PLOS Computational Biology

METHODS

Harmony-based data integration for distributed single-cell multi-omics data

Ruizhi Yuan[1], **Ziqi Rong**[2], **Haoran Hu**[1], **Tianhao Liu**[2], **Shiyue Tao**[1], **Wei Chen**[1,2*], **Lu Tang**[1*]

1 Department of Biostatistics and Health Data Science, University of Pittsburgh, Pittsburgh, Pennsylvania, United States of America, **2** Department of Pediatrics, University of Pittsburgh, Pittsburgh, Pennsylvania, United States of America

* wei.chen@pitt.edu (WC); lutang@pitt.edu (LT)

OPEN ACCESS

Citation: Yuan R, Rong Z, Hu H, Liu T, Tao S, Chen W, et al. (2025) Harmony-based data integration for distributed single-cell multi-omics data. PLoS Comput Biol 21(9): e1013526. https://doi.org/10.1371/journal.pcbi.1013526

Editor: Michael Domaratzki, University of Western Ontario: Western University, CANADA

Received: April 5, 2025

Accepted: September 15, 2025

Published: September 30, 2025

Data availability statement: The study uses various publicly available datasets. The URLs for each dataset are provided in S1 Table. The implementation code and tutorial are available: https://github.com/yrzzz/Federated-Harmony.

Funding: The research was partially supported by the NIH through grants R21DA055672

Abstract

Large-scale single-cell projects generate rapidly growing datasets, but downstream analysis is often confounded by data sources, requiring data integration methods to do correction. Existing data integration methods typically require data centralization, raising privacy and security concerns. Here, we introduce Federated Harmony, a novel method combining properties of federated learning with Harmony algorithm to integrate decentralized omics data. This approach preserves privacy by avoiding raw data sharing while maintaining integration performance comparable to Harmony. Experiments on various types of single-cell data showcase superior results, highlighting a novel data integration approach for distributed multi-omics data without compromising data privacy or analytical performance.

Author summary

In recent years, single-cell technologies have allowed scientists to study individual cells in great detail, helping us understand how tissues and organs function. As researchers around the world generate more data, combining these datasets has become important—but also challenging. One major issue is that data from different sources often vary due to technical differences, making integration tricky. Another growing concern is privacy: many institutions are hesitant to share sensitive biological data due to ethical, legal, and security risks. In our study, we introduce a method called Federated Harmony, which allows researchers to combine single-cell data from different institutions without directly sharing any raw data. By adapting a widely used data integration method (Harmony) into a federated learning framework, our approach preserves privacy while achieving comparable scientific results. We tested Federated Harmony on several types of single-cell data and found that it performs just as well as existing centralized methods, but with improved speed and security. We believe this method could

and R56LM014522 to L.T., P01AI106684 and R01DK138458 to W.C., by the NSF through grants 2225775 to W.C. and 2310217 to L.T., and by the University of Pittsburgh Medical Center through the Competitive Medical Research Fund to L.T.. R.Y., W.C. and L.T. received salary support from these grants. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing interests: The authors have declared that no competing interests exist.

help research teams around the world collaborate more safely and effectively, accelerating discoveries in biology and medicine.

Introduction

With the recent advances in single-cell technology [1], projects like Human Cell Atlas [2] are generating a rapidly growing collection of reference datasets from primary human tissues at different institutions, and these datasets significantly enhance our understanding of single-cell mechanisms. However, genuine biological signals in those datasets from different studies are often confounded by data source [3], which can significantly influence the downstream analysis.

Existing data integration methods [4–9] address above challenges by mapping cells from various experimental conditions and biological contexts into a unified, lower-dimensional space, facilitating the downstream analysis across different datasets. However, these methods typically require data centralization, which raises significant privacy and security concerns [10]. While policies such as the NIH Genomic Data Sharing Policy [11] mandate the sharing of transcriptomics and chromatin-related data, including raw files, to ensure reproducibility and foster collaboration, data sharing remains complex and constrained due to several additional challenges. Cross-border regulations, such as GDPR in Europe, PIPL in China, and LGPD in Brazil [12–15], impose strict limitations on genomic data transfer, creating legal barriers even when U.S. policies allow sharing. Furthermore, institutional privacy policies, ethical considerations—particularly for Indigenous and sensitive populations—and concerns over intellectual property and data ownership can further restrict data access. Additionally, cybersecurity risks associated with centralized repositories increase concerns over potential data breaches and re-identification threats [16]. Computational bottlenecks and storage limitations also make large-scale single-cell data centralization impractical, particularly for institutions with limited infrastructure under privacy regulation. These regulatory, ethical, and technical barriers create a fundamental conflict between the growing need to integrate rapidly expanding single-cell datasets and the constraints on data sharing. Thus, privacy-preserving methods are crucial for enabling secure, scalable, and globally collaborative single-cell research while ensuring compliance with diverse legal and institutional policies.

To address above challenges, we proposed Federated Harmony, a privacy-preserving method that combines federated learning and Harmony. Federated learning is a collaborative paradigm that enables institutions to collaboratively train models without raw data sharing [17], addressing privacy concerns [18–20] and reducing computational strain [21]. In addition, Harmony has also been proven as one of the best performing and most commonly used data integration methods in single-cell data analysis [22,23]. The basic idea of Federated Harmony is by incorporating Harmony into the federated learning framework. By performing local computations and sharing only aggregate statistics based on the Harmony algorithms, Federated Harmony addresses privacy concerns and reduces the computation time

while maintaining integration performance comparable to Harmony. We evaluate Federated Harmony on scRNA-seq, scATAC-seq and spatial transcriptomics data, and both visual and quantitative results are promising, highlighting Federated Harmony as a promising approach for secure and efficient distributed single-cell data integration.

Results and discussion

Federated Harmony begins with the output embeddings generated by Federated Principal Component Analysis (Federated PCA) [24]. It operates through an iterative four-step process (Fig 1). First, each institution conducts local computation on its dataset, deriving summary statistics (e.g., centroids or row sum of some matrices) without sharing the raw data between institutions (Step ①). Next, these summary statistics are sent to a central server (Step ②), which is responsible for aggregating and updating the received statistics (Step ③). The central server then sends the aggregated summaries back to the institutions (Step ④), allowing each institution to adjust its local model using information from other institutions. Finally, each institution obtains the corrected and integrated embeddings. Fig 1 also compares the process of Federated Harmony and Harmony, and the contrast highlights the ability of Federated Harmony to borrow information from other institutions and correct data without sharing raw data, ensuring privacy while maintaining integration performance.

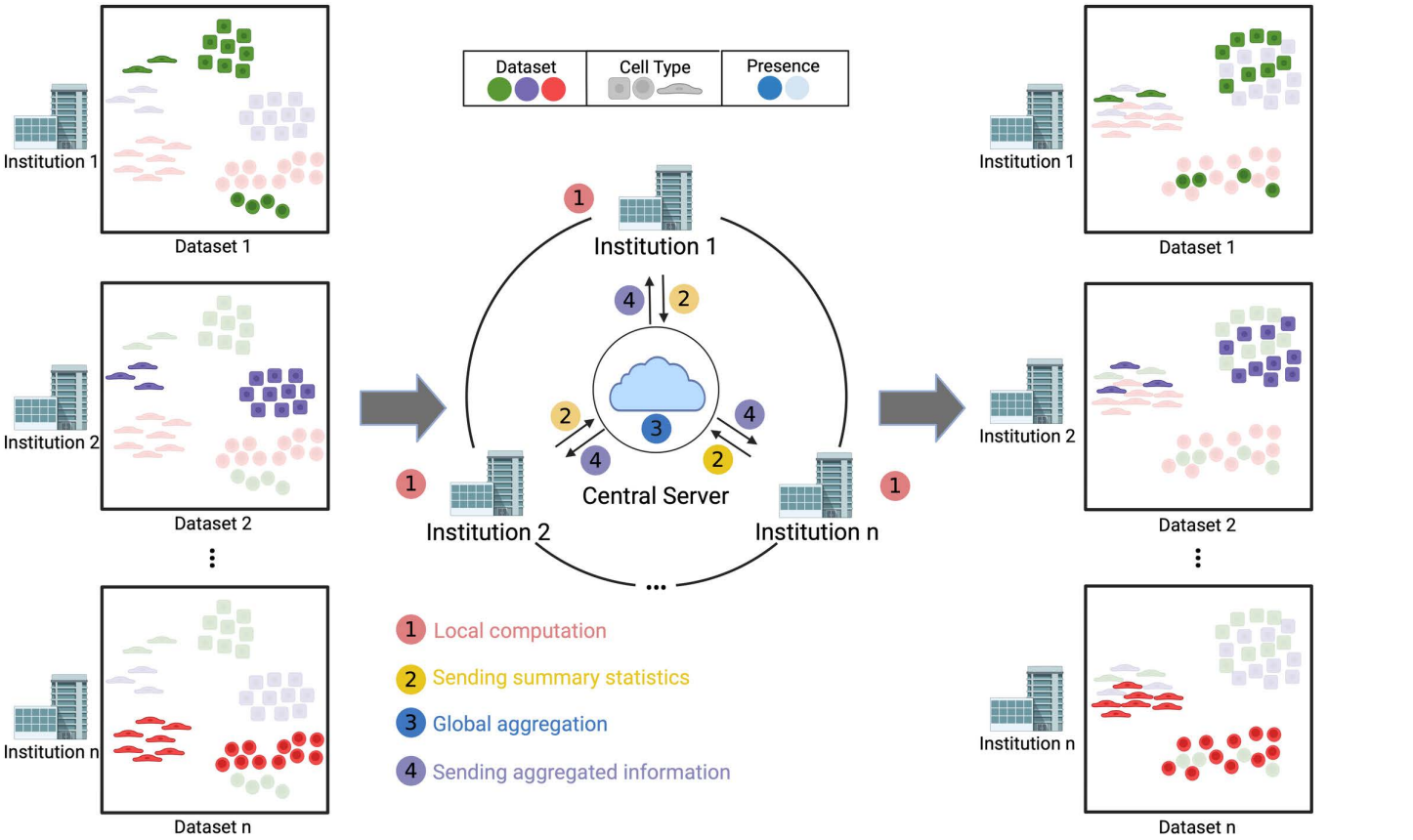

Fig 1. The Workflow of Federated Harmony. Federated Harmony operates through an iterative four-step process: ① Institutions perform local computations on their data; ② Summary statistics are sent to a central server; ③ The central server aggregates and updates the summaries; and ④ Aggregated summaries are returned to the institutions for further refinement. Each institution's dataset is represented using colored shapes to denote different cell types or datasets. Opaque shapes indicate data present within the institution, while semi-transparent shapes represent data that not in that institution but would be integrated if the data were centralized, offering a visual comparison between centralized Harmony and Federated Harmony.

https://doi.org/10.1371/journal.pcbi.1013526.g001

To evaluate the performance of Federated Harmony, we applied Federated Harmony to three types of datasets—scRNA-seq, spatial transcriptomics and scATAC-seq—and compared the results with those from Harmony. For Harmony, we followed the standard procedure. In contrast, Federated Harmony was tested through a simulation study where the dataset was split into distinct datasets based on their original batches, kept separate, and not integrated or shared, simulating real-world data privacy constraints. We then applied Federated Harmony to these split datasets to assess its effectiveness in integrating data in a decentralized manner. The Federated Harmony-integrated embeddings were then centralized solely for performance evaluation. All data source links can be found in S1 Table.

We first assess Federated Harmony on a scRNA-seq data. We use human peripheral blood mononuclear cells (PBMC) scRNA-seq data with 5 samples as an example. The whole blood was collected from four healthy donors, and scRNA-seq data were produced using the 10x Chromium platform. Furthermore, scRNA-seq data from an extra healthy donor from a pre-existing publicly available PBMC dataset was also included in the experiment. Within this group, samples 1 and 2 were sequenced together in one batch, while samples 3 and 4 were sequenced in a separate batch. The dataset for sample 5 was downloaded from a study previously conducted by 10x Genomics [25].

Focusing on the PBMC scRNA-seq panels in Fig 2a–2c, the lower part plot shows UMAP [26] embeddings of PBMC scRNA-seq data in a two-dimensional space: before integration, after Harmony integration, and after Federated Harmony integration. Before integration, we applied a standard PCA pipeline followed by UMAP embedding. Fig 2a (UMAP Before Integration) for PBMC scRNA-seq data shows that samples 1 and 2 cluster together, as do samples 3 and 4, while sample 5 forms a separate cluster. This pattern indicates that the cells are grouped based on their original batches. We also tested our method on two additional scRNA-seq datasets, with similarly positive results. Detailed findings are provided in S1 Fig.

We then applied Harmony to integrate them and used the result as a baseline. As shown in Fig 2b (Harmony Integration) for PBMC scRNA-seq data, cells from all five samples are mixed. The UMAP generated from Federated Harmony-integrated embeddings closely resembles that of the Harmony-integrated embedding, shown in Fig 2c (Federated Harmony Integration) for PBMC scRNA-seq data. While the orientations of clusters within the UMAP plots may vary, such rotations or reflections do not affect the biological interpretation. This visual similarity indicates that Federated Harmony performs well for scRNA-seq data.

We then evaluated Federated Harmony on spatial transcriptomics (ST) data from 10X genomics. This dataset consists of three batches (FFPE, fixed, and fresh) of mouse brain tissue, with around 20,000 spatial spots in total, each corresponding to single cells. Before integration, cells are clustered in three distinct clusters by their batch as shown in the lower plot of Fig 2a (Before Integration) for brain ST data. After applying both Harmony and Federated Harmony, lower plots of Fig 2b (Harmony Integration) for brain ST data and Fig 2c (Federated Harmony Integration) for brain ST data show that the UMAP visualizations for the Federated Harmony-integrated and Harmony-integrated embeddings are highly similar, and both integrate cells from different batches together.

We further evaluated Federated Harmony on scATAC-seq data using two single-cell chromatin datasets derived from human PBMCs. One dataset was generated with the 10x Genomics multiome technology, providing both DNA accessibility and gene expression information for each cell. The other dataset was profiled using 10x Genomics scATAC-seq, containing only DNA accessibility data. The lower plot of Fig 2a (Before Integration) for PBMC scATAC-seq data shows a clear separation between the two batches. After applying both Federated Harmony and Harmony, as seen in lower plots of Fig 2b (Harmony Integration) for PBMC scATAC-seq data and Fig 2c (Federated Harmony Integration) for PBMC scATAC-seq data, cells from both datasets are well integrated, demonstrating the effectiveness on scATAC-seq data.

To compare the similarity between the centralized Federated Harmony- and Harmony-integrated embeddings, we examined the Adjusted Rand Index (ARI) from the naive k-means clustering results for both embeddings (the Federated Harmony-integrated embeddings were centralized here solely for performance evaluation). We varied the number of clusters (ranging from 2 to 10) in the k-means algorithm and calculated the ARI values for each scenario. The boxplots in

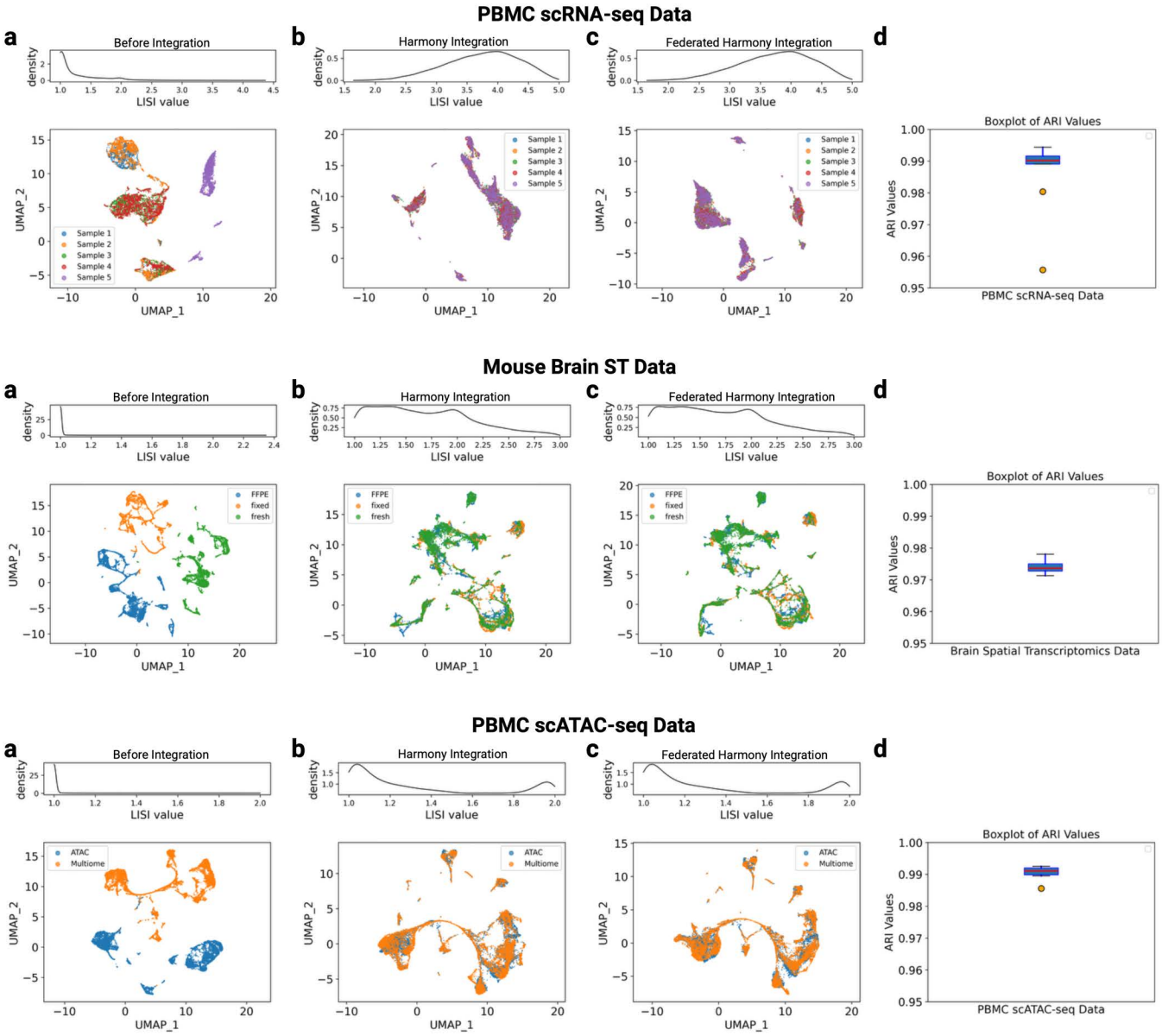

Fig 2. The performance of Federated Harmony on three types of single cell data. Each row represents results of a type of data. scRNA-seq data (first row); mouse brain tissue spatial transcriptomics data (second row); scATAC-seq data (third row). For **a-c**, the upper plot is the iLISI density plot, the lower one is the UMAP. **a**: iLISI density plot and UMAP before integration; **b**: iLISI density plot and UMAP after Harmony integration; **c**: iLISI density plot and UMAP after Federated Harmony integration; **d**: box plots of ARI values of naive k-means clustering results for Harmony-integrated and Federated Harmony-integrated embeddings.

https://doi.org/10.1371/journal.pcbi.1013526.g002

Fig 2d named Boxplot of ARI values present ARI values, which consistently exceeded 0.95, regardless of cluster number. These high ARI values indicate almost identical integrated embeddings produced by both methods. To more rigorously assess integration quality, we computed the integration local inverse Simpson's Index (iLISI) [8]. For each cell, iLISI measures the effective number of batches represented among its nearest neighbors. Values near 1 indicate single-batch neighborhoods (strong batch effect), whereas higher values indicate better mixing. As shown in Fig 2, the post-integration

iLISI density curves for Harmony and Federated Harmony are highly similar across all types of data. Quantitatively, median iLISI improved as follows: PBMC scRNA-seq from 1.07 (Before Integration) to 3.79 (Harmony Integration) and 3.76 (Federated Harmony Integration); brain spatial transcriptomics from 1.00 to 1.63 and 1.64; scATAC-seq from 1.00 to 1.35 and 1.35. These results indicate that Federated Harmony achieves mixing comparable to centralized Harmony.

To evaluate scalability in a realistic setting, we used a published multi-omic blood cohort profiling patients with varying COVID-19 severity alongside influenza, sepsis, and healthy controls [27]. The resource includes matched immune measurements across modalities and reports severity-linked signatures spanning myeloid and lymphoid subsets, inflammatory mediators, and acute-phase responses. For our integration test we focused on the scRNA-seq data and selected 40 donors (treating each donor as a separate site/batch).

The results of Federated Harmony are shown in Fig 3. Before integration, the combined UMAP was dominated by donor identity in some regions with fragmented cell-type structure, which can be proven by the iLISI density map. For density map before integration, the density peaks around 1 – 4 and then tapers off and only a tiny tail exceeds 10, which indicates strong batch effects for a 40-batch data. After Federated Harmony, the embedding organized by cell type while

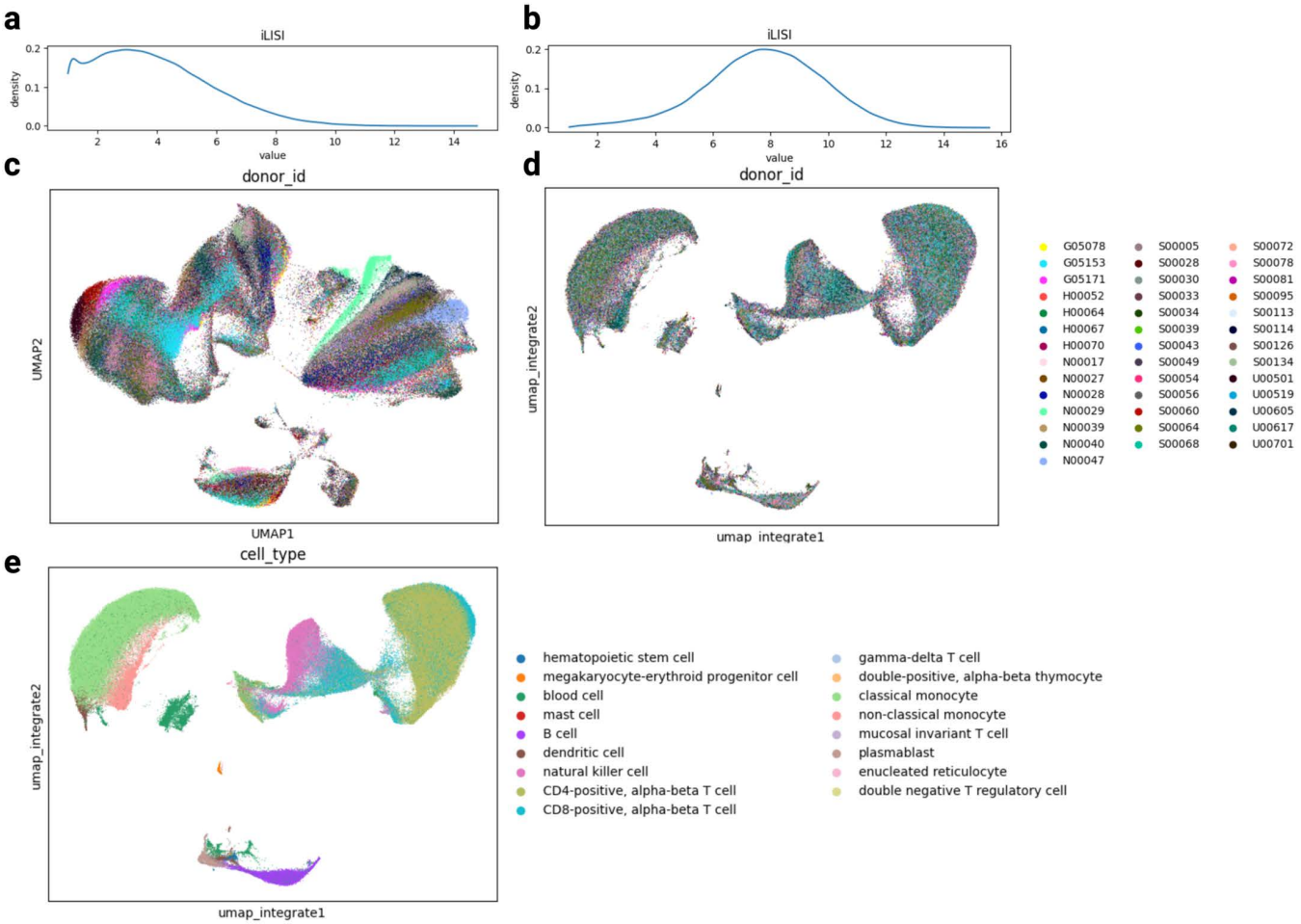

Fig 3. The results of Federated Harmony on large scale dataset. a: the iLISI density plot before integration; **b**: the iLISI density plot after Federated Harmony integration; **c**: UMAP before integration by donor; **d**: UMAP after Federated Harmony integration by donor; **e**: UMAP after Federated Harmony by cell type.

https://doi.org/10.1371/journal.pcbi.1013526.g003

the donors mixed across clusters. Quantitatively, median iLISI improved from 3.6 to 7.76. With the corrected embedding, each site can independently perform downstream analyses, such as cell type annotation. S3 Fig shows per-donor panels' cell type annotation results.

Computational efficiency, scalability and overhead

We also evaluated the computational efficiency of Federated Harmony and Harmony across different datasets, assessing running time (in seconds) and iterations required for convergence (S2 Fig). Federated Harmony consistently outperformed Harmony, requiring fewer iterations and shorter running times ideally. For instance, Federated Harmony converged in approximately 15 seconds and 10 iterations for PBMC scRNA-seq data (5 batches), while Harmony required 45 seconds and 25 iterations. Similarly, for brain spatial transcriptomics data (3 batches), Federated Harmony completed in 20 seconds with 9 iterations compared to Harmony's 44 seconds and 15 iterations. The efficiency gap became more pronounced with an increasing number of batches, highlighting Federated Harmony's scalability in multi-batch scenarios, where it offers a more computationally effective solution for large datasets. This improvement stems from two key factors. First, Federated Harmony performs some computations independently at each institute, parallelly distributing work that Harmony processes serially on a single server. Second, each federated round estimates batch-effect parameters from cluster-level averages, whose low variance allows the server to solve a closed-form weighted least squares problem and take larger steps toward the optimum compared to the small, per-cell adjustments used by Harmony (see Method). In our benchmarks this yields fewer global rounds to convergence.

Per institute, Federated Harmony only computes partial statistics in Harmony such as cluster centroids,co-occurrence matrix and intermediate values of correction matrix, so the cost in each institute is less than running Harmony once on the same data. For exchanging summary statistics between institutes and the center, only cluster-level sufficient statistics like centroids (d dimensions) and batch–cluster co-occurrence matrix (K clusters, B institutes) are transmitted, requiring no more than Kd or KB floats, whichever is larger. For example, with $d = 30$ PCs, $K = 400$ clusters, the centroids needed to be transmitted are approximately 12,000 floats (~48 KB float32). Thus, per-round transfer time (<0.01 s) and cost at 100 Mbps bandwidth is negligible in real-world application. Communication grows with K rather than the number of cells N_b, the network cost remains modest provided clustering resolution tracks biological heterogeneity even if raw data size increases. Consequently, the workload for each institute fits easily on a standard workstation or laptop, and because each round transmits usually no more than 1 MB of summaries, the network overhead remains negligible on any stable internet connection. The workflow therefore can easily scale to collaborations with many participating institutes.

To report the full cost of the workflow, we also measured the total end-to-end time including Federated PCA in data preprocessing steps followed by Federated Harmony: on PBMC scRNA-seq (1,000 genes) Federated PCA took 25 s and Federated Harmony 15 s (total 40 s); on brain spatial transcriptomics (19,465 genes) Federated PCA took 688 s and Federated Harmony 20 s (total 708 s); on scATAC-seq (108,377 peaks) Federated PCA took 68 min and Federated Harmony 22 s (total ~68 min). Empirically, Federated PCA time grows approximately linearly with the number of features and the number of samples. Note that all timings were obtained on a single workstation that emulates multiple institutes sequentially. Therefore, the reported wall-clock times are a conservative upper bound. In a real federated deployment, institutes compute in parallel, so per-round time is dominated by the slowest site plus a small communication term.

Incomplete and heterogeneous sites

In federated biomedical applications, incomplete or heterogenous data among different site is a common issue. Federated Harmony follows Harmony's assumption of a shared feature space, but it remains practical when deviations occur. When features are missing at some sites, we project each site onto the intersected set of highly variable genes (or another agreed feature subset) provided that this intersection still captures the key biology. If a site lacks a particular cell type, it contributes no statistics for that cluster; the parameters are updated from the remaining sites, which is analogous to how

Harmony handles batches in which that cell type is absent. Cases in which sites measure different modalities (for example, scRNA-seq versus scATAC-seq or spatial data) are beyond the current scope. Potential extensions include block-wise updates that skip unavailable modalities or representation-learning layers that map heterogeneous modalities into a shared latent space.

Privacy risk of sharing PCA-derived embeddings

Although principal-components are a convenient low-dimensional representation, they are not automatically outside the scope of data-protection law. Studies [28,29] have demonstrated that embeddings produced by PCA and other dimension reduction methods can be inverted to recover significant portions of the original high-dimensional data via a neural-network reconstruction attack, which means that sharing PCA embeddings carries a risk of raw data leakage or of inferring sensitive patient attributes. From a regulatory perspective, most frameworks including the EU GDPR (Recital 26; Art. 9), China's PIPL (Arts. 4, 28, 73), and the U.S. HIPAA de-identification rule define personal or sensitive information by identifiability, not by data type. In practice this means that any per-sample PCA embedding remains regulated if it reveals or can be used to reveal patient health/genetic attributes. Together, these considerations indicate that PCA cannot be assumed safe to share in federated settings. To guard against emerging inversion and attribute-inference threats, Federated Harmony never shares embeddings; instead, it exchanges only minimal aggregated summary statistics (e.g., cluster centroids, co-occurrence counts), ensuring compliance with the strictest privacy requirements. We introduce a four-level privacy taxonomy (Table 1). The tiers progress from the most identifiable data—raw sequence reads (Tier 0)—through processed count matrices (Tier 1) and per-sample low-dimensional embeddings such as PCA embeddings (Tier 2), to the least identifiable representation, coarse aggregate statistics like cluster centroids and co-occurrence counts (Tier 3). This framework makes explicit the re-identification risk associated with each data type and motivates our decision to share only Tier 3 outputs in Federated Harmony.

Conclusion

In this study, we developed Federated Harmony, a data integration method for distributed single-cell multi-omics data that preserves data privacy. Our results demonstrate that Federated Harmony successfully integrates multi-institutional datasets, producing embeddings comparable to those of the standard Harmony approach. UMAP visualizations, iLISI and ARI values confirm that Federated Harmony effectively removes the non-biological variations in the dataset without requiring direct data sharing. Additionally, Federated Harmony can reduce integration time and iterations theoretically. This work fills a gap in the literature, as federated learning methods for single-cell data integration remain underexplored.

However, there are limitations to our study. Our method inherently involves multiple communication rounds between institutes, which could be a challenge in real-world applications like some other federated learning methods. Additionally, for preprocessing scATAC-seq data, TF-IDF normalization and Latent Semantic Indexing (LSI) are also commonly used, whereas our pipeline employs log-normalization and Federated PCA for this step. Finally, for preprocessing, we employed Federated PCA, which, though occasionally time-consuming, is external to our core method and could be optimized further in future implementations.

Table 1. Privacy levels and information exchanged by Federated Harmony.

Tier	Example Data	Typical risk	Protection in Federated Harmony
Level 0	Raw reads	Direct identifiers & rare variant leakage	Never exchanged
Level 1	Count matrix, cell metadata	Gene- or sample-level linkage possible	Never exchanged
Level 2	PCA embeddings	Reconstruction/ attribute inference	Never exchanged
Level 3	Cluster centroids, co-occurrence counts, etc.	Minimal risk	Shared across sites

https://doi.org/10.1371/journal.pcbi.1013526.t001

Future work should prioritize reducing communication rounds without compromising accuracy, as this will be critical for enhancing the applicability of Federated Harmony. Improving the computational efficiency of Federated PCA is another essential step toward broadening the method's scalability. Additionally, optimizing the preprocessing steps for scATAC-seq data remains important to ensure consistency across data types. Once these advancements are achieved, the method can be tested on real-world applications, moving beyond simulations to validate its robustness in practical scenarios.

Method

We first define all notation of variables we will involve in our method. The dimensionality of the embedding (for example, the number of PCs) is denoted as d; the number of institutions/batches is denoted as B; N is the number of total samples; N_b is the number of samples institutions/batch b; K is the number of clusters.

$\phi \in \{0,1\}^{B\times N}$ The input one-hot assignment matrix of cells (columns) to batches/institutes (rows).

$Pr_b \in [0,1]^B$ Frequency of batches/institutes, i.e., N_b/N.

$R \in [0,1]^{K\times N}$ The soft cluster assignment matrix of cells (columns) to clusters (rows). Each column is a probability distribution and thus sums to 1.

$O \in [0,1]^{K\times B}$ The observed co-occurence matrix of cells in clusters (rows) and batches/institutes (columns).

$E \in [0,1]^{K\times B}$ The expected co-occurence matrix of cells in clusters and batches, under the assumption of independence between cluster and batches/institutes assignment.

$Z^{(b)} \in \mathbb{R}^{d\times N_b}$ The input embedding at institute b, to be corrected in Federated Harmony. The PCA embeddings of cells are often used.

$\widehat{Z^{(b)}} \in \mathbb{R}^{d\times N_b}$ The corrected embedding after batch correction at each institute, output by the proposed Federated Harmony.

$R^{(b)} \in [0,1]^{K\times N_b}$ The soft cluster assignment matrix of cells (columns) to clusters (rows) at institute b. Each column is a probability distribution and thus sums to 1.

$Y \in \mathbb{R}^{d\times K}$ Global cluster centroid locations.

$Y^{(b)} \in \mathbb{R}^{d\times K}$ Updated cluster centroids at institute b.

Harmony overview

The Harmony algorithm takes a PCA embedding of cells (Z) and their corresponding batch labels (ϕ) as inputs and produces a batch corrected embedding ($\hat{Z}$). It iterates between two stages: maximum diversity clustering and a mixture model based linear batch correction.

The objective function for maximum diversity clustering is defined by Eq. (1):

$$\min_{R,Y}\sum_{i,k} R_{ki}\ |Z_i - Y_k\ |^2 + \sigma R_{ki}\log R_{ki} + \sigma\theta R_{ki}\log\left(\frac{O_{ki}}{E_{ki}}\right)\phi_i,$$

$$\text{s.t.}\forall_i\forall_k R_{ki} > 0, \forall_i \sum_{k=1}^{K} R_{ki} = 1, \tag{1}$$

where σ is a hyperparameter. The optimization of the objective function is shown in Eq. (2):

$$R_{ki} = \frac{\left(\frac{O_{ki}}{E_{ki}}\right)^2 \exp\left(-\frac{2\left(1-Y_k^\top Z_i\right)}{\sigma}\right)}{\sum_{k=1}^{K}\left(\frac{O_{ki}}{E_{ki}}\right)^2 \exp\left(-\frac{2\left(1-Y_k^\top Z_i\right)}{\sigma}\right)}. \tag{2}$$

The mixture of experts correction calculates a correction factor for each cluster k is defined in Eq. (3):

$$W_k = \left(\phi^* diag\left(\mathrm{R_k}\right)\phi^{-\mathrm{T}} + \lambda \mathrm{I}\right)^{-1} \phi^* diag\left(\mathrm{R_k}\right)\mathrm{Z}^{\mathrm{T}}, \tag{3}$$

where $\phi^* \leftarrow 1\ ||\phi$, and then performs the batch correction by Eq. (4):

$$\hat{Z} = Z - W_k^T \phi^* diag\left(R_k\right). \tag{4}$$

Federated Harmony overview

The steps of Federated Harmony fundamentally retain the procedure of Harmony, yet it makes a significant difference in execution by leveraging the federated computing principles-namely, local computations, central aggregation, and privacy preservation, as shown in Fig 1. At the beginning, each institute log-normalizes their dataset. Then, like the procedure of Harmony [8], Federated Harmony begins with local low dimensional embeddings of cells. Under the assumption that the data cannot be shared, traditional dimension reduction methods cannot be used since they presuppose the data are centrally aggregated. In response to these constraints, we used Federated PCA [24] at the preprocessing step, a method that enables each institutes to derive principal components for their local data by leveraging statistical summaries from data distributed across different institutes, without necessitating direct data sharing. The outcome of Federated PCA closely approximates the results when data from all institutes is first pooled together for PCA and then the corresponding principal components (PCs) are distributed back to their original locations.

Federated Harmony (Algorithm in Fig 4) utilizes local embeddings and iterates between two stages, similar to Harmony. These stages include federated maximum diversity clustering and federated mixture model-based linear batch correction.

Initialization

Federated Harmony first initialize the global centroids using Federated k-means [30]. By the definition of the soft cluster assignment matrix of cells (columns) to clusters (rows) R defined in Harmony [8], the i^{th} column in R corresponds to the probability distribution of the i^{th} cell across different clusters. This means that for the i^{th} cell, each entry in the column of R indicates the likelihood of that cell belonging to each cluster, i.e., $\sum_{k=1}^{K} R_{ki} = 1$. The important aspect of calculating R is

Algorithm 1: Federated Harmony

Input: Local embeddings $Z^{(b)}$ at each site
Output: Integrated embeddings $\hat{Z}^{(b)}$.
$\hat{Z}^{(b)} \leftarrow Z^{(b)}$;
Initialization;
repeat
  Federated Maximum Diversity Clustering;
  Federated Mixture of Experts Correction;
until *convergence*;
return $\hat{Z}^{(b)}$;

Fig 4. Federated Harmony Algorithm.

https://doi.org/10.1371/journal.pcbi.1013526.g004

that its calculation for each column is based exclusively on the data corresponding to the individual cell, so R can also be represented as $R = [R^{(1)}\ R^{(2)} \dots\ R^{(B)}]$ if there are B batches of data in the dataset and we order the cells by their original batch. Thus, each individual institution can independently initialize its own soft cluster assignment matrix $R^{(b)}$ by using the same strategy as in Harmony shown in Eq. (5):

$$R^{(b)} = \frac{exp\left(-\frac{2\cdot\left(1-Y^T\cdot Z^{(b)}\right)}{\sigma}\right)}{\sum_i exp\left(-\frac{2\cdot\left(1-Y^T\cdot Z^{(b)}\right)}{\sigma}\right)}, \tag{5}$$

where Σ_i denotes the column-wise sum.

Initialization of the matrix O is achieved through the operation $O \leftarrow R\phi^T$. Within the Federated Harmony framework, direct access to batch (institute) indicators is absent. To circumvent this, each institution is assigned a unique index to signify the batch to which each cell belongs. Consequently, the center constructs a matrix ϕ based on the institute index $b \in [1, B]$ and the respective cell counts N_b for each institute b. The matrix ϕ is structured as follows, where each row corresponds to a distinct institute and the columns within a row are populated with ones to indicate the membership of cells to that institute, with the dimensionality of each segment determined by $N_1, N_2, \dots, N_B$, and each institution will also receive their own ϕ_b shown in Eq. (6):

$$\phi = [\phi_1\ \phi_2\ \cdots\ \phi_B] = \begin{bmatrix} 1\cdots1 & 0\cdots0 & \cdots & 0\cdots0 \\ 0\cdots0 & 1\cdots1 & \cdots & 0\cdots0 \\ \vdots & \vdots & \ddots & \vdots \\ \underbrace{0\cdots0}_{N_1} & \underbrace{0\cdots0}_{N_2} & \cdots & \underbrace{1\cdots1}_{N_B} \end{bmatrix}. \tag{6}$$

Given the representation of R as $R = [R^{(1)}\ R^{(2)} \dots\ R^{(B)}]$, the matrix O is derived by computing $O \leftarrow R\phi^T$, which yields Eq. (7):

$$O_{k,b} = \sum_{i=1}^{N_b} R_{k,i}^{(b)} \tag{7}$$

This formulation implies that the center requires only the summation across rows of $R^{(b)}$ for each institute b to construct the matrix O. The reason why we do not share $R^{(b)}$ to the center will be explained later in this section. For initialization of E in Harmony, $E \leftarrow R\mathbf{1}Pr_b^T$ where $R\mathbf{1}$ equals to a vector that sums each row of R in Eq. (8),

$$R\mathbf{1} = \left[\sum_{i=1}^{N} R_{1i}\ \ \sum_{i=1}^{N} R_{2i}\ \ \dots\ \ \sum_{i=1}^{N} R_{Ki}\right]^T. \tag{8}$$

In this case, $R\mathbf{1}$ equals to the row sum of O, and thus $E = O\mathbf{1}Pr_b^T$. In Federated Harmony settings, Pr_b^T can be easily obtained by collecting the total number of cells in each institute. The initialization process is summarized in Algorithm in Fig 5:

Federated maximum diversity clustering

After initializing $R^{(b)}$, E,and O, the next step is to conduct a federated maximum diversity clustering. First, we update Y using the same strategy as in Harmony shown in Eq. (9):

Algorithm 2: Initialization

Input: Local embeddings $Z^{(b)}$ at each site, number of clusters K, number of clusters at each site $K^{(b)}$.
Output: Updated matrices E and O.
[Institutes]:
$Y \leftarrow$ Federated k-means $\left(Z^{(b)}, K^{(b)}, K\right)$
Initialize local soft cluster assignment matrix $R^{(b)}$;
Send $\sum_{j=1}^{N_b} R_{ij}^{(b)}$ to the center;
[Center]:
Initialize O: $O_{[:,i]} = \sum_{j=1}^{N_b} R_{ij}^{(b)}$
Iniitialize E: $E = O\mathbf{1}Pr_b^T$
Send E and O to all institutes;

Fig 5. Initialization Algorithm.

https://doi.org/10.1371/journal.pcbi.1013526.g005

$$Y = ZR^T = \begin{bmatrix} Z^{(1)} & Z^{(2)} & \dots & Z^{(B)} \end{bmatrix} \begin{bmatrix} R^{(1)^T} & R^{(2)^T} & \dots & R^{(B)^T} \end{bmatrix} = \Sigma_{b=1}^{B} Z^{(b)} R^{(b)^T}. \tag{9}$$

This can be achieved by calculating $Y^{(b)} = Z^{(b)}R^{(b)^T}$ at institute b and then the center aggregates and sums $Y^{(b)}$ to get Y. The equation here shows why we cannot directly share $R^{(b)}$, if $R^{(b)}$ is shared, the center can easily get the original embedding $Z^{(b)}$using $Z^{(b)} = Y^{(b)} \left(R^{(b)^T}\right)^{-1}$, which yields privacy concerns since embedding level data are shared.

In Harmony, the objective function for the update of R is optimized using block updates as the values of O and E change with R. Following the same idea, we also use block updates to update $R^{(b)}$. However, in Harmony, cells in each block are randomly selected, but in our method, we create blocks locally in each institution, which means each block will contain data from one batch. We call it sequential block update. Although we do not randomly assign cells to blocks across all data, there will be only small errors.

In the context of sequential block updates, the introduction of error can be analyzed through a detailed mathematical framework. Let the block size $n_{\text{block}} = \alpha N$, where $0 < \alpha \ll 1$ (e.g., $\alpha = 0.05$ for a block size of 5%). This implies the total number of blocks is $M = \frac{N}{n_{\text{block}}} = \frac{1}{\alpha}$.

In the sequential block update process, for each block m, the contributions of cells in that block are temporarily removed from matrices O and E, yielding modified matrices O_m and E_m. The cluster assignments R_m for cells in block m are then updated using these modified matrices. Once the updates are made, the contributions of block m are re-added to O and E for subsequent block updates.

We define the error in the cluster assignment for block m as Eq. (10):

$$\delta R^{(m)} = R^{(m)}_{\text{sequential}} - R^{(m)}_{\text{ideal}}, \tag{10}$$

where $R^{(m)}_{\text{sequential}}$ represents the cluster assignments from the sequential update, and $R^{(m)}_{\text{ideal}}$ represents the assignments under an ideal, simultaneous update. This error arises because $O^{(m)}$ and $E^{(m)}$ differ from their ideal values due to the exclusion of cells in block m.

When $\alpha$is small, the proportion of excluded cells in each block is minimal, meaning that $O^{(m)}$ and $E^{(m)}$ deviate only slightly from their global values O and E. Using a first-order Taylor expansion, we approximate Eq. (11):

$$O^{(m)} = O - \Delta O^{(m)}, \quad E^{(m)} = E - \Delta E^{(m)}, \tag{11}$$

where $\Delta O^{(m)}$ and $\Delta E^{(m)}$ represent the contributions of cells in block m. Since $\Delta O^{(m)}$ and $\Delta E^{(m)}$ are proportional to $n_{\text{block},}$ they are small when α is small.

For each cell i in block m, the diversity penalty Ω_{ki} is given by Eq. (12):

$$\Omega_{ki} = \theta \log\left(\frac{O_{kb}^{(m)} + 1}{E_{kb}^{(m)} + 1}\right), \tag{12}$$

and the error in Ω_{ki} is defined in Eq. (13):

$$\delta\Omega_{ki} = \Omega_{ki}^{\text{sequential}} - \Omega_{ki}^{\text{ideal}}. \tag{13}$$

Using a first-order approximation, we get Eq. (14):

$$\delta\Omega_{ki} \approx \theta\left(\frac{\Delta E_{kb}^{(m)} - \Delta O_{kb}^{(m)}}{E_{kb} + 1}\right), \tag{14}$$

where E_{kb} and O_{kb} are global counts, and $\Delta E_{kb}^{(m)}$ and $\Delta O_{kb}^{(m)}$ are small perturbations.

The update rule for R_{ki} is shown in Eq. (15):

$$R_{ki} \propto \exp\left(-\frac{2\left(1 - Y_k^{\top} Z_i\right)}{\sigma}\right) \exp\left(-\Omega_{ki}\right). \tag{15}$$

Thus, the error in R_{ki} is shown in Eq. (16):

$$\delta R_{ki} \propto R_{ki}^{\text{ideal}}\left(\exp\left(-\delta\Omega_{ki}\right) - 1\right) \approx -R_{ki}^{\text{ideal}}\delta\Omega_{ki}. \tag{16}$$

Assuming $\delta\Omega_{ki}$ is small, we approximate $\exp\left(-\delta\Omega_{ki}\right) \approx 1 - \delta\Omega_{ki}$.

Since $\Delta O_{kb}^{(m)}$ and $\Delta E_{kb}^{(m)}$ are proportional to $n_{\text{block},}$ the error δR_{ki} is also proportional to α, leading to a per-block error of O (α).

To quantify the cumulative error over all blocks, we sum the per-block errors as in Eq. (17):

$$\delta R = \sum_{m=1}^{M} \delta R^{(m)}. \tag{17}$$

With $M = \frac{1}{\alpha}$ blocks, the total error is bounded as in Eq. (18):

$$|\delta R| \leq M \times C\alpha = \frac{1}{\alpha} \times C\alpha = C. \tag{18}$$

Thus, the total error $|\delta R|$ is bound by a constant C that is independent of α.

For the average error per cell, we have Eq. (19):

$$\text{Average}\left|\delta R_{ki}\right| = \frac{|\delta R|}{N} \leq \frac{C}{N}. \tag{19}$$

As N increases, the average error per cell decreases, indicating that the overall impact on clustering results is minimal. Additionally, the stochastic nature of the errors may lead to some cancellation over multiple blocks, further reducing the effect of the cumulative error on the clustering outcome.

After each sequential block update in one institution, updated O and E are sent to the next institution for calculation until all institutions finish the update. After $R^{(b)}$ at each institute is updated, we repeat the process until convergence. Algorithm in Fig 6 shows the overview of federated maximum diversity clustering.

Federated mixture of experts correction

The next step is to correct the embedding for each institute. In Harmony, the correction factors are calculated as $W_k = \left(\phi^* diag\left(\mathrm{R_k}\right)\phi^{*\mathsf{T}} + \lambda \mathrm{I}\right)^{-1}\phi^* diag\left(\mathrm{R_k}\right)\mathrm{Z}^{\mathsf{T}}$. In federated setting, the center can simply form a $\phi^* \leftarrow 1\ ||\phi$. If we order the cells by their batches, then the ϕ^* can be represented as: $\phi^* = [\phi_1^*\ \phi_2^* \cdots \phi_B^*]$ where $\phi_b^* = [\mathbf{1}\ \phi_b]^{\mathsf{T}} \in R^{(B+1)\times N_b}$, and $diag\left(R_k\right)$ can be represented as $diag\left(R_k\right) = diag\left(R_k^{(1)}\ R_k^{(2)}\ \cdots\ R_k^{(B)}\right)$.

Then we have Eqs (20) and (21)

Algorithm 3: Federated Maximum Diversity Clustering

Input: $Z^{(b)}$, ϕ.
Output: $R^{(b)}$.
repeat
  [Institute]:
  $Y^{(b)} \leftarrow Z^{(b)}R^{(b)^T}$;
  Send $Y^{(b)}$ to the centerr ;
  [Center]:
  $Y = \sum_b Y^{(b)}$
  $Y_{.,i} \leftarrow \frac{Y_{.,i}}{\|Y_{.,i}\|_2}$
  Generate update block fraction and send Y to all institutes ;
  foreach *Institute* **do**
    $in \leftarrow$ cells to update in block
    $E \leftarrow E - R_{in}1Pr_b^T$
    $O \leftarrow O - R_{in}\phi_{in}^T$
    $R_{in} \leftarrow \exp\left(-\frac{2\left(1-Y^T\hat{Z}_{in}\right)}{\sigma}\right)$
    $\Omega \leftarrow (E+1/O+1)^\theta\phi_{in}$
    $R_{in} \leftarrow R_{in} \circ \Omega$
    $R_{in} \leftarrow R_{in} \cdot \mathrm{diag}\left(\mathbf{1}^T R_{in}\right)^{-1}$
    $E \leftarrow E + R_{in}1Pr_b^T$
    $O \leftarrow O + R_{in}\phi_{in}^T$
    Send E, O to next institute
  until *the final institute*
until *convergence*;
return $R^{(b)}$

Fig 6. Federated Maximum Diversity Clustering Algorithm.

https://doi.org/10.1371/journal.pcbi.1013526.g006

$$\phi^* diag\left(R_k\right) Z^T = \sum_{b=1}^{B} \phi_b^* diag\left(R_k^{(b)}\right) Z^{(b)^T}, \tag{20}$$

$$\phi^* diag\left(R_k\right) \phi^{*T} = \sum_{b=1}^{B} \phi_b^* diag\left(R_k^{(b)}\right) \phi_b^{*T}. \tag{21}$$

For simplicity, we denote $T_k = \phi^* diag\left(R_k\right) Z^T$, $T_k^{(b)} = \phi_b^* diag\left(R_k^{(b)}\right) Z^{(b)^T}$, $S_k = \phi^* diag\left(R_k\right) \phi^{*T}$ and $S_k^{(b)} = \phi_b^* diag\left(R_k^{(b)}\right) \phi_b^{*T}$. Both $T_k^{(b)}$ and $S_k^{(b)}$ can be computed locally at each institution, so to solve W_k, we only need the center to aggregate these two terms from each institution. Once all W_k for $k = 1 \ldots k$ was calculated, we set $W_{k[0,\cdot]}$ to 0 to effectively remove batch-independent terms. Then $\left\{W_k\right\}_{k=1}^{K}$ are sent to each institution for data integration. In Harmony, the data are centralized and are corrected using Eq. (22)

$$\widehat{Z} = \left[\widehat{Z^1}\ \widehat{Z^2} \cdots \widehat{Z^B}\right] = Z - W_k^T \phi^* diag\left(R_k\right) = \left[Z^{(1)}\ Z^{(2)}\ \cdots\ Z^{(B)}\right] - W_k^T \left[\phi_1^* diag\left(R_k^{(1)}\right)\ \phi_2^* diag\left(R_k^{(2)}\right)\ \cdots\ \phi_2^* diag\left(R_k^{(B)}\right)\right], \tag{22}$$

but when the data are decentralized as in Eq. (23):

$$\widehat{Z^b} = Z^{(b)} - W_k^T \phi_b^* diag\left(R_k^{(b)}\right). \tag{23}$$

Each institution performs only matrix multiplication rather than engaging in compilated and time-consuming computations like matrix inverse. The whole process of federated mixture of experts correction can be summarized as Algorithm in Fig 7:

Supporting information

S1 Fig. The performance of Federated Harmony on two scRNA-seq data. For **a-c**, the upper plot is the iLISI density plot, the lower one is the UMAP. **a**: iLISI density plot and UMAP before integration; **b**: iLISI density plot and UMAP after Harmony integration; **c**: iLISI density plot and UMAP after Federated Harmony integration; **d**: box plots of ARI values of naive k-means clustering results for Harmony-integrated and Federated Harmony-integrated embeddings.
(TIF)

S2 Fig. Comparison of running time and iterations for Federated Harmony and Harmony across different datasets and batch conditions.
(TIF)

S3 Fig. UMAPs by cell type for each donor (institution). Sample S00030 contains only 13 cells, which is why its corresponding subplot appears sparse.
(TIF)

S1 Info. Further Result on scRNA-seq Data.
(DOCX)

S2 Info. Computational Efficiency Comparison.
(DOCX)

S3 Info. Per-donor (batch) Downstream Analysis.
(DOCX)

Algorithm 4: Federated Mixture of Experts Correction

Input: $Z^{(b)}$, $R^{(b)}$, and ϕ_b
Output: $\hat{Z}^{(b)}$
[Institute]:
for *each institute* b *and each cluster* $k = 1$ *to* K **do**
  $\phi_b^* \leftarrow \begin{bmatrix} \mathbf{1}_{1\times N_b} \\ \phi_b \end{bmatrix}$
  $\hat{Z}^{(b)} \leftarrow Z^{(b)}$
  $S_k^{(b)} = \phi_b^* \operatorname{diag}\left(R_k^{(b)}\right) \phi_b^{*T}$
  $T_k^{(b)} = \phi_b^* \operatorname{diag}\left(R_k^{(b)}\right) Z^{(b)^T}$
  Send $\{S_k^{(b)}, T_k^{(b)}\}$ for all k to the center
[Center]: **for** *each cluster* $k = 1$ *to* K **do**
  $S_k \leftarrow \sum_b S_k^{(b)}$
  $T_k \leftarrow \sum_b T_k^{(b)}$
  Compute Correction Coefficients W_k:
  $W_k \leftarrow (S_k + \lambda \mathbf{I})^{-1} T_k$
  $W_{k[0,\cdot]} \leftarrow 0$
  Send $\{W_k\}_{k=1}^K$ to all institutes
[Institute]:
for *each institute* b **do**
  for *each cluster* $k = 1$ *to* K **do**
    $\widehat{Z^b} = Z^{(b)} - W_k^T \phi_{\mathrm{b}}^* \operatorname{diag}\left(R_k^{(\mathrm{b})}\right)$
return $\hat{Z}^{(b)}$;

Fig 7. Federated Mixture of experts Correction Algorithm.

https://doi.org/10.1371/journal.pcbi.1013526.g007

S1 Table. URL for datasets being used.
(DOCX)

Acknowledgments

The research was partially supported by the University of Pittsburgh Center for Research Computing (RRID:SCR_022735) through the HTC cluster resources provided, which was supported by NIH award S10OD028483.

Author contributions

Conceptualization: Ruizhi Yuan, Wei Chen, Lu Tang.

Data curation: Ruizhi Yuan, Haoran Hu, Tianhao Liu, Shiyue Tao.

Formal analysis: Ruizhi Yuan, Ziqi Rong.

Funding acquisition: Wei Chen, Lu Tang.

Investigation: Ruizhi Yuan, Ziqi Rong.

Methodology: Ruizhi Yuan, Ziqi Rong.

Software: Ruizhi Yuan.

Supervision: Wei Chen, Lu Tang.

Validation: Ruizhi Yuan.

Visualization: Ruizhi Yuan.

Writing – original draft: Ruizhi Yuan, Wei Chen, Lu Tang.

Writing – review & editing: Ruizhi Yuan, Ziqi Rong, Haoran Hu, Tianhao Liu, Shiyue Tao, Wei Chen, Lu Tang.

References

1. Vandereyken K, Sifrim A, Thienpont B, Voet T. Methods and applications for single-cell and spatial multi-omics. Nat Rev Genet. 2023;24(8):494–515. https://doi.org/10.1038/s41576-023-00580-2 PMID: 36864178
2. Regev A, Teichmann SA, Lander ES, Amit I, Benoist C, Birney E, et al. The Human Cell Atlas. Elife. 2017;6:e27041. https://doi.org/10.7554/eLife.27041 PMID: 29206104
3. Goh WWB, Wang W, Wong L. Why Batch Effects Matter in Omics Data, and How to Avoid Them. Trends Biotechnol. 2017;35(6):498–507. https://doi.org/10.1016/j.tibtech.2017.02.012 PMID: 28351613
4. Butler A, Hoffman P, Smibert P, Papalexi E, Satija R. Integrating single-cell transcriptomic data across different conditions, technologies, and species. Nat Biotechnol. 2018;36(5):411–20. https://doi.org/10.1038/nbt.4096 PMID: 29608179
5. Haghverdi L, Lun ATL, Morgan MD, Marioni JC. Batch effects in single-cell RNA-sequencing data are corrected by matching mutual nearest neighbors. Nat Biotechnol. 2018;36(5):421–7. https://doi.org/10.1038/nbt.4091 PMID: 29608177
6. Hie B, Bryson B, Berger B. Efficient integration of heterogeneous single-cell transcriptomes using Scanorama. Nat Biotechnol. 2019;37(6):685–91. https://doi.org/10.1038/s41587-019-0113-3 PMID: 31061482
7. Polański K, Young MD, Miao Z, Meyer KB, Teichmann SA, Park J-E. BBKNN: fast batch alignment of single cell transcriptomes. Bioinformatics. 2020;36(3):964–5. https://doi.org/10.1093/bioinformatics/btz625 PMID: 31400197
8. Korsunsky I, Millard N, Fan J, Slowikowski K, Zhang F, Wei K, et al. Fast, sensitive and accurate integration of single-cell data with Harmony. Nat Methods. 2019;16(12):1289–96. https://doi.org/10.1038/s41592-019-0619-0 PMID: 31740819
9. Zhang Y, Parmigiani G, Johnson WE. ComBat-seq: batch effect adjustment for RNA-seq count data. NAR Genom Bioinform. 2020;2(3):lqaa078. https://doi.org/10.1093/nargab/lqaa078 PMID: 33015620
10. Ferguson DN, editor A privacy concern: Bioinformatics and storing biodata. In: The ADMI 2021 Symposium; 2021.
11. National Institutes of Health. Genomic Data Sharing Policy. Available from: https://sharing.nih.gov/genomic-data-sharing-policy
12. Voigt P, Von dem Bussche A. The eu general data protection regulation (gdpr). A Practical Guide. 1st Ed. Cham: Springer International Publishing; 2017;10(3152676):10-5555.
13. Personal Information Protection Law (PIPL), 2021.
14. Lei Geral de Proteção de Dados (LGPD), Law No. 13,709, 2018.
15. Oestreich M, Chen D, Schultze JL, Fritz M, Becker M. Privacy considerations for sharing genomics data. EXCLI J. 2021;20:1243–60. https://doi.org/10.17179/excli2021-4002 PMID: 34345236
16. Erlich Y, Narayanan A. Routes for breaching and protecting genetic privacy. Nat Rev Genet. 2014;15(6):409–21. https://doi.org/10.1038/nrg3723 PMID: 24805122
17. McMahan B, Moore E, Ramage D, Hampson S, y Arcas BA, editors. Communication-efficient learning of deep networks from decentralized data. In: Artificial intelligence and statistics. PMLR; 2017.
18. Rieke N, Hancox J, Li W, Milletarì F, Roth HR, Albarqouni S, et al. The future of digital health with federated learning. NPJ Digit Med. 2020;3:119. https://doi.org/10.1038/s41746-020-00323-1 PMID: 33015372
19. Li L, Fan Y, Tse M, Lin K-Y. A review of applications in federated learning. Comput Ind Eng. 2020;149:106854. https://doi.org/10.1016/j.cie.2020.106854
20. Li S, Yan M, Yuan R, Liu M, Liu N, Hong C. FedIMPUTE: Privacy-preserving missing value imputation for multi-site heterogeneous electronic health records. J Biomed Inform. 2025;165:104780. https://doi.org/10.1016/j.jbi.2025.104780 PMID: 40054590
21. Zhang C, Xie Y, Bai H, Yu B, Li W, Gao Y. A survey on federated learning. Knowl Based Syst. 2021;216:106775. https://doi.org/10.1016/j.knosys.2021.106775
22. Tran HTN, Ang KS, Chevrier M, Zhang X, Lee NYS, Goh M, et al. A benchmark of batch-effect correction methods for single-cell RNA sequencing data. Genome Biol. 2020;21(1):12. https://doi.org/10.1186/s13059-019-1850-9 PMID: 31948481

23. Luecken MD, Büttner M, Chaichoompu K, Danese A, Interlandi M, Mueller MF, et al. Benchmarking atlas-level data integration in single-cell genomics. Nat Methods. 2022;19(1):41–50. https://doi.org/10.1038/s41592-021-01336-8 PMID: 34949812
24. Liang Y, Balcan MF, Kanchanapally V, Woodruff D. Improved distributed principal component analysis. Adv Neural Inf Process Syst. 2014;27.
25. Zheng GXY, Terry JM, Belgrader P, Ryvkin P, Bent ZW, Wilson R, et al. Massively parallel digital transcriptional profiling of single cells. Nat Commun. 2017;8:14049. https://doi.org/10.1038/ncomms14049 PMID: 28091601
26. McInnes L, Healy J, Melville J. Umap: Uniform manifold approximation and projection for dimension reduction. arXiv preprint arXiv:180203426. 2018.
27. Ahern DJ, Ai Z, Ainsworth M, Allan C, Allcock A, Angus B, et al. A blood atlas of COVID-19 defines hallmarks of disease severity and specificity. Cell. 2022;185(5):916–38.e58.
28. Lumbut C, Ponnoprat D. Investigating privacy leakage in dimensionality reduction methods via reconstruction attack. J Inf Security Appl. 2025;92:104102. https://doi.org/10.1016/j.jisa.2025.104102
29. Kwatra S, Torra V, editors. Data reconstruction attack against principal component analysis. In: International Symposium on Security and Privacy in Social Networks and Big Data. Springer; 2023.
30. Dennis DK, Li T, Smith V, editors. Heterogeneity for the win: One-shot federated clustering. In: International Conference on Machine Learning. PMLR; 2021.