mathbf{Z} \sim p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta}_k),
\end{aligned}$$

but

$$\begin{aligned}
\mathbb{E}\tilde{\mathbf{X}} &= \int \frac{p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta}_{k+1})}{p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta}_k)} \cdot p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta}_k)\, \mathrm{d}\mathbf{z} \\
&= \int p(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta}_{k+1})\, \mathrm{d}\mathbf{z} \\
&= 1.
\end{aligned}$$

$\square$

2.2 Generalized EM (GEM)

By the proof, we know that if in the M-step we just find Q_{k+1} such that

$$Q(\boldsymbol{\theta} = \boldsymbol{\theta}_{k+1}) > Q(\boldsymbol{\theta} = \boldsymbol{\theta}_k),$$

then the EM algorithm still works.

Sometimes, GEM is preferred because the exact M-step May Not be simple.

2.2.1 Example 1. Gaussian Mixture Model.

$\mathbf{X}_i \overset{iid}{\sim} f(\mathbf{x}|\boldsymbol{\theta})$, where $\mathbf{X}_i = (X_{i1}, \ldots, X_{ip})^{\mathrm{T}}$, and

$$f(\mathbf{x}|\boldsymbol{\theta}) = \sum_{j=1}^{c} r_j \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j),$$

where $r_j > 0$, $\sum_{j=1}^{c} r_j = 1$, and $(\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)$ are not the same.

We can derive an EM algorithm to fit this model. One application of this Gaussian mixture model is model-based clustering. The likelihood function is

$$\begin{aligned} L(\boldsymbol{\theta}|\mathbf{X}) = f(\mathbf{X}|\boldsymbol{\theta}) &= \prod_{i=1}^{n} f(\mathbf{x}_i|\boldsymbol{\theta}) \\ &= \prod_{i=1}^{n} \left[\sum_{j=1}^{c} r_j \mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j) \right]. \end{aligned}$$

Given data $\{\mathbf{x}_i\}_{i=1}^n$, we maximize the log-likelihood function

$$\log L(\boldsymbol{\theta}|\mathbf{X}) = \sum_{i=1}^{n} \log \sum_{j=1}^{c} r_j \mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j).$$

$$\widehat{\boldsymbol{\theta}}^{\text{mle}} = \arg\max_{\boldsymbol{\theta}} \log L(\boldsymbol{\theta}|\mathbf{X}),$$

where $\boldsymbol{\theta} = (r_1, \ldots, r_c, \boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_c, \boldsymbol{\Sigma}_1, \ldots, \boldsymbol{\Sigma}_c)$.

Direct optimization is OK, just too many parameters involved together. EM can separate them.

We introduce "Missing Data" $\mathbf{Y}$:

$$\begin{aligned}
&\mathbf{X}_i|Y_i \sim \mathcal{N}(\boldsymbol{\mu}_{Y_i}, \boldsymbol{\Sigma}_{Y_i});\\
&Y_i \sim \text{Multinomial}(r_1, \ldots, r_c) \qquad p(y_i = j) = r_j,\ j = 1, \ldots, c,\ \sum_{j=1}^{c} r_j = 1;\\
&(\mathbf{X}_i, Y_i)\ i = 1, \ldots, n \text{ independent samples.}
\end{aligned}$$

The joint density is

$$f_{\mathbf{X},Y}(\mathbf{x}, y|\boldsymbol{\theta}) = \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_y, \boldsymbol{\Sigma}_y) r_y$$

The marginal density of $\mathbf{x}$ is

$$f(\mathbf{x}|\boldsymbol{\theta}) = \int f_{\mathbf{X},Y}(\mathbf{x}, y|\boldsymbol{\theta}) dy = \int f_{\mathbf{X}|Y}(\mathbf{x}|y, \boldsymbol{\theta}) f_Y(y) dy = \sum_{j=1}^{c} r_j \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)$$

The posterior density of $Y|\mathbf{X}$ is

$$f_{Y|\mathbf{X}}(y|\mathbf{x}, \boldsymbol{\theta}) = \frac{f_{\mathbf{X},Y}(\mathbf{x}, y|\boldsymbol{\theta})}{f(\mathbf{x}|\boldsymbol{\theta})} = \frac{\mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_y, \boldsymbol{\Sigma}_y) r_y}{\sum_{j=1}^{c} \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j) r_j}.$$

Therefore, the complete log-likelihood is

$$\begin{aligned}\log L(\boldsymbol{\theta}|\mathbf{X},\mathbf{y}) &= \sum_{i=1}^{n}\Big(\log\mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_{y_i},\boldsymbol{\Sigma}_{y_i}) + \log r_{y_i}\Big) \\ &= \sum_{i=1}^{n}\left(\sum_{j=1}^{c}\Big(\log\mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_j,\boldsymbol{\Sigma}_j) + \log r_j\Big) I(y_i = j)\right).\end{aligned}$$

- **E-step.**

$$\begin{aligned}Q(\boldsymbol{\theta}|\boldsymbol{\theta}_k) &= \mathbb{E}_{\mathbf{Y}|\mathbf{X},\boldsymbol{\theta}_k}\log L(\boldsymbol{\theta}|\mathbf{X},\mathbf{Y}) \\ &= \sum_{i=1}^{n}\left(\sum_{j=1}^{c}\Big(\log\mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_j,\boldsymbol{\Sigma}_j) + \log r_j\Big)\cdot\gamma_{ij}^{(k)}\right),\end{aligned}$$

  where

$$\gamma_{ij}^{(k)} = \hat{p}(y_i = j|\mathbf{x}_i,\boldsymbol{\theta}_k) = \frac{\mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_j^{(k)},\boldsymbol{\Sigma}_j^{(k)})r_j^{(k)}}{\sum_{j'=1}^{c}\mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_{j'}^{(k)},\boldsymbol{\Sigma}_{j'}^{(k)})r_{j'}^{(k)}}.$$

- **M-step.**

$$\boldsymbol{\theta}_{k+1} = \arg\max_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}|\boldsymbol{\theta}_k),$$

  where

$$\begin{aligned}Q(\boldsymbol{\theta}|\boldsymbol{\theta}_k) &= \sum_{i=1}^{n}\left(\sum_{j=1}^{c}\left[-\frac{(\mathbf{x}_i-\boldsymbol{\mu}_j)^{\mathrm{T}}\boldsymbol{\Sigma}_j^{-1}(\mathbf{x}_i-\boldsymbol{\mu}_j)}{2} + \frac{1}{2}\log\det\boldsymbol{\Sigma}_j^{-1}\right.\right. \\ &\qquad \left.\left. + \log(2\pi)^{p/2} + \log r_j\right]\cdot\gamma_{ij}^{(k)}\right) \\ &= \sum_{j=1}^{c}\log r_j\cdot\left(\sum_{i=1}^{n}\gamma_{ij}^{(k)}\right) + \sum_{j=1}^{c}\left(\sum_{i=1}^{n}\gamma_{ij}^{(k)}\right)\frac{1}{2}\log\det\boldsymbol{\Sigma}_j^{-1} \\ &\qquad + \sum_{j=1}^{c}\sum_{i=1}^{n}-\frac{1}{2}\gamma_{ij}^{(k)}(\mathbf{x}_i-\boldsymbol{\mu}_j)^{\mathrm{T}}\boldsymbol{\Sigma}_j^{-1}(\mathbf{x}_i-\boldsymbol{\mu}_j).\end{aligned}$$

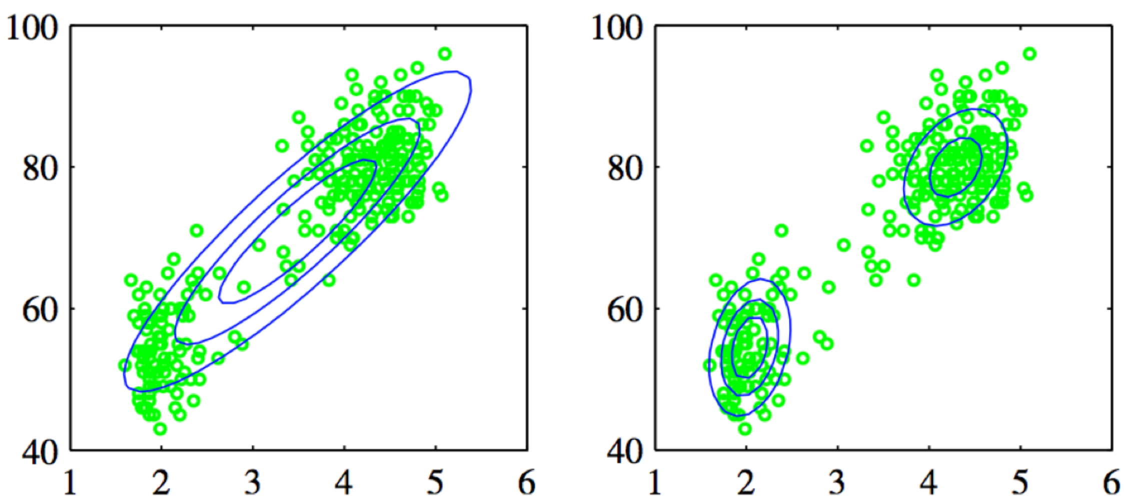
We maximize $Q(\boldsymbol{\theta}|\boldsymbol{\theta}_k)$ with the constraint $\sum_{j=1}^c r_j = 1$, which implies that

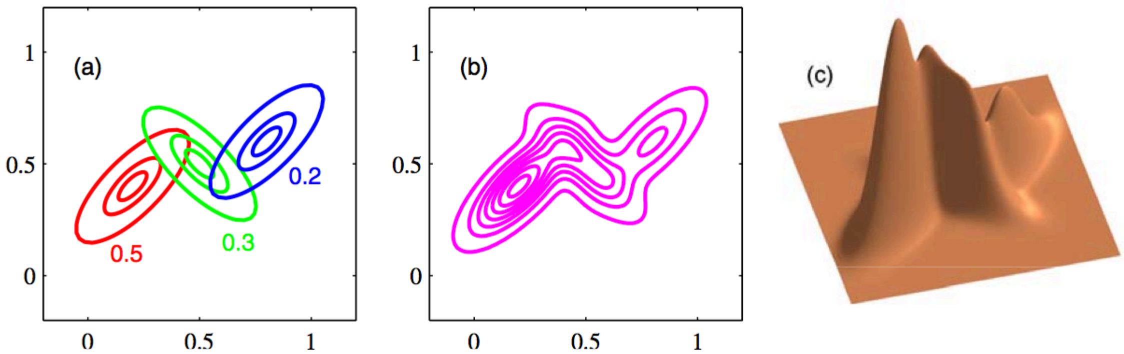
$$
\begin{aligned}
r_j^{(k+1)} &= \frac{\sum_{i=1}^n \gamma_{ij}^{(k)}}{\sum_{j=1}^c \left(\sum_{i=1}^n \gamma_{ij}^{(k)}\right)} = \frac{\sum_{i=1}^n \gamma_{ij}^{(k)}}{n} \\
\boldsymbol{\mu}_j^{(k+1)} &= \frac{\sum_{i=1}^n \gamma_{ij}^{(k)} \mathbf{x}_i}{\sum_{i=1}^n \gamma_{ij}^{(k)}} \\
\boldsymbol{\Sigma}_j^{(k+1)} &= \frac{\sum_{i=1}^n \gamma_{ij}^{(k)} (\mathbf{x}_i - \boldsymbol{\mu}_j^{(k+1)})(\mathbf{x}_i - \boldsymbol{\mu}_j^{(k+1)})^{\mathrm{T}}}{\sum_{i=1}^n \gamma_{ij}^{(k)}},
\end{aligned}
$$

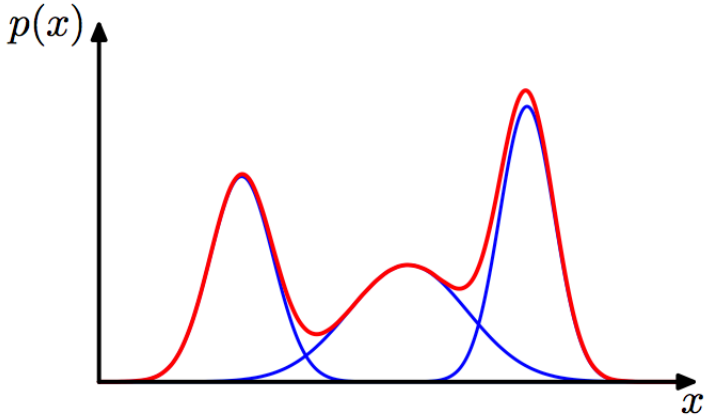
where we used the fact that

$$
\frac{\partial}{\partial \boldsymbol{\Omega}} \{-\mathrm{tr}(\mathbf{A}\boldsymbol{\Omega}) + \log\det \boldsymbol{\Omega}\} = -\mathbf{A} + \boldsymbol{\Omega}^{-1}.
$$

Try multiple random initial values for $\boldsymbol{\theta}^0$.








```
###############################################################
## Fit a Guassian mixture model with the EM algorithm
###############################################################
##  Arguments
##    X, an n row by p  column data matrix, each row is
##        an observed data point
##    C, the number of clusters/categories for Y (at least two)
```

```
##    tol, the convergence tolerance for the EM algorithm
##    maxit, the maximum number of iterations allowed
##    quiet, should the function stay quiet?
##
##  The function returns a list with elements
##    mu.mat, this is a p by C matrix, the yth column is
##            the estimate of the yth mean vector
##    Sigma, this is a list of the C covariance matrix estimates
##    pi.list, this is the vector of the C  estimated
##             probability masses for the maringal distribution of Y
##    P.mat, this is an n by C matrix who's (i,j)th element
##            is the final iterate's estimate of P(Y=j|X=x_i)
##    total.iterations, the total EM steps taken
#################################################################
gmmfit=function(X, C, tol=1e-7, maxit=1e3, quiet=TRUE)
{
  p=ncol(X)
  n=nrow(X)
  ## get starting values
  pi.list=rep(1/C, C)
  mu.mat=matrix(rnorm(p*C), nrow=p, ncol=C) +
apply(X, 2, mean)%*%matrix(1, nrow=1, ncol=C)
  Sigma=vector(length=C, mode="list")
```

```
for(y in 1:C)
{
  Sigma[[y]]=diag(p)
}

objfnval=-Inf

k=0
iterating=TRUE
while(iterating)
{
  k=k+1

  ###########################################################################
  ##  Compute the n by C matrix P.mat who's (i,j)th entry
  ##  is the current iterate's estimate of P(Y=j|X=x_i)
  logPhinum=matrix(NA, nrow=n, ncol=C)
  for (y in 1:C)
  {
   Xcentered=scale(X, scale=FALSE, center=mu.mat[,y])
   qf=-0.5*apply(t(Xcentered)*qr.solve(Sigma[[y]], t(Xcentered)), 2, sum)
   logPhinum[,y]=qf-0.5*determinant(Sigma[[y]],
logarithm=TRUE)$mod[1]-0.5*p*log(2*pi)+log(pi.list[y])
```

```
}

newobjfnval=sum(log(apply(exp(logPhinum), 1, sum)))
## control numerical stability by adjusting on the log scale:
##   subtract the row maximum from each element in the row
logPhinum = logPhinum - apply(logPhinum, 1, max)%*%matrix(1, nrow=1, ncol=C)
Phinum=exp(logPhinum)
Phiden=apply(Phinum, 1, sum)
P.mat=Phinum/Phiden
##########################################################################

if(!quiet) cat("k=", k, "f at kth iterate is" , newobjfnval, "\n")

if( ((newobjfnval - objfnval) < tol)  | (k > maxit) )
  iterating=FALSE

objfnval=newobjfnval

for(y in 1:C)
{
  ## update the pi's
  sumprob=sum(P.mat[,y])
  pi.list[y]=sumprob/n
```

```
      ## update the mu's
      weight.matrix=diag(P.mat[,y]/sumprob)
      mu.mat[,y] = apply(weight.matrix %*% X, 2, sum)

      ## update the Sigma's
      Xcentered=scale(X, scale=FALSE, center=mu.mat[,y])
      Sigma[[y]] =  crossprod(Xcentered, weight.matrix%*%Xcentered)
    }
  }
  return(list(mu.mat=mu.mat, Sigma=Sigma,
pi.list=pi.list, P.mat=P.mat,
total.iterations=k))
}


set.seed(5)
n=1000
p=2
C=3

## create the three covariance matrices
## Sigma 1 is diagonal with equal diagonal elements
```

```
Sigma1=0.05*diag(p)

## Sigma 2 is not diagonal:
Evectors=cbind(c(1,1), c(1,-1))/sqrt(2)
Sigma2=Evectors%*% diag(c(0.001, 0.1))%*%t(Evectors)

## Sigma 3 is diagonal with unequal diagonal elements
Sigma3=diag(c(0.1, 0.001))


## compute their matrix square roots for data generation
Sigma1.sqrt=sqrt(0.05)*diag(p)
Sigma2.sqrt=Evectors%*% diag(sqrt(c(0.001, 0.1)))%*%t(Evectors)
Sigma3.sqrt=diag(sqrt(c(0.1, 0.001)))


## Create the three mu's
mu1=c(0,0)
mu2=c(1,0)
mu3=c(0,1)

## Create the three pi's
pi1=1/3
```

```
pi2=1/3
pi3=1/3


## Generate the data matrix:
X=matrix(NA, nrow=n, ncol=p)
y=rep(0,n)
for(i in 1:n)
{
  ## perform a multinomial trial to
  ## generate the response cateogory
  mtrial=rmultinom(1, size=1, prob=c(pi1,pi2,pi3))
  if(mtrial[1])  ## resulted in category 1
  {
    X[i,] = mu1 + Sigma1.sqrt%*%rnorm(p)
    y[i]=1
  } else if(mtrial[2])  ## resulted in category 2
  {
    X[i,] = mu2 + Sigma2.sqrt%*%rnorm(p)
    y[i]=2
  } else  ## resulted in category 3
  {
    X[i,] = mu3 + Sigma3.sqrt%*%rnorm(p)
```

```
    y[i]=3
  }
}

## plot the points without class labels
plot(X)

## plot points with class labels
plot(X, col=y)

## fit the Gaussian mixture model
outfast=gmmfit(X=X, C=C, tol=1e-7)

## get the assigned cluster/class labels
labels=apply(outfast$P.mat, 1,  which.max)

## add these labels to the plot
points(X, pch=c("1", "2", "3")[labels]
```

2.2.2 Example 2. Factor Model.

$\mathbf{Y}_1, \ldots, bY_n$ iid in $\mathbb{R}^p$. The factor model is

$$\mathbf{Y}_i = \boldsymbol{\beta}^{\mathrm{T}} \mathbf{X}_i + \mathbf{e}_i,$$

where $\boldsymbol{\beta}$ is a $q \times p$ matrix and $\mathbf{X}_i$ is a q-dimensional random vector. Note that $\{\mathbf{X}_i\}$ are unobserved, $\mathbb{E}\mathbf{X}_i = \mathbf{0}$ and $\mathrm{Cov}(\mathbf{X}_i) = \mathbf{I}_q$. Moreover, $\mathbf{e}_i$ is a p-dimensional random error vector, $\mathbb{E}\mathbf{e}_i = \mathbf{0}$ and $\mathrm{Cov}(\mathbf{e}_i) = \mathrm{diag}(\tau_1^2, \ldots, \tau_p^2)$.

The factor model can be written as

$$\mathbf{Y}_{n\times p} = \mathbf{X}_{n\times q}\boldsymbol{\beta}_{q\times p} + \boldsymbol{\varepsilon}_{n\times p}.$$

Note that $\mathbf{X}\boldsymbol{\beta} = (\mathbf{X}\mathbf{U})(\mathbf{U}^{\mathrm{T}}\boldsymbol{\beta})$, $\mathbb{E}\mathbf{X}\mathbf{U} = \mathbf{0}$ and $\mathrm{Cov}(\mathbf{X}\mathbf{U}) = \mathbf{I}_q$ if $\mathbf{U}_{q\times q}$ is an orthogonal matrix.

Let us derive an E-M algorithm for fitting the factor model under the normality assumption. Later we can see that the E-M algorithm does not depend on the normality assumption.

Assume $\mathbf{X}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_q)$, $\boldsymbol{\varepsilon}_i \sim \mathcal{N}(\mathbf{0}, \mathrm{diag}(\tau_1^2, \ldots, \tau_p^2))$, then $\mathbf{Y}_i \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{\beta} + \boldsymbol{\tau}^2)$, where $\boldsymbol{\tau}^2 = \mathrm{diag}(\tau_1^2, \ldots, \tau_p^2)$. The log-likelihood is

$$\ell(\boldsymbol{\tau}^2, \boldsymbol{\beta}) = -\frac{n}{2}\log\det(\boldsymbol{\tau}^2 + \boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{\beta}) - \frac{1}{2}\sum_{i=1}^{n}\mathbf{y}_i^{\mathrm{T}}(\boldsymbol{\tau}^2 + \boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{\beta})^{-1}\mathbf{y}_i,$$

or equivalently

$$\ell(\boldsymbol{\tau}^2, \boldsymbol{\beta}) = -\frac{n}{2}\Big(\log\det(\boldsymbol{\tau}^2 + \boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{\beta}) + \mathrm{tr}\big((\boldsymbol{\tau}^2 + \boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{\beta})^{-1}\widehat{\boldsymbol{\Sigma}}^{s}\big)\Big),$$

where $\widehat{\boldsymbol{\Sigma}}^{s} = n^{-1}\sum_{i=1}^{n}\mathbf{Y}_i\mathbf{Y}_i^{\mathrm{T}}$ is the sample covariance matrix of $\mathbf{Y}$. Thus,

$$(\widehat{\boldsymbol{\beta}}, \boldsymbol{\tau}^2)^{\text{mle}} = \underset{\boldsymbol{\beta},\boldsymbol{\tau}}{\arg\min}\Big\{\log\det(\boldsymbol{\tau}^2 + \boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{\beta}) + \mathrm{tr}\big((\boldsymbol{\tau}^2 + \boldsymbol{\beta}^{\mathrm{T}}\boldsymbol{\beta})^{-1}\widehat{\boldsymbol{\Sigma}}^{s}\big)\Big\}.$$

Consider $\mathbf{X}$ as the "missing data". By $\mathbf{X}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_q)$, the joint likelihood of $(\mathbf{X}, \mathbf{Y})$ is

$$
\begin{aligned}
&L_{\mathbf{Y},\mathbf{X}}(\boldsymbol{\tau}^2, \boldsymbol{\beta}) \\
&= \left(2\pi \prod_{j=1}^{p} \tau_j^2\right)^{-n/2} \exp\left\{-\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{p} \frac{(Y_{ij} - \mathbf{X}[i,]\boldsymbol{\beta}[,j])^2}{\tau_j^2}\right\} \\
&\quad \times (2\pi \det \mathbf{I})^{-n/2} \exp\left\{-\frac{1}{2}\sum_{i=1}^{n} \mathbf{X}[i,]\mathbf{X}[i,]^{\mathrm{T}}\right\} \qquad \text{(R notation)}.
\end{aligned}
$$

- **E-step.**

$$
\begin{aligned}
Q(\boldsymbol{\tau}^2, \boldsymbol{\beta}|\boldsymbol{\tau}^2_{(k)}, \boldsymbol{\beta}_{(k)}) &= \mathbb{E}_{\mathbf{X}|\mathbf{Y},\boldsymbol{\tau}^2_{(k)},\boldsymbol{\beta}_{(k)}}[\log L_{\mathbf{Y},\mathbf{X}}(\boldsymbol{\tau}^2, \boldsymbol{\beta})] \\
&= -\frac{1}{2}\sum_{i=1}^{n}\sum_{i=1}^{p} \frac{1}{\tau_j^2}\Big(Y_{ij}^2 - 2Y_{ij}\mathbb{E}\big(\mathbf{X}[i,]|\mathbf{Y}, \boldsymbol{\beta}_{(k)}, \boldsymbol{\tau}_{(k)}\big)\boldsymbol{\beta}[,j] \\
&\qquad\qquad + \boldsymbol{\beta}[,j]^{\mathrm{T}}\mathbb{E}\big(\mathbf{X}[i,]^{\mathrm{T}}\mathbf{X}[i,]|\mathbf{Y}, \boldsymbol{\beta}_{(k)}, \boldsymbol{\tau}^2_{(k)}\big)\boldsymbol{\beta}[,j]\Big) \\
&\quad - \frac{1}{2}\sum_{i=1}^{n} \mathbb{E}\Big[\mathbf{X}[i,]\mathbf{X}[i,]^{\mathrm{T}}|\mathbf{Y}, \boldsymbol{\beta}_{(k)}, \boldsymbol{\tau}^2_{(k)}\Big] \\
&\quad - \frac{n}{2}\sum_{j=1}^{p} \log \tau_j^2 + \text{constant}
\end{aligned}
$$

Note that

$$
\mathbf{X}|\mathbf{Y}, \boldsymbol{\beta}_{(k)}, \boldsymbol{\tau}_k^2 \sim \mathcal{N}\Big(\mathbf{Y}(\boldsymbol{\tau}_{(k)} + \boldsymbol{\beta}_{(k)}^{\mathrm{T}}\boldsymbol{\beta}_{(k)})^{-1}\boldsymbol{\beta}_{(k)}^{\mathrm{T}}, \mathbf{I} - \boldsymbol{\beta}_{(k)}(\boldsymbol{\tau}^2_{(k)} + \boldsymbol{\beta}_{(k)}^{\mathrm{T}}\boldsymbol{\beta}_{(k)})^{-1}\boldsymbol{\beta}_{(k)}^{\mathrm{T}}\Big)
$$

imples that

$$
\mathbb{E}\big(\mathbf{X}[i,]|\mathbf{Y}, \boldsymbol{\beta}_{(k)}, \boldsymbol{\tau}_{(k)^2}\big) = \boldsymbol{\delta}^{\mathrm{T}}\mathbf{Y}[i,]^{\mathrm{T}},
$$

$$
\mathrm{Var}\big(\mathbf{X}[i,]|\mathbf{Y}, \boldsymbol{\beta}_{(k)}, \boldsymbol{\tau}^2_{(k)}\big) = \boldsymbol{\Delta},
$$

and that

$$\mathbb{E}\big(\mathbf{X}[i,]^{\mathrm{T}}\mathbf{X}[i,]|\mathbf{Y},\boldsymbol{\beta}_{(k)},\boldsymbol{\tau}^2_{(k)}\big)=\boldsymbol{\Delta}+\boldsymbol{\delta}^{\mathrm{T}}\mathbf{Y}[i,]^{\mathrm{T}}\mathbf{Y}[i,]\boldsymbol{\delta},$$

where

$$\boldsymbol{\delta}=\big(\boldsymbol{\tau}^2_{(k)}+\boldsymbol{\beta}^{\mathrm{T}}_{(k)}\boldsymbol{\beta}_{(k)}\big)^{-1}\boldsymbol{\beta}^{\mathrm{T}}_{(k)}$$

and

$$\boldsymbol{\Delta}=\mathbf{I}-\boldsymbol{\beta}_{(k)}\big(\boldsymbol{\tau}^2_{(k)}+\boldsymbol{\beta}^{\mathrm{T}}_{(k)}\boldsymbol{\beta}_{(k)}\big)^{-1}\boldsymbol{\beta}^{\mathrm{T}}_{(k)}.$$

Treat $\mathbb{E}\big(\mathbf{X}[i,]\mathbf{X}[i,]^{\mathrm{T}}|\mathbf{Y},\boldsymbol{\beta}_{(k)},\boldsymbol{\tau}^2_{(k)}\big)$ as another constant because it does not involve $\boldsymbol{\beta},\boldsymbol{\tau}^2$. Thus,

$$\begin{aligned}
&Q(\boldsymbol{\beta},\boldsymbol{\tau}^2|\boldsymbol{\beta}_{(k)},\boldsymbol{\tau}^2_{(k)})\\
= &-\frac{1}{2}\sum_{i=1}^{n} n\log\tau_j^2-\frac{1}{2}\sum_{j=1}^{p}\sum_{i=1}^{n}\frac{Y_{ij}^2-2Y_{ij}\mathbf{Y}[i,]\boldsymbol{\delta}\boldsymbol{\beta}[,j]}{\tau_j^2}\\
&-\frac{1}{2}\sum_{j=1}^{p}\sum_{i=1}^{n}\frac{\boldsymbol{\beta}[,j]^{\mathrm{T}}\big[\boldsymbol{\Delta}+\boldsymbol{\delta}^{\mathrm{T}}\mathbf{Y}[i,]^{\mathrm{T}}\mathbf{Y}[i,]\boldsymbol{\delta}\big]\boldsymbol{\beta}[,j]}{\tau_j^2}\\
&+\text{constant}.
\end{aligned}$$

- **M-step.**

$$(\boldsymbol{\beta}_{(k+1)},\boldsymbol{\tau}^2_{(k+1)})=\operatorname*{arg\,max}_{\boldsymbol{\beta},\boldsymbol{\tau}^2} Q(\boldsymbol{\beta},\boldsymbol{\tau}^2|\boldsymbol{\beta}_{(k)},\boldsymbol{\tau}^2_{(k)}).$$

First, we can do optimization for each j.

$$\Big(\boldsymbol{\beta}_{(k+1)}[,j], \boldsymbol{\tau}^2_{(k+1)_j}\Big) = \arg\max \Bigg\{ -\frac{1}{2} n \log \tau_j^2 \\ - \frac{1}{2}\sum_{i=1}^{n} \frac{Y_{ij}^2 - 2Y_{ij}\mathbf{Y}[i,]\boldsymbol{\delta}\boldsymbol{\beta}[,j]}{\tau_j^2} \\ - \frac{1}{2}\sum_{i=1}^{n} \frac{\boldsymbol{\beta}[,j]^{\mathsf{T}}\big[\boldsymbol{\Delta} + \boldsymbol{\delta}^{\mathsf{T}}\mathbf{Y}[i,]^{\mathsf{T}}\mathbf{Y}[i,]\boldsymbol{\delta}\big]\beta[,j]}{\tau_j^2}\Bigg\}.$$

Solve $\widehat{\boldsymbol{\beta}}_{(k+1)}[,j]$ first.

$$\widehat{\boldsymbol{\beta}}_{(k+1)}[,j] = \arg\min_{\boldsymbol{\beta}} \boldsymbol{\beta}^{\mathsf{T}}(n\boldsymbol{\Delta} + \boldsymbol{\delta}^{\mathsf{T}}\mathbf{Y}^{\mathsf{T}}\mathbf{Y}\boldsymbol{\delta})\boldsymbol{\beta} - 2\Big(\sum_{i=1}^{n} Y_{ij}\mathbf{Y}[i,]\Big)\boldsymbol{\delta}\boldsymbol{\beta}.$$

Let $\widehat{\boldsymbol{\Sigma}}^s = n^{-1}\mathbf{Y}^{\mathsf{T}}\mathbf{Y}$. Then,

$$\begin{aligned}
\widehat{\boldsymbol{\beta}}_{(k+1)}[,j] &= \arg\min_{\boldsymbol{\beta}} n\boldsymbol{\beta}^{\mathsf{T}}\big(\boldsymbol{\Delta} + \boldsymbol{\delta}^{\mathsf{T}}\widehat{\boldsymbol{\Sigma}}^s\boldsymbol{\delta}\big)\boldsymbol{\beta} - 2n\widehat{\boldsymbol{\Sigma}}^s[j,]\boldsymbol{\delta}\boldsymbol{\beta} \\
&= \big(\boldsymbol{\Delta} + \boldsymbol{\delta}^{\mathsf{T}}\widehat{\boldsymbol{\Sigma}}^s\boldsymbol{\delta}\big)^{-1}\big(\widehat{\boldsymbol{\Sigma}}^s[,j]\boldsymbol{\delta}\big)^{\mathsf{T}} \\
&= \big(\boldsymbol{\Delta} + \boldsymbol{\delta}^{\mathsf{T}}\widehat{\boldsymbol{\Sigma}}^s\boldsymbol{\delta}\big)^{-1}\boldsymbol{\delta}^{\mathsf{T}}\widehat{\boldsymbol{\Sigma}}^s[,j].
\end{aligned}$$

Then,

$$\begin{aligned}
\boldsymbol{\tau}^2_{(k+1)_j} &= \frac{1}{n}\sum_{i=1}^{n} Y_{ij}^2 + \widehat{\boldsymbol{\beta}}^{\mathsf{T}}\big(\boldsymbol{\Delta} + \boldsymbol{\delta}^{\mathsf{T}}\widehat{\boldsymbol{\Sigma}}^s\boldsymbol{\delta}\big)\widehat{\boldsymbol{\beta}} - 2\widehat{\boldsymbol{\Sigma}}^s[j,]\boldsymbol{\delta}\widehat{\boldsymbol{\beta}} \\
&= \widehat{\boldsymbol{\Sigma}}^s_{jj} - \widehat{\boldsymbol{\Sigma}}^s[j,]\boldsymbol{\delta}\Big(\boldsymbol{\Delta} + \boldsymbol{\delta}^{\mathsf{T}}\widehat{\boldsymbol{\Sigma}}^s\boldsymbol{\delta}\Big)^{-1}\boldsymbol{\delta}^{\mathsf{T}}\widehat{\boldsymbol{\Sigma}}^s[,j] \\
&= \widehat{\boldsymbol{\Sigma}}^s_{jj} - \widehat{\boldsymbol{\Sigma}}^s[j,]\boldsymbol{\delta}\widehat{\boldsymbol{\beta}}.
\end{aligned}$$

E-M algorithm for factor analysis:

(1) Initialization $\boldsymbol{\beta}^0, (\boldsymbol{\tau}^2)^0$.

(2) Compute $\widehat{\boldsymbol{\Sigma}}^s = n^{-1}\mathbf{Y}^{\mathsf{T}}\mathbf{Y}$.

For $k = 1, 2, \ldots,$ compute

$$\begin{aligned}
\boldsymbol{\delta} &= \left(\boldsymbol{\tau}^2_{(k-1)} + \boldsymbol{\beta}^{\mathsf{T}}_{(k-1)}\boldsymbol{\beta}_{(k-1)}\right)^{-1}\boldsymbol{\beta}^{\mathsf{T}}_{(k-1)} \\
\boldsymbol{\Delta} &= \mathbf{I} - \boldsymbol{\beta}_{(k-1)}\left(\boldsymbol{\tau}^2_{(k-1)} + \boldsymbol{\beta}^{\mathsf{T}}_{(k-1)}\boldsymbol{\beta}_{(k-1)}\right)\boldsymbol{\beta}^{\mathsf{T}}_{(k-1)}.
\end{aligned}$$

Compute $\boldsymbol{\Delta} + \boldsymbol{\delta}^{\mathsf{T}}\widehat{\boldsymbol{\Sigma}}^s\boldsymbol{\delta} = \mathbf{M}$. Do Cholesky of $\mathbf{M}$.

For $j = 1, 2, \ldots, q$,

$$\begin{aligned}
\boldsymbol{\beta}_{(k)_j} &= \mathbf{M}^{-1}\boldsymbol{\delta}^{\mathsf{T}}\widehat{\boldsymbol{\Sigma}}^s[, j] \\
\boldsymbol{\tau}^2_{(k+1)_j} &= \widehat{\boldsymbol{\Sigma}}^s_{jj} - \widehat{\boldsymbol{\Sigma}}^s[j,]\boldsymbol{\delta}\boldsymbol{\beta}_{(k)_j}.
\end{aligned}$$

2.2.3 Example 3. Censored Linear Model.

$$y_i = \beta_0 + \boldsymbol{\beta}^{\mathsf{T}}\mathbf{x}_i + \varepsilon_i,\ \varepsilon_i \sim \mathcal{N}(0, \sigma^2).$$

Only observed y_i when $y_i < C$, where C is known. Let

$$z_i = \begin{cases} y_i, & \text{if } y_i < C \\ C, & \text{if } y_i \geq C. \end{cases}$$

The observed data are $\{(\mathbf{x}_i, z_i)\}$. Without loss of generality, let

$$z_1 = y_1, \ldots, z_m = y_m,$$
$$z_{m+1} = C, \ldots, z_n = C.$$

Therefore, $y_1, \ldots, y_m$ are observed and non-random, and $y_{m+1}, \ldots, y_n$ are unobserved and random. Treat $y_{m+1}, \ldots, y_n$ as "missing data". The complete log-likelihood is

$$\ell(\boldsymbol{\theta}; \text{complete}) = -\frac{n}{2}\log\sigma^2 - \frac{1}{2\sigma^2}\sum_{i=1}^{n}(y_i - \beta_0 - \mathbf{x}_i^{\mathrm{T}}\boldsymbol{\beta})^2.$$

- **E-step.**

$$\begin{aligned}
Q(\boldsymbol{\theta}|\boldsymbol{\theta}_t) &= -\frac{n}{2}\log\sigma^2 - \frac{1}{2\sigma^2}\mathbb{E}_{Y|\mathbf{X},\boldsymbol{\theta}_t}\left\{\sum_{i=1}^{n}(y_i - \beta_0 - \mathbf{x}_i^{\mathrm{T}}\boldsymbol{\beta})^2 \;\middle|\; \begin{array}{l} y_1, \ldots, y_m \text{ are non} - \text{random} \\ y_{m+1} \geq C, \ldots, y_n \geq C \\ \boldsymbol{\theta}_t \end{array}\right\} \\
&= -\frac{n}{2}\log\sigma^2 - \frac{1}{2\sigma^2}\sum_{i=1}^{m}(y_i - \beta_0 - \mathbf{x}_i^{\mathrm{T}}\boldsymbol{\beta})^2 \\
&\quad - \frac{1}{2\sigma^2}\sum_{i=m+1}^{n}\mathbb{E}_{Y|\mathbf{X},\boldsymbol{\theta}_t}\left[(y_i - \beta_0 - \mathbf{x}_i^{\mathrm{T}}\boldsymbol{\beta})^2\,|y_i \geq C,\ \boldsymbol{\theta}_t\right]
\end{aligned} \tag{7}$$

In the above equation, $\sum_{i=1}^{m}(y_i - \beta_0 - \mathbf{x}_i^{\mathrm{T}}\boldsymbol{\beta})^2 = \mathbb{E}_{Y|\mathbf{X},\boldsymbol{\theta}_t}\left[\sum_{i=1}^{m}(y_i - \beta_0 - \mathbf{x}_i^{\mathrm{T}}\boldsymbol{\beta})^2\right]$ since $y_1, \ldots, y_m$ are observed and non-random. Note that

$$\begin{aligned}
&\mathbb{E}\left[(y_i - \beta_0 - \mathbf{x}_i^{\mathrm{T}}\boldsymbol{\beta})^2\,|y_i \geq C,\ \boldsymbol{\theta}_t\right] \\
&= \mathbb{E}\left[y_i^2|y_i \geq C,\ \boldsymbol{\theta}_t\right] - 2\mathbb{E}(y_i|y_i \geq C,\ \boldsymbol{\theta}_t)(\beta_0 + \mathbf{x}_i^{\mathrm{T}}\boldsymbol{\beta}) + (\beta_0 + \mathbf{x}_i^{\mathrm{T}}\boldsymbol{\beta})^2
\end{aligned} \tag{8}$$

We know that

$$\mathbb{E}(y_i^2|y_i \geq C,\ \boldsymbol{\theta}_t) = [\mathbb{E}(y_i|y_i \geq C,\ \boldsymbol{\theta}_t)]^2 + \mathrm{Var}(y_i|y_i \geq C,\ \boldsymbol{\theta}_t) = [\mathbb{E}(y_i|y_i \geq C,\ \boldsymbol{\theta}_t)]^2 + \text{const}, \tag{9}$$

Let $\tilde{y}_i = y_i$ for $i \leq m$ and $\tilde{y}_i = \mathbb{E}(\tilde{y}_i|y_i \geq C, \boldsymbol{\theta}_t)$ for $i \geq m+1$. Plug in (9) into (8), we can show that

$$\begin{aligned}
&\mathbb{E}\left[(y_i - \beta_0 - \mathbf{x}_i^{\mathrm{T}}\boldsymbol{\beta})^2 \,|y_i \geq C,\ \boldsymbol{\theta}_t\right] \\
&= \left[\tilde{y}_i^2 - 2\tilde{y}_i(\beta_0 + \mathbf{x}_i^{\mathrm{T}}\boldsymbol{\beta}) + (\beta_0 + \mathbf{x}_i^{\mathrm{T}}\boldsymbol{\beta})^2 + \text{const}\right] \mid (y_i \geq C,\ \boldsymbol{\theta}_t) \qquad (10) \\
&= \left[(\tilde{y}_i - \beta_0 - \mathbf{x}_i^{\mathrm{T}}\boldsymbol{\beta})^2 + \text{const}\right] \mid (y_i \geq C,\ \boldsymbol{\theta}_t)
\end{aligned}$$

Then it follows from (7) and (10) that

$$\begin{aligned}
Q(\boldsymbol{\theta}|\boldsymbol{\theta}_t) &= -\frac{n}{2}\log\sigma^2 - \frac{1}{2\sigma^2}\sum_{i=1}^{m}(y_i - \beta_0 - \mathbf{x}_i^{\mathrm{T}}\boldsymbol{\beta})^2 - \frac{1}{2\sigma^2}\sum_{i\geq m+1}(\tilde{y}_i - \beta_0 - \mathbf{x}_i^{\mathrm{T}}\boldsymbol{\beta})^2 + \text{const} \\
&= -\frac{n}{2}\log\sigma^2 - \sum_{i=1}^{n}(\tilde{y}_i - \beta_0 - \mathbf{x}_i^{\mathrm{T}}\boldsymbol{\beta})^2 + \text{const},
\end{aligned}$$

- **M-step.**

To compute $\boldsymbol{\beta}_t$.

$$\begin{aligned}
\boldsymbol{\beta}_{t+1} &= \underset{(\boldsymbol{\beta},\beta_0)}{\arg\max} Q(\boldsymbol{\beta}, \beta_0 \mid \boldsymbol{\beta}_t, \beta_{0t}), \\
&= \underset{(\boldsymbol{\beta},\beta_0)}{\arg\max} \sum_{i=1}^{n}(\tilde{y}_i - \beta_0 - \mathbf{x}_i^{\mathrm{T}}\boldsymbol{\beta})^2,
\end{aligned}$$

where $\tilde{y}_i = y_i$ for $i \leq m$ and $\tilde{y}_i = \mathbb{E}(\tilde{y}_i|y_i \geq C, \boldsymbol{\theta}_t)$ for $i \geq m+1$.

After $\boldsymbol{\beta}_{t+1}$, then update

$$\sigma_{t+1}^2 = \frac{1}{n}\sum_{i=1}^{n}(\tilde{y}_i - (\beta_0)_{t+1} - \mathbf{x}_i^{\mathrm{T}}\boldsymbol{\beta}_{t+1})^2$$