图正则化的高斯混合模型学习

1st Shamsiiat Abdurakhmanova
dept. of Computer Science
Aalto University
Helsinki, Finland
shamsiiat.abdurakhmanova@aalto.fi

2nd Alex Jung
dept. of Computer Science
Aalto University
Helsinki, Finland
alex.jung@aalto.fi

摘要—我们提出了一种在分布式设置下具有异构和有限本地数据的高斯混合模型（GMM）的图正则化学习方法。该方法利用提供的相似性图来指导节点之间的参数共享，避免传输原始数据。得到的模型允许灵活地聚合邻居的参数，并在异构、低样本量的情况下优于集中式和局部训练的 GMM。

***Index Terms*—高斯混合模型，联邦学习，图正则化，期望最大化，分布式学习等术语**

I. 介绍

我们提出了 GraphFed-EM，这是一种联邦高斯混合模型，在该模型中，局部节点通过基于图的正则化协同学习个性化的概率模型，而无需交换原始数据。算法适应了分散式设置，并包含了促进连接节点间参数相似性的聚合步骤。

GraphFed-EM 对异构的本地数据集集合有益，在这些情况下，单一全局 GMM 可能因数据集差异而失效，并且在小样本条件下（样本量小或高维特征）进行本地训练具有挑战性。我们通过合成聚类数据集来说明这一点，在每个聚类内的数据集共享相同的 GMM 参数，但在不同的聚类之间有所不同。我们还在具有偏斜特征分布的合成和 MNIST 数据集上测试了我们的算法。与先前假设客户端间 GMM 参数相同的联邦 EM 方法不同，GraphFed-EM 通过相似性图灵活聚合参数，同时尊重客户端特定的变化。

论文的其余部分组织如下：第 II 节回顾了高斯混合模型和 EM 算法；第 III 节介绍了 GraphFed-EM；第 IV 节讨论了相关工作；第 V 节报告了实验结果；第 VI 节解释了将 GraphFed-EM 视为正则化 EM 的解读；第 VII 节进行了总结。

Funded by the Research Council of Finland (Decision #363624), Jane and Aatos Erkko Foundation (Decision #A835), Business Finland

II. 有限高斯混合模型的预备知识

高斯混合分布可以写成 K 高斯分布的加权和 [1]：

$$p(\boldsymbol{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\boldsymbol{x} \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \tag{1}$$

其中，π_k 是成分 k 的混合权重，$\boldsymbol{\mu}_k$ 和 $\boldsymbol{\Sigma}_k$ 分别是成分 k 的均值向量和协方差矩阵，而 $\mathcal{N}(\boldsymbol{x} \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$ 是多元高斯概率密度函数。给定一组独立同分布的观测值 $\boldsymbol{X} = (\boldsymbol{x}_1^T, \ldots, \boldsymbol{x}_N^T)$，数据可以使用高斯混合模型进行建模，对数似然为

$$\ln p(\boldsymbol{X} \mid \boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \sum_{n=1}^{N} \ln \left\{ \sum_{k=1}^{K} \pi_k \mathcal{N}(\boldsymbol{x} \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \right\}$$

。最大化此函数最常用的方法是期望最大化（EM）算法 [2]。简而言之，算法的 E 步包括计算每个数据点 $\boldsymbol{x}_n$ 和组件 k 的责任 γ_{kn}：

$$\gamma_{kn} = \frac{\pi_k \mathcal{N}\left(\boldsymbol{x}_n \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\right)}{\sum_{j=1}^{K} \pi_j \mathcal{N}\left(\boldsymbol{x}_n \mid \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j\right)}.$$

和 M 步则根据封闭形式的解进行参数更新步骤：

$$\pi_k = \frac{N_k}{N}, \quad \text{with } N_k = \sum_{i=1}^{N} \gamma_{nk}$$

$$\boldsymbol{\mu}_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma_{nk} \boldsymbol{x}_n$$

$$\boldsymbol{\Sigma}_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma_{nk} (\boldsymbol{x}_n - \boldsymbol{\mu}_k)(\boldsymbol{x}_n - \boldsymbol{\mu}_k)^T$$

我们通过将局部 GMM 均值设置为 KMeans 中心点来初始化 EM 算法，并从由 KMeans 聚类分配诱导的数据集分割中估计协方差和混合权重。为了在训练期间确保数值稳定性，我们使用收缩和对角加载技术来正则化协方差矩阵。

III. 图正则化 EM 算法

考虑一个由节点（客户端）通过邻接矩阵 A 连接的网络（图）$\mathcal{G}$。每个节点 i 持有一个局部数据集 $\mathcal{D}^{(i)} := \{\boldsymbol{x}_1^{(i)}, \ldots, \boldsymbol{x}_{N_i}^{(i)}\}$，其中每个样本 $\boldsymbol{x}_r^{(i)} \in \mathbb{R}^d$ 都是从方程 1 中的高斯混合模型中抽取的。节点 i 通过权重为 A_{ij} 的无向边与其邻居 $j \in \Gamma(i)$ 连接。假设该图编码了相连节点之间的 GMM 参数相似性。

GraphFed-EM 训练包括 T_i 次本地 EM 更新（第 II 节），随后进行一次聚合步骤。每个节点 i 共享其局部估计的 GMM 参数 $\boldsymbol{\theta}^i = \{\boldsymbol{\mu}^i, \boldsymbol{\Sigma}^i, \boldsymbol{\pi}^i\}$ 与邻居，以计算自身和邻居参数的加权平均值。由于组件的顺序是任意的，我们首先使用来自 SciPy 的线性求和分配求解器对 Bhattacharyya 距离矩阵上的 K 组件进行对齐。对齐的参数按如下方式聚合：

$$\boldsymbol{\theta}^i_{aggr,k} = \frac{N_k^i \boldsymbol{\theta}_k^i + \sum_{j\in\Gamma(i)} A_{ij} N_k^j \boldsymbol{\theta}_k^j}{N_k^i + \sum_{j\in\Gamma(i)} A_{ij} N_k^j} \tag{2}$$

通过使用 $\sum_{k=1}^K N_k^i + \sum_{j\in\Gamma(i)} A_{ij} N_k^j$ 归一化的混合系数来确保 $\sum_{k=1}^K \pi_k^i = 1$。换句话说，这对应于 GMM 参数的责任加权聚合。此外，参数聚合的程度由 α 控制：

$$\boldsymbol{\theta}_k^i \leftarrow (1-\alpha) * \boldsymbol{\theta}_k^i + \alpha * \boldsymbol{\theta}^i_{aggr,k} \tag{3}$$

聚合后，协方差矩阵被对称化并进行对角正则化以确保正定性。

IV. 相关工作

我们的 GraphFed-EM 算法是正则化 EM 的一个实例（参见第 VI 节）。在样本量较少的情况下，EM 中的正则化已被用于提高稳定性和收敛性 [3], [4]，并简化具有冗余组件的混合模型 [5]。

随着数据量的增长和设备的限制，分布式 EM 算法变得必要。大多数方法采用客户端-服务器设置来训练并共享单一全局模型 [6]，通常假设数据是独立同分布的。一些工作解决了数据异构性的问题：例如，FedGenGMM [7] 通过重新加权组件聚合客户机训练的 GMMs，生成合成数据，训练全局模型，并重新分配它。[8] 解决了共享全球 GMM 参数的估计问题，同时引入了一种适用于非独立同分布数据和部分参与的联邦学习中通信高效的 EM 算法。与我们的方法密切相关，FedGrEM [9] 使用带有正则化项的基于梯度的 EM 算法，该正则化项惩罚偏离中心聚合参数的行为。

Algorithm 1 图正则化 EM 算法

输入： 经验图 $\mathcal{G} := \{\mathcal{V}, \mathcal{E}\}$ 具有 N 个节点和邻接矩阵 A。本地数据集 $\mathcal{D}^{(i)}$ 在每个节点上，$|\mathcal{D}^{(i)}| = N_i$，组件数量 K，迭代次数 T，局部迭代次数 T_i，聚合强度 α
初始化： 高斯混合模型参数 $\boldsymbol{\theta}^{(i,0)} = (\boldsymbol{\pi}^{(i,0)}, \boldsymbol{\mu}^{(i,0)}, \boldsymbol{\Sigma}^{(i,0)})$ 对应于每个节点 $i \in \mathcal{V}$
for $t = 1$ to T **do**
  for each node i **do**
    for $t_i = 1$ to T_i **do**
      Compute $\boldsymbol{\theta}^{(i,t_i)}$ with local EM updates
    end for
    Share local estimates $\boldsymbol{\theta}^{(i,T_i)}$ with neighbors $\Gamma(i)$
  end for
  for each node i **do**
    更新 $\boldsymbol{\theta}^{(i,T_i)} \leftarrow \boldsymbol{\theta}^{(i,T_i)}$ 通过 3
  end for
end for
return $\{\boldsymbol{\theta}^{(i,T)}\}_{i\in\mathcal{V}}$

V. 数值实验

我们在合成数据集和 MNIST 数据集上评估 GraphFed-EM，这些数据集模拟了常见的客户端异质性。合成数据集包括：(i) 参数为 $\{\boldsymbol{\theta}^{(c)}\}_{c=1}^C$ 的聚类 GMMs，其中具有相同参数的数据集形成一个集群 $\mathcal{C}^{(c)}$；以及 (ii) 一种共享混合模型，带有节点特定的混合系数 $\boldsymbol{\pi}^i$。MNIST 用于在客户端之间创建偏斜的标签分布。性能通过平均对数似然和归一化互信息 (NMI) [10] 在本地验证集（样本大小 $N_i^{(val)} = 500$）上进行评估，以量化聚类准确性。我们也报告了质心估计误差：

$$\mu_{err} = \frac{1}{NK} \sum_{i\in\mathcal{V}} \sum_{k=1}^K \left\| \boldsymbol{\mu}_k^{c_i} - \boldsymbol{\mu}_k^i \right\|_2^2 \tag{4}$$

其中 $\boldsymbol{\mu}_k^{c_i}$ 表示用于生成节点 i 数据的真实聚类质心。组件使用 Bhattacharyya 距离进行匹配（第 III 节）。

所有实验重复进行 10 次，均值和标准误差以 y 误差线或阴影区域显示。除非另有说明，GraphFed-EM 使用聚合强度 $\alpha = 1$，$T = 10$ 迭代次数，以及 $T_i = 5$ 局部 EM 步骤。这些和其他实验的代码可以在 GitHub 找到。

A. 合成数据集 - 聚类设置

我们在一个由图 $\mathcal{G}$ 表示的合成数据集上测试 GraphFed-EM，该数据集包含 N 个节点，这些节点被划分为 C 个等大小的簇 $\{\mathcal{C}^{(c)}\}_{c=1}^{C}$，其中节点 i 属于一个簇 $\mathcal{C}^{(c_i)}$。边是独立的伯努利变量 $b_{ij} \in \{0, 1\}$，如果 i 和 j 共享一个聚类，则以概率 p_{in} 出现，否则为 p_{out}；边是没有权重的，所以当 $b_{ij} = 1$ 时为 $A_{ij} = 1$。每个节点 i 持有从具有参数 $(\mu_k^{(c_i)}, \Sigma_k^{(c_i)}, \pi_k^{(c_i)})$ 的 K 组件 GMM 中采样的本地数据 $\mathcal{D}^{(i)} = \{\boldsymbol{x}^{(i,1)}, \ldots, \boldsymbol{x}^{(i,N_i)}\}$，这些参数在同一簇中的节点间共享。为了引入异质性，均值和协方差矩阵在所有簇中一致地经历了随机旋转和平移变换。

在什么情况下汇集局部数据集是有意义的？我们将局部训练（每个本地数据集独立进行）、集中式训练（所有数据集合并）以及 GraphFed-EM（采用直接聚类池化或聚合步骤（算法 1））进行了比较；后两者，给定真实的聚类身份，则被称为 Oracle。为了孤立地分析聚合相对于池化的益处，我们还在基于聚类的池化数据集上训练了 GraphFed-EM。

实验使用了 $C = 5$ 个集群，每个集群有 $N = 25$ 个节点（每群 5 个），$K = 3$ 个组件，特征维度为 $d \in \{2, 6, 10\}$，局部训练样本大小为 $N_i \in \{10, 50, 100\}$。连接矩阵设置为 $p_{in} = 1, p_{out} = 0$。验证集大小为 $N_i^{(val)} = 500$，按聚类标签进行了分层。

结果（图 1）显示各指标之间存在差异。集中式 EM 匹配了 Oracle 对数似然值为 $d = 2, 6$，但一直产生较低的 NMI，可能是由于低维度下聚类 GMM 之间的重叠导致。局部训练在 $N_i = 10$ 时表现不佳，但在 N_i 增加时接近 GraphFed-EM。参数误差 μ_{err} 对于 Oracle 是最小的，无论是池化还是聚合。

总体而言，当 d 较高且 N_i 较小时，合作最为有利。直接汇聚和 GraphFed-EM 在 $\alpha = 1$ 上表现相似，除了在 $N_i = 10$ 处，汇聚产生的对数似然更高但在 NMI 或 μ_{err} 方面没有明显优势。

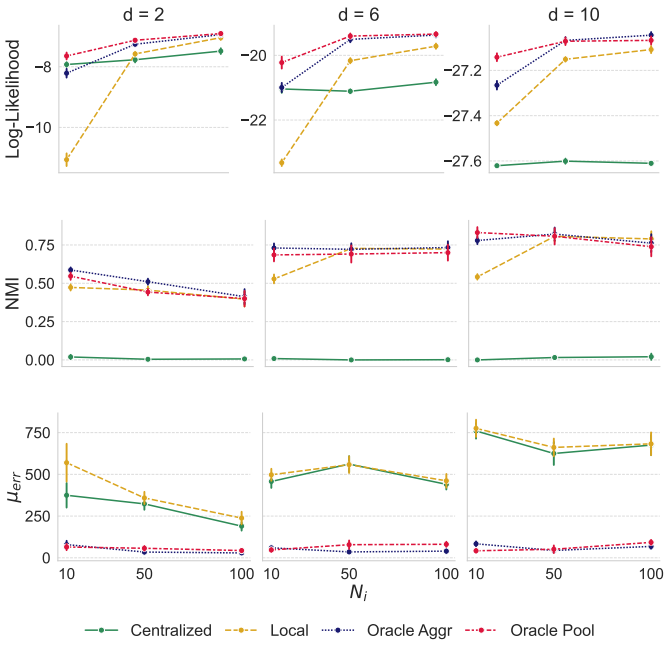


图 1: 不同维度的方法比较。本地、集中式和 GraphFed-EM 训练，通过聚集步骤直接或间接地将局部数据集按簇汇集。包含 $C = 5$ 簇，有 $N = 25$ 节点，以及 $K = 3, p_{in} = 1, p_{out} = 0$。

聚类中心共识。接下来，我们研究了聚合强度 α 和图的连通性如何影响聚类质心的共识。我们计算估计均值向量 $\boldsymbol{\mu}_k^i$ 与邻居节点共识 $\boldsymbol{\mu}_k^{i,cons}$ 之间的加权平方范数：

$$\mu_{err}^{cons} = \frac{1}{NN_i} \sum_{i \in \mathcal{V}} \sum_{k=1}^{K} N_k^i \left\| \boldsymbol{\mu}_k^i - \boldsymbol{\mu}_k^{i,cons} \right\|_2^2 \tag{5}$$

其中

$$\boldsymbol{\mu}_k^{i,cons} = \frac{\sum_{j \in \Gamma(i)} A_{ij} N_k^j \boldsymbol{\mu}_k^j}{\sum_{j \in \Gamma(i)} A_{ij} N_k^j} \tag{6}$$

图 2 显示，增加连接概率 p_{in} 和聚集强度 α 会导致共识误差 μ_{err}^{cons} 更小。

***GraphFed-EM* 对集群外连接的鲁棒性**。So far, we assumed the "true" connectivity matrix, where only nodes in the same cluster are connected. Next, we investigate how spurious connections between different clusters affect performance in a low sample support setting. We test out-of-cluster connectivity probabilities $p_{out} \in \{0, 0.2, 0.4\}$ and varying aggregation strength

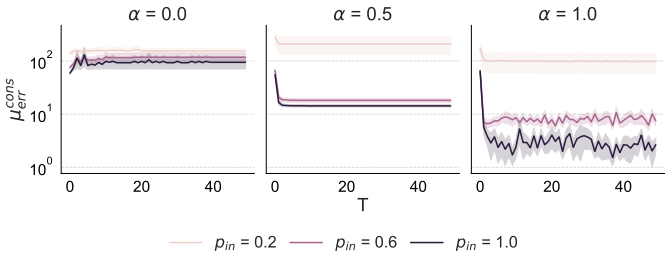


图 2: GraphFed-EM 共识误差在一个包含 $N = 10$ 个节点的集群上，以及 $N_i = 10, d = 10, K = 3, T = 50$。

α. The number of components is set to $K = 3$, local sample size $N_i = 10$, $d = 10$, and $p_{in} = 1$.

如图 3 所示，当 $p_{out} = 0$ 时，增加 α 可提高 NMI。对于 $p_{out} > 0$，较大的 α 则会导致性能下降。在这些情况下，中等的聚合强度达到最佳结果，与 Oracle 性能相匹配（GraphFed-EM 使用真实的连通性矩阵）。具体来说，$\alpha = 0.4$ 对于 $p_{out} \in 0.2, 0.4$ 是最优的。

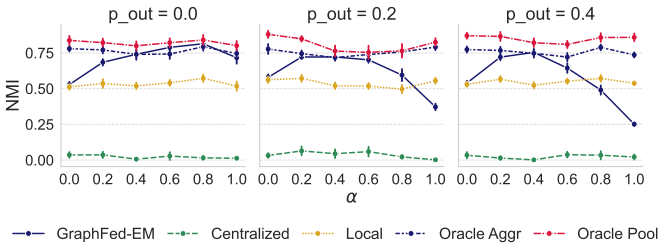


图 3: 簇外连接 p_{out} 和聚合强度 α 对 GraphFed-EM 性能的影响。

B. 合成数据集 - 客户特定的先验 π_k^i

我们使用一个具有共享 GMM 参数但因不同的混合系数而特征分布偏斜的数据集 π_k^i。与 [7] 不同，我们假设这种异质性在推理过程中保持固定，并且使用相同的客户端特定的先验 π_k^i 生成验证集以供训练。我们通过测量分布之间的重叠来定义邻接矩阵 A:

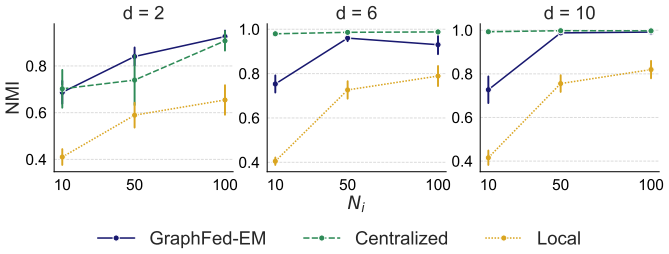
$$A_{ij} = \frac{\sum_{k=1}^{K} \min(\pi_k^i, \pi_k^j)}{\sum_{j \in \Gamma(i)} \sum_{k=1}^{K} \min(\pi_k^i, \pi_k^j)} \tag{7}$$

我们可以从图 4a 看出，集中式模型在更高维度下最符合数据 $d = 6, 10$。GraphFed-EM 超过了局部训练，但不如集中式训练。因此，在客户端特定先验的情况下，对于样本量较小的局部数据集 $N_i = 10$，集中式解决方案是最优的。

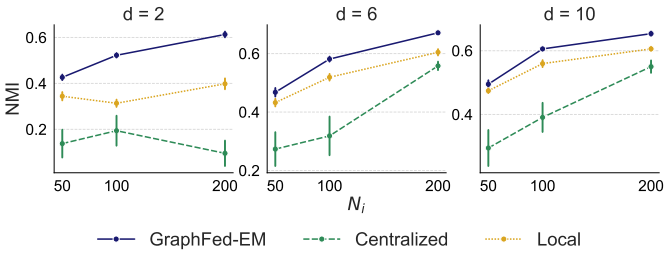
C. MNIST - 偏斜的标签分布

我们构建了一个使用 MNIST 的具有客户特定先验的类似数据集 π_k^i。该数据集由 28x28 灰度数字图像（0 到 9）组成。我们首先应用 UMAP [11] 进行降维到 $d = \{2, 6, 10\}$，并设置 $K = 10$，假设每个数字有一个组件。数据分布在使用狄利克雷分布（集中参数 $\alpha = 0.3$）的 $N = 10$ 节点上，标签比例倾斜。局部样本量是 $N_i = \{50, 100, 200\}$。相似性图的构建如第 V-B 节所述，基于节点之间的标签重叠。

如图 4b 所示，GraphFed-EM 在所有 (d, N_i) 对中均实现了高于局部和集中式 GMM 的 NMI 值。



(a) GraphFed-EM 在具有客户端特定先验的 数据集 π^i 上的表现



(b) GraphFed-EM 在带有偏斜标签分布的 MNIST 数据集上的性能。$N = 10, K = 10$ 和 $T = 10$。

图 4: GraphFed-EM 在具有客户端特定先验的 数据集和标签分布偏斜的 MNIST 数据集上的性能。

VI. 基于图的聚合作为正则化 EM

令 θ_k^i 表示节点 i 中组件 k 的一个通用参数（例如，$\boldsymbol{\mu}_k^i$、$\boldsymbol{\Sigma}_k^i$ 或 π_k^i），并令 $\Gamma(i)$ 表示经验图中节点 i 的邻居集合。令 N_k^i 为节点 i 中组件 k 的责任加权样本数。为了简化，我们假设连接节点的 GMM 组件完全匹配。

我们可以将 GraphFed-EM 视为正则化 EM 的一个实例，其中局部 EM 目标函数通过图上的平滑惩罚得到增强:

$$\ell_i(\theta_k^i) + \frac{\lambda}{2} \sum_{j \in \Gamma(i)} A_{ij} N_k^j \left\| \theta_k^i - \theta_k^j \right\|_2^2 \tag{8}$$

其中第一项是局部的负 Q-函数（对数似然 [1] 的上限），$\ell_i(\theta_k^i) = -Q_i(\theta_k^i)$，第二项则惩罚相邻节点之间的差异。

如果 $\theta_k^{i,\mathrm{EM}}$ 是未惩罚的 EM 极大化器 $\ell_i(\theta_k^i)$，那么 $\nabla \ell_i(\theta_k^{i,\mathrm{EM}}) = 0$。关于在 $\theta_k^{i,\mathrm{EM}}$ 处的惩罚项的近端梯度步是：

$$\begin{aligned}\theta_k^{i,\mathrm{new}} &= \theta_k^{i,\mathrm{EM}} - \eta\lambda \sum_{j\in\Gamma(i)} A_{ij} N_k^j \big(\theta_k^{i,\mathrm{EM}} - \theta_k^j\big) \\ &= (1-\alpha)\,\theta_k^{i,\mathrm{EM}} + \alpha \frac{N_k^i \theta_k^{i,\mathrm{EM}} + \sum_j A_{ij} N_k^j \theta_k^j}{N_k^i + \sum_j A_{ij} N_k^j}\end{aligned}$$

其中 $\alpha = \eta\lambda(N_k^i + \sum_j A_{ij} N_k^j)$。这等价于聚合步骤方程 3。这表明 GraphFed-EM 中的基于图的聚合步骤可以解释为正则化 EM 目标上的近似更新：每个节点将其参数更新为其邻居参数的加权平均值，权重由有效样本量 N_k^j 和边权重 A_{ij} 给出。因此，这种聚合平衡了局部证据（来自节点 i 自身的数据）与基于图的平滑处理（来自邻居的信息），这为 GraphFed-EM 与正则化 EM 之间建立了联系。

VII. 结论

我们介绍了一种适用于高斯混合模型的简单分散式图正则化联邦学习算法。我们的方法专为点对点协作学习设计，并在异构客户端环境中特别有效。与集中式解决方案不同，它利用基于图形的正则化来促进相关客户端之间的信息共享，同时尊重本地数据特征。数值实验表明，GraphFed-EM 有效地平衡了连接节点之间达成共识和优化局部目标之间的权衡。特别是在样本量较低的情况下，当数据维度与本地样本数量相当或更大时，它在性能上超过了本地训练和集中式训练。我们的方法不要求所有客户端全部参与，并且可以根据客户端特定的通信和计算约束进行调整。未来扩展包括在通信轮次中动态推断相似性图，类似于集群联邦学习 [12]，以及分析保证恢复局部 GMM 参数和算法收敛所需的连通性条件。

VIII. 遵守伦理标准

这是一项数值模拟研究，无需伦理审批。

参考文献

[1] C. M. Bishop and N. M. Nasrabadi, *Pattern recognition and machine learning*. Springer, 2006, vol. 4, no. 4.

[2] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the royal statistical society: series B (methodological)*, vol. 39, no. 1, pp. 1–22, 1977.

[3] P. Houdouin, E. Ollila, and F. Pascal, "Regularized em algorithm," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023, pp. 1–5.

[4] X. Yi and C. Caramanis, "Regularized em algorithms: A unified framework and statistical guarantees," *Advances in neural information processing systems*, vol. 28, 2015.

[5] H. Li, K. Zhang, and T. Jiang, "The regularized em algorithm," in *AAAI*, 2005, pp. 807–812.

[6] Q. Zhang and J. Chen, "Distributed learning of finite gaussian mixtures," *Journal of Machine Learning Research*, vol. 23, no. 99, pp. 1–40, 2022.

[7] S. Z. Pettersson, K.-Y. Liang, and J. C. Andresen, "Federated gaussian mixture models," *arXiv preprint arXiv:2506.01780*, 2025.

[8] A. Dieuleveut, G. Fort, E. Moulines, G. Robin, and F. Evry-Courcouronnes, "Federated expectation maximization with heterogeneity mitigation and variance reduction," *arXiv preprint arXiv:2111.02083*.

[9] Y. Tian, H. Weng, and Y. Feng, "Towards the theory of unsupervised federated learning: Non-asymptotic analysis of federated em algorithms," *arXiv preprint arXiv:2310.15330*, 2023.

[10] A. Strehl and J. Ghosh, "Cluster ensembles—a knowledge reuse framework for combining multiple partitions," *Journal of machine learning research*, vol. 3, no. Dec, pp. 583–617, 2002.

[11] L. McInnes, J. Healy, and J. Melville, "Umap: Uniform manifold approximation and projection for dimension reduction," 2020. [Online]. Available: https://arxiv.org/abs/1802.03426

[12] A. Ghosh, J. Chung, D. Yin, and K. Ramchandran, "An efficient framework for clustered federated learning," *Advances in neural information processing systems*, vol. 33, pp. 19 586–19 597, 2020.