Fed-SC: One-Shot Federated Subspace Clustering over High-Dimensional Data

Songjie Xie[1], Youlong Wu[1], Kenwen Liao[2], Lu Chen[3], Chengfei Liu[3], Haifeng Shen[2], MingJian Tang[4], Lu Sun[1]

[1]ShanghaiTech University, [2]Australian Catholic University, [3]Swinburne University of Technology, [4]Atlassian

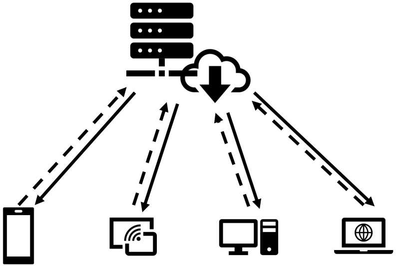
Apr 4, 2023

Outline

1 Introduction

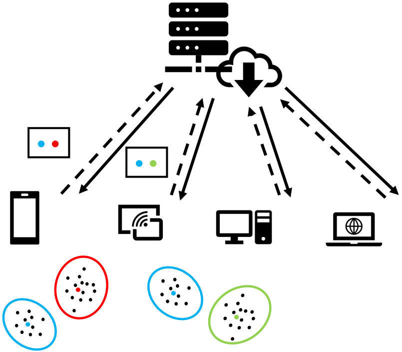
Introduction: Federated Learning

- Decentralized approach to ML
- Cooperative training without sharing raw data
- Widely applicable in supervised ML models



Introduction: Federated Clustering

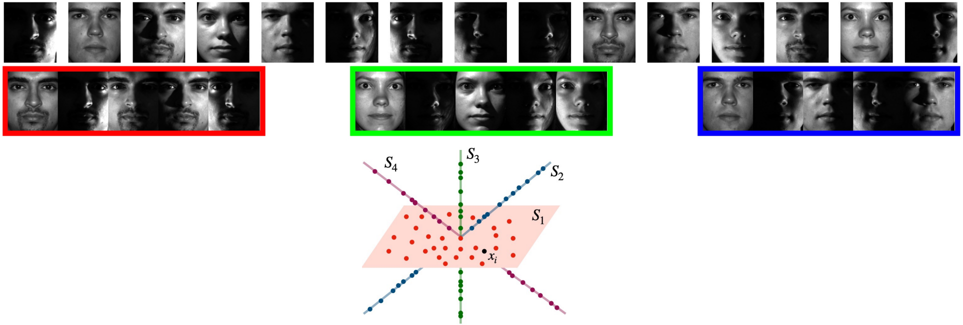
- k-means based one-shot federated clustering: k-FED.
- **Practical applications**: clustering medical, image, or genomics data resided at different nodes.



D. K. Dennis, T. Li, and V. Smith, "Heterogeneity for the win: One-shot federated clustering," in International Conference on Machine Learning. PMLR, 2021, pp. 2611-2620.

Introduction: Subspace Clustering

In many applications, high-dimensional data can be well represented by a union of low-dimensional subspaces.



E. Elhamifar and R. Vidal, "Sparse subspace clustering: Algorithm, theory, and applications," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 35, no. 11, pp. 2765-2781, 2013.

Motivation and Challenge – Federated Learning meets Subspace Clustering

- Lack of previous studies on federated clustering for high-dimensional data.
- Unique requirements of federated learning
  - Communication efficiency
  - Privacy-preserving: learning without sharing data
- Effectiveness of subspace clustering
  - Empirical performance on real-world high-dimensional datasets
  - Theoretical guarantee

2 Problem Formulation

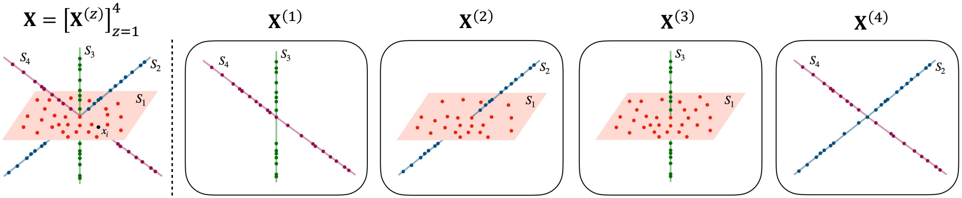
Centralized Subspace Clustering

1. Constructing an affinity graph $\mathbf{W} \in \mathbb{R}^{n\times n}$
   - Sparse Subspace Clustering (SSC): $\mathbf{W} = |\mathbf{C}|+|\mathbf{C}|^T$, $\mathbf{C} = [\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_N]$,
   $$\min_{\mathbf{c}_i\in\mathbb{R}^N} \frac{\lambda}{2}\|\mathbf{X}\mathbf{c}_i - \mathbf{x}_i\|_2^2 + \|\mathbf{c}_i\|_1, \quad \text{s.t.} \quad c_{ii} = 0.$$
   - Thresholding-based subspace clustering (TSC): calculate and threshold the cosine distances between data points
2. Applying spectral clustering on $\mathbf{W}$ to generate L clusters

Federated Subspace Clustering

Given data $\mathbf{X}$ residing in a federated network with Z devices, federated SC aims to cluster $\mathbf{X}$ into L classes according to the global subspaces $\{\mathcal{S}_\ell\}_{\ell=1}^{L}$ they lie.

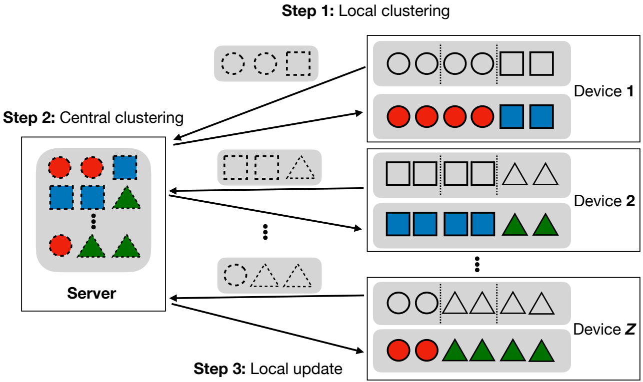
Statistical heterogeneity: there exists at least one device z such that the number of local clusters is smaller than the number of total clusters, $L^{(z)} < L$.



3 Method: Fed-SC

Main Steps:

- Local clustering and sampling
- Central clustering
- Local update



Fed-SC: Local Clustering and Sampling

Local clustering (at device z):

1. Run SSC on $\mathbf{X}^{(z)}$ to obtain $\mathbf{C}^{(z)}$ and form an affinity graph $\mathbf{W}^{(z)} = |\mathbf{C}^{(z)}| + |\mathbf{C}^{(z)}|^T$
2. Use eigengap heuristic to estimate the number of clusters $r^{(z)}$
3. Apply spectral clustering to segment local data points into $r^{(z)}$ clusters $T^{(z)} = (T_i^{(z)})_{t=1}^{r^{(z)}}$

Random sampling:

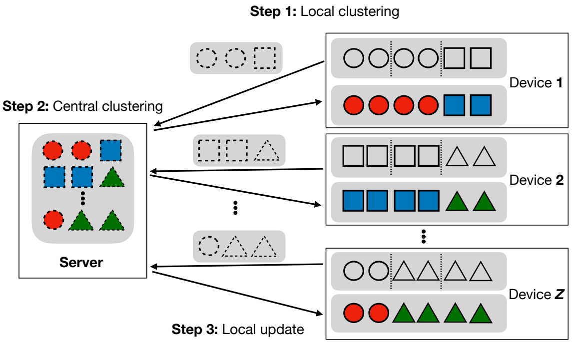
1. Estimate the orthogonal basis $\mathbf{U}_{d_t}^{(z)}$ from data points in $T_t^{(z)}$.
2. Sample the coefficient $\boldsymbol{\alpha}_t^{(z)} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, and generate $\boldsymbol{\theta}_t^{(z)} \in \text{span}(\{\mathbf{x}_i^{(z)}\}_{i \in T_t^{(z)}})$ by

$$\boldsymbol{\theta}_t^{(z)} = \frac{\mathbf{U}_{d_t}^{(z)} \boldsymbol{\alpha}_t^{(z)}}{\|\mathbf{U}_{d_t}^{(z)} \boldsymbol{\alpha}_t^{(z)}\|_2}.$$

$$\boldsymbol{\Theta}^{(z)} = [\boldsymbol{\theta}_1^{(z)}, \boldsymbol{\theta}_2^{(z)}, \ldots, \boldsymbol{\theta}_{r^{(z)}}^{(z)}]$$

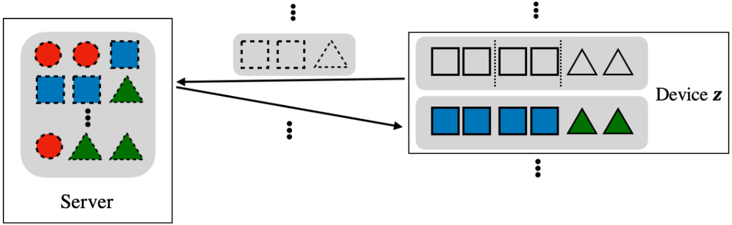
Fed-SC: Central Clustering and Local Update

- Central clustering: The server runs SC algorithms (TSC or SSC) to segment $[\boldsymbol{\Theta}^{(z)}]_{z=1}^{Z}$ into L clusters
- Local update: Each client z updates $T^{(z)}$ into $\hat{T}^{(z)} = (\hat{T}_\ell^{(z)})_{\ell=1}^{L}$ by $\hat{T}_\ell^{(z)} = \{i : i \in T_t^{(z)} \text{ and } \tau_t^{(z)} = \ell\}$



Fed-SC (SSC) and *Fed-SC (TSC)* to denote the Fed-SC methods where SSC and TSC are implemented at the central server, respectively.

4 Effectiveness Guarantees



	Central Clustering	Local Clustering
Data	$\mathbf{\Theta}$	$\mathbf{X}^{(z)}$
Data model	Semi-random model	Deterministic model
Algorithm	SSC/TSC	SSC
Criteria	SEP/Exacting Clustering	SEP

Local clustering: Active Deterministic Condition

Assume that each $\mathbf{X}^{(z)}$ is in general position and the non-zero $N_\ell^{(z)} \geq d_\ell + 1$ for all $\ell \in [L]$ and $z \in [Z]$. Let $r' = \max_{z\in[Z]} r^{(z)}$, $N'_\ell = \min\{N_\ell^{(z)} | N_\ell^{(z)} > 0\}_{z\in[Z]}$ and $\mathbb{W}_\ell^{N'_\ell}$ be the set of all sub-matrices of X_ℓ with N'_ℓ columns. If for each $\ell \in [L]$, The active deterministic condition

$$\min_{\tilde{\mathbf{X}}_\ell \in \mathbb{W}_\ell^{N'_\ell}} \min_{i:x_i \in \tilde{\mathbf{X}}_\ell} r(\mathcal{P}(\tilde{\mathbf{X}}_{\ell,-i})) > \tilde{\mu}(\mathbf{X}_\ell), \text{ for each } \ell \in [L]$$

Central clustering: Global Semi-random Condition

Z_ℓ: Number of subspaces where the local data $\mathbf{X}^{(z)}$ is distributed, d_ℓ:Dimension of subspace $\mathcal{S}_\ell$.

- Fed-SC (SSC):

$$\boxed{c\sqrt{\log\frac{Z_\ell - 1}{d_\ell}} > \max_{k:k\neq l}\ t\log[Lr'Z_\ell(r'Z_k+1)]\frac{\operatorname{aff}(\mathcal{S}_\ell,\mathcal{S}_k)}{\sqrt{d_k}}}$$

- Fed-SC (TSC):

$$\boxed{\max_{\ell,k:k\neq\ell}\frac{\operatorname{aff}(\mathcal{S}_\ell,\mathcal{S}_k)}{\sqrt{d_\ell\wedge d_k}} \leq (15\log\sum_{\ell\in[L]} r'Z_\ell)^{-1}}$$

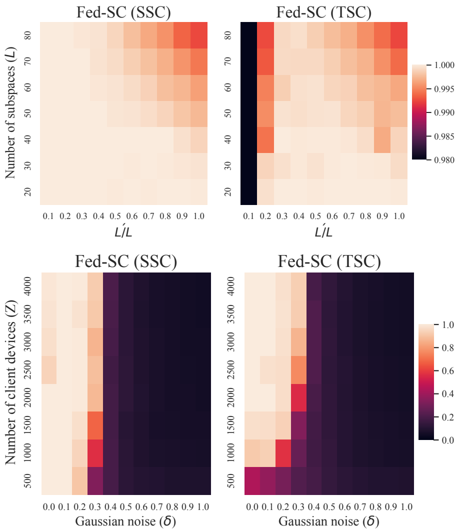
5 Experiments

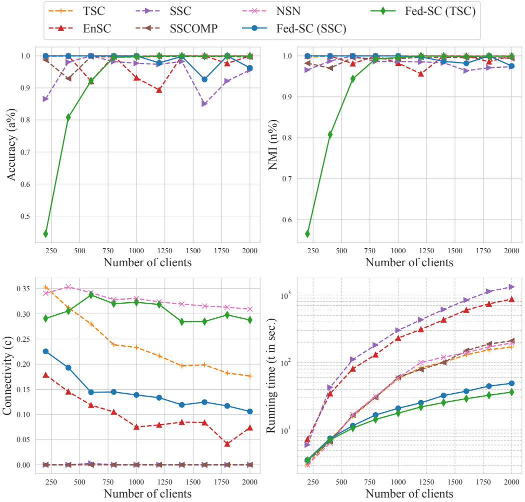
We set up Z devices and randomly distribute the data among Z devices such that each device z receives data points from $L' \leq L$ clusters.

- **Baseline**: The centralized methods include SSC, NSN, TSC, SSCOMP, and EnSC. The state-of-the-art one-shot federated clustering method is k-FED.

- **Datasets**: EMNIST and Augmented COIL100.
- **Evaluation metrics**: All algorithms are evaluated by the *clustering accuracy* (ACC: $a\%$), *normalized mutual information* (NMI: $n\%$), *connectivity* of the affinity graph (CONN: c), and running time.

Evaluation on Synthetic Data





Empirical Evaluation on Real World Datasets

TABLE III
PERFORMANCE COMPARISON ON EMNIST AND CIFAR-10 WHERE '-' DENOTES THE METRIC CANNOT BE COMPUTED PROPERLY. *: THE RUNNING TIME OF SSC FOR EMNIST EXCEEDS THE TIME LIMIT OF 1 DAY.

EMNIST ($2 \le L^{(z)} \le 4, z \in [Z]$)				
Methds	ACC($a\%$)	NMI($n\%$)	CONN($\bar{c}$)	T(sec.)
Fed-SC (SSC)	<u>85.77</u>	**88.28**	0.0019	262.83
Fed-SC (TSC)	**86.17**	<u>87.00</u>	0.0186	237.31
k-FED	56.68	67.18	-	<u>16.00</u>
k-FED + PCA-10	11.47	31.23	-	**7.95**
k-FED + PCA-100	11.64	31.28	-	16.18
SSC*	-	-	-	-
SSCOMP	56.17	70.26	0.000	12943.46
EnSC	60.83	74.00	<u>0.0317</u>	29459.42
TSC	49.04	66.92	0.0131	2511.73
NSN	41.68	63.82	**0.1571**	8117.37
Augmented COIL100 ($2 \le L^{(z)} \le 4, z \in [Z]$)				
Methds	ACC($a\%$)	NMI($n\%$)	CONN($\bar{c}$)	T(sec.)
Fed-SC (SSC)	**74.43**	**85.09**	0.0104	96.65
Fed-SC (TSC)	<u>57.54</u>	75.24	0.0579	78.12
k-FED	31.52	52.05	-	<u>3.03</u>
k-FED + PCA-10	8.59	26.18	-	**1.44**
k-FED + PCA-100	8.43	26.44	-	3.64
SSC	45.25	71.93	0.0006	31676.33
SSCOMP	41.17	68.26	0.0118	1616.64
EnSC	51.55	76.91	0.0324	3842.41
TSC	53.06	<u>78.99</u>	<u>0.1859</u>	809.27
NSN	30.46	46.97	**0.4280**	1765.18

TABLE IV
CLUSTERING ACCURACIES ($a\%$) WITH DIFFERENT NUMBER OF LOCAL CLUSTERS L'

EMNIST					
L'	2	4	6	8	10
Fed-SC (SSC)	**88.96**	**82.74**	**75.58**	**72.66**	**69.76**
Fed-SC (TSC)	<u>86.03</u>	<u>81.37</u>	<u>71.95</u>	<u>69.24</u>	<u>65.57</u>
k-FED	67.70	57.25	46.56	38.19	25.29
k-FED + PCA-10	13.41	9.02	7.62	7.82	7.14
k-FED + PCA-100	13.13	9.39	7.93	7.61	7.19
Augmented COIL100					
L'	2	4	6	8	10
Fed-SC (SSC)	**82.07**	**72.44**	**49.15**	**45.83**	**39.31**
Fed-SC (TSC)	<u>75.33</u>	<u>66.54</u>	<u>47.99</u>	<u>44.48</u>	<u>38.09</u>
k-FED	37.08	25.56	19.60	19.12	17.88
k-FED + PCA-10	10.78	7.01	5.40	5.56	5.61
k-FED + PCA-100	11.40	7.08	5.54	5.84	5.47

6 Conclusion and Future Work

- We investigated federated clustering for high-dimensional data and proposed the solution of one-shot federated subspace clustering.
- We theoretically and empirically guarantee the effectiveness of federated schemes for subspace clustering, especially with the benefit of statistical heterogeneity.
- The promising future directions are to theoretically guarantee privacy-preserving and to consider privacy-utility tradeoffs in federated clustering.

Thank you