APPENDIX

A DERIVATIONS

Variational lower bound Here we derive the variational lower bound Eq. (2) for the log-likelihood objective Eq. (1). For each $i \in [K]$,

$$\log \sum_{z_i=1}^{K} p(D_i, z_i; \Theta) = \log \sum_{z_i=1}^{K} q(z_i) \cdot \frac{p(D_i, z_i; \Theta)}{q(z_i)} \tag{9}$$

$$= \log \mathbb{E}_{q(z_i)} \left[\frac{p(D_i, z_i; \Theta)}{q(z_i)} \right] \tag{10}$$

$$\geq \mathbb{E}_{q(z_i)} \left[\log \frac{p(D_i, z_i; \Theta)}{q(z_i)} \right] \tag{11}$$

$$= \mathbb{E}_{q(z_i)} \left[\log p(D_i, z_i; \Theta) \right] - \mathbb{E}_{q(z_i)} [\log q(z_i)], \tag{12}$$

where q is an alternative distribution, the inequality is due to Jensen's Inequality and the last term $\mathbb{E}_{q(z_i)}[\log q(z_i)]$ is constant independent of the parameter Θ.

Derivations of the EM steps Given the assumptions in the main text about $p_i(y|\mathbf{x})$ and $p_i(\mathbf{x})$, we know that

$$-\log p(D_i | z_i = j; \Phi) = \sum_{s=1}^{n_i} \ell(h_{\boldsymbol{\phi}_j}(\mathbf{x}_s)^{(i)}, y_s^{(i)}) - \log p(\mathbf{x}_s^{(i)}) + c. \tag{13}$$

- **E-step:** Find the best q for each client given the current parameters $\Theta^{(t-1)}$:

$$w_{ij}^{(t)} := q^{(t)}(z_i = j) = p(z_i = j | D_i; \Theta^{(t-1)}) \tag{14}$$

$$= \frac{p(z_i = j | \Pi^{(t-1)}) \cdot p(D_i | z_i = j; \Phi^{(t-1)})}{\sum_{j'=1}^{K} p(z_i = j' | \Pi^{(t-1)}) \cdot p(D_i | z_i = j'; \Phi^{(t-1)})} \tag{15}$$

$$= \frac{\Pi_{ij}^{(t-1)} \cdot p(D_i | z_i = j; \Phi^{(t-1)})}{\sum_{j'=1}^{K} \Pi_{ij'}^{(t-1)} \cdot p(D_i | z_i = j'; \Phi^{(t-1)})} \tag{16}$$

$$\propto \Pi_{ij}^{(t-1)} \exp \left[-\sum_{s=1}^{n_i} \ell \left(h_{\boldsymbol{\phi}_j^{(t-1)}}(\mathbf{x}_s^{(i)}),\ y_s^{(i)} \right) \right]. \tag{17}$$

  Then the variational lower bound becomes

$$\mathcal{L}(q^{(t)}, \Theta) = \frac{1}{n} \sum_i \sum_j w_{ij}^{(t)} \cdot \log p(D_i, z_i = j; \Theta) + C \tag{18}$$

$$= \frac{1}{n} \sum_i \sum_j w_{ij}^{(t)} \cdot (\ \log p(z_i = j; \Pi) + \log p(D_i | z_i = j; \Phi)\ ) + C \tag{19}$$

$$= \frac{1}{n} \sum_i \sum_j w_{ij}^{(t)} \cdot (\ \log \Pi_{ij} + \log p(D_i | z_i = j; \Phi)\ ) + C. \tag{20}$$

- **M-step:** Given the posterior $w_{ij}^{(t)}$ from the E-step, we need to maximize $\mathcal{L}$ w.r.t. $\Theta = (\Phi, \Pi)$. For the priors Π, we can optimize each row i of Π individually since they are decoupled in Eq. (20). Note that each row of Π is also a probability distribution, so the optimum solution is given by $\Pi_{ij}^{(t)} = w_{ij}^{(t)}$. This is because the first term of Eq. (20) for each i is the negative cross entropy, which is maximized when Π_{ij} matches $w_{ij}^{(t)}$.
  Optimizing Eq. (20) w.r.t. Φ gives

$$\Phi^{(t)} \in \operatorname*{argmax}_{\Phi} \mathcal{L}(q^{(t)}, \Theta) = \operatorname*{argmin}_{\Phi} \frac{1}{n} \sum_{i=1}^{K} \sum_{j=1}^{K} w_{ij}^{(t)} \sum_{s=1}^{n_i} \ell \left(h_{\boldsymbol{\phi}_j}(\mathbf{x}_s^{(i)}),\ y_s^{(i)} \right). \tag{21}$$

Posterior and accumulative loss Here we show an alternative implementation for Eq. (3) using accumulative loss. To shorten notations, let $\ell_{ij}^{(t)} := \sum_{s=1}^{n_i} \ell\left(h_{\boldsymbol{\phi}_j^{(t)}}(\mathbf{x}_s^{(i)}),\ y_s^{(i)}\right)$. Combining Eq. (3) and Eq. (4) gives

$$w_{ij}^{(t)} = p(z_i = j | D_i; \Theta^{(t-1)}) \tag{22}$$

$$\propto w_{ij}^{(t-1)} \exp\left[-\ell_{ij}^{(t-1)}\right] \tag{23}$$

$$\propto w_{ij}^{(t-2)} \exp\left[-\left(\ell_{ij}^{(t-2)} + \ell_{ij}^{(t-1)}\right)\right]. \tag{24}$$

We can see that it is accumulating the losses of previous models (e.g., $\boldsymbol{\phi}_j^{(t-2)}$, $\boldsymbol{\phi}_j^{(t-1)}$ and so on) inside the exponential. Therefore, assuming the uniform prior $\Pi_{ij}^{(0)} = 1/K, \forall j$, $w^{(t)}$ is the softmax transformation of the negative of the accumulative loss $L_{ij}^{(t)} := \sum_{\tau=1}^{t-1} \ell_{ij}^{(\tau)}$ up until round t.

B THE FEDERICO ALGORITHM

Algorithm 1 describes our proposed FedeRiCo algorithm.

Algorithm 1: FedeRiCo: Federating with the Right Collaborators

Input: Client local datasets $\{D_i\}_{i=1}^K$, number of communication rounds r, number of neighbors M, ϵ-greedy sampling probability ϵ, momentum for exponential moving average loss tracking β, learning rate η.

Output: Client models $\{\boldsymbol{\phi}_i\}_{i=1}^K$ and client weights w_{ij}.

`// Initialization`

Randomly initialize $\{\boldsymbol{\phi}_i\}_{i=1}^K$;

for *client* C_i *in* $\{C_i\}_{i=1}^K$ **do**

  Initialize $\widehat{L}_{ij}^{(0)} = 0, \ell_{ij}^{(0)} = 0, w_{ij}^{(0)} = \frac{1}{K}$;

end

for *iterations* $t = 1 \dots T$ **do**

  for *client* C_i *in* $\{C_i\}_{i=1}^K$ **do**

    Sample M neighbors of this round B^t according to ϵ-greedy selection w.r.t. $w_{ij}^{(t-1)}$;

    Send $\boldsymbol{\phi}_i$ to other clients that sampled C_i;

    Receive $\boldsymbol{\phi}_j$ from sampled neighbors B^t;

    `// E-step`

    $\ell_{ij}^{(t)} = \ell_{ij}^{(t-1)}$; `// Keep the loss from previous round`

    for *b in* B^t **do**

      $\ell_{ib}^{(t)} = \sum_{s=1}^{n_i} \ell\left(h_{\boldsymbol{\phi}_b^{(t)}}(\mathbf{x}_s^{(i)}),\, y_s^{(i)}\right)$; `// Update the sampled ones`

    end

    $\widehat{L}_{ij}^{(t)} = (1-\beta)\widehat{L}_{ij}^{(t-1)} + \beta\ell_{ij}^{(t)}$; `// Update exponential moving averages`

    $w_{ij}^{(t)} = \frac{\exp(-\widehat{L}_{ij}^{(t)})}{\sum_{j'=1}^K \exp(-\widehat{L}_{ij'}^{(t)})}$;

    `// M-step`

    for C_b *in* B^t **do**

      `// Could also do multiple gradient steps instead`

      Compute and send $\mathbf{g}_{bi} = w_{ib}^{(t)} \nabla_{\boldsymbol{\phi}_b} \sum_{s=1}^{n_i} \ell\left(h_{\boldsymbol{\phi}_b}(\mathbf{x}_s^{(i)}),\, y_s^{(i)}\right)$ to C_b;

    end

    for C_j *that sampled* C_i **do**

      Receive $\mathbf{g}_{ij} = w_{ji}^{(t)} \sum_{s=1}^{n_j} \nabla_{\boldsymbol{\phi}_i} \ell\left(h_{\boldsymbol{\phi}_i}(\mathbf{x}_s^{(j)}),\, y_s^{(j)}\right)$;

    end

    $\boldsymbol{\phi}_i^t = \boldsymbol{\phi}_i^{(t-1)} - \eta \sum_j \mathbf{g}_{ij}$; `// Or any other gradient-based method`

  end

end

C ADDITIONAL EXPERIMENTAL RESULTS

Dirichlet data split Here we compare FedeRiCo with the other baselines with Office-Home dataset using a different data split approach. Specifically, we firstly partition the data labels into 4 clustersm and then distribute data within the same clusters across different clients using a symmetric Dirichlet distribution with parameter of 0.4, as in FedEM Marfoq et al. (2021)[5]. As a result, each client contains a slightly different mixture of the 4 distributions. The results are reported over a single run.

Method	FedAvg	FedAvg+	Local Training	Clustered FL	FedEM	FedFomo	FedeRiCo
Accuracy	69.73 $\pm$ 11.02	71.20 $\pm$ 24.41	68.32 $\pm$ 19.43	69.73 $\pm$ 11.02	47.15 $\pm$ 25.43	75.78 $\pm$ 6.20	**83.90** $\pm$ **4.11**

Table 2: Accuracy of different algorithms with Office-Home dataset and Dirichlet distribution.

Client collaboration Here we include more client weight plots of our proposed FedeRiCo on CIFAR100 with four client distributions using different data partition and training seeds. As shown in Fig. 8 and Fig. 9, clients from the same distribution collaborates has more client weights and more collaboration.

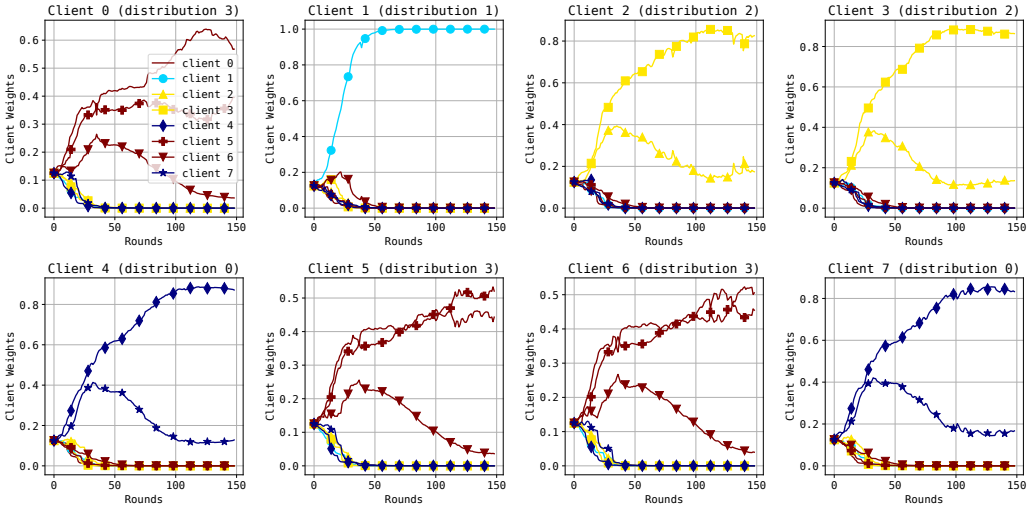

Figure 8: Client weights over time of FedeRiCo with CIFAR100 data and four different client distributions. Clients are color coded by their private data's distribution.

[5]We use the implementation from `https://github.com/omarfoq/FedEM`

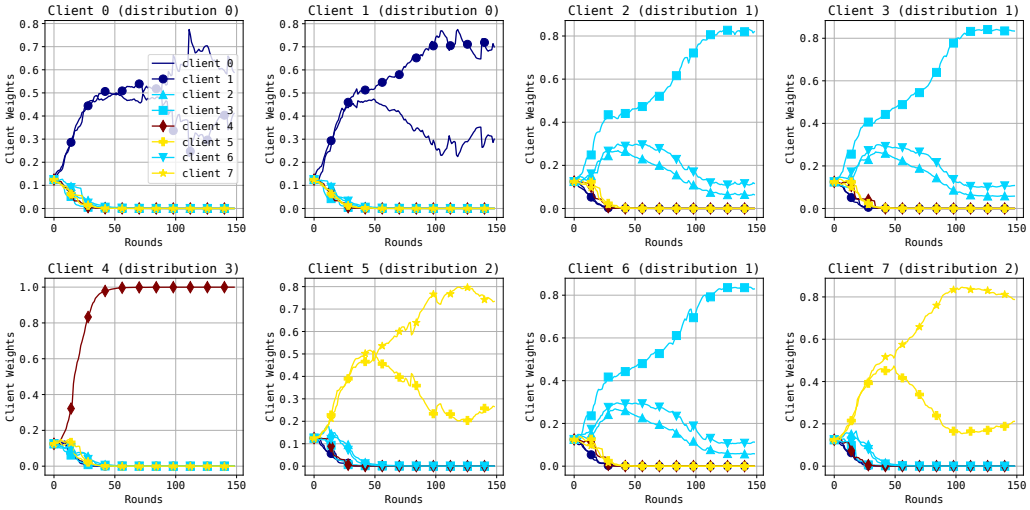

Figure 9: Client weights over time of FedeRiCo with CIFAR100 data and four different client distributions. Clients are color coded by their private data's distribution.

D Additional Experiment Details

Dataset To speed up training, we take 10%, and 15% of the training data from CIFAR-10, and CIFAR-100 respectively. For the Office-Home dataset, we merge images from all domains to get the training dataset, and use the features extracted from the penultimate layer of ResNet-18 pretrained on ImageNet.

Models and Methods For CIFAR-10, we use the CNN2 from Shen et al. (2020) with three 3x3 convolution layers (each with 128 channels followed with 2x2 max pooling and ReLu activation) and one FC layer. For CIFAR-100, we use ResNet-18 as in Marfoq et al. (2021). For Office-Home, the model is an MLP with two hidden layers (1000 and 200 hidden units). The batch size is 50 for CIFAR, and 100 for Office-Home. For FedFomo, we use 5 local epochs in CIFAR-100 to adapt to the noisiness of training and 1 local epoch per communication round for all other experiments.

Settings CIFAR experiments use 8 clients and Office-Home experiments use 10 clients.

Computational resources and software We summarize the computational resources used for the experiments in Table 3 and software versions in Table 4.

Table 3: Summary of computational resource

Operating System	Memory	CPU	GPU
Ubuntu 18.04.5	700GB	Intel(R) Xeon(R) Platinum 8168@2.70GHz	8 Tesla V100-SXM2

Table 4: Software versions

Python	Pytorch	mpi4py
3.9	1.9.0	3.1.2

E CONVERGENCE PROOF

We adapt assumptions 2 to 7 of Marfoq et al. (2021) to our setting as follows:

Assumption E.1. $\forall i \in [K], p_i(x) = p(x)$.

Assumption E.2. *The conditional probability $p_i(y|x)$ satisfies*

$$-\log p_i(y|x) = \ell(h_{\boldsymbol{\phi}_i^*}(x), y) + c, \tag{25}$$

for some parameters $\boldsymbol{\phi}_i^* \in \mathbb{R}^d$, loss function $\ell : \mathcal{Y} \times \mathcal{Y} \mapsto \mathbb{R}^+$ and normalization constant c.

Let $f(\Phi, \Pi) := \frac{1}{n} \log p(D; \Phi, \Pi)$ be the log-likelihood objective as in Eq. (1).

Assumption E.3. *f is bounded below by $f^* \in \mathbb{R}$.*

Assumption E.4 (Smoothness and bounded gradient)**.** *For all x, y, the function $\boldsymbol{\phi} \mapsto \ell(h_{\boldsymbol{\phi}}(x), y)$ is L-smooth, twice continuously differentiable and has bounded gradient: there exists $B < \infty$ such that $\|\nabla_{\boldsymbol{\phi}} \ell(h_{\boldsymbol{\phi}}(x), y)\| \leq B$.*

Assumption E.5 (Unbiased gradients and bounded variance)**.** *Each client $i \in [K]$ can sample a random batch ξ and compute an unbiased estimator $\mathbf{g}_i(\boldsymbol{\phi}, \xi)$ of the local gradient with bounded variance, i.e., $\mathbb{E}_\xi[\mathbf{g}_i(\boldsymbol{\phi}, \xi)] = \frac{1}{n_i} \sum_{s=1}^{n_i} \nabla \ell(h_{\boldsymbol{\phi}}(\mathbf{x}_i^{(s)}), y_i^{(s)})$ and $\mathbb{E}_\xi \|\mathbf{g}_t(\boldsymbol{\phi}, \xi) - \frac{1}{n_i} \sum_{s=1}^{n_i} \nabla \ell(h_{\boldsymbol{\phi}}(\mathbf{x}_i^{(s)}), y_i^{(s)})\| \leq \sigma^2$.*

Assumption E.6 (Bounded dissimilarity)**.** *There exist β and G such that any set of weights $\boldsymbol{\gamma} \in \Delta^K$:*

$$\sum_{i=1}^{K} \frac{n_i}{n} \left\| \frac{1}{n_i} \sum_{s=1}^{n_i} \sum_{j=1}^{K} \gamma_j \nabla \ell(h_{\boldsymbol{\phi}}(\mathbf{x}_i^{(s)}), y_i^{(s)}) \right\|^2 \leq G^2 + \beta^2 \left\| \frac{1}{n} \sum_{i=1}^{K} \sum_{s=1}^{n_i} \sum_{j=1}^{K} \gamma_j \nabla \ell(h_{\boldsymbol{\phi}}(\mathbf{x}_i^{(s)}), y_i^{(s)}) \right\|^2. \tag{26}$$

Theorem 3.1. *[Convergence] Under Assumptions E.1-E.6, when the clients use SGD with learning rate $\eta = \frac{a_0}{\sqrt{T}}$, and after sufficient rounds T, the iterates of our algorithm satisfy*

$$\frac{1}{T} \sum_{t=1}^{T} \mathbb{E} \|\nabla_\Phi f(\Phi^t, \Pi^t)\|_F^2 \leq \mathcal{O}\left(\frac{1}{\sqrt{T}}\right), \qquad \frac{1}{T} \sum_{t=1}^{T} \Delta_\Pi f(\Phi^t, \Pi^t) \leq \mathcal{O}\left(\frac{1}{T^{3/4}}\right), \tag{7}$$

where the expectation is over the random batch samples and $\Delta_\Pi f(\Phi^t, \Pi^t) := f(\Phi^t, \Pi^t) - f(\Phi^t, \Pi^{t+1}) \geq 0$.

Proof: At a high level, we apply the generic convergence result from Marfoq et al. (2021, Thm.3.2') for the proof. Whereas other conditions can be easily verified, we need to find *partial first-order surrogates* (Marfoq et al., 2021, Def.1) g_i and g for f_i and f, respectively, where

$$f_i(\Theta) = f_i(\Phi, \pi_i) := -\frac{1}{n_i} \log p(D_i|\Phi, \pi_i) = -\frac{1}{n_i} \sum_{s=1}^{n_i} \log p(x_i^{(s)}, y_i^{(s)}|\Phi, \pi_i), \tag{27}$$

is the local objective function. In the following, we will verify that

$$g_i^{(t)}(\Phi, \Pi) := g_i^{(t)}(\Phi, \boldsymbol{\pi}_i) \tag{28}$$

$$:= \frac{1}{n_i} \sum_{s=1}^{n_i} \sum_{j=1}^{K} q_j^{(t)} \left[\ell\left(h_{\boldsymbol{\phi}_j}(x_i^{(s)}), y_i^{(s)}\right) - \log p_j(x_i^{(s)}) - \log \pi_{ij} + \log q_j^{(t)} - c \right], \tag{29}$$

$$g^{(t)}(\Phi, \Pi) := \sum_{i=1}^{K} \frac{n_i}{n} g_i^{(t)}(\Phi, \boldsymbol{\pi}_i), \tag{30}$$

satisfy the three conditions of partial first-order surrogates near $(\Phi^{(t-1)}, \Pi^{(t-1)})$: (similarly defined for $g^{(t)}$ and f)

1. $g_i^{(t)}(\Phi, \Pi) \geq f_i(\Phi, \Pi), \forall t, \Phi, \Pi$;

2. $r_i^{(t)}(\Phi, \Pi) := g_i^{(t)}(\Phi, \Pi) - f_i(\Phi, \Pi)$ is differentiable and $\widetilde{L}$-smooth w.r.t. Φ (for some $\widetilde{L} < \infty$). Moreover, $r_i^{(t)}(\Phi^{(t-1)}, \Pi^{(t-1)}) = 0$ and $\nabla_\Phi r_i(\Phi^{(t-1)}, \Pi^{(t-1)}) = \mathbf{0}$;
3. $g_i^{(t)}(\Phi, \Pi^{(t-1)}) - g_i(\Phi, \Pi) = \mathsf{d}(\Pi^{(t-1)}, \Pi)$ for all Φ and $\Pi \in \operatorname{argmin}_{\Pi'} g(\Phi, \Pi')$ where d is non-negative and $\mathsf{d}(\Pi, \Pi') = 0$ iff $\Pi = \Pi'$.

To simplify notations, define the following (the dependency on round t is ignored when it is clear from context)

$$q_j := q_i(z_i = j), \tag{31}$$

$$\mathcal{L}_j := \sum_{s=1}^{n_i} \ell\left(h_{\boldsymbol{\phi}_j}(x_i^{(s)}), y_i^{(s)}\right), \tag{32}$$

$$\gamma_j := p_i(z_i = j | D_i, \Phi, \boldsymbol{\pi}_i). \tag{33}$$

(1) To start verifying the first condition,

$$g_i(\Phi, \boldsymbol{\pi}_i) = \frac{1}{n_i} \sum_{s=1}^{n_i} \sum_{j=1}^{K} q_j \left[\ell\left(h_{\boldsymbol{\phi}_j}(x_i^{(s)}), y_i^{(s)}\right) - \log p_j(x_i^{(s)}) - \log \pi_{ij} + \log q_j - c\right] \tag{34}$$

$$= \frac{1}{n_i} \sum_{s=1}^{n_i} \sum_{j} q_j \left[-\log\left(p_j(y_i^{(s)} | x_i^{(s)}, \boldsymbol{\phi}_j) \cdot p_j(x_i^{(s)}) \cdot p_i(z_i = j)\right) + \log q_j\right] \tag{35}$$

$$= \frac{1}{n_i} \sum_{s=1}^{n_i} \sum_{j} q_j \left[-\log p_i\left(x_i^{(s)}, y_i^{(s)}, z_i = j \middle| \Phi, \boldsymbol{\pi}_i\right) + \log q_j\right] \tag{36}$$

$$= \frac{1}{n_i} \sum_{j} q_j \left[-\log p_i\left(D_i, z_i = j | \Phi, \boldsymbol{\pi}_i\right) + \log q_j\right]. \tag{37}$$

Then

$$r_i(\Phi, \boldsymbol{\pi}_i) = g_i(\Phi, \boldsymbol{\pi}_i) - f_i(\Phi, \boldsymbol{\pi}_i) \tag{38}$$

$$= \frac{1}{n_i} \mathcal{KL}\left(\, q(\cdot) \,\|\, p_t(\cdot | D_i, \Phi, \boldsymbol{\pi}_i) \,\right), \tag{39}$$

where $\mathcal{KL}$ is the KL-divergence. This verifies the first condition of partial first-order surrogates since the KL-divergence is non-negative.

(2) Now we verify the second condition. Note that r_t is twice continuously differentiable due to Assumption E.4. With Assumption E.1

$$\gamma_j = p_i(z_i = j | D_i, \Phi, \boldsymbol{\pi}_i) = \frac{\exp\left[-\mathcal{L}_{j'} + \log \pi_{ij}\right]}{\sum_{j'} \exp\left[-\mathcal{L}_{j'} + \log \pi_{ij'}\right]}, \tag{40}$$

$$\nabla_{\boldsymbol{\phi}_{j'}} \gamma_j = \begin{cases} (-\gamma_j + \gamma_j^2) \nabla \mathcal{L}_j & \text{if } j' = j \\ \gamma_j \gamma_{j'} \nabla \mathcal{L}_{j'} & \text{if } j' \neq j, \end{cases} \tag{41}$$

where $\nabla \mathcal{L}_j$ is shorthand for $\nabla_{\boldsymbol{\phi}_j} \mathcal{L}_j$. Then

$$\nabla_{\boldsymbol{\phi}_{j'}} r_i = \frac{1}{n_i} \nabla_{\boldsymbol{\phi}_{j'}} \sum_{j} (-q_j \log \gamma_j) \qquad \text{Definition of } \mathcal{KL} \tag{42}$$

$$= \frac{1}{n_i} \sum_{j} \left(-\frac{q_j}{\gamma_j} \nabla_{\boldsymbol{\phi}_{j'}} \gamma_j\right) \tag{43}$$

$$= \frac{1}{n_i} \left[q_{j'}(1 - \gamma_{j'}) - \sum_{j \neq j'} q_j \gamma_{j'}\right] \nabla \mathcal{L}_{j'} \qquad \text{When } j = j' \text{ vs } j \neq j' \tag{44}$$

$$= \frac{1}{n_i} \left[q_{j'}(1 - \gamma_{j'}) - (1 - q_{j'}) \gamma_{j'}\right] \nabla \mathcal{L}_{j'} \qquad \sum_j q_j = 1 \tag{45}$$

$$= \frac{1}{n_i} (q_{j'} - \gamma_{j'}) \nabla \mathcal{L}_{j'}. \tag{46}$$

The Hessian of r_i, $\mathbf{H}(r_i) \in \mathbb{R}^{dK \times dK}$ w.r.t. Φ, is a block matrix, with blocks given by

$$\Big(\mathbf{H}(r_t)\Big)_{j,j'} = \begin{cases} \frac{1}{n_i}\left[(q_j - \gamma_j)\mathbf{H}(\mathcal{L}_j) + (\gamma_j - \gamma_j^2)(\nabla\mathcal{L}_j)(\nabla\mathcal{L}_j)^\top\right] \\ -\frac{1}{n_i}\gamma_j\gamma_{j'}(\nabla\mathcal{L}_j)(\nabla\mathcal{L}_{j'})^\top \qquad \text{when } j \neq j', \end{cases} \tag{47}$$

where $\mathbf{H}(\mathcal{L}_j) \in \mathbb{R}^{d\times d}$ is the Hessian of $\mathcal{L}_{\boldsymbol{\phi}_j}(D_t)$ w.r.t. $\boldsymbol{\phi}_j$. Introduce block matrices $\widetilde{\mathbf{H}}, \widehat{\mathbf{H}} \in \mathbb{R}^{dK\times dK}$ as

$$\begin{aligned} \widetilde{\mathbf{H}}_{j,j'} &= \begin{cases} \frac{1}{n_i}(\gamma_j - \gamma_j^2)(\nabla\mathcal{L}_j)(\nabla\mathcal{L}_j)^\top \\ -\frac{1}{n_i}\gamma_j\gamma_{j'}(\nabla\mathcal{L}_j)(\nabla\mathcal{L}_{j'})^\top \qquad \text{when } j \neq j', \end{cases} \\ \widehat{\mathbf{H}}_{j,j'} &= \begin{cases} \frac{1}{n_i}(q_j - \gamma_j)\mathbf{H}(\mathcal{L}_j) \\ \mathbf{0} \qquad \text{when } j \neq j'. \end{cases} \end{aligned} \tag{48}$$

Since $q_j, \gamma_j \in [0,1]$ and ℓ is L-smooth by Assumption E.4, we have $-L\cdot I_{dK} \preccurlyeq \widehat{\mathbf{H}} \preccurlyeq L\cdot I_{dK}$. Using Lemma E.7 (see below), we have $\mathbf{0} \preccurlyeq \widetilde{\mathbf{H}} \preccurlyeq B^2\cdot I_{dK}$ (note that $\nabla\mathcal{L}_j$ is the sum of n_i individual gradients and $\mathbf{H}(r_t)$ has $1/n_i$). As a result, $-\widetilde{L}\cdot I_{dK} \preccurlyeq \mathbf{H}(r_t) \preccurlyeq \widetilde{L}\cdot I_{dK}$ (where $\widetilde{L} = L + B^2 < \infty$) and therefore r_t is $\widetilde{L}$-smooth.

Finally, $q_j^{(t)} = p_i(z_i = j | D_i, \Phi^{(t-1)}, \boldsymbol{\pi}_i^{(t-1)}), \forall t > 0$ by the algorithm, which means

$$r_i^{(t)}(\Phi^{(t-1)}, \Pi^{(t-1)}) = r_i^{(t)}(\Phi^{(t-1)}, \boldsymbol{\pi}_i^{(t-1)}) = 0. \tag{49}$$

Additionally, from Eq. (39) we know that $r_i^{(t)}(\Phi, \boldsymbol{\pi}_i)$ is a (non-negative) KL-divergence for all Φ, Π. Recall that $r_i^{(t)}$ is differentiable. It follows that $\Phi^{(t-1)}$ is a minimizer of the function $\{\Phi \mapsto r_i^{(t)}(\Phi, \boldsymbol{\pi}_i^{(t-1)})\}$ and

$$\nabla_\Phi r_i^{(t)}(\Phi^{(t-1)}, \boldsymbol{\pi}_i^{(t-1)}) = \mathbf{0}. \tag{50}$$

This verifies the second condition of the partial first-order surrogate.

(3) Note that $\boldsymbol{\pi}_i^{(t)} = \operatorname{argmin}_{\boldsymbol{\pi}} g_i^{(t)}(\Phi, \boldsymbol{\pi})$ due to the choice of $q_i^{(t)}$ by the algorithm. Then for any $\boldsymbol{\pi}_i$ and $i \in [K]$,

$$\begin{aligned} g_i^{(t)}(\Phi, \boldsymbol{\pi}_i) - g_i^{(t)}(\Phi, \boldsymbol{\pi}_i^{(t)}) &= \sum_j q_j^{(t)}(\log \pi_{ij}^{(t)} - \log \pi_{ij}) \\ &= \sum_j \pi_{ij}^{(t)}(\log \pi_{ij}^{(t)} - \log \pi_{ij}) \\ &= \mathcal{KL}(\boldsymbol{\pi}_i^{(t)} \| \boldsymbol{\pi}_i), \end{aligned} \tag{51}$$

which is non-negative and equals zero iff $\boldsymbol{\pi}_i^{(t)} = \boldsymbol{\pi}_i$. This verifies the third condition of partial first-order surrogate.

At last, g, f are convex combinations of $\{g_i\}_{i=1}^K, \{f_i\}_{i=1}^K$, respectively, thus the same properties hold between g and f. This completes the proof. ■

Lemma E.7. *Suppose $\mathbf{g}_1, \ldots, \mathbf{g}_K \in \mathbb{R}^d$ and $\boldsymbol{\gamma} = (\gamma_1, \ldots, \gamma_K) \in \Delta^K$. The block matrix $\mathbf{H} \in \mathbb{R}^{dK}$:*

$$\mathbf{H}_{j,j'} = \begin{cases} (\gamma_j - \gamma_j^2)\mathbf{g}_j\mathbf{g}_j^\top \\ -\gamma_j\gamma_{j'}\mathbf{g}_j\mathbf{g}_{j'}^\top \qquad \textit{when } j \neq j', \end{cases} \tag{52}$$

is positive semi-definite (PSD). If in addition $\|\mathbf{g}_j\| \leq B < \infty, \forall j \in [K]$, then $\mathbf{H} \preccurlyeq B^2\cdot I_{dK}$

Proof: Let $\mathbf{x} = [\mathbf{x}_1, \dots, \mathbf{x}_K] \in \mathbb{R}^{dK}$, then

$$\mathbf{x}^\top \mathbf{H} \mathbf{x} = \sum_{j,j'=1}^{K} \mathbf{x}_j^\top \mathbf{H}_{j,j'} \mathbf{x}_j \tag{53}$$

$$= \sum_{j=1}^{K} \left(\mathbf{x}_j^\top \mathbf{H}_{j,j} \mathbf{x}_j + \sum_{j' \neq j} \mathbf{x}_j^\top \mathbf{H}_{j,j'} \mathbf{x}_{j'} \right) \tag{54}$$

$$= \sum_{j=1}^{K} (\gamma_j - \gamma_j)^2 \cdot (\mathbf{x}_j^\top \mathbf{g}_j)^2 - \sum_{j=1}^{K} \left(\sum_{j' \neq j} \gamma_j \gamma_{j'} \cdot (\mathbf{x}_j^\top \mathbf{g}_j) \cdot (\mathbf{x}_{j'}^\top \mathbf{g}_{j'}) \right) \tag{55}$$

$$= \sum_{j=1}^{K} \gamma_j (1 - \gamma_j) \cdot (\mathbf{x}_j^\top \mathbf{g}_j)^2 - \sum_{j=1}^{K} \left(\gamma_j (\mathbf{x}_j^\top \mathbf{g}_j) \cdot \sum_{j' \neq j} \gamma_{j'} \cdot (\mathbf{x}_{j'}^\top \mathbf{g}_{j'}) \right) \tag{56}$$

$$= \sum_{j=1}^{K} \gamma_j \left(\sum_{j' \neq j} \gamma_{j'} \right) \cdot (\mathbf{x}_j^\top \mathbf{g}_j)^2 - \sum_{j=1}^{K} \left(\gamma_j (\mathbf{x}_j^\top \mathbf{g}_j) \cdot \sum_{j' \neq j} \gamma_{j'} \cdot (\mathbf{x}_{j'}^\top \mathbf{g}_{j'}) \right) \tag{57}$$

$$= \sum_{j=1}^{K} \gamma_j (\mathbf{x}_j^\top \mathbf{g}_j) \cdot \sum_{j' \neq j} \gamma_{j'} \left(\mathbf{x}_j^\top \mathbf{g}_j - \mathbf{x}_{j'}^\top \mathbf{g}_{j'} \right) \tag{58}$$

$$= \sum_{j=1}^{K} \gamma_j (\mathbf{x}_j^\top \mathbf{g}_j) \cdot \sum_{j'=1}^{K} \gamma_{j'} \left(\mathbf{x}_j^\top \mathbf{g}_j - \mathbf{x}_{j'}^\top \mathbf{g}_{j'} \right) \tag{59}$$

$$= \sum_{j=1}^{K} \gamma_j (\mathbf{x}_j^\top \mathbf{g}_j)^2 - \left(\sum_{j=1}^{K} \gamma_j \mathbf{x}_j^\top \mathbf{g}_j \right)^2 \tag{60}$$

$$= \mathbb{E}_{j \sim \boldsymbol{\gamma}} [(\mathbf{x}_j^\top \mathbf{g}_j)^2] - \left(\mathbb{E}_{j \sim \boldsymbol{\gamma}} [\mathbf{x}_j^\top \mathbf{g}_j] \right)^2 \tag{61}$$

$$= \mathbb{V}_{j \sim \boldsymbol{\gamma}} [\mathbf{x}_j^\top \mathbf{g}_j] \geq 0, \tag{62}$$

where we have repeatedly applied $\sum \gamma_j = 1$ and $\mathbb{E}, \mathbb{V}$ denote expectation and variance, treating $\mathbf{x}_j^\top \mathbf{g}_j$ as a random variable. As a result, $\mathbf{H}$ is PSD.

Suppose in addition $\|\mathbf{g}_j\| \leq B < \infty, \forall j \in [K]$. Using the Cauchy-Schwarz inequality, we have

$$-B \cdot \|\mathbf{x}_j\| \leq -\|\mathbf{x}_j\| \cdot \|\mathbf{g}_j\| \leq \mathbf{x}_j^\top \mathbf{g}_j \leq \|\mathbf{x}_j\| \cdot \|\mathbf{g}_j\| \leq B \cdot \|\mathbf{x}_j\|. \tag{63}$$

Since $\|\mathbf{x}_j\| \leq \|\mathbf{x}\|, \forall j \in [K]$, we have

$$-B \cdot \|\mathbf{x}\| \leq \mathbf{x}_j^\top \mathbf{g}_j \leq B \cdot \|\mathbf{x}\|. \tag{64}$$

Finally, with the Popoviciu's inequality on variances, we have

$$\mathbf{x}^\top \mathbf{H} \mathbf{x} = \mathbb{V}_{j \sim \boldsymbol{\gamma}} [\mathbf{x}_j^\top \mathbf{g}_j] \leq \frac{1}{4} (B \cdot \|\mathbf{x}\| + B \cdot \|\mathbf{x}\|)^2 = B^2 \|\mathbf{x}\|^2, \tag{65}$$

which means $\mathbf{H} \preccurlyeq B^2 I_{dK}$. ■