STAT 7650 - Computational Statistics Lecture Slides

EM Algorithm

Elvan Ceyhan

Updated: February, 2025

AU

- Based on parts of: Chapter 4 in Givens & Hoeting (Computational Statistics), and Chapter 13 of Lange (Numerical Analysis for Statisticians).

Outline

Notion of "Missing Data"

- Let **X** denote the observable data and θ the parameter to be estimated.
- The EM algorithm is particularly suited for problems in which there is a notion of "missing data".
- The missing data can be actual data that is missing, or some "imaginary" data that exists only in our minds (and necessarily missing).
- The point is that **IF** the missing data were available, then finding the MLE for θ would be relatively straightforward.

Notation

- **Observable Data:** Again, $\mathbf{X}$ is the observable data.
- **Complete Data:** Let $\mathbf{Y}$ denote the complete data[1].
- Usually, we think of $\mathbf{Y}$ as being composed of observable data $\mathbf{X}$ and missing data $\mathbf{Z}$, that is, $\mathbf{Y} = (\mathbf{X}, \mathbf{Z})$.
- Perhaps, more generally, we think of the observable data $\mathbf{X}$ as a sort of projection of the complete data, i.e., $\mathbf{X} = M(\mathbf{Y})$.
- This suggests a notion of **marginalization**...
- The basic idea behind the EM algorithm is to iteratively impute the missing data.

[1] This is the notation used in G&H which, as they admit, is not standard in the EM literature.

Example: Mixture Model

- Consider an example where $\mathbf{X} = (X_1, \ldots, X_n)$ consists of i.i.d samples from the mixture:

$$\pi N(\mu_1, 1) + (1 - \pi) N(\mu_2, 1),$$

where $\boldsymbol{\theta} = (\pi, \mu_1, \mu_2)$ is to be estimated.

- **Missing Data:** If we knew which of the two groups X_i was from, estimating θ would be straightforward—simply calculate the group means.
- The missing part $\mathbf{Z} = (Z_1, \ldots, Z_n)$ represents the group label, where:

$$Z_i = \begin{cases} 1 & \text{if } X_i \sim N(\mu_1, 1) \\ 0 & \text{if } X_i \sim N(\mu_2, 1) \end{cases}$$

- Here, the "missing data" is not real but hypothetical, helping in the estimation process.

Outline

More Notation

- **Complete Data:** $\mathbf{Y} = (\mathbf{X}, \mathbf{Z})$ — Splits into the observed data $\mathbf{X}$ and missing data $\mathbf{Z}$.
- **Complete Data Likelihood:** $\theta \mapsto L_{\mathbf{Y}}(\theta)$ — The joint distribution of $(\mathbf{X}, \mathbf{Z})$: $L_{\mathbf{Y}}(\theta) = f_{\mathbf{Y}}(\mathbf{y}) = f_{\mathbf{X},\mathbf{Z}}(\mathbf{x}, \mathbf{z})$.
- **Observed Likelihood:** $\theta \mapsto L_{\mathbf{X}}(\theta)$ — Obtained by *marginalizing* the joint distribution of $(\mathbf{X}, \mathbf{Z})$: $L_{\mathbf{X}}(\theta) = f_{\mathbf{X}}(\mathbf{x}) = \int f_{\mathbf{X},\mathbf{Z}}(\mathbf{x}, \mathbf{z}) d\mathbf{z}$.
- **Conditional Distribution of Z, Given X:** $\theta \mapsto L_{\mathbf{Z}|\mathbf{X}}(\theta)$ — An essential piece for understanding the relationship between observed and missing data: $L_{\mathbf{Z}|\mathbf{X}}(\theta) = f_{\mathbf{Z}|\mathbf{X}}(\mathbf{z}) = \frac{f_{\mathbf{X},\mathbf{Z}}(\mathbf{x},\mathbf{z})}{f_{\mathbf{X}}(\mathbf{x})}$.
- Although the same notation "L" is used for all the likelihoods, it's crucial to recognize that these represent distinct functions of $\boldsymbol{\theta}$.

Mixture Model Example (Cont'd)

- **Complete Data Y** $= (Y_1, \ldots, Y_n)$, where each Y_i consists of the observed data X_i with the missing group label Z_i.
- **Observed Data Likelihood**:

$$L_{\mathbf{X}}(\boldsymbol{\theta}) = \prod_{i=1}^{n} \left[\pi N(X_i|\mu_1, 1) + (1-\pi)N(X_i|\mu_2, 1)\right],$$

  not a nice function—the sum is inside the product.
- **Complete Data Likelihood** is much nicer — Letting $\pi_1 = \pi$ and $\pi_2 = 1 - \pi$, we have

$$L_{\mathbf{Y}}(\boldsymbol{\theta}) = \prod_{k=1}^{2} \prod_{i=1}^{n} \left(\pi_k N(X_i|\mu_k, 1)\right)^{I(Z_i=k)}.$$

- **Conditional Distribution of Z, Given X**, is determined by the conditional probabilities:

$$P_{\boldsymbol{\theta}}(Z_i = 1|X_i) = \frac{\pi N(X_i|\mu_1, 1)}{\pi N(X_i|\mu_1, 1) + (1-\pi)N(X_i|\mu_2, 1)}.$$

EM Formulation

- In general, the EM algorithm works with a specific function:

$$Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) = \mathbf{E}_{\boldsymbol{\theta}^{(t)}}[\log L_{\mathbf{Y}}(\boldsymbol{\theta})|\mathbf{X}],$$

  the conditional expectation of the complete data log likelihood, at $\boldsymbol{\theta}$, given $\mathbf{X}$ and the value at t-th iteration $\boldsymbol{\theta}^{(t)}$ of the EM Algorithm.
- Implicit in this expression is that, given $\mathbf{X}$, the only "random" part of $\mathbf{Y}$ is the missing data $\mathbf{Z}$.
- Thus, in this expression, the expectation is actually with respect to $\mathbf{Z}$, given $\mathbf{X}$, i.e.,

$$Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) = \mathbf{E}_{\mathbf{Z}|\mathbf{X}}[\log L_{\mathbf{Y}}(\boldsymbol{\theta})|\mathbf{X}] = \int \log L_{(\mathbf{X},\mathbf{z})}(\boldsymbol{\theta}) L_{\mathbf{z}|\mathbf{X}}(\boldsymbol{\theta}^{(t)}) d\mathbf{z}.$$

EM Formulation (Cont'd)

The EM algorithm iterates computing $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})$, which involves an **E**xpectation, and then **M**aximizing it.

Procedure

1. **Start** with an initial estimate $\boldsymbol{\theta}^{(0)}$.
2. **At iteration** $t = 1, 2, \ldots$ do:
   - **E-step:** Evaluate $Q_t(\boldsymbol{\theta}) := Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t-1)})$.
   - **M-step:** Update $\boldsymbol{\theta}^{(t)} = \arg\max_{\boldsymbol{\theta}} Q_t(\boldsymbol{\theta})$.
3. **Repeat** these steps until practical convergence is reached.

- **Goal:** Maximize the observed data likelihood.
- But EM iteratively maximizes some other function, so it's not clear that we are doing something reasonable.
- Before we get to theory, it helps to consider a few simple examples to see that EM is doing the right thing.

Trivial Example - 1:

- Let $Y_1, Y_2 \overset{iid}{\sim} \text{Exp}(\theta)$ with $y_1 = 5$ observed but y_2 missing.
- The complete data log likelihood function is

$$\log L(\theta|\mathbf{y}) = \log f_{\mathbf{Y}}(\mathbf{y}|\theta) = 2\log\theta - \theta y_1 - \theta y_2.$$

- So,

$$Q(\theta|\theta^{(t)}) = 2\log\theta - 5\theta - \theta/\theta^{(t)}$$

since $\mathbf{E}[Y_2|y_1, \theta^{(t)}] = \mathbf{E}[Y_2|\theta^{(t)}] = 1/\theta^{(t)}$ follows from independence.

- The maximizer of $Q(\theta|\theta^{(t)})$ is the root of $Q'(\theta|\theta^{(t)}) = 2/\theta - 5 - 1/\theta^{(t)} = 0$.
- Thus,

$$\theta^{(t+1)} = \frac{2\theta^{(t)}}{5\theta^{(t)} + 1}.$$

Converges quickly to $\hat{\theta} = 0.2$.

Trivial Example - 1 - Comments

- The *E-step* and *M-step* do not need to be re-derived at each iteration!
- This example is not realistic. An easy analytic solution exists (how?).
- Taking the required expectation is trickier in real applications because one needs to know the conditional distribution of the missing data given the observed data.

Trivial Example - 2

- **Example:**
  - Let $\mathbf{Y} = (X, Z)$, where X, Z are i.i.d. from $N(\theta, 1)$, but Z is missing.
  - Observed data MLE $\hat{\theta} = X$.
  - The Q function in the E-step is given by:

$$Q(\theta|\theta^{(t)}) = -\frac{1}{2}\left[(\theta - X)^2 + (\theta - \theta^{(t)})^2\right].$$

- Find the **M-step Update** — what should happen as $t \to \infty$?

Outline

The Nature of EM

Ascent Property: Each M step increases observed-data the log likelihood.

Convergence: Linear (slow!). Rate is inversely related to the proportion of missing data.

Optimization transfer:

$$\ell(\boldsymbol{\theta}|\mathbf{x}) \geq Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) + \ell(\boldsymbol{\theta}^{(t)}|\mathbf{x}) - Q(\boldsymbol{\theta}^{(t)}|\boldsymbol{\theta}^{(t)}) = G(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}).$$

The last two terms in $G(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})$ are constant with respect to $\boldsymbol{\theta}$, so Q and G are maximized at the same $\boldsymbol{\theta}$. Further, G is tangent to ℓ at $\boldsymbol{\theta}^{(t)}$, and lies everywhere below ℓ. We say that G is a *minorizing function* for ℓ. At each iteration, EM transfers optimization from ℓ to the surrogate function G, which is more convenient to maximize.

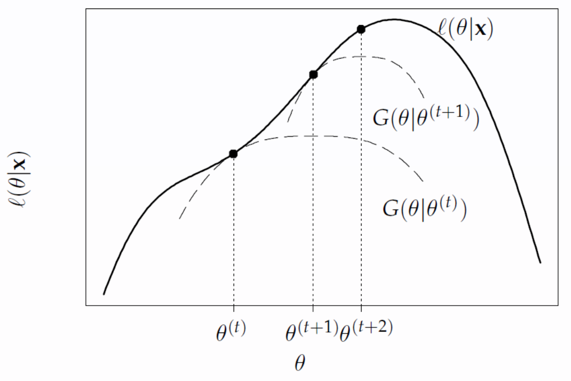


Figure 1: One-dimensional illustration of EM algorithm as a minorization or optimization transfer strategy.

Each E-step forms a minorizing function G, and each M-step maximizes it to provide an uphill step.

Convergence of EM Algorithm

Objective: Investigate the convergence properties of the Expectation-Maximization (EM) algorithm.

- Each maximization step increases the observed-data log likelihood, $\ell(\boldsymbol{\theta}|\mathbf{x})$.
- The log of the observed-data density can be rewritten as:

$$\log f_{\mathbf{X}}(\mathbf{x}|\boldsymbol{\theta}) = \log f_{\mathbf{Y}}(\mathbf{y}|\boldsymbol{\theta}) - \log f_{\mathbf{Z}|\mathbf{X}}(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta}). \tag{1}$$

- Taking expectation w.r.t. the conditional distribution of $\mathbf{Z}|(\mathbf{x}, \boldsymbol{\theta}^{(t)})$:

$$\mathbf{E}[\log f_{\mathbf{X}}(\mathbf{x}|\boldsymbol{\theta})|\mathbf{x}, \boldsymbol{\theta}^{(t)}] = Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) - H(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}), \tag{2}$$

  where

$$H(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) = \mathbf{E}[\log f_{\mathbf{Z}|\mathbf{X}}(\mathbf{Z}|\mathbf{x}, \boldsymbol{\theta})|\mathbf{x}, \boldsymbol{\theta}^{(t)}]. \tag{3}$$

Maximization of $H(\theta|\theta^{(t)})$

Key Observation: $H(\theta|\theta^{(t)})$ is maximized at $\boldsymbol{\theta} = \boldsymbol{\theta}^{(t)}$.
Proof:

$$
\begin{aligned}
H(\boldsymbol{\theta}^{(t)}|\boldsymbol{\theta}^{(t)}) - H(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) = \\
& \mathbf{E}\left[\log f_{\mathbf{Z}|\mathbf{X}}(\mathbf{Z}|\mathbf{x}, \boldsymbol{\theta}^{(t)}) - \log f_{\mathbf{Z}|\mathbf{X}}(\mathbf{Z}|\mathbf{x}, \boldsymbol{\theta}) \,\middle|\, \mathbf{x}, \boldsymbol{\theta}^{(t)}\right] \\
&= \int \left[-\log \frac{f_{\mathbf{Z}|\mathbf{X}}(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta})}{f_{\mathbf{Z}|\mathbf{X}}(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta}^{(t)})}\right] f_{\mathbf{Z}|\mathbf{X}}(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta}^{(t)}) d\mathbf{z} \\
&\geq -\log \int f_{\mathbf{Z}|\mathbf{X}}(\mathbf{z}|\mathbf{x}, \boldsymbol{\theta}^{(t)}) d\mathbf{z} \\
&= 0.
\end{aligned}
$$

Thus,

$$
H(\boldsymbol{\theta}^{(t)}|\boldsymbol{\theta}^{(t)}) - H(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) \geq 0. \tag{4}
$$

This inequality follows from an application of Jensen's inequality, since $-\log u$ is strictly convex in u.

Implication of $H(\theta|\theta^{(t)})$ Maximization

Key Result: $H(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})$ is maximized at $\boldsymbol{\theta} = \boldsymbol{\theta}^{(t)}$.

Implication: For any $\boldsymbol{\theta} \neq \boldsymbol{\theta}^{(t)}$, we have:

$$H(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) < H(\boldsymbol{\theta}^{(t)}|\boldsymbol{\theta}^{(t)}).$$

Guaranteed Ascent in EM:

- Since $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})$ is maximized at $\boldsymbol{\theta} = \boldsymbol{\theta}^{(t+1)}$, we obtain:

$$\log f_{\mathbf{X}}(\mathbf{x}|\boldsymbol{\theta}^{(t+1)}) - \log f_{\mathbf{X}}(\mathbf{x}|\boldsymbol{\theta}^{(t)}) \geq 0.$$

- This follows because $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})$ increases and $H(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})$ decreases at each step.
- If $Q(\boldsymbol{\theta}^{(t+1)}|\boldsymbol{\theta}^{(t)}) > Q(\boldsymbol{\theta}^{(t)}|\boldsymbol{\theta}^{(t)})$, the inequality is strict.

Conclusion: Each EM iteration ensures a non-decreasing log-likelihood, establishing its ascent property.

Derivation of Optimization Transfer in EM - I

Recall Equation (2), where the function $H(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})$ captures the expected log-conditional distribution of the missing data.

Also, Equation (4) is

$$H(\boldsymbol{\theta}^{(t)} \mid \boldsymbol{\theta}^{(t)}) - H(\boldsymbol{\theta} \mid \boldsymbol{\theta}^{(t)}) \geq 0.$$

which implies that $H(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})$ is maximized at $\boldsymbol{\theta} = \boldsymbol{\theta}^{(t)}$.

Derivation of Optimization Transfer in EM - II

Manipulating the Expression for $\log f_{\mathbf{X}}(\mathbf{x}|\theta)$

$$
\begin{aligned}
\log f_{\mathbf{X}}(\mathbf{x}|\theta) &= Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) - H(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) \\
&= Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) - H(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) + H(\boldsymbol{\theta}^{(t)}|\boldsymbol{\theta}^{(t)}) - H(\boldsymbol{\theta}^{(t)}|\boldsymbol{\theta}^{(t)}) \\
&= Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) - \Big[H(\boldsymbol{\theta}^{(t)}|\boldsymbol{\theta}^{(t)}) - (H(\boldsymbol{\theta}^{(t)}|\boldsymbol{\theta}^{(t)}) - H(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}))\Big].
\end{aligned}
$$

Applying the Jensen Bound:

$$H(\boldsymbol{\theta}^{(t)}|\boldsymbol{\theta}^{(t)}) - H(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) \geq 0.$$

Thus, we obtain:

$$\log f_{\mathbf{X}}(\mathbf{x}|\theta) \geq Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) - H(\boldsymbol{\theta}^{(t)}|\boldsymbol{\theta}^{(t)}). \qquad (5)$$

Derivation of Optimization Transfer in EM - III

From:

$$\log f_{\mathbf{X}}(\mathbf{x}|\theta^{(t)}) = Q(\theta^{(t)}|\theta^{(t)}) - H(\theta^{(t)}|\theta^{(t)}),$$

we obtain:

$$H(\theta^{(t)}|\theta^{(t)}) = Q(\theta^{(t)}|\theta^{(t)}) - \log f_{\mathbf{X}}(\mathbf{x}|\theta^{(t)}).$$

Substituting this into the previous result in Equation (5):

$$\log f_{\mathbf{X}}(\mathbf{x}|\theta) \geq Q(\theta|\theta^{(t)}) + \log f_{\mathbf{X}}(\mathbf{x}|\theta^{(t)}) - Q(\theta^{(t)}|\theta^{(t)}).$$

Further Properties

- EM updates can be expressed through an abstract mapping, Ψ, i.e., $\theta^{(t+1)} = \Psi(\theta^{(t)})$.
- If EM converges to $\hat{\theta}$, then $\hat{\theta}$ must be a fixed-point of Ψ.
- Do a **Taylor approximation** of $\Psi(\theta^{(t)})$ around $\hat{\theta}$:

$$\Psi(\theta^{(t)}) \approx \Psi(\hat{\theta}) + \Psi'(\hat{\theta})(\theta^{(t)} - \hat{\theta})$$

- Hence,

$$\Psi(\theta^{(t)}) - \Psi(\hat{\theta}) \approx \Psi'(\hat{\theta})(\theta^{(t)} - \hat{\theta})$$
$$\implies \theta^{(t+1)} - \hat{\theta} \approx \Psi'(\hat{\theta})(\theta^{(t)} - \hat{\theta})$$

- If the parameter is one-dimensional, then the convergence order can be seen to be linear, provided that $\hat{\theta}$ is a (local) maximum.

Further (Asymptotic) Properties

Asymptotic Normality: If the model is correctly specified and certain regularity conditions are met, the maximum likelihood estimates obtained by the EM algorithm are asymptotically normal. That is, as the sample size n approaches infinity, the distribution of the estimate around the true parameter value follows a normal distribution. This result is similar to the asymptotic normality of maximum likelihood estimators more generally.

Asymptotic Efficiency: Under suitable conditions, the EM estimates are asymptotically efficient, meaning that they achieve the Cramér-Rao lower bound. This implies that the estimates have the smallest possible variance among all unbiased estimators, at least asymptotically.

EM for Exponential Family Models

- Recall that a model/joint distribution $f_{\boldsymbol{\theta}}$ for data $\mathbf{Y}$ is a natural exponential family if the log-likelihood is of the form:

$$\log L_{\mathbf{Y}}(\boldsymbol{\theta}) = \text{constant} + \log a(\boldsymbol{\theta}) + \boldsymbol{\theta}^{\top} s(\mathbf{y}),$$

  where $s(\mathbf{y})$ is the "sufficient statistic."
- For problems where the complete data $\mathbf{Y}$ is modeled as an exponential family, EM takes a relatively simple form.
- This is an important case since many examples involve exponential families, simplifying the implementation of EM and interpretation of its results.

EM for Exponential Family Models (Cont'd)

For exponential families, the Q function is expressed as:

$$Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) = \text{constant} + \log a(\boldsymbol{\theta}) + \int \boldsymbol{\theta}^T s(\mathbf{y}) L_{\mathbf{z}|\mathbf{X}}(\boldsymbol{\theta}^{(t)}) d\mathbf{z}.$$

- To maximize this, take derivative w.r.t. $\boldsymbol{\theta}$ and set to zero:

$$\Rightarrow -\frac{a'(\boldsymbol{\theta})}{a(\boldsymbol{\theta})} = \int s(\mathbf{y}) L_{\mathbf{z}|\mathbf{X}}(\boldsymbol{\theta}^{(t)}) d\mathbf{z}.$$

- From STAT 7600, you know that the left-hand side is $\mathbf{E}_{\boldsymbol{\theta}}[s(\mathbf{Y})]$.
- Let $s^{(t)}$ be the right-hand side.
- M-step updates $\boldsymbol{\theta}^{(t)} \to \boldsymbol{\theta}^{(t+1)}$ by solving the equation:

$$\mathbf{E}_{\boldsymbol{\theta}}[s(\mathbf{Y})] = s^{(t)}.$$

EM for Exponential Family Models (Cont'd)

E-step
Compute $s^{(t)}$ based on guess $\boldsymbol{\theta}^{(t)}$.

M-step
Update guess to $\boldsymbol{\theta}^{(t+1)}$ by solving the equation

$$\mathbf{E}_{\boldsymbol{\theta}}[s(\mathbf{Y})] = s^{(t)}.$$

Outline

Example 1: Censored Exponential Model

- **Complete Data** $Y_1, \ldots, Y_n \overset{iid}{\sim} \text{Exp}(\theta)$, with the rate parameter.
- **Complete Data Log-Likelihood:**

$$\log L_{\mathbf{Y}}(\theta) = n \log \theta - \theta \underbrace{\sum_{i=1}^{n} Y_i}_{S(\mathbf{Y})}.$$

- **Censoring:** Suppose some observations are right-censored, i.e., only a lower bound is observed.
- Write the observed data as pairs (X_i, δ_i), where

$$X_i = \min(Y_i, c_i), \quad \text{and} \quad \delta_i = \mathbf{I}_{\{X_i = Y_i\}}$$

where c_i's are non-random censoring thresholds.
- **Missing Data Z:** Consists of the actual event times for the censored observations.

Example 1: Censored Exponential Model (Cont'd)

- For the EM algorithm, we focus on censored cases for computing $s^{(t)}$.
- **Observations:** If an observation Y_i is right-censored at c_i, then c_i is a lower bound.
- Recall the **memory-less property of exponential:** This property is crucial for the E-step computation.
- **E-step Computation:**

$$s^{(t)} = \sum_{i=1}^{n} \left[\delta_i X_i + (1-\delta_i) E_{\theta^{(t)}}[Y_i | \text{censored}]\right],$$

which simplifies to

$$= \sum_{i=1}^{n} \left[\delta_i X_i + (1-\delta_i)(X_i + \frac{1}{\theta^{(t)}})\right] = n\bar{X} + \frac{1}{\theta^{(t)}} \sum_{i=1}^{n} (1-\delta_i).$$

Example 1: Censored Exponential Model (Cont'd)

M-step Computation

Given that $E_\theta[s(\mathbf{Y})] = n/\theta$, the M-step requires solving for θ in

$$\frac{n}{\theta} = n\bar{X} + \frac{1}{\theta^{(t)}} \sum_{i=1}^{n} (1 - \delta_i).$$

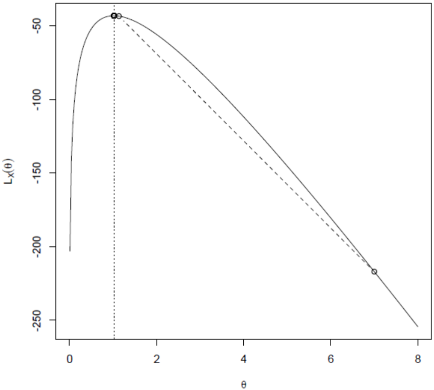
In particular, the **EM Update Formula** in this case is

$$\theta^{(t+1)} = \frac{n}{n\bar{X} + \frac{1}{\theta^{(t)}} \sum_{i=1}^{n} (1 - \delta_i)}.$$

Iterate this update until convergence is achieved.

Example 1: Censored Exponential Model (Cont'd)

- **Simulated Data:** Number of observations $n = 30$, rate parameter $\theta = 3$, and censored at 0.632.
- **EM Algorithm Initialization:** Starting point for $\theta^{(0)} = 7$.
- The picture below illustrates the observed data likelihood and the EM steps.



Example 2: Probit Regression

- Recall the **probit regression model**: $X_i \sim \text{Ber}(\Phi(\mathbf{u}_i^T \boldsymbol{\theta}))$.
- The EM algorithm facilitates obtaining the MLE of $\boldsymbol{\theta}$.
- **Complete Data Representation**: $\mathbf{Y} = (Y_1, \ldots, Y_n)$, where $Y_i \sim N(\mathbf{u}_i^T \boldsymbol{\theta}, 1)$, and X_i is defined as:

$$X_i = \begin{cases} 1 & \text{if } Y_i > 0 \\ 0 & \text{if } Y_i \leq 0 \end{cases}$$

- **Exercise**: Verify that X_i, defined in this way, has the same distribution as that given by the probit model.
- Essentially, we observe the sign of the complete data, but the actual values are missing.

Example 2: Probit Regression (Cont'd)

- The complete-data problem is easy, just a normal linear regression with known variance—exponential family.
- $s(\mathbf{Y}) = U^T \mathbf{Y}$, where U is the design matrix.
- Observed data provides the sign of Y_i, leading to the conditional expectation of a truncated normal distribution[2]:

$$\mathbf{E}_{\boldsymbol{\theta}^{(t)}}(Y_i | X_i) = \mu_i^{(t)} + w_i \frac{\phi(\mu_i^{(t)})}{\Phi(w_i \mu_i^{(t)})} = \mu_i^{(t)} + v_i^{(t)},$$

where $\mu_i^{(t)} = \mathbf{u}_i^T \boldsymbol{\theta}^{(t)}$, $w_i = 2X_i - 1$, and $v_i^{(t)} = w_i \frac{\phi(\mu_i^{(t)})}{\Phi(w_i \mu_i^{(t)})}$.

- This completes the **E-step**; **M-step** requires solving:

$$\underbrace{U^T U \boldsymbol{\theta}}_{\mathbf{E}_\theta[s(\mathbf{Y})]} = \underbrace{U^T U \boldsymbol{\theta}^{(t)} + U^T \mathbf{v}^{(t)}}_{s^{(t)}},$$

[2] https://en.wikipedia.org/wiki/Truncated_normal_distribution

Example 2: Probit Regression (Recap)

Complete-Data Problem

- In this example the complete-data problem is treated as a normal linear regression problem with known variance, falling into the exponential family of distributions.
- Because in this case, the relationship between Y_i and the covariates $\mathbf{u}_i$ is linear, with Y_i having a normal distribution centered around $\mathbf{u}_i^T \boldsymbol{\theta}$ with variance 1.

Sufficient Statistic $s(\mathbf{Y})$

- The sufficient statistic for this complete-data problem is $s(\mathbf{Y}) = U^T \mathbf{Y}$, where U is the design matrix comprising all covariate vectors $\mathbf{u}_i$ as its rows.
- This follows from the log-likelihood function for a normal distribution, which involves the sum of the product of observed values and their corresponding model-predicted values.

Example 2: Probit Regression (Recap)

Observed Data and Conditional Expectation

- Given that the observed data only provide the sign of Y_i, the EM algorithm computes the conditional expectation of Y_i given X_i, corresponding to the expectation of a truncated normal distribution.
- This conditional expectation is $\mathbf{E}_{\boldsymbol{\theta}^{(t)}}(Y_i|X_i) = \mu_i^{(t)} + v_i^{(t)}$, where:
  - $\mu_i^{(t)} = \mathbf{u}_i^T \boldsymbol{\theta}^{(t)}$ is the mean of Y_i under the current estimate $\boldsymbol{\theta}^{(t)}$,
  - $w_i = 2X_i - 1$ adjusts the direction of the adjustment based on whether X_i is 0 or 1,
  - $v_i^{(t)} = w_i \frac{\phi(\mu_i^{(t)})}{\Phi(w_i \mu_i^{(t)})}$ represents the adjustment based on the standard normal distribution's density and cdfs.

Example 2: Probit Regression (Recap)

E-step and M-step

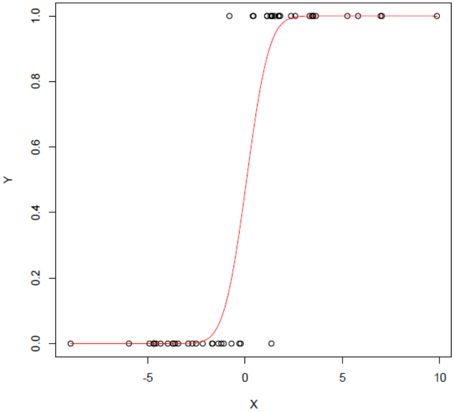
- **E-step**: Computes the conditional expectation of the latent variable Y_i given the observed data X_i and the current estimate of $\boldsymbol{\theta}$.
- **M-step**: Updates the estimate of $\boldsymbol{\theta}$ by solving the equation:

$$U^T U\boldsymbol{\theta} = U^T U\boldsymbol{\theta}^{(t)} + U^T \mathbf{v}^{(t)},$$

where $\mathbf{v}^{(t)}$ is the vector of adjustments for each observation.

Example 2: Probit Regression (Cont'd)

- **Simulated Data:** Number of observations: $n = 50$, intercept $\theta_1 = 0$, slope $\theta_2 = 1$, and predictor variables are i.i.d $N(0, 4^2)$.
- The plot below illustrates the observed data alongside the EM fitted probit regression line.



Example 3: Robust Regression

- Consider the linear model $y_i = \mathbf{x}_i^T \boldsymbol{\beta} + \varepsilon_i$ where y_i is the response, $\mathbf{x}_i$ is the predictor vector, $\boldsymbol{\beta}$ represents coefficients, and $\varepsilon_i \overset{iid}{\sim} N(0, \sigma^2)$ denotes model errors.
- Least-squares estimators are sensitive (i.e. not robust) to "outlier" observations due to fitting at the mean $\mathbf{x}_i^T \boldsymbol{\beta}$ (which undermines the assumption of normal errors).
- **Remedy:** Fit a model with heavier-than-normal tails, such as modeling ε with a Student-t distribution with a small number of degrees of freedom (i.e., $\varepsilon_i \overset{iid}{\sim} t_\nu$ with small ν).
- This model can be fitted using standard optimization tools, but a clever application of EM significantly simplifies the process.
- **Key Observation:** The Student-t distribution is a scale mixture of normals:

$$f(\varepsilon) = \int N(\varepsilon | 0, \sigma^2 / z) \text{ ChiSq}(z | \nu) dz.$$

Example 3: Robust Regression (Cont'd)

- For simplicity, we assume $\nu = df$ (degrees of freedom) is known.
- The Z_i values, related to the Student-t error distribution for ε_i (i.e. representing scale factors in the Student-t error model for each observation), are considered as "missing data".
- **If we knew $\mathbf{Z} = (Z_1, \ldots, Z_n)$**, the problem would essentially be a simple modification of the normal model.
- Define model parameters as $\boldsymbol{\theta} = (\boldsymbol{\beta}, \sigma)$. The complete data log-likelihood is:

$$\log L_{\mathbf{Y}}(\boldsymbol{\theta}) = \sum_{i=1}^{n} \log N(y_i - \mathbf{x}_i^T \boldsymbol{\beta} \big| 0, \nu\sigma^2 / Z_i).$$

- **E-step:** Requires computing the expectation with respect to the conditional distribution of Z, given the data and a guess $\boldsymbol{\theta}^{(t)}$...

Robust Regression - E-step

- It can be shown that **conditional distribution of Z_i, given observed data and $\theta^{(t)}$** is

$$\left[\frac{1}{\nu}\left(\frac{y_i - \mathbf{x}_i^T\beta^{(t)}}{\sigma^{(t)}}\right)^2 + 1\right]^{-1} \times \text{ChiSq}(\nu+1) \quad \text{for } i = 1, \ldots, n.$$

- **Q Function:**

$$Q(\theta|\theta^{(t)}) = -\frac{n}{2}\log\sigma^2 - \frac{1}{2\sigma^2}\sum_{i=1}^{n} w_i^{(t)}(y_i - \mathbf{x}_i^T\beta)^2,$$

where weights $w_i^{(t)}$ are specifically designed to mitigate the impact of outliers by adjusting each observation's influence on the model based on its alignment with current estimates:

$$w_i^{(t)} = (\nu+1)\left(\left(\frac{y_i - \mathbf{x}_i^T\beta^{(t)}}{\sigma^{(t)}}\right)^2 + \nu\right)^{-1}, \quad i = 1, \ldots, n.$$

Robust Regression - M-step: Weighted Least Squares

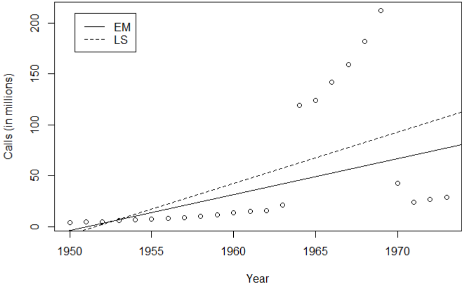
- The M-step is equivalent to solving a weighted least squares problem, where weights are $w_i^{(t)}$ from the E-step.
- This leads to updates for $\boldsymbol{\theta}$ that account for the robustness against outliers by adjusting the influence of each observation.

Insights and Applications

- **Adjusting for Variability:** The conditional distribution of Z_i reflects how individual observations' variability is adjusted in the presence of outliers, directly influencing the robustness of the regression model.
- **Optimization via Weights:** The Q function illustrates the method for incorporating the calculated weights into the optimization process, ensuring that the updated parameter estimates $(\boldsymbol{\theta})$ in the M-step are influenced appropriately by the data's underlying structure.

Belgian Phone Call Data Analysis

- Analysis of Belgian phone call data from `R`'s MASS library.
- **Objective:** Compare the fitting of Least Squares (LS) versus Student-t distribution via the EM algorithm ($df = 4$) for the years 1950 and 1955.
- Notice the robustness of the Student-t model against outliers compared to the traditional LS approach.



Outline

A Challenge in EM Algorithm

Background: The Core of EM Algorithm

- The EM algorithm excels at finding the Maximum Likelihood Estimate (MLE) $\hat{\boldsymbol{\theta}}$ in scenarios complicated by incomplete data or latent variables.
- It does not offer a direct method for estimating the standard errors of the estimated parameters $\hat{\boldsymbol{\theta}}$.

The Challenge

- Recall that if we run, say, `BFGS` via the function `optim` in R, then we can request that the Hessian at the MLE be returned, which can be used to approximate the standard errors of $\hat{\boldsymbol{\theta}}$.
- The challenge is that the EM doesn't work directly with the observed data log-likelihood.

The Question: How do we integrate standard error computation into the EM algorithm's framework to extend its utility in statistical analysis?

Analytical Calculation of Standard Errors

Background: Importance of Standard Errors

- Standard errors measure the variability of parameter estimates.
- They are estimated using the negative second derivative of the log-likelihood, $-\log L_{\mathbf{X}}(\boldsymbol{\theta})$, or the Fisher information, $I(\hat{\boldsymbol{\theta}})$.

Probit Regression Model: Fisher Information

For the probit regression model, the Fisher information is given by:

$$I_n(\boldsymbol{\theta}) = \sum_{i=1}^{n} \frac{\phi(\mathbf{u}_i^T\boldsymbol{\theta})^2}{\Phi(\mathbf{u}_i^T\boldsymbol{\theta})(1-\Phi(\mathbf{u}_i^T\boldsymbol{\theta}))}\mathbf{u}_i\mathbf{u}_i^T.$$

Plugging in our MLE $\hat{\boldsymbol{\theta}}$ from the EM algorithm into this formula allows us to (numerically) invert the matrix to estimate standard errors.

Alternative Approach: Numerical Differentiation

Numerically differentiating $-\log L_{\mathbf{X}}(\boldsymbol{\theta})$ provides a versatile method to estimate standard errors without explicit formulas.

Bootstrap: A Preliminary Overview

Main Idea (more on this later)
The bootstrap method aims to estimate the variance (hence standard error) of $\hat{\theta}$, obtained via the EM algorithm, in situations where we only have a single value of $\hat{\theta}$.

Challenge

- Estimating the variance of $\hat{\theta}$ is difficult with only one sample.
- If multiple samples or copies of $\hat{\theta}$ were available, variance estimation would be straightforward.

Solution: Bootstrap Principle

- Generate multiple copies of $\hat{\theta}$ by resampling (with replacement) from the observed data $\mathbf{X} = (X_1, \ldots, X_n)$, many times.

Bootstrap Method (Cont'd)

Procedure:

1. Choose a large number B for bootstrap samples.
2. For $b = 1, \ldots, B$:
   - Sample $\mathbf{X}_b^* = (X_{b1}^*, \ldots, X_{bn}^*)$ with replacement from the observed data $\mathbf{X} = (X_1, \ldots, X_n)$.
   - Compute $\hat{\boldsymbol{\theta}}_b$ by applying the EM algorithm to $\mathbf{X}_b^*$.
3. Estimate the variance of $\hat{\boldsymbol{\theta}}$ using the sample variance (or covariance) of $\hat{\boldsymbol{\theta}}_1, \ldots, \hat{\boldsymbol{\theta}}_B$.

Considerations:

- The empirical distribution of $\mathbf{X}$ should resemble the true sampling model for large n, motivating the bootstrap approach.
- This method may be computationally intensive in the EM context due to the requirement for B separate EM algorithm runs.

Other (Advanced) Methods for Analyzing the EM Algorithm

Numerical Differentiation for Score Function $\frac{\partial}{\partial \boldsymbol{\theta}} \log L_{\mathbf{X}}(\theta)$, at parameter estimates $\hat{\boldsymbol{\theta}}$, the EM solution or estimate.

Applications:

1. **Standard Error Estimation:** For the precision of parameter estimates, confidence intervals and significance tests.
2. **Sensitivity Analysis:** Evaluates parameter sensitivity to changes, aiding in model specification and diagnostics.
3. **Model Comparison and Selection:** Construction of likelihood ratio tests and information criteria (AIC/BIC).
4. **Gradient-Based Optimization:** Enhances EM algorithm extensions through gradient optimization, improving convergence in complex models.
5. **Robustness Checks:** Verifies the reliability of EM solutions by examining log-likelihood changes around $\hat{\boldsymbol{\theta}}$.

Other (Advanced) Methods for Analyzing the EM Algorithm

Supplemented EM (SEM) for Enhanced Stability

- **Enhancement over EM**: Multiple iterations of EM, each from different starting points, to improve stability.
- **Benefit**: Mitigates risk of local maxima, enhancing reliability.

Louis's Method: Missing Information Principle

- **Theoretical Foundation**: Uses $i_{\mathbf{X}}(\theta) = i_{\mathbf{Y}}(\theta) - i_{\mathbf{Z}|\mathbf{X}}(\theta)$ to partition information into observed and missing.
- **Application**: Complex computation but offers valuable insights for standard errors and confidence intervals.

Empirical Information

- **Simplicity and Practicality**: Uses empirical variance as an estimator for the Fisher information.
- **Advantage**: Straightforward and minimally demanding, enhancing accessibility.

Outline

Problem and Motivation

Definition of the EM Algorithm

Properties of EM

Examples

Estimating Standard Errors

Different Versions of EM

Summary

Considerations in EM Algorithm Design

Computational Challenges:

- In each of the two main steps of the EM algorithm, there are potentially some non-trivial computations involved.
- The **E-step** requires computing an expectation, which often cannot be done analytically.
- The **M-step** involves optimization, which also frequently cannot be solved analytically.

Numerical Approaches:

- Both integration (for the E-step) and optimization (for the M-step) can be performed numerically.
- However, this introduces concerns about **efficiency**, particularly due to the nested loops in these computations.

Efficient Design:

- There are ongoing questions about how to efficiently design EM algorithms to mitigate these computational challenges.

Modifying the E-step

- **E-step Challenge:** Compute an expectation with respect to the conditional distribution of $\mathbf{Z}$, given $\mathbf{X}$.
- In some cases, this boils down to several one-dimensional integrals, which we could possibly do with quadrature.
- **Alternative: Monte Carlo Integration**
  - Replace numerical integration with Monte Carlo simulation (more on this later) to estimate these expectations.
  - Attractive for its general applicability but may increase computational costs, requiring Monte Carlo simulations at every E-step.
  - Adds a Bayesian flavor to the EM algorithm.
- **Acronyms in EM:** In line with the EM community's fondness for acronyms, this approach is known as Monte Carlo EM (MCEM).

Modifying the M-step

Challenge in M-step
To maximize $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})$ w.r.t. $\boldsymbol{\theta}$, especially when an analytical solution is not available.

Numerical Optimization

- If not doable analytically, then consider using numerical optimization routines, though they may be computationally expensive.

Alternative Approaches

- **ECM Algorithm:** Maximize Q one component at a time for more manageable optimization.
- **EM Gradient:** Perform just one iteration of Newton's method at each M-step to gradually approach the maximum.

One Specific Extension: PX-EM

Introduction to PX-EM: Ordinary EM algorithm simplifies computations under the assumption that some "missing data" were known.

Counter-Intuitive Idea: A counter-intuitive approach that involves introducing additional parameters (i.e. expanding and reparametrizing the parameter space) and mapping expanded parameters back to the original space.

This approach is called **PX-EM**, where PX stands for "Parameter Expansion".

Convergence Properties

- The PX-EM algorithm maintains the same ascent property as the traditional EM algorithm.
- Its rate of convergence is guaranteed to be no slower than that of the standard EM algorithm.

PX-EM (Cont'd)

Parameter (Space) Expansion
Treat parameter $\boldsymbol{\theta}$ as a function of additional parameters (ψ, ϕ), with the intuition that the original model corresponds to ϕ being fixed at a specified value ϕ_0, i.e., $\boldsymbol{\theta} = f(\psi, \phi_0)$.

Complete-Data Log Likelihood
Begin with the complete-data log likelihood for (ψ, ϕ), expressed as $\log L_{\mathbf{Y}}(\psi, \phi)$. For exponential families, this likelihood is a linear function of the sufficient statistics for the expanded (ψ, ϕ)-model.

Iterative Process
Proceed with iterative computation of conditional expectation and maximization, similar to traditional EM. However, the PX E-step includes a slight difference, adapting to the expanded model's structure.

PX-EM (Cont'd)

Iteration Process
At iteration t, suppose we have $(\psi^{(t)}, \phi^{(t)})$, which defines $\boldsymbol{\theta}^{(t)}$.

PX E-step
Set $Q(\psi, \phi|\psi^{(t)}, \phi_0)$, the conditional expectation of the complete-data log-likelihood, using ϕ_0 instead of the current guess $\phi^{(t)}$.

PX M-step
Maximize Q to obtain $(\psi^{(t+1)}, \phi^{(t+1)})$ and compute $\boldsymbol{\theta}^{(t+1)} = f(\psi^{(t+1)}, \phi^{(t+1)})$.

Advantage of PX-EM
This version improves the M-step by using extra information from the enlarged model, potentially enhancing convergence properties.

Example 2: Probit Regression with PX-EM

- **Probit Regression Model:** $X_i \sim \text{Ber}(\Phi(\mathbf{u}_i^T \boldsymbol{\theta}))$.
- **Complete Data:** $Y_i \sim N(\mathbf{u}_i^T \boldsymbol{\theta}, 1)$.
- **Parameter Expansion:** Introduce a variance parameter to expand $\boldsymbol{\theta}$, resulting in $Y_i \sim N(\mathbf{u}_i^T \boldsymbol{\theta}, \phi^2)$ with $\phi_0 = 1$.
- **Sufficient Statistics:** For the complete-data model are $s(\mathbf{Y}) = (U^T \mathbf{Y}, \mathbf{Y}^T \mathbf{Y})$.
- **PX E-step:** Utilizes properties of the truncated normal distribution.
- **PX M-step:** Straightforward, akin to the ordinary EM, with enhancements from the parameter expansion.

Note: For implementation details, see the R code.

Outline

Remarks on the EM Algorithm

- EM is a powerful tool for maximizing complex likelihood functions, especially with "missing data".
- Deriving E- and M-steps requires effort but is facilitated by extensive literature.
- Applications include mixture models and censored-data problems.
- *Data augmentation* (ideas of sort of "randomly imputing" missing values) is clever and introduces a Bayesian flavor.
- **Main challenge:** Potentially slow convergence, with several remedies available.
- **Open question:** Is it feasible to parallelize EM?
- The original EM paper has garnered significant attention, indicating its impact.[3]

[3] As of March 2024, the original EM paper (Dempster, Laird, and Rubin, JRSS-B 1977) has been cited over 72,000 times!