Notations (common)

Each row is the recording for neuron i, $\boldsymbol{y}_i = (y_{i1}, \dots, y_{iT})'$, $i = 1, \dots N$. Denote the cluster index for neuron i as $z_i \in \{1, \dots\}$. The number of neurons in cluster j is $n_j = \sum_{i=1}^{N} I(z_i = j)$, and $\sum_{j=1,2,\dots} n_j = N$. The proportion/ probability in cluster z_i is ρ_{z_i}.

Mixture Model (MM)

The number of cluster is J for MM. The full likelihood for these N neurons is

$$L = \prod_{i=1}^{N} \rho_{z_i} f(\boldsymbol{y}_i | \boldsymbol{\Theta}_{z_i}) = \prod_{j=1}^{J} \rho_j^{n_j} \left[\prod_{i:z_i=j} f(\boldsymbol{y}_i | \boldsymbol{\Theta}_j) \right]$$

Where $\boldsymbol{\Theta}_j$ is all parameters in cluster j defined by the model. The model details & likelihood can be found in "models" folder, and model XXX is named as "MCMC_XXX".

The parameters need to update:

(1) Cluster indicator: $\{z_i\}_{i=1}^{N}$
(2) Cluster proportion: $\boldsymbol{\rho} = (\rho_1, \dots \rho_J)'$
(3) Model parameters: $\boldsymbol{\Theta}_j$

The (conditional) priors:

(1) Cluster indicator $\{z_i\}_{i=1}^{N}$:

$$P(z_i = j) = \rho_j$$

(2) Cluster proportion $\boldsymbol{\rho} = (\rho_1, \dots \rho_J)'$:

$$\boldsymbol{\rho} \sim Dir(\delta_1, \dots \delta_J)$$

Where $\delta_1 = \cdots = \delta_J = 1$

MCMC iteration:

(1) Update $\{z_i\}_{i=1}^{N}$:

$$P\left(z_i = j \middle| \boldsymbol{y}_i, \{\boldsymbol{\Theta}_j\}_{j=1}^{J}\right) \propto \rho_j f(\boldsymbol{y}_i | \boldsymbol{\Theta}_j)$$

(2) Update $\boldsymbol{\rho} = \left(\rho_1, \dots \rho_J\right)'$

$$\boldsymbol{\rho} | \{\boldsymbol{y}_i\}_{i=1}^{N}, \{z_i\}_{i=1}^{N}, \{\boldsymbol{\Theta}_j\}_{j=1}^{J} \sim Dir(\delta_1 + n_1, \dots \delta_J + n_J)$$

(3) Update $\boldsymbol{\Theta}_j$:
See details of the chosen model in "~/documents/models/MCMC_XXX.docx". when there's no $z_i = j$, just sample $\boldsymbol{\Theta}_j$ from prior.

Dirichlet Process (DP)

Use slice sampler (Walker 2007, https://www.tandfonline.com/doi/full/10.1080/03610910601096262).

Represent cluster proportion by "stick-breaking", i.e.

$$\rho_1 = \eta_1$$

$$\rho_j = (1-\eta_1)\cdot \ldots \cdot (1-\eta_{j-1})\eta_j$$

$$\eta_j \sim Beta(1,\alpha)$$

The parameters need to update:

(1) "stick-breaking" elements: η_j
(2) Augment latent variable: $\{u_i\}_{i=1}^N$
(3) Model parameters: $\mathbf{\Theta}_j$
(4) Cluster indicator: $\{z_i\}_{i=1}^N$

MCMC iteration:

(1) update η_j, for $j = 1, \ldots, \mathrm{z}^* = \max\{z_i\}_{i=1}^N$ as

$$\eta_j|\{z_i\}_{i=1}^N, \ldots \sim Beta(n_j + 1, N - \sum_{l=1}^{j} n_l + \alpha)$$

(2) update $\{u_i\}_{i=1}^N$

$$u_i|\boldsymbol{\rho}, \ldots \sim U(0, \rho_{z_i})$$

(3) update η_j, for $j = \mathrm{z}^* + 1, \ldots, s^*$. s^* is the smallest value, s.t. $\sum_{j=1}^{s^*} \rho_j > 1 - \min\{u_1, \ldots, u_N\}$

$$\eta_j \sim Beta(1,\alpha)$$

(4) update state vectors $\{\mathbf{\Theta}_j\}_{j=1,\ldots}$

See details of the chosen model in "~/documents/models/MCMC_XXX.docx". when there's no $z_i = j$, just sample $\mathbf{\Theta}_j$ from prior.

(5) Update $\{z_i\}_{i=1}^N$

$$P(z_i = j|\boldsymbol{y}_i, \{\mathbf{\Theta}_j\}, \boldsymbol{\rho}, \{u_i\}_{i=1}^N) = \frac{f(\boldsymbol{y}_i|\mathbf{\Theta}_j)}{\sum_{j:\rho_j > u_i} f(\boldsymbol{y}_i|\mathbf{\Theta}_j)}$$