Supplementary Materials

scFedVI: A Privacy-Preserving Approach to Mitigating Batch Effects in Single-Cell RNA-Sequencing Data

Parishad Mokhber[1 †], Alireza Gargoori Motlagh[2 †],
Babak H. Khalaj[2 *]

[1]Department of Computer Science, Sharif University of Technology, Tehran, Iran
[2]Department of Electrical Engineering, Sharif University of Technology, Tehran, Iran

[†]These authors contributed equally to this work.
*Corresponding Author. Email: khalaj@sharif.edu

1 Figures

Please refer to the next pages.

2 Data and Code Availability

The raw pancreatic dataset can be retrieved from this publicly available link as *h5ad* format: `https://figshare.com/ndownloader/files/24539828`. *scFedVI* is implemented in Python and available on this GitHub repository: `https://github.com/parishadmk/scFedVI`.

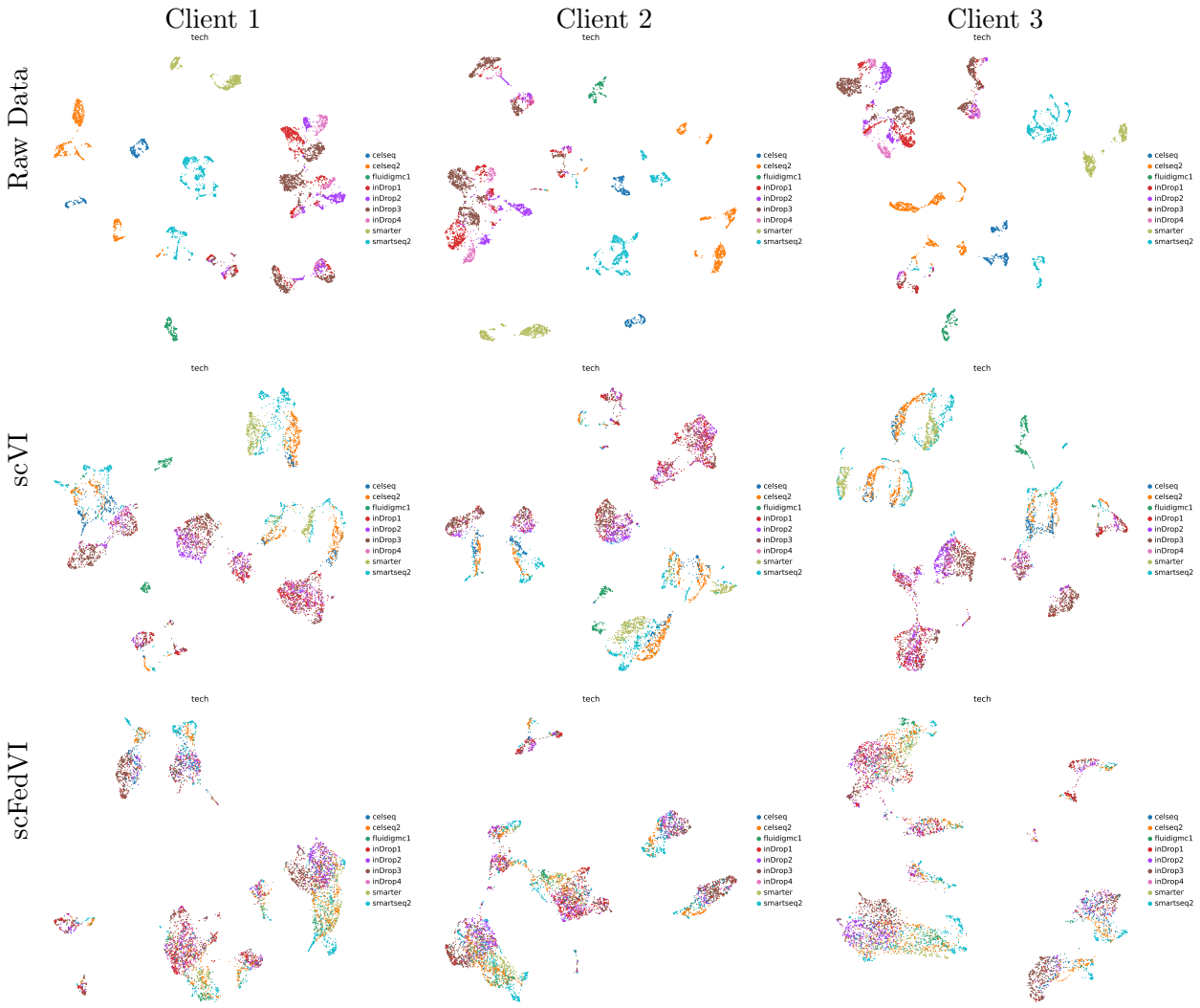

Figure 1: **UMAP Visualization of Batches (i.e. Sequencing Technologies) for the 3-Client Case**: In the first row, raw data (not batch corrected) for each client is plotted. The second row indicates the results of independent scVI batch corrections for clients. The third line, scFedVI, contains the results of our approach which uses federated-averaging among clients without privacy concerns. As it can be seen, our method has successfully accounted for batches, while preserving the biological insights, indicated by the next figure.

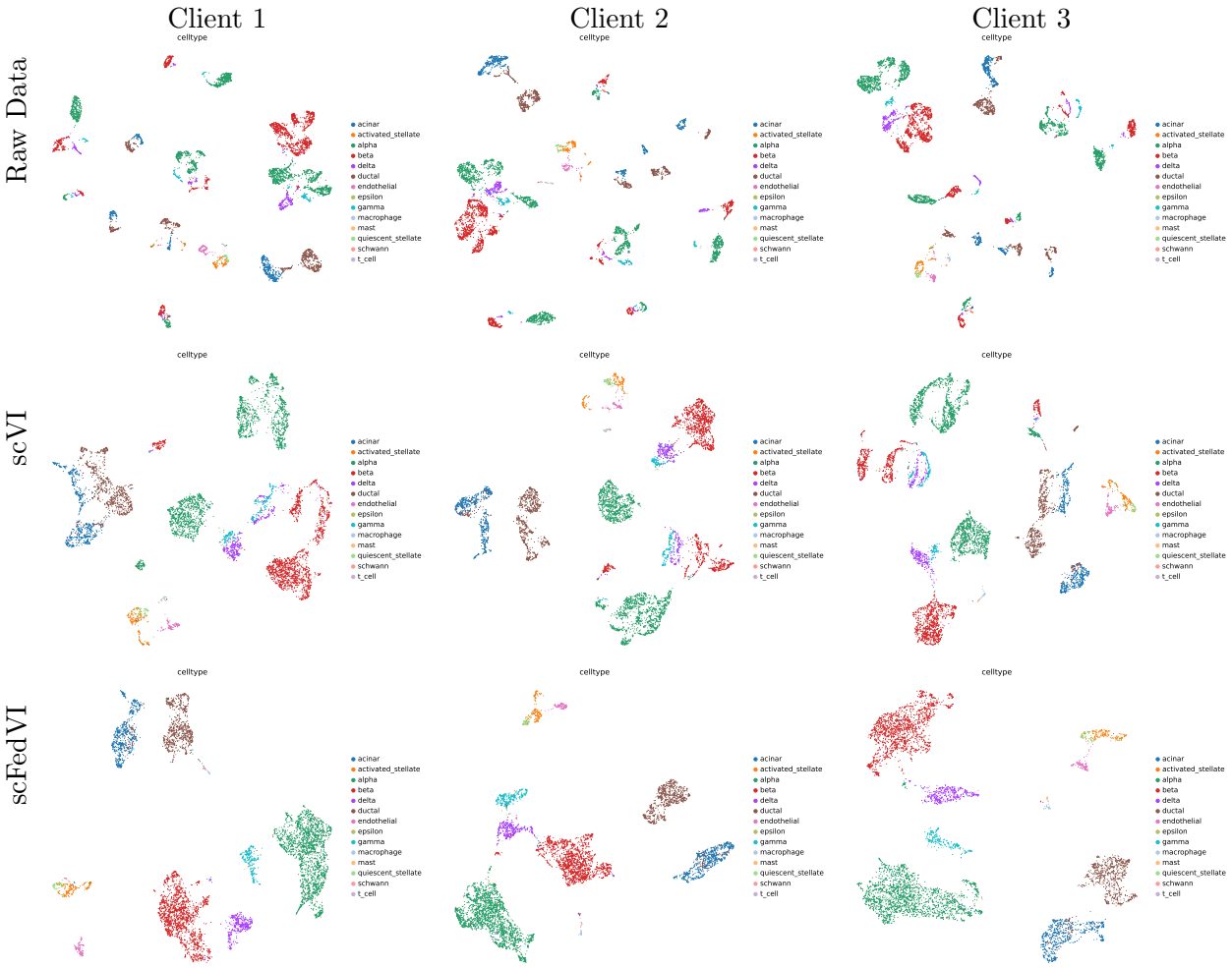

Figure 2: **UMAP Visualization of Cell Types for the 3-Client Case**: In the first row, raw data (not batch corrected) for each client is plotted. The second row indicates the results of independent scVI cell types for clients. The third line, scFedVI, contains the results of our approach which uses federated-averaging among clients without privacy concerns. As it can be seen, our method has successfully accounted for batches, while preserving the cell types distributions for downstream analysis such as clustering.

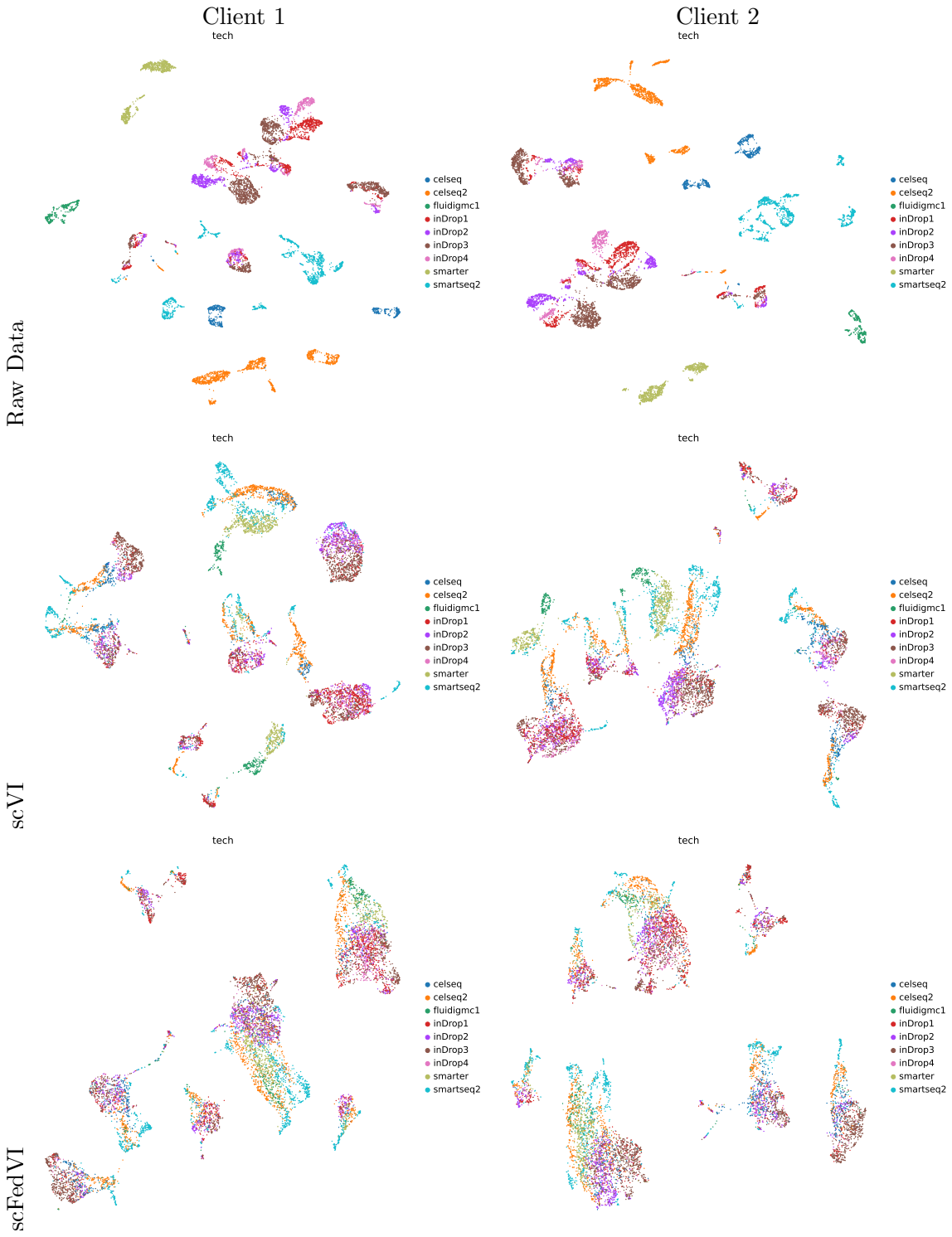

Figure 3: **UMAP Visualization of Batches (i.e., Sequencing Technologies) for the 2-Client Case**: In the first row, raw data (not batch corrected) for each client is plotted. The second row indicates the results of independent scVI batch corrections for clients. The third line, scFedVI, contains the results of our approach which uses federated-averaging among clients without privacy concerns. This demonstrates how our method successfully accounts for batches, while preserving the biological insights.

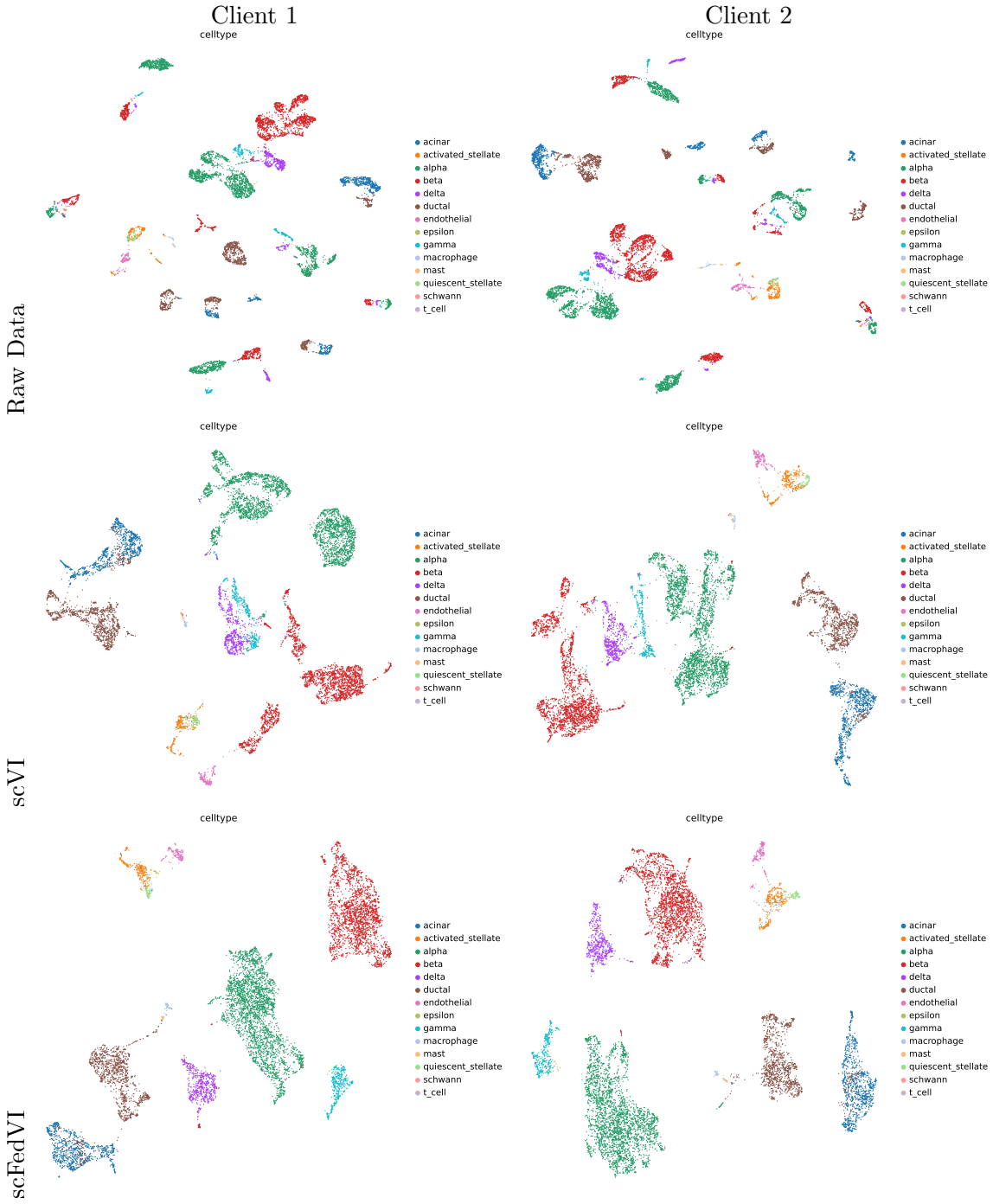

Figure 4: **UMAP Visualization of Cell Types for the 2-Client Case**: In the first row, raw data (not batch corrected) for each client is plotted. The second row indicates the results of independent scVI cell types for clients. The third line, scFedVI, contains the results of our approach which uses federated-averaging among clients without privacy concerns. As it can be seen, our method has successfully accounted for batches, while preserving the cell types distributions for downstream analysis such as clustering.

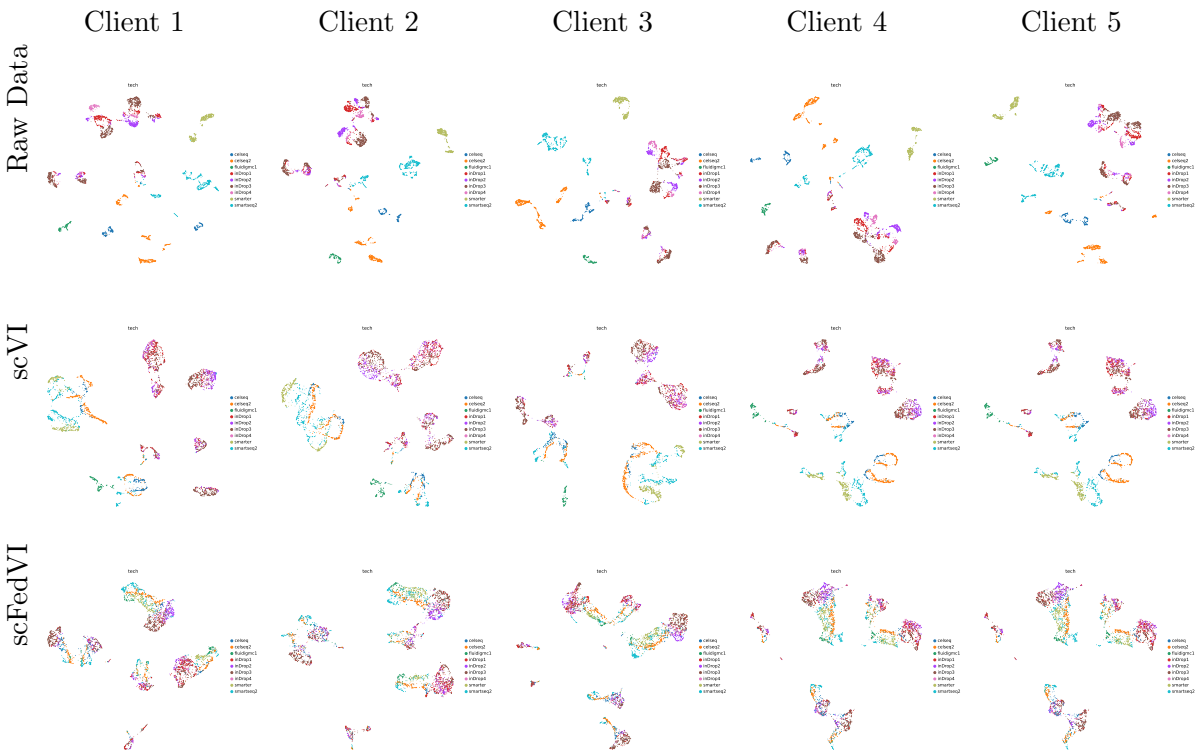

Figure 5: **UMAP Visualization of Batches (i.e. Sequencing Technologies) for the 5-Client Case**: In the first row, raw data (not batch corrected) for each client is plotted. The second row indicates the results of independent scVI batch corrections for clients. The third line, scFedVI, contains the results of our approach which uses federated-averaging among clients without privacy concerns. As it can be seen, our method has successfully accounted for batches, while preserving the biological insights, indicated by the next figure. (**Notice the sharp decline in batch correction effectiveness of scVI while the average number of samples per client decrease, while scFedVI is robust to the this criteria.**

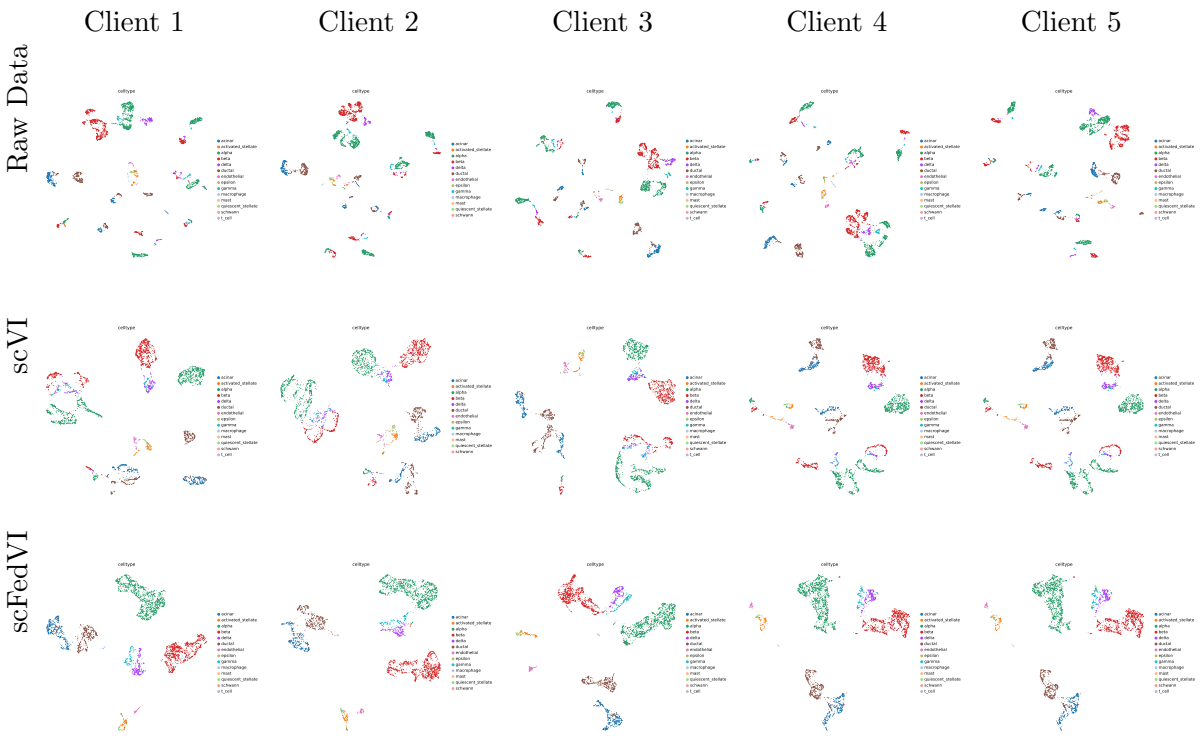

Figure 6: **UMAP Visualization of Cell Types for the 5-Client Case**: In the first row, raw data (not batch corrected) for each client is plotted. The second row indicates the results of independent scVI cell types for clients. The third line, scFedVI, contains the results of our approach which uses federated-averaging among clients without privacy concerns. As it can be seen, our method has successfully accounted for batches, while preserving the cell types distributions for downstream analysis such as clustering.

3 Methods

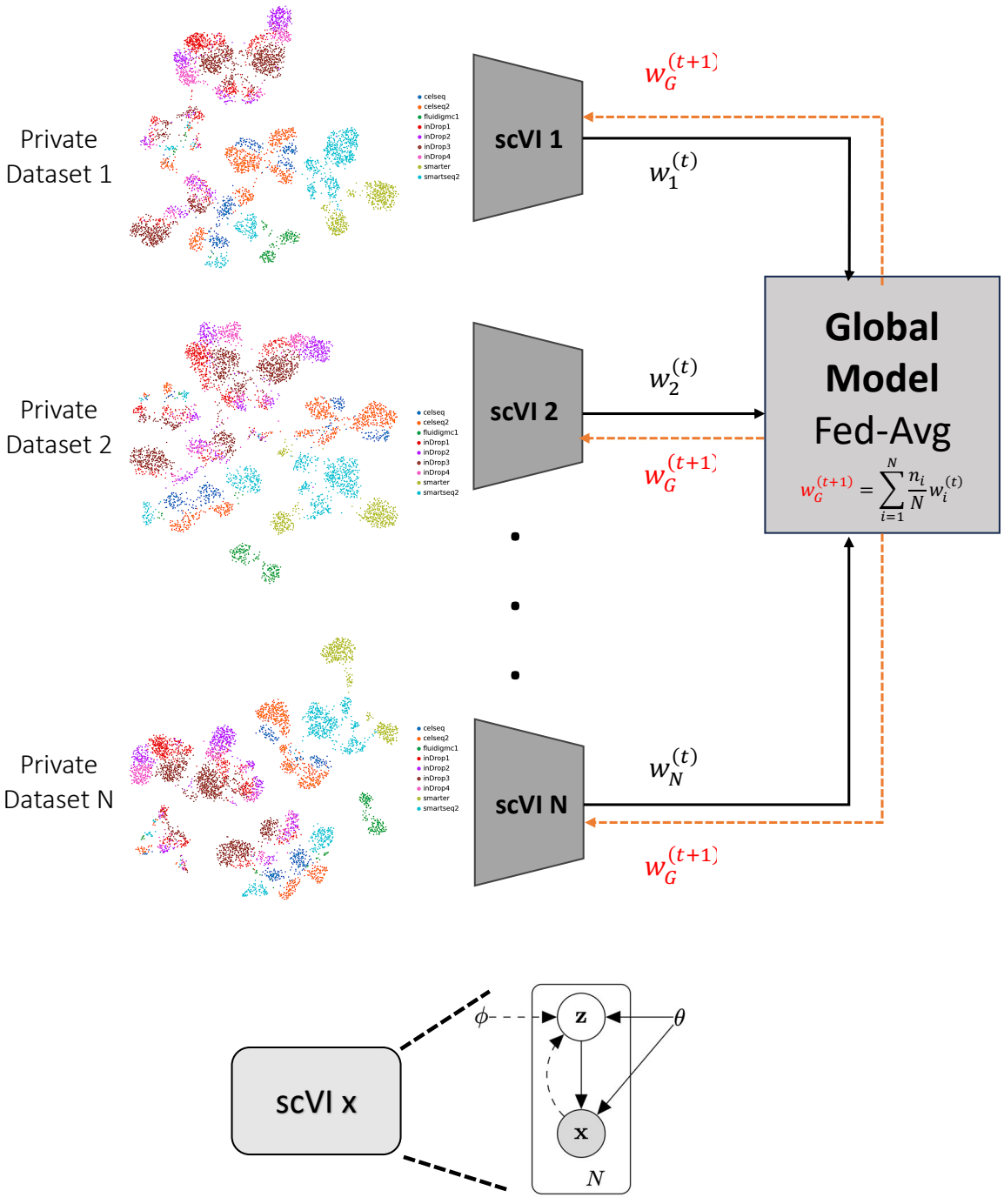

Figure 7: scFedVI Model

Algorithm 1 scFedVI Algorithm

Require: N clients, each with dataset D_i of size n_i, $i = 1, \ldots, N$
Ensure: Global VAE model parameters updated with Fed-Avg
  for $round = 1$ to Fed_rounds **do**
    for each client $i = 1$ to N in parallel **do**
      Initialize VAE model with parameters $\theta_i^{(round)}$
      Train VAE on dataset D_i minimizing ELBO loss:
      $\mathcal{L}(\theta_i^{(round)}, D_i) = -E_{q_{\theta_i^{(round)}}(z|x)}[\log p_{\theta_i^{(round)}}(x|z)] + KL(q_{\theta_i^{(round)}}(z|x)||p(z))$
      Store updated parameters $\theta_i^{(round+1)}$
    end for
    Aggregate parameters with Federated Averaging:
    $\theta^{(round+1)} = \sum_{i=1}^{N} \frac{n_i}{N} \theta_i^{(round+1)}$
    for each client $i = 1$ to N **do**
      Update client model parameters: $\theta_i^{(round+1)} = \theta^{(round+1)}$
    end for
  end for=0