Our problem: How to get the lifetime characteristics of components from system lifetimes?

- In reliability study, experimenters are interested in the lifetime distribution of the components as well as the lifetime distribution of the system.
- We only observe the system lifetime, not the component lifetime.
- Our mission is to do statistical inference on the component lifetime distribution based on the SEM algorithm.

**SEM Algorithm**

Suppose $T$ is the observed data and $Z$ is the missing data, the complete data is $X = (T, Z)$ and the likelihood function based on complete data is $L(\theta; X)$, where $\theta$ is the parameter of interest. SEM algorithm is an iterative procedure that replaces the missing data $Z$ with a single draw from conditional distribution of $Z$ [5-step] and maximizes the complete-data likelihood to update the parameter estimate (M-step).

**SEM algorithm for complete data**

- **SEM algorithm based on ordinary system signature:**
  - Right-truncated distribution
    $$\phi_k(z; \theta_k) = \frac{f_k(z; \theta_k)}{S_k(z; \theta_k)}$$
  - Left-truncated distribution
    $$\psi_k(z; \theta_k) = \frac{F_k(z; \theta_k)}{S_k(z; \theta_k)}$$

  Where $f_k$, $F_k$, $S_k$ are the density function, cumulative density function and survival function of component lifetime $X_k$, respectively. Assume there are $n$ components in each system. For the $(k+1)$-th iteration, given the current value $\theta^{(m)}$ and observed $m$ system lifetimes, the 5-step and M-step are:

  **5-step:**
  1. For $k$-th system, generate a discrete random variable $\Delta_k$ based on signature $x_k$ with probability $Pr(\Delta_k = 0) = s_0$, $\Delta = 1, \ldots, n$
  2. Generate $\delta - 1$ random variables from distribution (1), with $\theta = \theta^{(m)}$, $s_k$, $x_k(\delta - 1)$
  3. Generate $n - \delta$ random variables from distribution (2), with $\theta = \theta^{(m)}$, $s_k$, $x_k(n - \delta)$
  4. The pseudo-complete sample for system $k$ is $x_k = (x_{k1}, x_{k2}, \ldots, x_{k\delta}, x_{k\delta+1}, \ldots, x_{kn})$
  5. Repeat steps 1-4 for $k = 1, \ldots, m$

  **M-step:**
  - Maximize the log-likelihood function
    $$L(\theta; x_1, x_2, \ldots, x_m) = \sum_{k=1}^{m} \log f(x_k; \theta)$$
  - With respect to $\theta$ to obtain $\theta^{(m+1)}$.

- **SEM based on ordered system signature (SEM-OSS)**
  - S-step: 1. For $k$-th ordered system, generate a discrete random variable $\Delta$ based on ordered signature $x(k)$ with probability
    $$Pr(\Delta = 0) = s_k^{(m)}(x(k, \delta))$$
  - 2-step: M-step: Maximize the log-likelihood function

- **SEM algorithm for Type-II censored data**
  - Impulse component lifetimes directly (SEM-I)
  - Suppose system is working at time $t_r$, $y^{(m+1)}(t_r)$ out of $n$ components failed $[T_1 > t_r]$
    $$p^{(m)}_r(t) = \left( \sum_{k=1}^{m} \frac{n}{\prod_{j=1}^{n} F_k(y^{(m+1)}(t))} \left( \prod_{j=1}^{n} F_k(y^{(m+1)}(t)) \right)^{r-1} \prod_{j=1}^{n} \left( 1 - F_k(y^{(m+1)}(t)) \right) \right)$$

  If observed system lifetime is $t_{c, m} < t_r$, then $t = t_{c, m}$

**Example**

- Signature: $1, 1, 1, 1, 1, 0$.
- Sample size: $n = 10$.
- Location parameter = 1.09861, Scale parameter $= 5$ (1), with underlying SEV distribution.

- **Results**
  - Monte-Carlo simulation is carried out to evaluate the performance of the proposed methods, comparing to MLE, in terms of difference and mean squared error (MSE), with different signatures $x_k = (1, 1, 1, 1, 1, 0), \ y_k = (0, 0, 0, 0, 0)$
  - The underlying component lifetime distribution is smallest extreme value (SEV) distribution with $\mu = 0$ and $\sigma = 1$

  I. Performance of SEM algorithm for complete system lifetime data comparing to MLE
  
  - SEM-OSS vs. MLE
  - SEM-OSS vs. MLE
  - SEM-OSS vs. MLE

  **Discussion**

  1. For complete case, the SEM-OSS algorithm performs almost the same as the MLE in the SEM-I algorithm.
  2. For Type-II censored case, SEM-I algorithm and SEM-II algorithm perform similarly in the sense that both approximate the MLE very well.
  3. When censoring proportion is small to moderate (say, $q \leq 0.4$), SEM-II algorithm is recommended.
  4. When censoring proportion is large (say, $q > 0.4$), SEM-I algorithm is recommended.

**Reference**