Structural Estimation: EM Algorithm

Christine Braun

Last Time

- Non-Parametric estimation of mixing distribution
  - We discretize G
  - $\{\nu_j\}_{j=1}^K$: set of points in G
  - $\{\pi_j\}_{j=1}^K$: the probability of point j
- Sum over the points to get the full distribution of durations

$$f(t|x) = \sum_{j=1}^{K} \pi_j \times f(t|x, \nu_j)$$

- The likelihood function we be a function of $\{\nu_j\}_{j=1}^K$ and $\{\pi_j\}_{j=1}^K$ and we get ML estimates of each point and it's probability.

Last Time

- **Problem:** can no longer log the likelihood function
- Increases computational burden
    - In our case $f(t|x, \nu_j)$ was "simple" enough
- **Solution:** Expectation-Maximization Algorithm

General Latent Variable Problem

- x is an observed random variable
- z is an unobserved (latent) variable
- The joint probability is parameterized by $\theta \in \Theta$

$$p(x, z; \theta)$$

- There are two sets of unknowns: z and θ
- EM algorithm
  - Guess z, maximize w.r.t. θ
  - Use the estimate of θ to get a better guess for z

Simple Example: Gaussian Mixture Model

- We observe data $x = \{x_1, ...x_n\}$ which are i.i.d. draws
  - $N(\mu_1, \sigma_1)$ with probability π
  - $N(\mu_2, \sigma_2)$ with probability $1 - \pi$
- We do not know which distribution each x_i came from
- So we need to estimate $\theta = \{\pi, \mu_1, \mu_2, \sigma_1, \sigma_2\}$

Simple Example: Gaussian Mixture Model

- The likelihood function, $\phi(\cdot)$ is the normal pdf

$$
\begin{aligned}
L(\theta; x) &= \prod_{i=1}^{N} p(x_i; \theta) \\
&= \prod_{i=1}^{N} \pi\phi(x_i; \mu_1, \sigma_1) + (1-\pi)\phi(x_i; \mu_2, \sigma_2)
\end{aligned}
$$

- The log-likelihood

$$
\mathcal{L}(\theta; x) = \sum_{i=1}^{N} \log[\pi\phi(x_i; \mu_1, \sigma_1) + (1-\pi)\phi(x_i; \mu_2, \sigma_2)]
$$

- **Problem:** can't distribute the log any further

Simple Example: Gaussian Mixture Model

- **Solution:** introduce a latent variable

$$z_i = \begin{cases} 1 & x_i \text{ is a draw from } N(\mu_1, \sigma_1) \\ 0 & x_i \text{ is a draw from } N(\mu_2, \sigma_2) \end{cases}$$

  where $P(z_i = 1) = \pi$

- If we observed z_i then the likelihood is

$$L(\theta; x, z) = \prod_{i=1}^{N} [\pi\phi(x_i; \mu_1, \sigma_1)]^{z_i}[(1-\pi)\phi(x_i; \mu_2, \sigma_2)]^{1-z_i}$$

$$\mathcal{L}(\theta; x, z) = \sum_{i=1}^{N} z_i log[\phi(x_i; \mu_1, \sigma_1)] + (1 - z_i) log[\phi(x_i; \mu_2, \sigma_2)]$$
$$+ z_i log(\pi) + (1 - z_i) log(1 - \pi)$$

Simple Example: Gaussian Mixture Model

- EM Algorithm

  1. Guess initial values, $\hat{\theta}_0$
  2. Expectations Step (E-Step)
     - Given $\hat{\theta}_0$ estimate $\hat{z}_i$ (we will see how to do this later)
     - Construct $\mathcal{L}(\theta; x, \hat{z})$
  3. Maximization Step (M-Step)

  $$\hat{\theta}_1 = argmax \mathcal{L}(\theta; x, \hat{z})$$

  4. Repeat 2-3 until $|\hat{\theta}_{j+1} - \hat{\theta}_j| < \varepsilon$

EM Algorithm: Why does it work?

- For any θ guess, $\mathcal{L}(\theta; x, \hat{z})$ is a lower bound to $\mathcal{L}(\theta; x)$
- The algorithm is repeated maximization of lower bounds
- Two caveats
  - convergence is often slow
  - converges to local max (initial guess matters!)

EM Algorithm: Why does it work?

$$
\begin{aligned}
\mathcal{L}(\theta; x) &= log\ P(x; \theta) \\
&= log \sum_z P(x, z; \theta) \\
&= log \sum_z P(z) \left(\frac{P(x, z; \theta)}{P(z)} \right) \\
&\geq \underbrace{\sum_z P(z) log \left(\frac{P(x, z; \theta)}{P(z)} \right)}_{\mathcal{L}(\theta; x, z)} \quad \text{(Jensen's Inequality)}
\end{aligned}
$$

- So $\mathcal{L}(\theta; x, z)$ is a lower bound for any choice of z

EM Algorithm: The best lower bound

$$
\begin{aligned}
\mathcal{L}(\theta; x, z) &= \sum_z P(z) log\left(\frac{P(x, z; \theta)}{P(z)}\right) \\
&= \sum_z P(z) log\left(\frac{P(z|x; \theta)P(x; \theta)}{P(z)}\right) \\
&= \sum_z P(z) log\left(\frac{P(z|x; \theta)}{P(z)}\right) + \sum_z P(z) log\ P(x; \theta) \\
&= -KL\big(P(z)||P(z|x; \theta)\big) + \mathcal{L}(\theta; x)
\end{aligned}
$$

- $KL\big(P(z)||P(z|x;\theta)\big)$ is the Kullbeck-Leibler divergence
- $KL\big(P(z)||P(z|x;\theta)\big) = 0$ when $P(z) = P(z|x;\theta)$.

EM Algorithm

(E-Step) With $\hat{\theta}_j$ and compute the probabilities of z

$$P(z|x;\hat{\theta}_j) = \frac{P(x|z;\hat{\theta}_j)P(z|\hat{\theta}_j)}{\sum_z P(x|z;\hat{\theta}_j)P(z|\hat{\theta}_j)}$$

(M-Step) Maximize the lower-bound to get new estimate

$$\hat{\theta}_{j+1} = argmax \sum_z P(z|x;\hat{\theta}_j) log \left(\frac{P(x,z;\theta)}{P(z|x;\hat{\theta}_j)} \right)$$

$$\hat{\theta}_{j+1} = argmax \sum_z P(z|x;\hat{\theta}_j) log [P(x|z;\theta)P(z;\theta)]$$

EM Algorithm: Gaussian Mixture Model

- We observe $\{x_1, ..., x_n\}$ that are i.i.d. draws from
  - $\phi(x_i, \mu_1, \sigma_1) \sim N(\mu_1, \sigma_1)$ with probability π
  - $\phi(x_i, \mu_2, \sigma_2) \sim N(\mu_2, \sigma_2)$ with probability $1 - \pi$
- If want to estimate $\theta = \{\pi, \mu_1, \mu_2, \sigma_1, \sigma_2\}$
- Introduce a latent variable

$$z_i = \begin{cases} 1 & x_i \text{ is a draw from } N(\mu_1, \sigma_1) \\ 0 & x_i \text{ is a draw from } N(\mu_2, \sigma_2) \end{cases}$$

- Start with an initial guess $\hat{\theta} = \{\hat{\pi}, \hat{\mu}_1, \hat{\mu}_2, \hat{\sigma}_1, \hat{\sigma}_2\}$

EM Algorithm

(E-Step) With $\hat{\theta}_j$ and compute the probabilities of z

$$P(z|x;\hat{\theta}_j) = \frac{P(x|z;\hat{\theta}_j)P(z|\hat{\theta}_j)}{\sum_z P(x|z;\hat{\theta}_j)P(z|\hat{\theta}_j)}$$

$$P(z=1|x;\hat{\theta}) = \frac{\hat{\pi}\phi(x_i,\hat{\mu}_1,\hat{\sigma}_1)}{\hat{\pi}\phi(x_i,\hat{\mu}_1,\hat{\sigma}_1)+(1-\hat{\pi})\phi(x_i,\hat{\mu}_2,\hat{\sigma}_2)}$$

$$P(z=0|x;\hat{\theta}) = \frac{(1-\hat{\pi})\phi(x_i,\hat{\mu}_2,\hat{\sigma}_2)}{\hat{\pi}\phi(x_i,\hat{\mu}_1,\hat{\sigma}_1)+(1-\hat{\pi})\phi(x_i,\hat{\mu}_2,\hat{\sigma}_2)}$$

EM Algorithm

(M-Step) Maximize the lower-bound to get new estimate

$$\hat{\theta}_{j+1} = argmax \sum_z P(z|x;\hat{\theta}_j) log[P(x|z;\theta)P(z;\theta)]$$

$$= argmax \Big(P(z=1|x;\hat{\theta}) log[\pi\phi(x_i,\mu_1,\sigma_1)] + P(z=0|x;\hat{\theta}) log[(1-\pi)\phi(x_i,\mu_2,\sigma_2)] \Big)$$

(Check) $|\hat{\theta}_{j+1} - \hat{\theta}_j| < \varepsilon$

Matlab Estimation

- using data5.csv
- File 1: SE5_main
  - Part 1: estimate the Gaussian mixture model
  - pick a ε as stopping criterion
- File 2: log_like_GM.m
  - inputs ?
  - outputs ?

Matlab Estimation: Part 1 Answer

- init guess $= [1, 1, 0, 1, 0.5]$

Parameter	Value	Estimate
μ_1	5	4.9760
		(0.0255)
σ_1	1.2	1.1859
		(0.0177)
μ_2	0	0.0029
		(0.0081)
σ_2	1	1.0029
		(0.0081)
π	0.2	0.1973
		(0.0040)

EM Algorithm - Mixed Proportional Hazard Model

- From Last Time

$$f(t_i|x_i;\alpha,\beta,\nu_1) = \nu_1 \exp(x_i'\beta)\alpha t_i^{\alpha-1} e^{-\nu_1 \exp(x_i'\beta) t_i^{\alpha}}$$
$$f(t_i|x_i;\alpha,\beta,\nu_2) = \nu_2 \exp(x_i'\beta)\alpha t_i^{\alpha-1} e^{-\nu_2 \exp(x_i'\beta) t_i^{\alpha}}$$
$$f(t_i|x_i;\alpha,\beta,\nu_3) = \nu_2 \exp(x_i'\beta)\alpha t_i^{\alpha-1} e^{-\nu_3 \exp(x_i'\beta) t_i^{\alpha}}$$

- $\theta = \{\alpha, \beta, \{\nu_j\}, \{\pi_j\}\}$

- ν is our latent variable "z"

EM Algorithm

(E-Step) With $\hat{\theta}_j = \{\hat{\alpha}, \hat{\beta}, \{\hat{\nu}_j\}, \{\hat{\pi}_j\}\}$ and compute the probabilities of ν_k

$$P(z|x;\hat{\theta}_j) = \frac{P(x|z;\hat{\theta}_j)P(z|\hat{\theta}_j)}{\sum_z P(x|z;\hat{\theta}_j)P(z|\hat{\theta}_j)}$$

$$P(\nu_k|x;\hat{\theta}) = \frac{\hat{\pi}_k \hat{\nu}_k \exp(x_i'\hat{\beta})\hat{\alpha} t_i^{\hat{\alpha}-1} e^{-\hat{\nu}_k \exp(x_i'\hat{\beta}) t_i^{\hat{\alpha}}}}{\sum_k \hat{\pi}_k \hat{\nu}_k \exp(x_i'\hat{\beta})\hat{\alpha} t_i^{\hat{\alpha}-1} e^{-\hat{\nu}_k \exp(x_i'\hat{\beta}) t_i^{\hat{\alpha}}}}$$

EM Algorithm

(M-Step) Maximize the lower-bound to get new estimate

$$\hat{\theta}_{j+1} = argmax \sum_{z} P(z|x; \hat{\theta}_j) log[P(x|z; \theta) P(z; \theta)]$$

$$= argmax \sum_{k} P(\nu_k|x; \hat{\theta}_j) log[\pi_k \nu_k \exp(x_i' \beta) \alpha t_i^{\alpha-1} e^{-\nu_k \exp(x_i' \beta) t_i^{\alpha}}]$$

(Check) $|\hat{\theta}_{j+1} - \hat{\theta}_j| < \varepsilon$

Estimation Answer

- Estimates and Standard Errors

Parameter	Estimate	Standard Error
α	0.9675	0.0275
ν_1	0.0574	0.0076
ν_2	0.2565	0.0354
ν_2	0.7543	0.1310
π_1	0.0523	0.0899
π_2	0.6649	1.1531
π_3	0.2827	0.4891
β_{FE}	0.0629	0.0260
β_{educ2}	0.0044	0.1167
β_{educ3}	0.0277	0.1132

- Log-Likelihood Value

$$logL = -2.9267e + 04$$