Supplementary material

1 Details of parameter estimation

1.1 E-step

Set $\{\boldsymbol{y}_{s,t}\}_1^N = (\boldsymbol{y}_{s,t_1}, \boldsymbol{y}_{s,t_1+1}, \cdots, \boldsymbol{y}_{s,t_2})$ for $t_1 < t_2$. Here, we assume that $\boldsymbol{x}_{s,0} = \boldsymbol{0},\ s = 1, \cdots, S$. In the E-step, we calculate the conditional expectation of the joint log-likelihood given the guesses for the parameters as:

$$\begin{aligned}
&E\big\{\log \Pr\big(\{\boldsymbol{x}_{s,t}\}, \{\boldsymbol{y}_{s,t}\}|A, \{\boldsymbol{b}_s\}, C, \sigma^2\big)\big\}\\
=& -\frac{1}{2\sigma^2}\sum_{s=1}^{S}\sum_{t=1}^{T}\Big(\boldsymbol{y}'_{s,t}\boldsymbol{y}_{s,t} - 2\boldsymbol{y}'_{s,t}C\bar{\boldsymbol{x}}_{s,t} + \mathrm{Tr}\big(C'CP_{s,t}\big)\Big)\\
&-\frac{1}{2}\Big(\mathrm{Tr}\big(R_1\big) - 2\mathrm{Tr}\big(\Psi' R_2\big) + \mathrm{Tr}\big(\Psi'\Psi R_3\big)\Big)\\
&-\frac{NST}{2}\log\sigma^2 - \frac{KST}{2}\log\tau^2,
\end{aligned}$$

where

$$\begin{aligned}
\bar{\boldsymbol{x}}_{s,t} &= E\big[\boldsymbol{x}_{s,t}\big|\{\boldsymbol{y}_{s,t}\}_1^T\big],\\
P_{s,t} &= E\big[\boldsymbol{x}_{s,t}\boldsymbol{x}'_{s,t}\big|\{\boldsymbol{y}_{s,t}\}_1^T\big],\\
P_{s,t,t-1} &= E\big[\boldsymbol{x}_{s,t}\boldsymbol{x}'_{s,t-1}\big|\{\boldsymbol{y}_{s,t}\}_1^T\big],\\
\Psi &= \begin{bmatrix} A & \boldsymbol{b}_1 & \cdots & \boldsymbol{b}_S \\ \boldsymbol{0} & & I & \end{bmatrix},
\end{aligned}$$

$$R_1 = \begin{bmatrix} \sum_{s=1}^{S}\sum_{t=1}^{T} P_{s,t} & \sum_{t=1}^{T}\bar{\boldsymbol{x}}_{1,t} \cdots \sum_{t=1}^{T}\bar{\boldsymbol{x}}_{S,t} \\ \sum_{t=1}^{T}\bar{\boldsymbol{x}}'_{1,t} & \\ \vdots & T\times I \\ \sum_{t=1}^{T}\bar{\boldsymbol{x}}'_{S,t} & \end{bmatrix}.$$

$$R_2 = \begin{bmatrix} \sum_{s=1}^{S}\sum_{t=1}^{T} P_{s,t,t-1} & \sum_{t=1}^{T}\bar{\boldsymbol{x}}_{1,t} \cdots \sum_{t=1}^{T}\bar{\boldsymbol{x}}_{S,t} \\ \sum_{t=1}^{T}\bar{\boldsymbol{x}}'_{1,t-1} & \\ \vdots & T\times I \\ \sum_{t=1}^{T}\bar{\boldsymbol{x}}'_{S,t-1} & \end{bmatrix},$$

$$R_3 = \begin{bmatrix} \sum_{s=1}^{S}\sum_{t=1}^{T} P_{s,t-1} & \sum_{t=1}^{T}\bar{\boldsymbol{x}}_{1,t-1} \cdots \sum_{t=1}^{T}\bar{\boldsymbol{x}}_{S,t-1} \\ \sum_{t=1}^{T}\bar{\boldsymbol{x}}'_{1,t-1} & \\ \vdots & T\times I \\ \sum_{t=1}^{T}\bar{\boldsymbol{x}}'_{S,t-1} & \end{bmatrix},$$

$\{\bar{\boldsymbol{x}}_{s,t}\}$, $\{P_{s,t}\}$ and $\{P_{s,t,t-1}\}$ can be obtained by performing Kalman Filter algorithm. Since our model is slightly different from usual state space models, we derived the Kalman Filter algorithm for our model.

Set $\bar{\boldsymbol{x}}_{s,t|t^*} = E\big[\boldsymbol{x}_{s,t}\big|\{\boldsymbol{y}_{s,t}\}_1^{t^*}\big]$, and $V_{s,t|t^*} = \mathrm{Var}\big[\boldsymbol{x}_{s,t}\big|\{\boldsymbol{y}_{s,t}\}_1^{t^*}\big]$. The forward recursions is as follows:

$$\begin{aligned}
\boldsymbol{x}_{s,t|t-1} &= A\boldsymbol{x}_{s,t-1|t-1} + \boldsymbol{b}_s\\
V_{s,t|t-1} &= AV_{s,t-1|t-1}A' + \tau^2 I\\
\boldsymbol{x}_{s,t|t} &= \boldsymbol{x}_{s,t|t-1} + K_{s,t}(\boldsymbol{y}_{s,t} - C\boldsymbol{x}_{s,t|t-1})\\
V_{s,t|t} &= (I - K_{s,t}C)V_{s,t|t-1}\\
K_{s,t} &= V_{s,t|t-1}C'(\sigma^2 I + CV_{s,t|t-1}C')^{-1},
\end{aligned}$$

Then, $\bar{\boldsymbol{x}}_{s,t} = \boldsymbol{x}_{s,t|T}$ and $P_{s,t} = V_{s,t|T} + \boldsymbol{x}_{s,t|T}\boldsymbol{x}'_{s,t|T}$ can be obtained via the backward recursion

$$\begin{aligned}
J_{s,t-1} &= V_{s,t-1|t-1}A'V^{-1}_{s,t-1|t-1} \\
\boldsymbol{x}_{s,t-1|T} &= \boldsymbol{x}_{s,t-1|t-1} + J_{s,t-1}(\boldsymbol{x}_{s,t|T} - x_{s,t|t-1}) \\
V_{s,t-1|T} &= V_{s,t-1|t-1}J_{s,t-1}(V_{s,t|T} - V_{s,t-1|T})J^{-1}_{s,t-1}.
\end{aligned}$$

Furthermore, $P_{s,t,t-1} = V_{s,t,t-1|T} + \boldsymbol{x}_{s,t|T}\boldsymbol{x}'_{s,t-1|T}$ can be calculated by the recursion

$$V_{s,t-1,t-2|T} = V_{s,t-1|t-1}J'_{s,t-2} + J_{s,t-1}(V_{s,t,t-1|T} - AV_{s,t-1|t-1})J'_{s,t-2},$$

where $V_{s,T,T-1|T} = (I - K_{s,T}C)AV_{s,T-1|T-1}$.

1.2 M-step

In the M-step, each of parameters is re-estimated by maximizing the conditional likelihood obtained in the E-step. The results are as follows:

- C

  The term of the expected likelihood relevant to C or d can be rewritten as:

  $$\sum_{n=1}^{N}\Big(\boldsymbol{c}'_n\bar{P}\boldsymbol{c}_n - 2\boldsymbol{\phi}'_n\boldsymbol{c}_n\Big),$$

  where $\bar{P} = \sum_{s=1}^{S}\sum_{t=1}^{T}P_{s,t}$, $\boldsymbol{\phi}_n = \sum_{s=1}^{S}\sum_{t=1}^{T}y_{n,s,t}\bar{\boldsymbol{x}}_{s,t}$ and $\boldsymbol{c}'_n$ is the n-th row of C. Therefore, each $\boldsymbol{c}_n$ is updated as Algorithm 1.

- A, b

  The term relevant to A or b can be rewritten as:

  $$\sum_{k=1}^{K+S}\boldsymbol{\psi}_k R_3\boldsymbol{\psi}'_k - 2\sum_{k=1}^{K+S}\boldsymbol{r}_k\boldsymbol{\psi}'_k + \lambda\tau^2\sum_{k=1}^{K}\sum_{i=1}^{K+S}\chi(\psi \neq 0),$$

  where $\boldsymbol{\psi}_k = (\psi_{k,1}, \cdots, \psi_{k,K+S})$ is the k-th row of Ψ and $R_2 = [\boldsymbol{r}_1, \cdots, \boldsymbol{r}_{K+S}]'$. Therefore, each $\boldsymbol{\psi}_k$ each $\boldsymbol{\psi}_k$ is updated as Algorithm 2.

- σ^2, τ^2

  Taking the corresponding partial derivative of the expected log-likelihood, σ^2 is estimated as:

  $$\hat{\sigma}^2 \Leftarrow \frac{\sum_{s=1}^{S}\sum_{t=1}^{T}\Big(\boldsymbol{y}'_{s,t}\boldsymbol{y}_{s,t} - 2\boldsymbol{y}'_{s,t}C\bar{\boldsymbol{x}}_{s,t} + \mathrm{Tr}\big(C'CP_{s,t}\big)\Big)}{NST}.$$

  Similarly, τ^2 is estimated as:

  $$\hat{\tau}^2 \Leftarrow \frac{\Big(\mathrm{Tr}\big(R_1\big) - 2\mathrm{Tr}\big(\Psi'R_2\big) + \mathrm{Tr}\big(\Psi'\Psi R_3\big)\Big)}{KS(T-1)}.$$

Algorithm 1 Update $\boldsymbol{c}_n$

for $n = 1$ to N **do**
  $k^* = \arg\max_{k=1,\cdots,K}\big(\bar{p}_{k,k} - 2\phi_{n,k}\big)$ ($\bar{P} = \{\bar{p}_{i,j}\}$, $\boldsymbol{\phi}_n = (\phi_{n,1}, \cdots, \phi_{n,K})'$)
  $\boldsymbol{c}_n = \mathbf{1}_{k^*}$ ($\mathbf{1}_{k^*}$ is indicator vector whose k^*-th value is 1 and the others are zero)
end for
return $\boldsymbol{c}_n$

Algorithm 2 Update $\boldsymbol{\psi}$ (In the following, we omit indices k of $\boldsymbol{\psi}_k$, $\boldsymbol{r}_k$ for simplicity)

$v^* \Leftarrow \frac{1}{2}\boldsymbol{\psi}' R_3 \boldsymbol{\psi}_k - \boldsymbol{r}'\boldsymbol{\psi} + \frac{1}{2}\lambda\tau^2 \sum_{k=1}^{K+S} \chi(\psi_k \neq 0)$ // Calculating the present value of objective function
$\mathcal{K} = \{\}$
$\boldsymbol{\zeta} \Leftarrow \arg\min\left(\frac{1}{2}\boldsymbol{\xi}' R_3 \boldsymbol{\xi} - \boldsymbol{r}'\boldsymbol{\xi}\right)$
$v \Leftarrow \left(\frac{1}{2}\boldsymbol{\zeta}' R_3 \boldsymbol{\zeta} - \boldsymbol{r}'\boldsymbol{\zeta}\right)$
for $n = 1$ to $2K$ **do**
  $\tilde{k} \Leftarrow \text{mod}(\text{rand}, K) + 1$ // Draw a number at random from 1 to K
  if $\tilde{k} \notin \mathcal{K}$ **then**
    $\tilde{\mathcal{K}} \Leftarrow \mathcal{K} \cup \{\tilde{k}\}$
    $\tilde{\boldsymbol{\zeta}} \Leftarrow \arg\min_{\zeta_k = 0, k \in \tilde{\mathcal{K}}}\left(\frac{1}{2}\boldsymbol{\xi}' R_3 \boldsymbol{\xi} - \boldsymbol{r}'\boldsymbol{\xi}\right)$
    $\tilde{v} \Leftarrow \left(\frac{1}{2}\tilde{\boldsymbol{\zeta}}' R_3 \tilde{\boldsymbol{\zeta}} - \boldsymbol{r}'\tilde{\boldsymbol{\zeta}} + \frac{1}{2}\lambda\tau^2 |\tilde{\mathcal{K}}|\right)$
    if $\tilde{v} < v$ **then**
      $v \Leftarrow \tilde{v}$
      $\zeta \Leftarrow \tilde{\zeta}$
      $\mathcal{K} \Leftarrow \tilde{\mathcal{K}}$ // $|\tilde{\mathcal{K}}|$ is cardinality of $\tilde{\mathcal{K}}$
    end if
  end if
end for
if $v < v*$ **then**
  $\boldsymbol{\psi} \Leftarrow \boldsymbol{\zeta}$
end if
return $\boldsymbol{\psi}$

2 Details of split-merge procedure

Let $\text{Fit}(k), k = 1, \cdots, K$ denote the degree of fitness of cluster k as

$$\text{Fit}(k) = \sum_{n; c_{n,k}=1} \sum_{s=1}^{S} \sum_{t=1}^{T} (y_{n,s,t} - x_{k,s,t} - b_{k,s})^2,$$

and let $\text{Fit}(k, l), k \neq l$ denote the expected degree of fitness of the clusters k, l.

$$\text{Fit}(k,l) = \sum_{n; c_{n,k}+c_{n,l}=1} \sum_{s=1}^{S} \sum_{t=1}^{T} \left(y_{n,s,t} - \frac{N_k x_{k,s,t} + N_l x_{l,s,t}}{N_k + N_l} - \frac{N_k b_{k,s} + N_l b_{l,s}}{N_k + N_l} \right)^2,$$

where N_k is the number of elements in cluster k. We use $\text{Fit}(k)$ and

$$\text{dFit}(k,l) = \text{Fit}(k,l) - \text{Fit}(k) - \text{Fit}(l)$$

as criteria for splitting and merging, respectively. $\text{dFit}(k, l)$ is a estimation of the degree of increased fitness when the clusters k, l are merged. First, we select J_m merge candidates in decreasing order of $\text{dFit}(k, l)$, Then, for J_m merge candidates, we select J_s split candidates in decreasing order of $\text{Fit}(k)$ excluding that merge candidate. Then combining these sorted candidates, we obtain an ordered split-merge candidates. Split into two clusters is done based on the first eigenvector of the variance-covariance matrix of the elements in the cluster to be splitted. The whole procedure is as follows:

1. Estimate parameters from some initial value until convergence. Let Θ^*, L^* denote the estimated parameters and the corresponding loss function.

2. For resulting Θ^*, obtain the split-merge candidates via the above criteria.

3. For each split-merge candidates, perform split-merge operation, and optimize the loss function until convergence. Let Θ^{**}, L^{**} denote the estimated parameters and the corresponding loss function. If L^{**} is larger than L^*, then set $L^* \leftarrow L^{**}$, $\Theta^* \leftarrow \Theta^{**}$ and go to step 2. If no L^{**} exceeds L^*, return Θ^* as the result.

Table 1: The number of misspecification out of 50 trials

S	1	2	4
# misspecification	8	0	0

3 Numerical experiments on the determination of the number of clusters

We examined whether the proposed method can choose the true number (4 in this data) for 50 trial, changing the number of stimuli S over 1, 2, 4. The result, which is shown in Table 1, shows that our method works well, especially when the number of stimulus is large.