# scMoMaT: Mosaic integration of single cell multi-omics matrices using matrix trifactorization

**Ziqi Zhang[1], Haoran Sun[2], Xinyu Chen[3], Ragunathan Mariappan[4], Xi Chen[5], Mika S Jain[6], Mirjana Efremova[7], Vaibhav Rajan[4], Sarah A Teichmann[6], and Xiuwei Zhang[1,*]**

[1]School of Computational Science and Engineering, Georgia Institute of Technology, Atlanta, GA 30332

[2]School of Mathematics, Georgia Institute of Technology, Atlanta, GA 30332

[3]Bioengineering Program, Georgia Institute of Technology, Atlanta, Georgia, USA, 30332

[4]Department of Information Systems and Analytics, National University of Singapore, Singapore

[5]Department of Biology, Southern University of Science and Technology, China

[6]Wellcome Sanger Institute, United Kingdom

[7]Cancer Research UK Barts Center

[*]Correspondence should be addressed to xiuwei.zhang@gatech.edu

## ABSTRACT

Single-cell multi-omics technology is able to measure cells from multiple data modalities. Data integration method aims to integrate cells across data batches and modalities, and learn a comprehensive view of the cells. Data integration can be categorized into horizontal, vertical, diagonal, and mosaic integration, where mosaic integration is the most general and the most challenging case. Most of the existing data integration methods can only work with the first three cases, which constrains their applicability to various data integration tasks. We propose scMoMaT, a method that is able to integrate single cell multi-omics data under the mosaic scenario using matrix tri-factorization. The framework of scMoMaT makes it possible to uncover the cell type specific bio-markers at the same time when learning a unified cell representation. Moreover, scMoMaT can integrate cell batches with unequal cell type composition. Applying scMoMaT to multiple biological datasets shows that scMoMaT has superior performance compared to baseline methods.

## 1 Introduction

The advance in single-cell multi-omics technology makes it possible to measure the activities of a single cell from different modalities. Single-cell RNA-sequencing (scRNA-seq) technology is able to measure the gene expression of every individual cell, whereas single-cell ATAC-sequencing (scATAC-seq) measures the chromatin accessibility of individual cells. On the other hand, new sequencing technologies have been proposed to profile more than one modality in a cell simultaneously. There exist sequencing technologies that are able to profile both protein abundance and gene expression[1], chromatin accessibility and gene expression[2], or chromatin accessibility and

protein abundance[3] within a cell at the same time. Jointly analyzing cells from multiple modalities provides a comprehensive view of cellular identity and its marking features (e.g. chromatin regions, genes, proteins, etc), and also help to better understand the underlying cross-modalities interactions that define the cell identity. An ideal way for such an analysis would be directly studying the cell population where all data modalities are simultaneously measured. However, the bulk of existing single-cell datasets are measured with only one modality. Simultaneously profiled datasets, even exist, are usually only measured with two modalities, which makes it hard to derive a comprehensive view of cells from a single dataset.

Single-cell data integration methods integrate cells from the same biological system that are sequenced from different data batches where one or multiple data modalities are measured, and learn a unitary representation for each cell that encodes the biological information learned from multiple modalities. The sequencing data from one batch and one modality corresponds to one data matrix. For example, if scRNA-seq is performed on a batch of cells, we obtain an mRNA count matrix where rows correspond to genes and columns correspond to cells. Similarly, performing scATAC-seq on a batch of cells gives rise to a matrix where rows are chromatin regions and columns are cells. We term the object that each dimension of the matrix represents as an *entity*. For example, "cells", "genes", "chromatin regions" are entities. Genes and chromatin regions are feature entities as they are the features of a cell. Similarly, proteins can also be a feature entity when protein abundance is measured.

Data integration tasks on such single cell data matrices can be separated into four different scenarios according to[4]: *horizontal integration*, or termed batch effect removal, refers to the case where all data batches have the same modalities. In this scenario, the feature entity between data matrices is shared, whereas cell entities are different between data matrices. *Vertical integration* refers to the case where a data batch is measured with different modalities. In such a scenario, the cell entity is shared but feature entities are different between data matrices. *Diagonal integration* refers to the case that neither cells nor features are shared between data batches. In such a scenario, both the cell entities and feature entities are different between data matrices. *Mosaic integration* is the most general case and can be any combination of horizontal, vertical, and diagonal integration. Considering an $m \times b$ grid corresponding to $m$ modalities and $b$ batches, mosaic integration methods aim to integrate any subset of data matrices from this grid.

Various methods have been proposed to deal with different integration scenarios. LIGER[5] and Seurat v3[6] were developed for horizontal and diagonal integration. CoupleNMF[7], UnionCom[8], MMD-MA[9], and scDART[10] were developed specifically to deal with diagonal integration task. Seurat v4[11], scAI[12], and MultiVI[13] were developed for vertical integration task. Recently, new methods have been proposed to work with less restricted integration scenarios. MultiMap[14] is generalized from UMAP, which constructs a neighborhood graph between cells from all data batches using the shared and unshared features, and projects the graph onto low dimensional space. Bridge[15], on the other hand, uses one jointly profiled data batch that includes all modalities as the bridges and integrates all

data batches using dictionary learning. MultiMap and Bridge still have limitations on the integration scenarios that they can be applied to. Bridge always needs one batch of cells where all modalities are measured (to serve as the "bridge"). Such batch might not exist in real-world integration tasks. MultiMap mainly works on the scenario where only one modality is measured for each batch. It does not discuss how to utilize the information from all modalities for the batches that are measured with more than one modality. UINMF[16] uses a matrix bi-factorization framework to integrate data matrices with shared and unshared features. UINMF works with most mosaic integration cases, but it focuses on learning cell embedding and does not simultaneously learn a feature embedding along with the marker features of cell identities.

Here we propose scMoMaT (**s**ingle **c**ell **M**ulti-**o**mics integration using **Ma**trix **T**rifactorization), a data integration framework that is designed to integrate an arbitrary number of data matrices under mosaic integration scenario. Apart from integrating cells, scMoMaT is also able to extract cell type specific bio-marker features. scMoMaT extracts the bio-marker not only from the original feature of the data matrices, but also from the curated features that are generated by other methods. For example, users can provide the motif deviation matrix learned from the original scATAC-seq matrix using chromVAR, and scMoMaT can extract the motif marker from the matrix. In addition, scMoMaT does not assume cells to have similar distribution across batches, which makes scMoMaT capable of integrating cell batches with disproportionate cell type composition.

scMoMaT uses a matrix tri-factorization framework, which treats each single cell data matrix as a relationship matrix between the cell and feature entity. It factorizes a data matrix (which corresponds to a batch) into batch-specific cell factor, feature factor, and a factor association matrix. It connects data matrices of different data batches and modalities by enforcing the factor of the same entity to be shared among tri-factorization terms.

We tested scMoMaT on four real datasets covering various kinds of integration scenarios, including one human PBMC dataset, one mouse brain cortex dataset, one human bone marrow dataset, and one mouse spleen dataset. We compare scMoMaT with state-of-the-art data integration methods, and the results show that scMoMaT has superior performance in learning cell embedding, discovering biomarkers, and dealing with disproportionate cell type composition between batches.

## 2 Results

### 2.1 Framework of scMoMaT

scMoMaT uses matrix tri-factorization to learn the low dimensional cell factor and feature factor from data matrices. Given a single-cell data matrix $\mathbf{X}_{ij}$ where $i$ refers to the cell batch and $j$ refers to the feature modality, matrix tri-factorization factorizes the data matrix into a cell factor $\mathbf{C}_i$, a feature factor $\mathbf{C}_j$, and an association matrix $\Sigma_{ij}$ by

minimizing the reconstruction loss:

$$\hat{\mathbf{C}}_i, \hat{\Sigma}_{ij}, \hat{\mathbf{C}}_j = \arg\min_{\mathbf{C}_i, \mathbf{C}_j, \Sigma_{ij}} \|\mathbf{X}_{ij} - \mathbf{C}_i \Sigma_{ij} \mathbf{C}_j^T\|_F^2 \tag{1}$$

In order for the factorization to only capture the major biological variation within the data, the number of latent dimensions $d$ (number of columns in $\mathbf{C}_i$ and $\mathbf{C}_j$) should be much smaller than the number of cells or features in the data matrix. We fix the latent dimension $d$ to be 30 in all test results of the manuscript. We assume that each latent dimension encodes a distinct biological factor of the dataset, and the factor values of each cell or feature (row vectors of $\mathbf{C}_i$ or $\mathbf{C}_j$) encode the proportion of each biological factor with the cell or feature identity. Under such an assumption, we constrain the factor values of each cell to be simplex, i.e. they are all non-negative and sum up to 1 for each cell or feature. As the data matrix is always non-negative, we constrain the association matrix $\Sigma_{ij}$ to be non-negative too. We also include the bias and scaling terms into the reconstruction loss to better accommodate the cell- and feature-specific bias and scaling of the data matrix. The objective function can be written as:

$$\begin{aligned} \hat{\mathbf{C}}_i, \hat{\Sigma}_{ij}, \hat{\mathbf{C}}_j, \hat{\mathbf{b}}_i, \hat{\mathbf{b}}_j, \hat{\alpha}_{ij} &= \arg\min_{\mathbf{C}_i, \mathbf{C}_j, \Sigma_{ij}, \mathbf{b}_i, \mathbf{b}_j, \alpha_{ij}} \|\mathbf{X}_{ij} - \alpha_{ij} \mathbf{C}_i \Sigma_{ij} \mathbf{C}_j^T - \mathbf{b}_i - \mathbf{b}_j^T\|_F^2 \\ s.t. \quad & \mathbf{C}_\mathrm{x} \cdot \mathbf{1} = \mathbf{1}, \mathbf{C}_\mathrm{x} \geq 0, \Sigma_{ij} \geq 0 \end{aligned} \tag{2}$$

$\mathbf{b}_i$ and $\mathbf{b}_i$ are 1 dimensional bias vectors for cell batch $i$ and feature modality $j$ respectively. $\alpha_{ij}$ is the matrix scaling parameter. $\mathbf{C}_\mathrm{x} \cdot \mathbf{1} = \mathbf{1}$, $\mathbf{C}_\mathrm{x} \geq 0$ constrains each row of the factor ($\mathbf{C}_i$ and $\mathbf{C}_j$) to be a simplex. A graphical illustration of the factorization is shown in Fig. 1a. Compared to matrix bi-factorization[17], the use of tri-factorization allows scMoMaT to have a more biologically interpretable framework and learns a feature factor as accurate as the cell factor, which is better for extracting marker features and other downstream analysis.

We have discussed how scMoMaT works on one data matrix, now consider the case where we have multiple single-cell data matrices from different data batches and modalities. We use the case where we have multiple single-cell gene expression, chromatin accessibility and protein abundance matrices for ease of explanation. Denoting the gene expression matrices as $\{\mathbf{G}_i\}_{i \in S_g}$ where rows correspond to cells and columns correspond to genes, the chromatin accessibility matrices by $\{\mathbf{R}_j\}_{j \in S_r}$ where rows correspond to cells and columns correspond to chromatin regions, and the protein abundance matrices by $\{\mathbf{P}_k\}_{k \in S_p}$ where rows correspond to cells and columns correspond to proteins. $S_g$, $S_r$ and $S_p$ are the batch indices corresponding to gene expression, chromatin accessibility, and protein abundance matrices. The objective function of scMoMaT can be written as:

$$\begin{aligned} & \arg\min_{\mathbf{C}_\mathrm{x}, \Sigma} L \\ s.t. \quad & \mathbf{C}_\mathrm{x} \cdot \mathbf{1} = \mathbf{1}, \mathbf{C}_\mathrm{x} > 0, \Sigma \geq 0 \end{aligned} \tag{3}$$

And

$$\begin{aligned} L = & \sum_{i \in S_g} \|\mathbf{G}_i - \alpha_{ig} \mathbf{C}_i (\Sigma + \Sigma_{ig}) \mathbf{C}_g^T - \mathbf{b}_{1i} - \mathbf{b}_{gi}^T\|_F^2 + \sum_{j \in S_r} \|\mathbf{R}_j - \alpha_{jr} \mathbf{C}_j (\Sigma + \Sigma_{jr}) \mathbf{C}_r^T - \mathbf{b}_{2j} - \mathbf{b}_{rj}^T\|_F^2 \\ & + \sum_{k \in S_p} \|\mathbf{P}_k - \alpha_{kp} \mathbf{C}_k (\Sigma + \Sigma_{kp}) \mathbf{C}_p^T - \mathbf{b}_{3k} - \mathbf{b}_{pk}^T\|_F^2 + \lambda \left( \sum_{i \in S_g} \|\Sigma_{ig}\|_F^2 + \sum_{j \in S_r} \|\Sigma_{jr}\|_F^2 + \sum_{k \in S_p} \|\Sigma_{kp}\|_F^2 \right) \end{aligned} \tag{4}$$

where $\mathbf{C}_{\mathrm{x}}$s denote factor matrices within the objective function, $\mathbf{C}_i$s are the factors for cell batches that have gene expression matrices, $\mathbf{C}_j$s are the factors for cell batches that have chromatin accessibility matrices, and $\mathbf{C}_k$s are the factors for cell batches that have protein abundance matrices. $\mathbf{C}_g$, $\mathbf{C}_r$ and $\mathbf{C}_p$ are the factors for genes, regions, and proteins. The factor of the same cell batch or feature modality is shared across the reconstruction loss of data matrices. $\Sigma$ is the shared association matrix across all data matrices, and $\Sigma_{ig}, \Sigma_{jr}, \Sigma_{kp}$ are data matrix-specific association matrices. $\mathbf{b}_{\mathrm{xx}}$s are the cell or feature specific bias vectors for each data matrix. $\alpha_{ig}, \alpha_{jr}, \alpha_{kp}$ are data matrix-specific scaling parameters. $\lambda$ is the regularization weight that regularize how much data matrix-specific association matrix should vary. We set $\lambda = 0.001$ for all test results in the manuscript.

The aforementioned framework work for the case where there always exists a common feature modality between batches (Fig. 1b), additional assumption is needed when dealing with the cases where there is no common modality between batches (Fig. 1c). We can resolve the issue by constructing pseudo-count matrices to make the corresponding modality shared across all batches when running the model. The constructing process is similar to the method used in Seurat and Liger which doesn't require additional input for the model (Methods).

After the factors are learned by minimizing the objective function, we further include an additional post-processing step to better match the cells from different batches (Methods). The post-processing step constructs a neighborhood graph of all cells, which can be visualized using UMAP and clustered using Leiden cluster algorithm[18]. After obtaining the cluster identity of the cells, we retrain the model to learn the feature factors again. Compared to the feature factor learned in the first training stage, the retrained feature factor has each latent factor dimension corresponding to one specific cell cluster, which can be used to extract the cluster-specific marker features across modalities that jointly define cell type identities (Methods).

### 2.2 scMoMaT mosaic integration on human PBMC data

We applied scMoMaT to a human PBMC dataset[3]. The dataset includes 4 batches of cells: the first 2 batches of cells are measured with gene and protein abundance simultaneously using CITE-seq[19,20] (batch 1 includes 5023 cells, and batch 2 includes 3666 cells); The last 2 batches of cells are measured with protein abundance and chromatin accessibility simultaneously using ASAP-seq[3] (batch 3 includes 3517 cells, and batch 4 includes 4849 cells). In total, there are 8 data matrices as shown in Fig. 2a. We ran scMoMaT on the dataset and visualize the cell factors of all batches using UMAP (Figs. 2b,c,d). In Fig. 2b, the cells are colored with the cell type labels obtained from the original data paper[3], where different cell types are clearly separated. In Fig. 2c, cells from the four batches are mixed in the latent space.

To compare scMoMaT with baseline methods, we ran two recently published methods which can also work with this integration scenario: MultiMap[21] and UINMF[16] (Methods, visualization of latent embedding in Figs. S1a,b), and quantitatively measured the overall performance of all four methods with Graph connectivity (GC) score, NMI

score, and ARI score (cite, Methods) using the label in the original data paper as ground truth. Graph connectivity score measures how well the cells from the same cell type are matched among different data batches, whereas NMI and ARI scores measure the separation of different cell types as clusters in the latent space. The result (Fig. 2e) shows that scMoMaT performs better than the two baseline methods with all metrics. We further validate the learned feature factors (gene, region, and protein) by re-annotating the cell types using the feature factors and comparing them with the ground truth labels. We first ran Leiden cluster algorithm on the latent space of scMoMaT (Fig. 2d) and fed the cluster label into scMoMaT for retraining.

The factor value of each gene for each cluster can be obtained from the learned gene factor, and for each cluster, the genes with highest factor values are considered marker genes of this cluster. Indeed, the top 20 genes with the largest factor values for clusters 3, 4, and 6 contain known marker genes of corresponding cell types (Fig. 2f). The top 20 genes in cluster 3 includes *GNLY*, *NKG7*, *KLRB1*, *KLRD1* and *KLRF1*, which are the marker genes of Natural Killer (NK) cell[22,23]. The top 20 genes in cluster 4 includes *CD79A* and *CD37*, which are the marker genes of B cells[24]. The top-20 genes of cluster 6 includes *S100A9*, *LYZ*,ge *NRP1*, and *CD68*, which are the marker genes of Myeloid cells. For clusters $0, 1, 2, 5, 7$, we find high factor values for genes *CD4*, *CD8A*, and *CD8B*, which are the marker genes of T cells. The annotation of cell type learned from scMoMaT gene factor matches the label in the original data paper (Fig. 2b).

The feature factors learned with scMoMaT can be used to further identify T cell subtypes in the integrated data. Clusters $0, 1, 7$ have higher *CD4* factor values, which shows that they correspond to $CD4^+$ T cells. Clusters $2, 5$ have higher *CD8A* and *CD8B* factor values. Those two genes have high expression only in $CD8^+$ T cells. The distribution of the expression value of *CD4*, *CD8A*, and *CD8B* also matches the analysis of these three genes using gene factor (Fig. S1c). Within $CD4^+$ and $CD8^+$ T cells, protein markers can be used to further separate them into naive T cells and activated T cells. Naive T cells have high abundance of surface protein *CD45RA* and low abundance of surface protein *CD45RO*. Activated T cells, on the contrary, have high *CD45RO* and low *CD45RA*[25,26]. Using the learned protein factor, we annotate cluster 0 and 2 to be naive T cells, with low *CD45RO* factor value and high *CD45RA* value (upper two plots in Fig. S1d). Cluster $1, 5, 7, 8$ are activated T cell with higher *CD45RO* and lower *CD45RA* (lower two plots in Fig. S1d). The high factor value of Naive T cell marker genes (*CCR7*, *CD27*, *TCF7*) in cluster 0 and 2 also matches the annotation of Naive T cell from protein factor[26,27] (Fig. S2a). Among the activated T cell clusters, scMoMaT is able to discover more cell sub-types using gene factors. Cluster 1 is shown to correspond to $CD4^+$ regulatory T cell (Treg) using the factor value of marker genes *Foxp3*, *CTLA4*, and *IL2RA*[27] (Fig. S2b). Cluster 7 is annotated as $CD4^+$ effector T cell with marker *IL2* and *TNF*[27] (Fig. S2c). Cluster 5 has high *GZMK* and *GZMB*[27], which are $CD8^+$ cytotoxicity markers (Fig. S2d). Cluster 8 is shown to have high factor value of $CD4^+$ tissue resident memory T cells (TRM) markers, including *ITGAE* and *ITGA1*[27] (Fig. S2e). Fig. S2f shows the complete cell type annotation of the clusters.

### 2.3 scMoMaT mosaic integration on mouse cortex data

We then applied scMoMaT on a mouse brain cortex dataset[2,28]. We collected three batches of mouse brain cortex datasets published by different research groups[2,28]. The first data batch simultaneously measures the chromatin accessibility and gene expression of $10,309$ cells using SNARE-Seq[2], whereas the second batch profiles the gene expression of $40,166$ cells using 10x v3 single-nucleus RNA-Sequencing technology (snRNA-seq) and the third batch profiles the chromatin accessibility of 8718 cells using single-nucleus ATAC-Sequencing (snATAC-seq) (Totally four data matrices organized as Fig. 3a). We visualize each data matrix separately using UMAP (without applying integration), and color the cells using the cell type labels curated and re-organized from the original data paper (Methods, Fig. 3b). The visualization shows a strong mismatch of cell type structures among different batches and modalities. We take the four data matrices as the input of scMoMaT, where an additional pseudo-scRNA-seq matrix is calculated from scATAC-seq matrix for data batch 3 (Methods) and the cells factors are learned (Methods, $\mathbf{C}_1$, $\mathbf{C}_2$, $\mathbf{C}_3$ in Fig. 3a). We visualize the cell factors of all batches using UMAP after post-processing, and cluster the cells by running Leiden clustering algorithm on the cell factors[29] (Figs. 3c,d). In Figs. 3c,d, cells are well mixed between batches, and the cell types (annotated with our learned gene factors) are separated. The cell types in Fig. 3d are annotated with the bio-markers learned from the retrained gene factors (Methods), which will be discussed below.

We visualize the factor values of different marker genes using barplots (Fig. 3e, Fig. S3a, cluster with the largest factor value is colored red). The factor values of gene markers clearly detect the cluster id that correspond to different cell types. The high factor value of *Foxp2* in cluster 0 shows that the cluster correspond to L6 excitatory neuron, and *Calb1* shows that cluster 1 matches L2/3 excitatory neuron, *Tshz2* shows that cluster 8 matches near-projecting excitatory neurons(NP), *Rorb* shows that cluster 2 and part of cluster 4, 5, 7 correspond to L4/5 excitatory neuron[28]. Non-neuron cell type are also well detected through the gene factor values, including *Csf1r* in Macrophage (cluster 11), *Lhfpl3* in oligodendrocyte precursors (OPC, cluster 12), *Slc1a2* in Astrocytes, and *Plp1* in oligodendrocyte (Oligo, cluster 9), etc[28,30].

scMoMaT can also incorporate additional data matrices learned from the original input data. For example, given a scATAC-seq data matrix, ChromVAR[31] can learn the accessibility of motifs in every single cell and output a "cell by motif" matrix. During the retraining step, we incorporated the "cell by motif" matrix to learn the motif factors (Methods). We sorted the motifs according to their factor values for each cluster, where multiple cluster-specific marker motifs are uncovered (Fig. 3f, Fig. S3b), including MA0062.2_Gabpa, MA0117.2_Mafb, MA0002.2_RUNX1 for Macrophage (cluster 11), MA0463.1_Bcl6, MA0518.1_Stat4, MA0631.1_Six3 for L6 excitatory neuron (cluster 0), MA0515.1_Sox6, MA0442.1_SOX10, MA0514.1_Sox3 for oligodendrocyte (cluster 9), etc[32]. The overall cell type annotation from motif factor shows strong consistency with the annotation from

gene factors. In addition, we measure the consistency between region factor and gene factor by summing the factor of all regions that lie within 2000 base-pair upstream of each gene and measuring the cosine similarity between the region-transferred gene factor and the original gene factor for each gene. The boxplot of cosine similarity is shown in Fig. S3c. We also color the cell using the literature curated cell type label in the cell factor visualization (Fig. S3d), where the cell type label matches the label obtained from scMoMaT analysis. We didn't use the literature curated cell type label to measure the integration accuracy of scMoMat since the label of different batches, being obtained from different research papers, is highly inconsistent in clustering standards and cell type naming.

### 2.4 scMoMaT integrates batches with no shared modalities

It is a very challenging scenario for integration if the batches do not share any features. The most common example of such a scenario is the integration of a scATAC-seq matrix and a scRNA-seq matrix obtained from different data batches (also called diagonal integration). In this scenario, a valid integration cannot be achieved without additional assumptions or information (e.g. cross-modalities relationship) being provided. Some methods integrate data batches assuming that the latent distribution of cells is similar between batches, which fails to accommodate the cases where there exist missing or disproportionate cell types in certain data batches. Other methods transform the scATAC-seq matrices into pseudo-scRNA-seq matrices (also termed as *gene activity score* in some literature) and integrate the scRNA-seq matrices and pseudo-scRNA-seq matrices. Using the pseudo-scRNA-seq instead of the scATAC-seq matrices, these methods may suffer from the errors introduced during the process of calculating the pseudo-scRNA-seq matrix and do not fully utilize the epigenomic information in the scATAC-seq matrix. scMoMaT, on the other hand, keeps both the scATAC-seq matrix and the pseudo-scRNA-seq matrix in its framework in order to better exploit the scATAC-seq information. Moreover, scMoMaT can integrate scATAC-seq and scRNA-seq data batches with disproportionate cell type composition.

We applied scMoMaT to a healthy human bone marrow mononuclear cells (BMMC) dataset[33]. The dataset includes two batches of cells, where the first batch has 16510 cells that are sequenced with scATAC-seq and the second batch has 12601 cells sequenced with scRNA-seq (Fig. 4a). scMoMaT takes as input both matrices, and generates a pseudo-scRNA-seq matrix for the second batch using its scATAC-seq data matrix (Methods). In order to fully utilize the epigenomic information, scMoMaT factorizes the scATAC-seq matrix together with the pseudo-scRNA-seq and scRNA-seq matrices. We visualize the cell factors learned from scMoMaT (Figs. 4b,c,d) using UMAP, and color the cells using the literature derived labels (Fig. 4b), data modalities (Fig. 4c), and cell label obtained from Leiden clustering algorithm (Fig. 4d). In the visualization plot, cell batches are well integrated in the latent space, while cell-type identities in each data batch are also well preserved.

We also ran baseline methods on the dataset, including UINMF, MultiMap, and Liger[5] (UMAP visualization in Figs. S4a, b, c). Liger can be considered as the version of UINMF without using epigenomic information that only

integrates pseudo-scRNA-seq matrix and scRNA-seq matrix. We quantitatively measured the overall performance of four methods using GC, NMI, and ARI scores (Methods), and summarized the scores of the methods in Fig. 4e. scMoMaT has the highest GC score, which shows that scMoMaT better matches the same cell type between batches. scMoMaT and UINMF have similar ARI and NMI scores, which are higher than the other baseline methods. The ARI and NMI scores show that scMoMaT better separates cells with different cell types compared to baseline methods. scMoMaT, UINMF, and Liger are all matrix-factorization-based methods. Compared to Liger, scMoMaT and UINMF shows superior performance in all three metrics, which shows that using epigenomic information can help to better integrate data batches. MultiMap mixes cells from different cell types in the latent space, which might due to the fact that the cell types in BMMC dataset are closely located in the original dataset as they follow the trajectories of the hematopoiesis process and MultiMap failed to distinguish the closely located cell types.

We fed the Leiden cluster labels (Fig. 4d) into scMoMaT which is retrained to learn the feature factor. To better capture the epigenomic information in each cell type, we also input the motif deviation matrix obtained by ChromVAR into scMoMaT for retraining. scMoMaT successfully capture multiple cell-type specific marker genes and relevent motifs, especially in the differentiated cell types. In cluster 9 (corresponding to natural killer cells (NK)), scMoMaT found high factor value of marker gene *GNLY*[22,23], which matches the gene expression pattern in the dataset (Fig. 4f). Similarly, scMoMaT found T-cell marker gene *CD3D*[27] in clusters 0, 3, and 5 (Fig. 4f), B-cell marker gene *CD79A*[24] in clusters 6 and 7 (Fig. 4g), Monocyte marker gene *S100A9*[34] in clusters 1, 2, and 8 (Fig. 4g), and Plasmacytoid Dendritic Cell (pDC) marker gene *PTPRS*[35] in cluster 10 (Fig. 4g). On the other hand, the motif factor also reveals multiple TF-motifs that are relevent to different cell-type formations. scMoMaT found high value of *MA0466.2_CEBPB* in Monocytes which is related to Monocyte-differentiation regulator *CEBPB*[33,36] (Fig. S4d). In addition, motif *MA0800.1_EOMES* has the highest factor value in cluster 9 (NK), which also links to *EOMES* that regulate the maturation of NK cells[37]. scMoMaT also found motifs including *MA0850.1_FOXP3*, *MA0523.1_TCF7L2*, and *MA0033.2_FOXL1* in clusters correspond to T cells, which correspond to marker TF families in T cells including FOX family, and TCF family[27].

### 2.5 scMoMaT integrates batches with unequal cell type compositions

Now that we have shown that scMoMaT generates better integration after fully utilizing the epigenomic information, we further test how scMoMaT performs when there is disproportionate cell type composition between batches using a mouse spleen dataset[21,38]. The original dataset includes two batches of cells, where the first batch has 4382 cells that are sequenced with scRNA-seq, and the second batch has 3166 cells that are sequenced with scATAC-seq (Fig. 5a). The dataset mainly consists of T cells (1190 cells in the first batch, and 990 cells in the second batch), B cells (2621 cells in the first batch, and 1835 cells in the second batch), and some other cell types that reside in mouse spleen. The original two data batches have similar cell type composition (Fig. 5b). We created data batches with

disproportionate cell type composition by taking the most populated cell type, B cells, in batch 1, and sub-sampling only 100 B cells, which drastically change the cell type composition of batch 1 (Fig. 5c, B cells correspond to blue and darker blue regions).

We ran our methods along with baseline methods including UINMF, MultiMap, and Liger on the disproportional dataset. The visualization shows that scMoMaT can correctly match cell types in two data batches regardless of the disproportionate cell type composition between two batches, especially B cells which barely exist in the first batch (Figs. 5d,e,f). The cell factors of two batches are separately plotted in Figs. 5d,e, where B cells lie within the box. The visualization of UINMF and Liger also shows good cell integration results, but the cells of different cell types are no longer clearly separated in MultiMap (Fig. S5a,b). Indeed, although MultiMap has the highest GC score, it has the lowest ARI and NMI score (Fig. 5g). scMoMaT, UINMF, and Liger showed robust performance towards disproportionate cell type composition with scMoMaT having the best overall performance among all methods. The result is reasonable as scMoMaT, UINMF, and Liger are all designed based on joint matrix factorization, which integrates cells from different batches using their biological information instead of cell position in the data manifold. Therefore, they should not be heavily affected by the disproportionate cell type composition which causes the mismatch of data manifold between batches.

## 3 Discussion

In this study, we introduce scMoMaT, a single-cell data integration method that works on mosaic integration scenarios. We applied scMoMaT on different mosaic integration tasks. The results validated the broad applicability of scMoMaT under various types of data integration scenarios, and show that scMoMaT also is capable of capturing cluster-specific bio-marker across modalities during integration, which can be used to accurately annotate cell types in exploratory data analysis tasks where cell type annotation is not known in advance. The results on mouse spleen dataset also show that scMoMaT is able to integrate batches that have disproportionate cell type composition. With the increasing availability of various types of single-cell multi-omics datasets, we expect that scMoMaT will be widely applied to various types of data integration tasks.

The advances of single cell multi-omics technology provide an opportunity to comprehensively study the regulation mechanism across different modalities within cells. While existing data integration methods mainly learn cell embedding from single cell multi-omics datasets, scMoMaT explores the direction of learning from feature embedding too. Future works should explore the potential of using single cell multi-omics data to uncover cross-modalities regulatory mechanisms while performing data integration.

## 4 Methods

### 4.1 Training procedure of scMoMaT

We minimize the loss function of scMoMaT (Equ. 3 and 4) using mini-batch stochastic gradient descent. Within each iteration, we pick one parameter matrix from cell and feature factors $\mathbf{C}_{\mathrm{x}}$s, shared and data matrix-specific association matrix $\{\Sigma, \Sigma_{\mathrm{xx}}\}$s, bias $\mathbf{b}_{\mathrm{xx}}$s, and scaling parameter $\alpha_{\mathrm{xx}}$s and fix the other parameter matrices. Then, we update a mini-batch of the selected parameter matrix using gradient descent. Each minibatch is constructed by subsampling 10 percent of cells and features in each data matrix. Then we loop through all parameter matrices and update them using gradient descent in order.

In order to enforce the simplex constraint on the factor matrices, we transform the original factor matrices using a softmax function before using it to calculate the reconstruction loss and use the softmax-transformed factor matrices as the output factor matrices of the model. We enforce the non-negativity constraint on the shared association matrices by changing all its negative values to zero after every time that it is updated.

In each iteration, we update the bias terms and scaling parameters directly using closed-form solutions by setting its gradient to 0. Taking the data matrix $\mathbf{G}_i$ as an example, the closed-form solution of its cell bias term $\mathbf{b}_{1i}$ follows:

$$\mathbf{b}_{1i} = \frac{1}{n_{\text{feats}}} \sum_{n=1}^{n_{\text{feats}}} (\mathbf{G}_i[:,n] - \alpha_{ig}\mathbf{C}_i(\Sigma + \Sigma_{ig})\mathbf{C}_g^T[:,n] - b_{gi}[n]) \tag{5}$$

where $n_{\text{feats}}$ is the total number features in $\mathbf{G}_i$. Similarly, its feature bias term $\mathbf{b}_{gi}$ follows:

$$\mathbf{b}_{gi} = \frac{1}{n_{\text{cell}}} \sum_{n=1}^{n_{\text{cell}}} (\mathbf{G}_i[n,:] - \alpha_{ig}\mathbf{C}_i(\Sigma + \Sigma_{ig})\mathbf{C}_g^T[n,:] - b_{1i}[n])^T \tag{6}$$

where $n_{\text{cell}}$ is the total number of cells in $\mathbf{G}_i$. The scaling parameter can be calculated as

$$\alpha_{ig} = \frac{tr(\mathbf{G}_i - \mathbf{b}_{1i} - \mathbf{b}_{gi}^T)^T(\mathbf{C}_i(\Sigma + \Sigma_{ig})\mathbf{C}_g^T)}{tr((\mathbf{C}_i(\Sigma + \Sigma_{ig})\mathbf{C}_g^T)^T(\mathbf{C}_i(\Sigma + \Sigma_{ig})\mathbf{C}_g^T))} \tag{7}$$

### 4.2 Calculating pseudo-count matrix to fill in the missing modalities

When the data matrices do not have a common cell batch and feature entity, there is no way to directly integrate these data matrices without knowing any relationship between different feature entities or making any assumption on the data manifold. scMoMaT uses the relationship between different feature entities to create pseudo-count matrices that can fill in the position of missed modalities for the batches during integration. When integrating scATAC-seq matrix and scRNA-seq matrix from different batches, scMoMaT creates a pseudo-scRNA-seq for each batch that has only scATAC-seq matrix. scMoMaT constructs pseudo-scRNA-seq matrix from scATAC-seq matrix by summing up the region counts from all regions that lie within the 2000 base-pair upstream from the TSS of the gene and the regions that lie within the gene body on the genome. Different from the gene activity matrix that is used in Seurat and Liger, scMoMaT further binarizes the pseudo-scRNA-seq instead of directly using it for integration. The

additional binarization step is based on two reasons: (1). the relationship between region counts and gene counts is not linear. The activations of the promoting regions of a gene have a strong correlation with the activation of the transcription process of the gene, but there is not enough evidence showing that the gene expression level is positively correlated with the number of activated promoting regions. (2). binarized gene count is shown to also have enough ability in distinguishing cell types in the data matrix[39]. A similar procedure can also be conducted between other modalities as long as a valid relationship is known between two data modalities.

### 4.3 Post-processing procedure

After training the model, we calculate a pairwise distance matrix between cells from all batches using cell factor values. We then construct a neighborhood graph from the distance matrix by connecting each cell with both its within-batch nearest neighbors and its cross-batches nearest neighbors. Denoting the overall number of nearest neighbors for each cell to be $k$ ($k = 30$ works for most cases), we assign the number of nearest neighbors of the cell in each batch according to the total number of cells within the batch. More specifically, the number of neighbors $k_i$ for batch $i$ can be calculated as

$$k_i = \frac{N_i}{N_{\text{total}}} \cdot k \tag{8}$$

where $N_i$ is the number of cells in batch $i$, and $N_{\text{total}}$ is the total number of cells in all batches.

After obtaining the neighborhood graph, we then normalize the distance value between cells in the graph. We first calculate the *mean within-batch distance* and *mean cross-batches distances* for each cell using the distance of the cells to its within-batch nearest neighbors and cross-batches nearest neighbors. Then we normalize the distances between the cell and its cross-batches nearest neighbors, which makes the *mean within-batch distance* and *mean cross-batches distances* for the cell to be the same. Considering cell $m$ and cell $n$ are nearest neighbor calculated from batch $i$ and batch $j$, the distance $d_{mn}$ between $m$ and $n$ can be normalized following:

$$\hat{d}(mn) = \frac{\bar{d}_{ii}}{\bar{d}_{ij}} d(mn) \tag{9}$$

where $\hat{d}(mn)$ is the normalized distance between cell $m$ and cell $n$, $\bar{d}_{ii}$ is the *mean within-batch distance* of cell $m$ and its neighbors in batch $i$, $\bar{d}_{ij}$ is the *mean cross-batches distance* of cell $m$ and its neighbors in batch $j$. The normalized neighborhood graph can be used for visualization and clustering purposes.

### 4.4 Visualizing and Clustering cells using post-processed graphs

Since the cell factors are summarized into a neighborhood graph after post-processing, we should visualize the cells using graph-based dimension reduction methods. In all our test results, we use UMAP for visualization, and use the neighborhood graph as the input graph of UMAP. The cells should also be clustered using graph-based clustering algorithms. In our test results, we run Leiden clustering algorithm on the neighborhood graph to obtain the cell cluster identities.

### 4.5 Retraining procedure

After clustering the cells, we use the cluster label for the retraining of scMoMaT. We first construct binary cell factor matrices from the cluster label by making each column dimension of the cell factor matrices match one specific cell cluster, and by assigning 1 to the corresponding cluster dimension and 0 to the other dimensions for each cell.

We then put binary cell factor matrix into scMoMaT and fix them while retraining scMoMaT to minimize the loss (Equ. 3). The retraining procedure is the same as the training procedure mentioned above, and the only difference is that we no longer update the cell factor matrix during the whole retraining process. This retraining process trains the feature factor matrices and association matrices to better capture the biological variation within the dataset. The retrained feature factor matrix can be used to build the marker scoring matrix that includes the marker score for each feature in each cell cluster. The top-scoring features in each cluster are considered to be the marker of the cluster. Given the retrained feature factor matrix $\mathbf{C}_{\text{feat}}$, the marker scoring matrix $\mathbf{M}_{\text{feat}}$ can be calculated as

$$\mathbf{M}_{\text{feat}} = \mathbf{C}_{\text{feat}} \cdot \Sigma^T \tag{10}$$

where $\Sigma$ is the shared association matrix, and each column of $\mathbf{M}_{\text{feat}}$ can be considered as the marker scoring of all features in the corresponding cell clusters.

During the retraining process, scMoMaT is flexible on the data matrices that are used for each data batch. One can incorporate additional data matrices that measure different data modalities of the existing data batches into the retraining process and learn the factor of the newly added data modalities through scMoMaT. In the testing result of mouse brain cortex dataset, PBMC dataset, BMMC dataset, we obtained the motif deviation matrices (cell by motif matrices) calculated from scATAC-seq matrix using ChromVAR, and included the motif deviation matrices in the retraining process to learn the motif factor of the dataset.

### 4.6 Pre-processing procedure

We preprocess the data matrices before inputting them into scMoMaT. For scRNA-seq matrices, we preprocess the matrix by first filtering the genes and selecting the highly variable genes within the matrix. The number of genes that should be kept after the filtering process is decided based on the trade-off between running time and the accuracy of the learned factors. More genes being used provide more information during the factorization, but also takes a longer time to obtain the result. $1000-2000$ genes are recommended for most of the cases. Then we quantile-log transform the gene count before sending it to scMoMaT. The transformation step can be separated into quantile normalization[40] and log-transform. The protein abundance matrix is also preprocessed through quantile-log transform, but no protein filtering step is conducted as there is only a limited number of proteins that are measured. The scATAC-seq is filtered by only selecting the regions that lie within the 2000 base-pair upstream activation region and the gene body of all genes in scRNA-seq on the genome. When dealing with multiple scATAC-seq matrices,

peak calling is conducted separately, and region features can be completely different between matrices. We then remap the fragment file of other scATAC-seq matrices using the peaks from one scATAC-seq matrix that we select. All region features are matched between batches after the remapping. The preprocessing procedure for each dataset in our test results is described as follows.

#### *4.6.1 Preprocessing steps for human PBMC dataset*

We first filter the genes and select top-7000 highly variable genes using *scanpy* for each scRNA-seq matrix separately. We do not remap the scATAC-seq matrix as the scATAC-seq matrices share the same region features. We further filter the regions in scATAC-seq and use the regions that lie within the 2000 base-pair upstream activation region and the gene body of the filtered genes on the genome and filter the genes that do not have any connected regions. We finally obtained 4768 genes, 17442 regions and 216 proteins. We quantile-log transform the scRNA-seq and protein matrix, and binarize the scATAC-seq matrix before sending them to scMoMaT.

#### *4.6.2 Preprocessing steps for mouse brain cortex dataset*

We first filter the genes and select top-2000 highly variable genes using *scanpy* for the scRNA-seq matrix in the second batch of the dataset. We then filter the genes in the scRNA-seq matrix in the first batch according to the top-2000 highly variable genes selected from the second batch. Since there are two scATAC-seq matrices that share different region features, we remap the fragment file of scATAC-seq matrix in the first batch according to the region in the third batch, and replace the old scATAC-seq matrix of the first batch with the remapped matrix. We calculate the pseudo-scRNA-seq matrix for the third batch using the method that we mentioned above. Finally, we use the overlapped genes of the first batch, the second batch, and the third batch, which gives 1709 genes in total, as the common gene for all three batches. We further filter the scATAC-seq in the first and third batches by only selecting the regions that lie within the 2000 base-pair upstream activation region and the gene body of the 1709 genes on the genome, which gives us 26125 regions. We binarize the filtered scATAC-seq matrices, and quantile-noramlize the filtered scRNA-seq matrices before sending them to scMoMaT.

We download the cell label from the original data manuscripts, reorganize the labels to make them as consistent as possible. We re-annotate the “E2Rasgrf2”, “E3Rmst” and “E3Rorb” as “L2/3”, “E4Il1rapl2”, “E4Thsd7a”, “E5Galnt14”, “E5Parm1”, “E5Sulf1”, and “E5Tshz2” as “L4/5”, “E6Tle4” as “L6”, “OliM” and “OliI” as “Oligo”, “InV” as “CGE”, “InS” as “Sst”, “InP” as “Pvalb”, “InN” as “Npy”, and “Mic” as “MGC” in the first batch. We re-annotate “Lamp5”, “Vip” and “Sncg” as “CGE”, “L4”, “L5 ET” and “L5 IT” as “L4/5”, “L6 CT”, “L6 IT” and “L6b” as “L6”, “L5/6 NP” as “NP”, “Macrophage” as “MGC” in the second batch. We re-annotate “L5.IT.a”, “L5.IT.b” and “L4” as “L4/5”, “L6.CT” and “L6.IT” as “L6”, “L23.a”, “L23.b”, and “L23.c” as “L2/3”, “OGC” as “Oligo”, “ASC” as “Astro”, and “Pv” as “Pvalb” in the third batch.

#### 4.6.3 Preprocessing steps for healthy human BMMC dataset

We first filter the genes and select top-1000 highly variable genes using *scanpy* for scRNA-seq matrix. We further filter the regions in scATAC-seq and use the regions that lie within the 2000 base-pair upstream activation region and the gene body of the filtered genes on the genome and filter the genes that do not have any connected regions. The filtering process gives us 924 genes 22133 regions. We also generate the pseudo-scRNA-seq matrix from the scATAC-seq matrix. We quantile-log transform the scRNA-seq, and binarize the scATAC-seq matrix before sending them to scMoMaT.

#### 4.6.4 Preprocessing steps for mouse spleen dataset

We first filter the genes and select top-3000 highly variable genes using *scanpy* for scRNA-seq matrix. We further filter the regions in scATAC-seq and use the regions that lie within the 2000 base-pair upstream activation region and the gene body of the filtered genes on the genome and filter the genes that do not have any connected regions. The filtering process gives us 2708 genes 20435 regions. We also generate the pseudo-scRNA-seq matrix from the scATAC-seq matrix. We quantile-log transform the scRNA-seq, and binarize the scATAC-seq matrix before sending them to scMoMaT.

### 4.7 Evaluation metrics

#### 4.7.1 Graph connectivity score

The graph connectivity (GC) score measures how well the cells of the same cell type between batches are mixed in the latent space. GC score is calculated by first constructing a knn graph using cells from all batches. Then for each cell type identity, we select the cells that belong to the cell type and extract the subgraph with only the selected cells. Denoting the subgraph as $G_c(N_c, E_c)$ where $c$ denotes the cell type identity, the GC score can be calculated as

$$\mathrm{GC} = \frac{1}{|C|} \frac{|\mathrm{LCC}(G_c)|}{|N_c|} \tag{11}$$

$|\mathrm{LCC}(G_c)|$ denotes the largest number of connected cells within the subgraph $G_c$, and $|N_c|$ denotes the total number of cells in the subgraph. The GC score uses the average of $\frac{|\mathrm{LCC}(G_c)|}{|N_c|}$ for all cell types as the final score.

#### 4.7.2 Adjusted Rand Index (ARI) score

The ARI score measures how well cells from different cell types can be correctly clustered regardless of batches using the latent embedding. After clustering the cells using the cell latent embedding obtained from different methods, we calculate the Adjusted Rand Index[41] by comparing it with the ground truth cell label. Leiden clustering algorithm has one resolution parameter that decides the number of clusters, so we ran Leiden clustering with different resolution parameters (from 0.1 to 1 with stepsize 0.5) and used the highest ARI score for all resolution parameters as the final result.

#### *4.7.3 Normalized Mutual Information (NMI) score*

Similar to ARI score, NMI score also measures how well cells from different cell types can be correctly clustered using the latent embedding. NMI is calculated with both the cluster label and ground-truth label. We obtained the cluster label using Leiden clustering algorithm, ran the clustering algorithm with different resolution parameters (from 0.1 to 1 with stepsize 0.5) and used the highest ARI score for all resolution parameters as the final result.

### 4.8 Data and code availability

The human PBMC dataset is available at Gene Expression Omnibus under accession number GSE156478. The first batch of mouse brain cortex dataset that we used can be accessed at Gene Expression Omnibus under accession number GSE126074. The second and the third batches can be accessed at NeMO Archive with accession number nemo:dat-ch1nqb7. The healthy human BMMC dataset is available at Gene Expression Omnibus under accession number GSE139369. The scATAC-seq and scRNA-seq matrix of mouse spleen dataset are available at ArrayExpress under accession numbers E-MTAB-6714 and E-MTAB-9769.

The code of scMoMaT is available at https://github.com/PeterZZQ/scMoMaT.

## 4.9 Figures and Legends

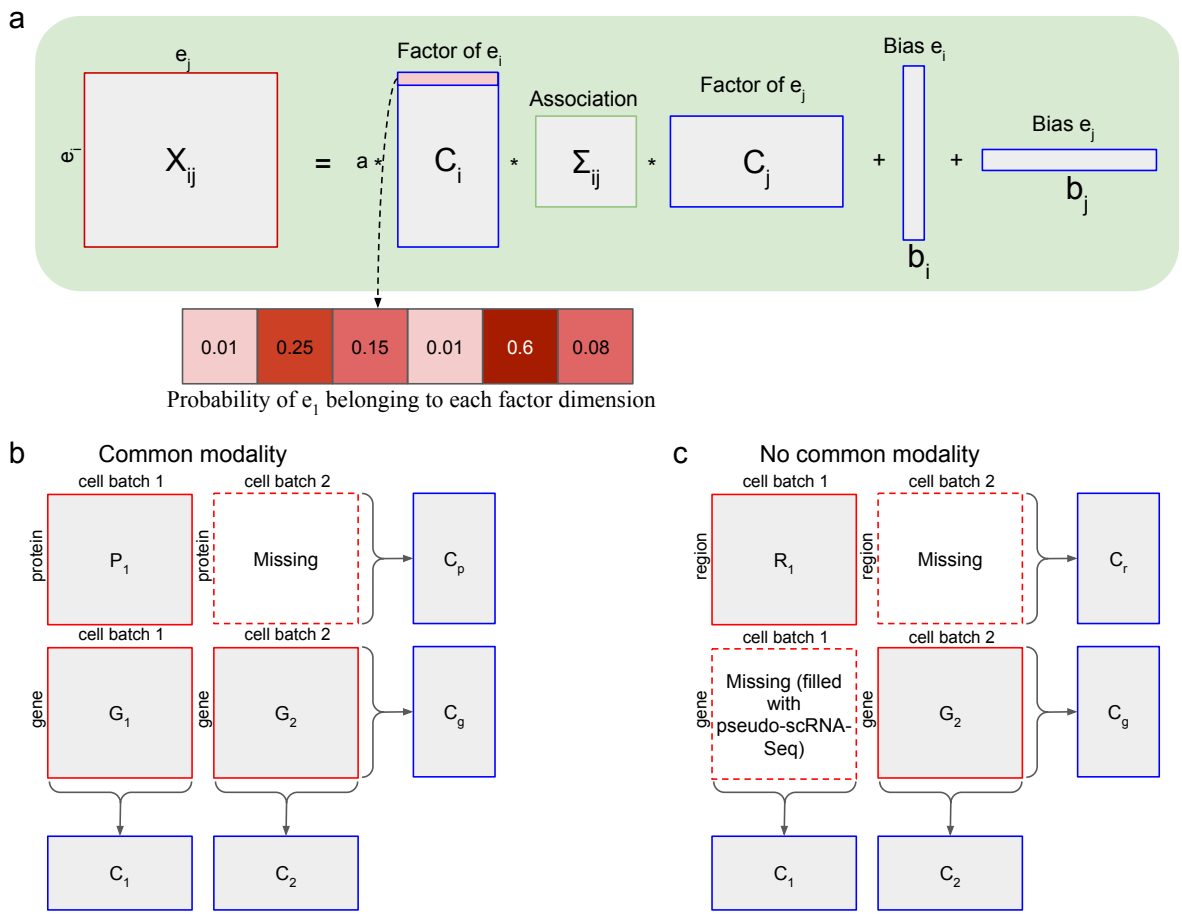



**Figure 1.** `scMoMaT` pipeline. **a**. Graph illustration showing the factorization of matrix $\mathbf{X}_{ij}$ in scMoMaT. **b**. An example integration scenario where data batches have common modality. **c**. An example integration scenario where data batches do not have common modality. scMoMaT fills in the missing modality using a pseudo-scRNA-seq matrix, and jointly factorizes the data matrix along with pseudo-scRNA-seq matrix.

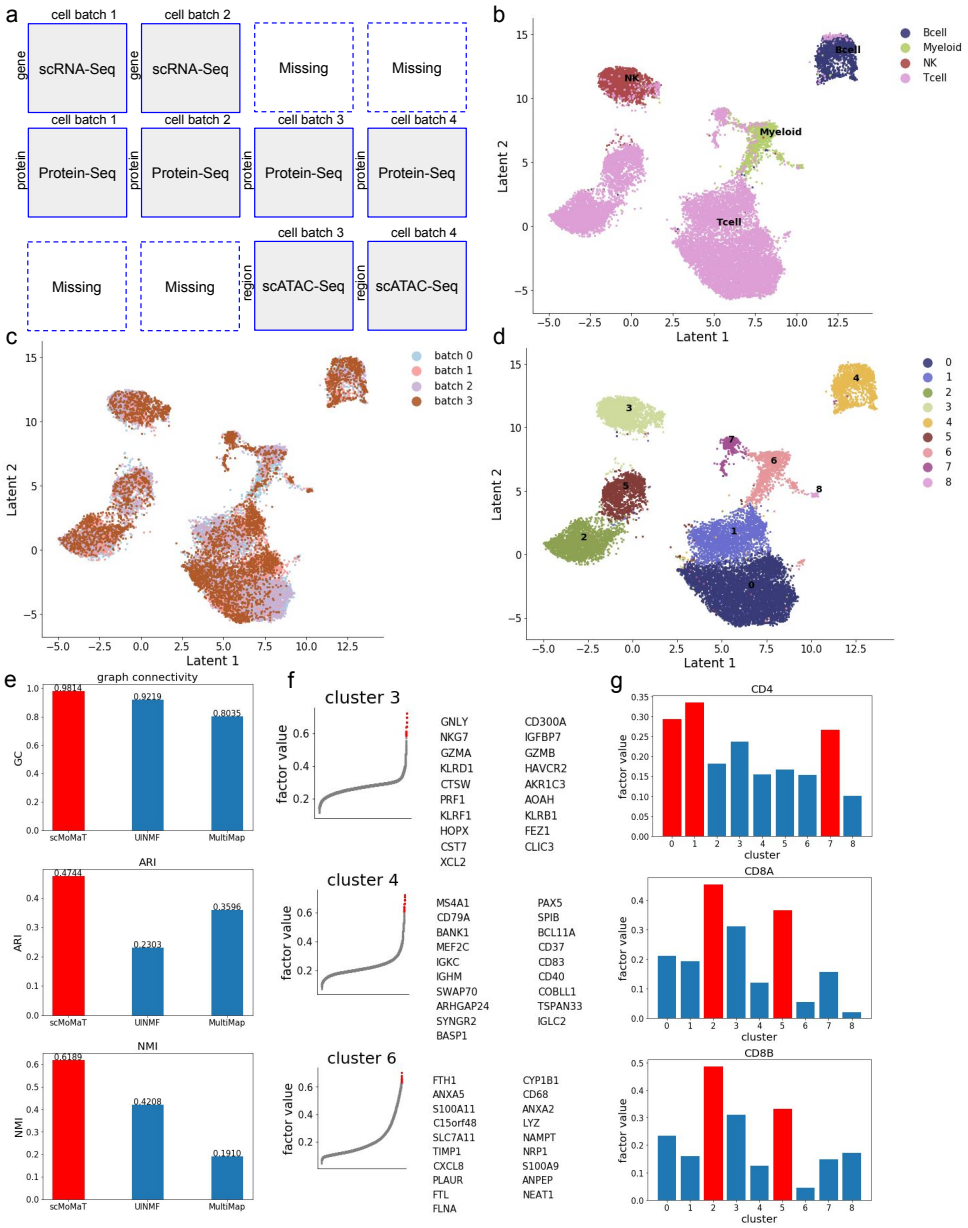



**Figure 2.** Results of scMoMaT on the human PBMC dataset. **a**. Relationship between data matrices in human PBMC dataset. **b**-**d**. The UMAP visualization of cell factor learned by scMoMaT, where cells are colored by (**b**) ground truth cell type, (**c**) data batches, and (**d**) Leiden cluster result. **e**. The graph connectivity score (GC), ARI score, and NMI score of scMoMaT along with baseline methods. Top-scoring methods are colored red. **f**. The gene factor values of cluster 3, 4, and 5. Top 20 genes are listed for each cluster. **g**. The factor value of marker genes *CD4*, *CD8A*, and *CD8B* in different clusters, where labels on x-axis correspond to Leiden cluster labels. The clusters with the highest values are colored red.

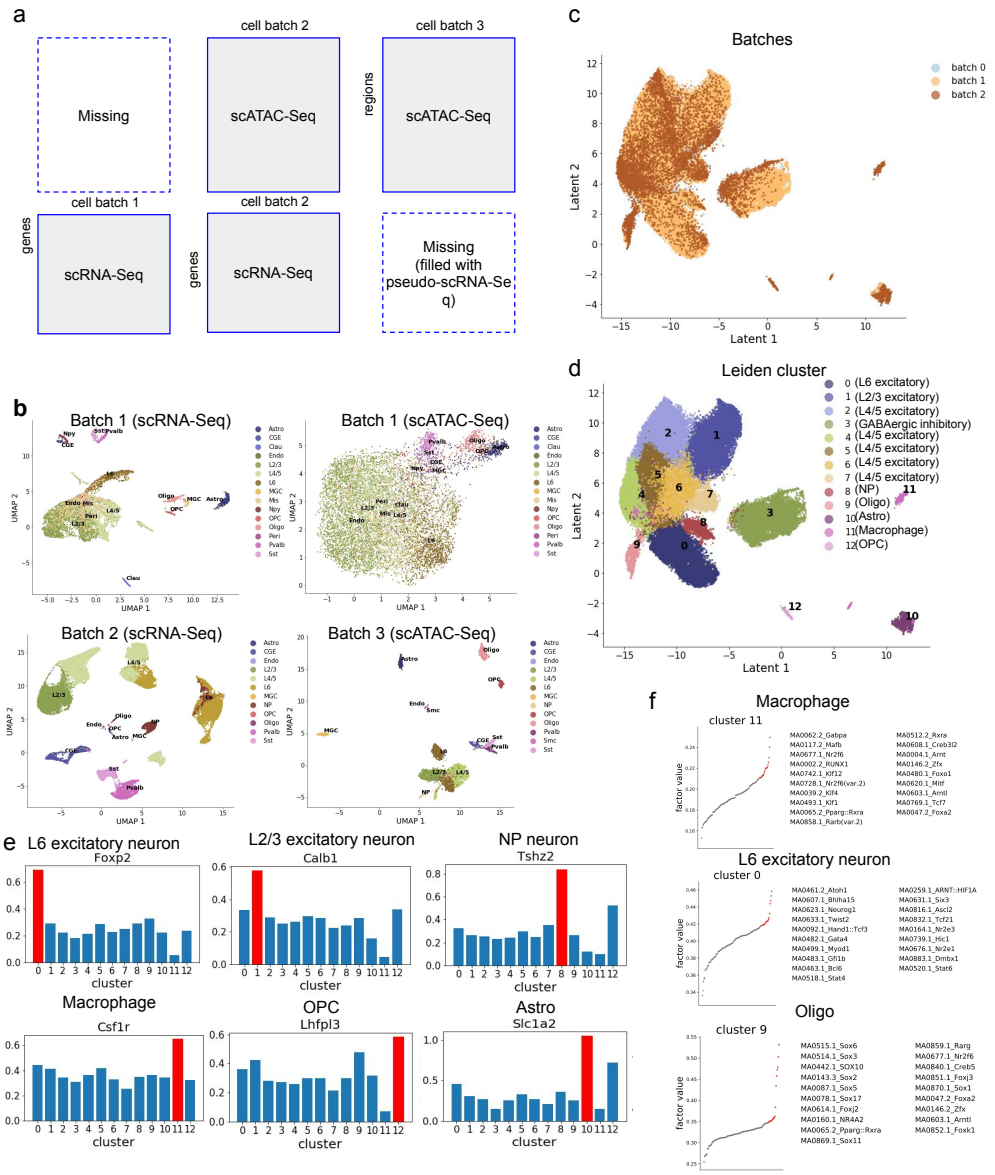
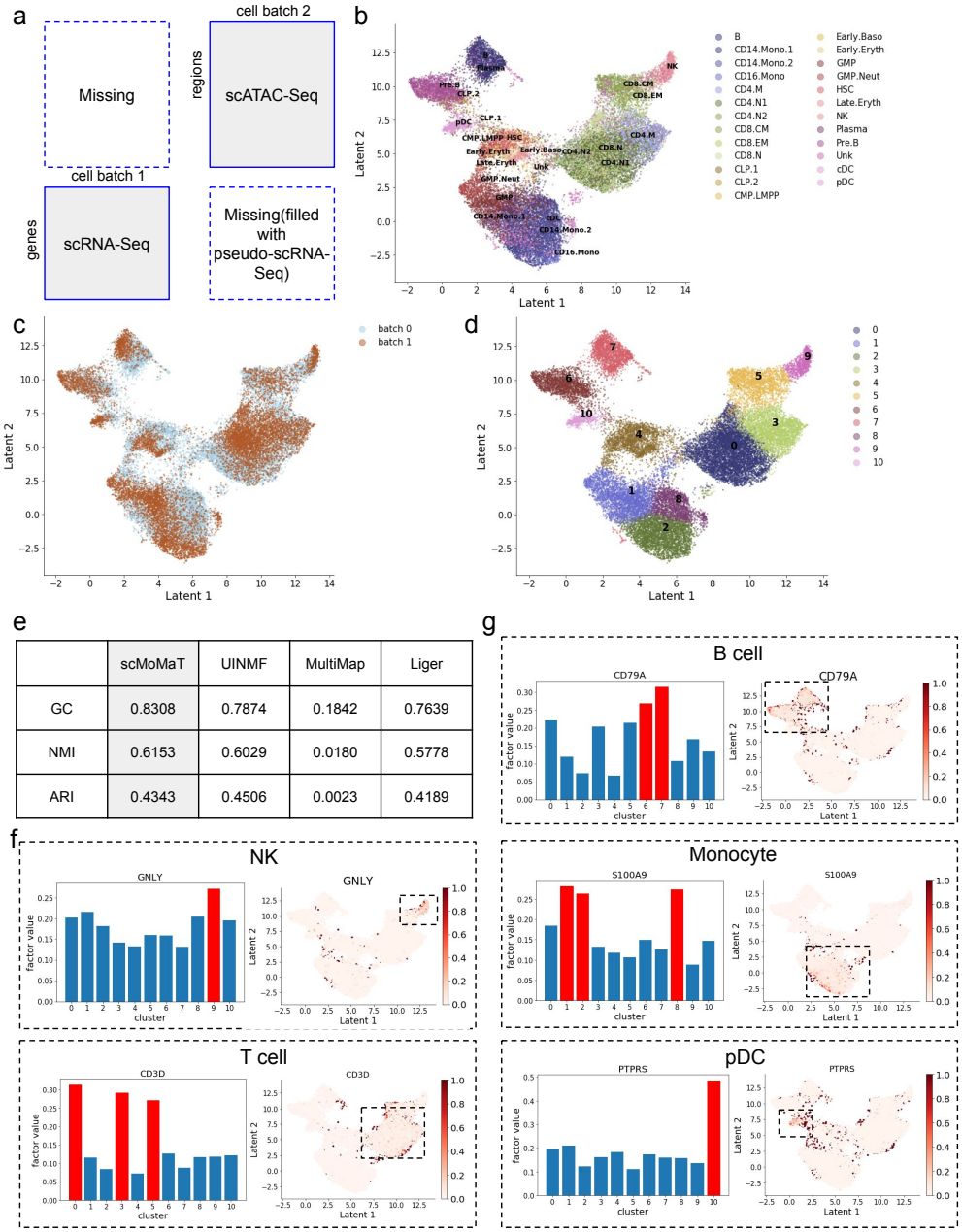
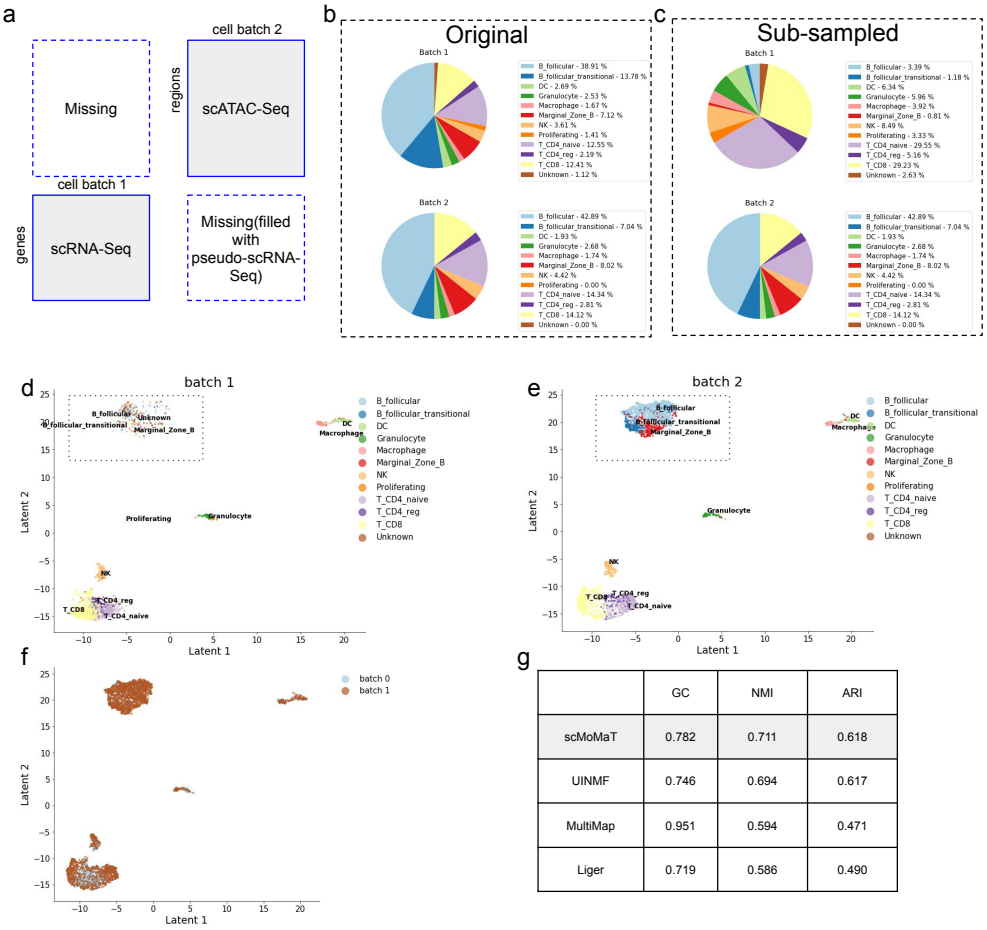



**Figure 3.** Results of scMoMaT on mouse brain cortex dataset. **a**. Relationship between data matrices in mouse brain cortex dataset. **b**. The UMAP visualization of data matrices, including the scRNA-seq matrix in batch 1 (top-left), scATAC-seq matrix in batch 1 (top-right), scRNA-seq matrix in batch 2 (bottom-left), and the scATAC-seq matrix in batch 3 (bottom-right). **c,d**. The UMAP visualization of cell factor learned by scMoMaT, where cells are colored by (**c**) data batches, and (**d**) Leiden cluster result (with scMoMaT-annotated cell types). **e**. The factor value of marker genes (for differnt cell types) in different clusters, where labels on x-axis correspond to Leiden cluster labels. The value of the marker genes in each cluster can be used to annotate cell types, and the clusters with the highest gene factor values are colored red. **f**. The motif factor value of Macrophage (cluster 11), L6 excitatory neuron (cluster 0), and Oligodendrocyte (cluster 9). Top 20 motifs are listed for each cluster.




| | scMoMaT | UINMF | MultiMap | Liger |
|---|---|---|---|---|
| GC | 0.8308 | 0.7874 | 0.1842 | 0.7639 |
| NMI | 0.6153 | 0.6029 | 0.0180 | 0.5778 |
| ARI | 0.4343 | 0.4506 | 0.0023 | 0.4189 |

**Figure 4.** Results of scMoMaT on the human bone marrow dataset. **a**. Relationship between data matrices in human bone marrow dataset. **b-d**. The UMAP visualization of cell factor learned by scMoMaT, where cells are colored by (**b**) ground truth cell type, (**c**) data batches, and (**d**) Leiden cluster result. **e**. The graph connectivity score (GC), ARI score, and NMI score of scMoMaT along with baseline methods. **f, g**. The left barplots show the factor values of (**f**) NK cell marker gene *GNLY*, and T cell marker gene *CD3D*, and (**g**) B cell marker gene *CD79A*, Monocyte marker gene *S100A9*, pDC marker gene *PTPRS* in different clusters, where labels on x-axis correspond to Leiden cluster labels. The top-scoring clusters are colored red. The right plot shows the UMAP visualization of cell factor where cells are colored by the expression level of (**f**) *GNLY*, *CD3D* and (**g**) *CD79A*, *S100A9*, *PTPRS*. The highly expressed region (in the box) is consistent with the top-scoring clusters in the barplots.




g

| | GC | NMI | ARI |
|---|---|---|---|
| scMoMaT | 0.782 | 0.711 | 0.618 |
| UINMF | 0.746 | 0.694 | 0.617 |
| MultiMap | 0.951 | 0.594 | 0.471 |
| Liger | 0.719 | 0.586 | 0.490 |

**Figure 5.** Results of scMoMaT on the mouse spleen dataset. **a**. Relationship between data matrices in mouse spleen dataset. **b, c**. The cell type composition of each batch in (**b**) original dataset and (**c**) sub-sampled dataset. **d, e**. The UMAP visualization of cell factors learned from sub-sampled dataset, where **d** shows the visualization of batch 1 and **e** shows the visualization of batch 2. The box in each plot shows the distribution of disproportionate B follicular cells in each batch. **f**. The UMAP visualization of cell factor learned from sub-sampled dataset, where cells are colored by data batches. **g**. The graph connectivity score (GC), ARI score, and NMI score of scMoMaT along with baseline methods on sub-sampled dataset.

## Supplementary Figures

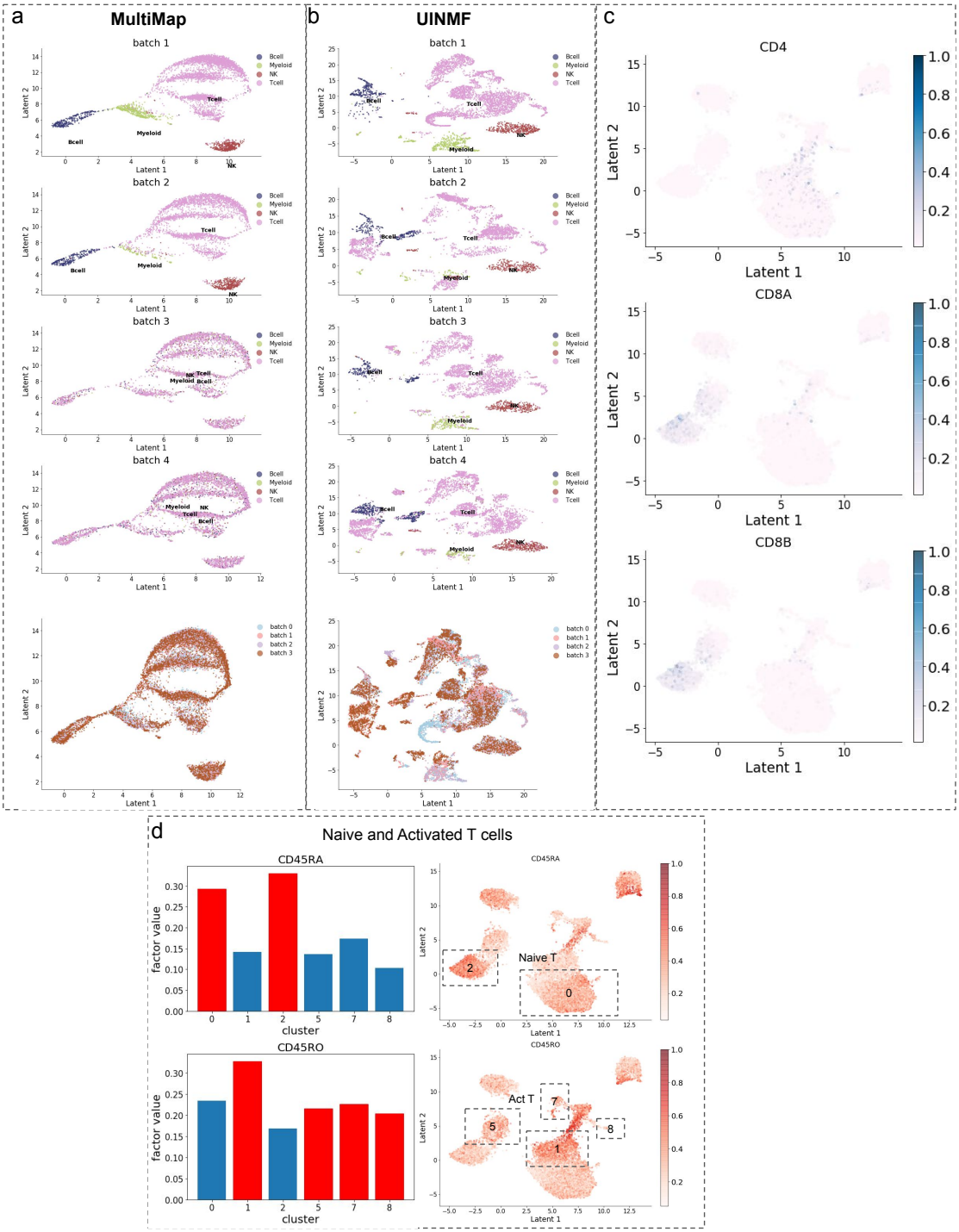
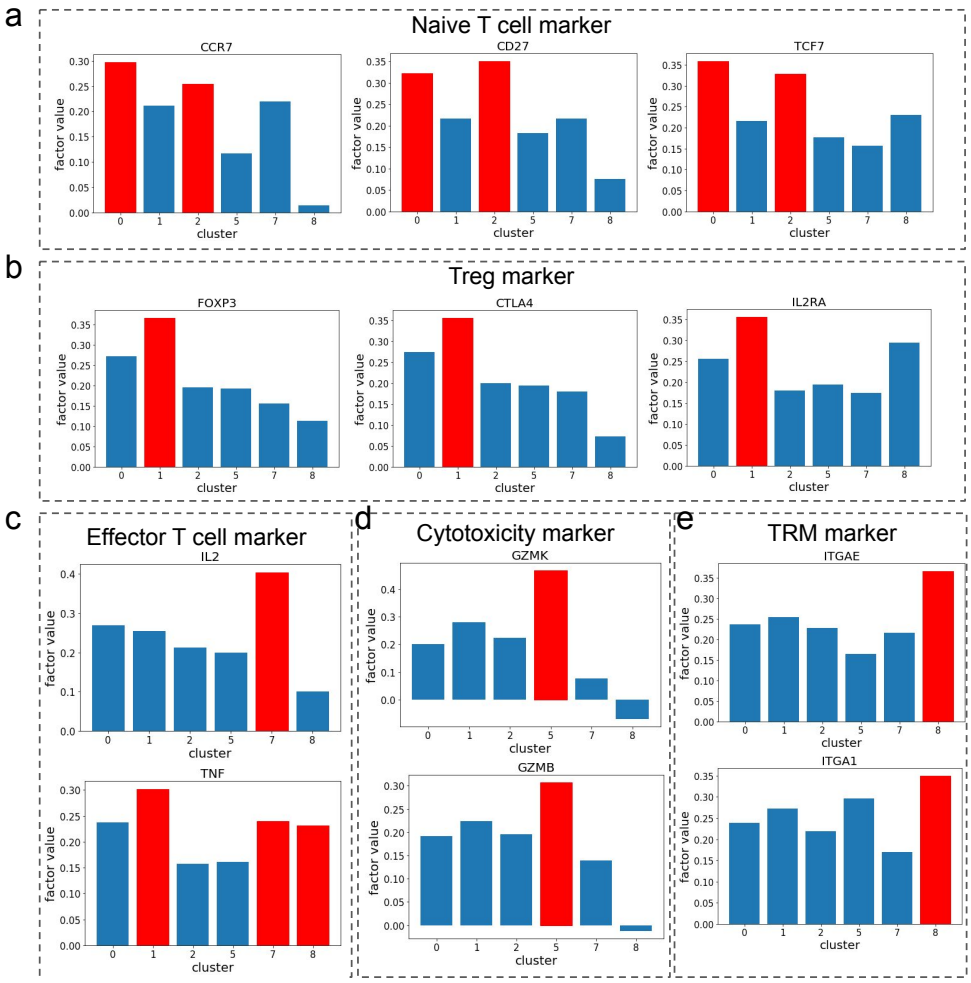



**Figure S1.** Additional test results on human PBMC dataset. **a,b**. The UMAP visualization of cell embedding learned by (**a**) MultiMap and (**b**) UINMF. The top four plots show the cell embedding of different batches, where cells are colored using the ground truth cell type. The bottom plot shows the cells that are colored by cell batches. **c**. The UMAP visualizatino of cell factor of scMoMaT, where the cells are colored by the expression level of marker genes *CD4*, *CD8A*, and *CD8B*. **d**. The left barplots show factor value of marker protein *CD45RA* and *CD45RO* in different clusters, where x-axis labels corresponds to Leiden cluster labels. The top-scoring clusters are colored red. The right plots show the abundance of *CD45RA* and *CD45RO*, where the top-scoring clusters are shown in the boxes.




f PBMC cluster annotation

| | Cluster ID |
|---|---|
| **Naive CD4+ T cell** | 0 |
| **Naive CD8+ T cell** | 2 |
| **CD4 + regulatory T cell** | 1 |
| **Activated CD4+ T cell** | 7 |
| **CD4+ memory T cell** | 8 |
| **CD8+ cytotoxicity T cell** | 5 |
| **B cell** | 4 |
| **Natural Killer (NK)** | 3 |
| **Myeloid** | 6 |

**Figure S2.** Additional test result on human PBMC dataset. **a-e**. Barplots showing the factor values of (**a**) naive T cell marker genes, (**b**) Treg marker genes, (**c**) Effector T cell marker genes, (**d**) Cytotoxicity T cell marker genes, and (**e**) TRM marker genes in different clusters, where x-axis correspond to Leiden cluster labels. Clusters with the highest factor values are colored red. **f**. Table of the cluster annotation that is learned from the feature factor in scMoMaT.

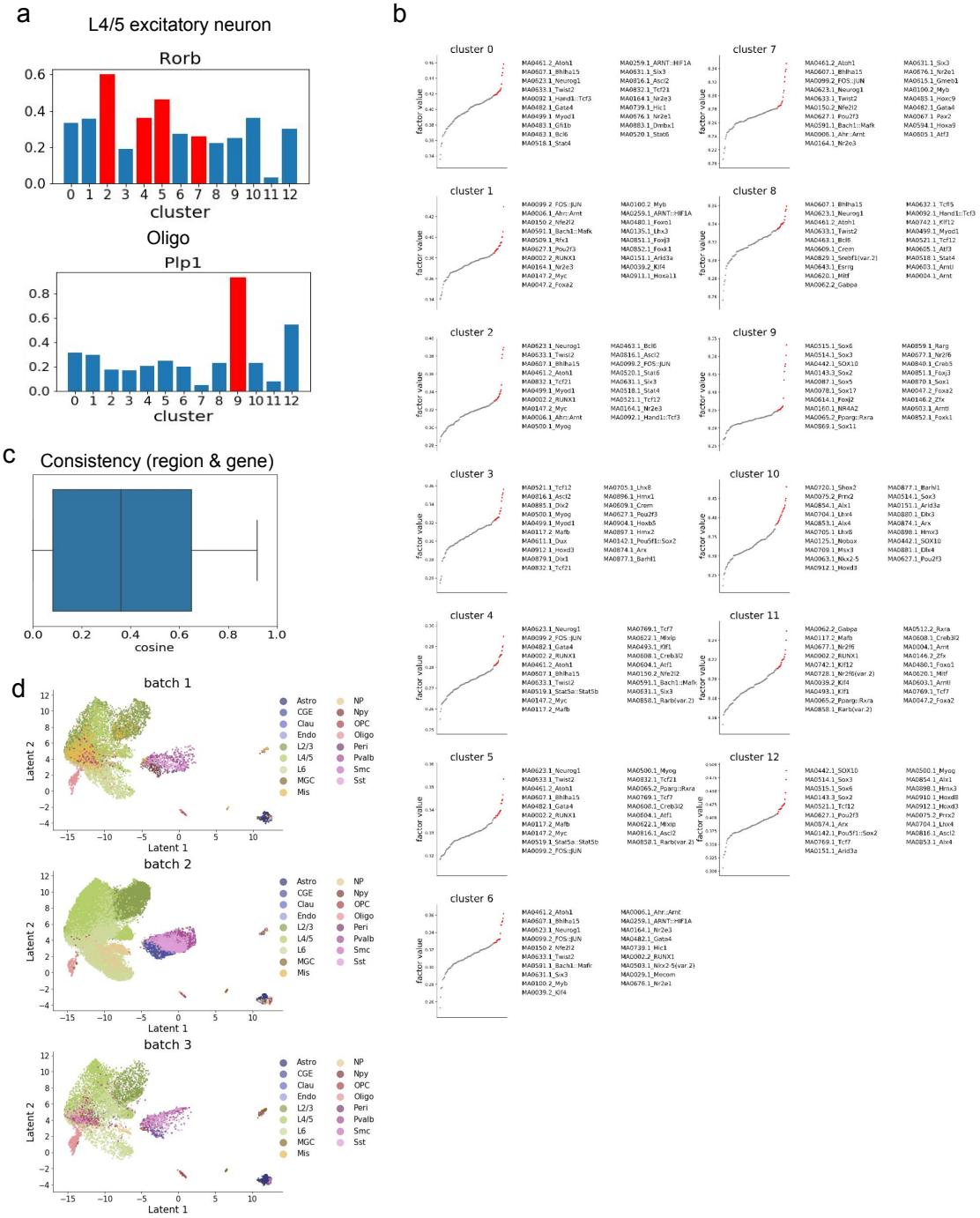



**Figure S3.** Additional test result on mouse brain cortex dataset. **a**. Barplots showing the factor values of L4/5 excitatory neuron marker gene *Rorb* (top), and Oligodendrocyte marker gene *Plp1* (bottom) in different clusters, where x-axis correspond to Leiden cluster labels. Clusters with the highest factor values are colored red. **b**. The motif factor values in different Leiden clusters. The top 20 motifs are listed for each cluster. **c**. The boxplot of consistency (cosine similarity) values between regions and genes. **d**. The UMAP visualization of cell factor learned by scMoMaT. Cell factor of different batches are shown separately. Cells are colored using the cell type annotation obtained from the original data paper.

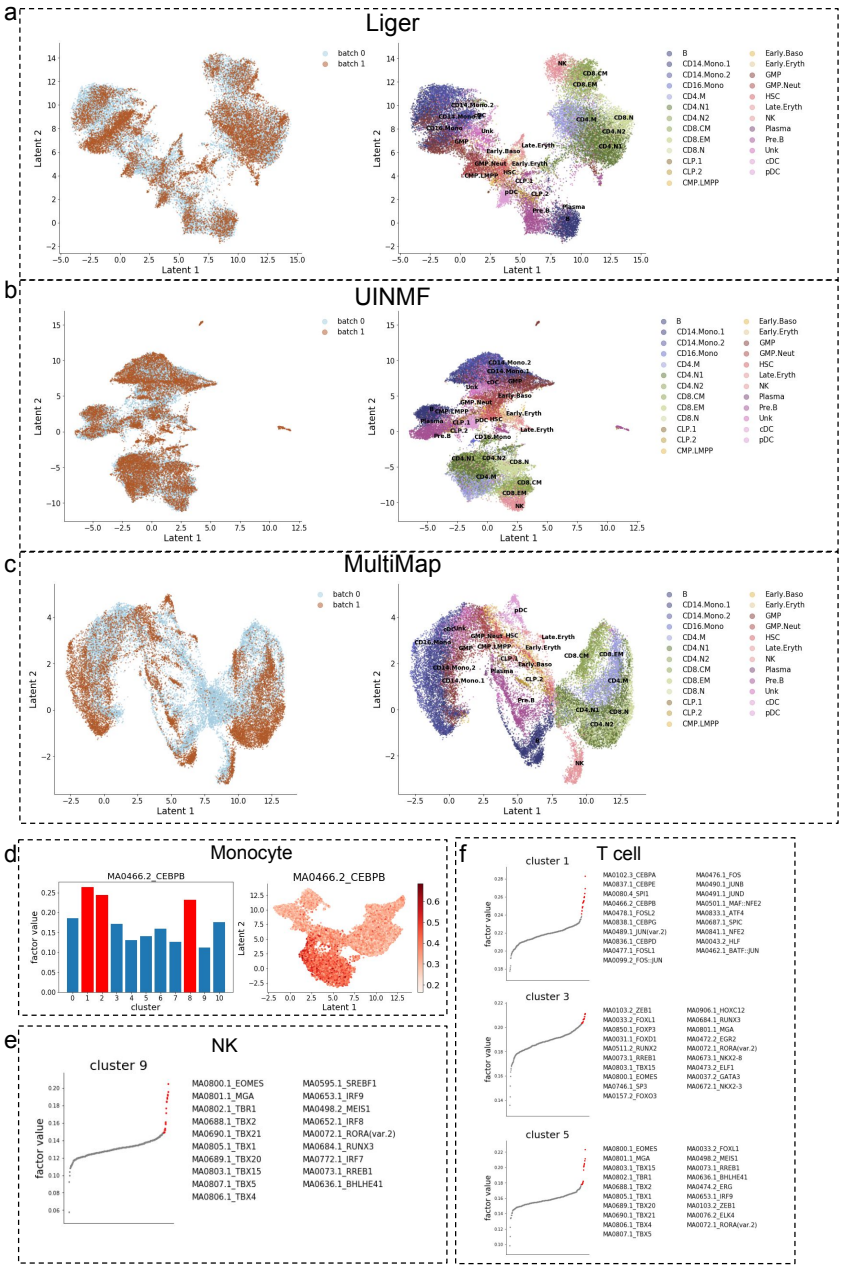



**Figure S4.** Additional test result on human bone marrow dataset. **a-c**. The UMAP visualization of cell embedding learned by baseline methods including: (**a**) Liger, (**b**) MultiMap, and (**c**) UINMF. Cells in the left are colored by ground truth cell type annotation, in the right are colored by data batches. **d**. (Left) The barplot showing the factor value of marker motif "MA0466.2_CEBPB" in different clusters, where x-axis correspond to Leiden cluster labels. Clusters with high factor value are colored red. (Right) The umap visualization of cell factor from scMoMaT, where cells are colored by the motif deviation value of "MA0466.2_CEBPB". The distribution of motif deviation value is consistent with the factor value in barplot. **e-f**. The motif factor values in NK (**e**, cluster 9) and T cells (**f**, cluster 1, 3, 5). For each cluster, the top 20 motifs with highest factor values are listed.

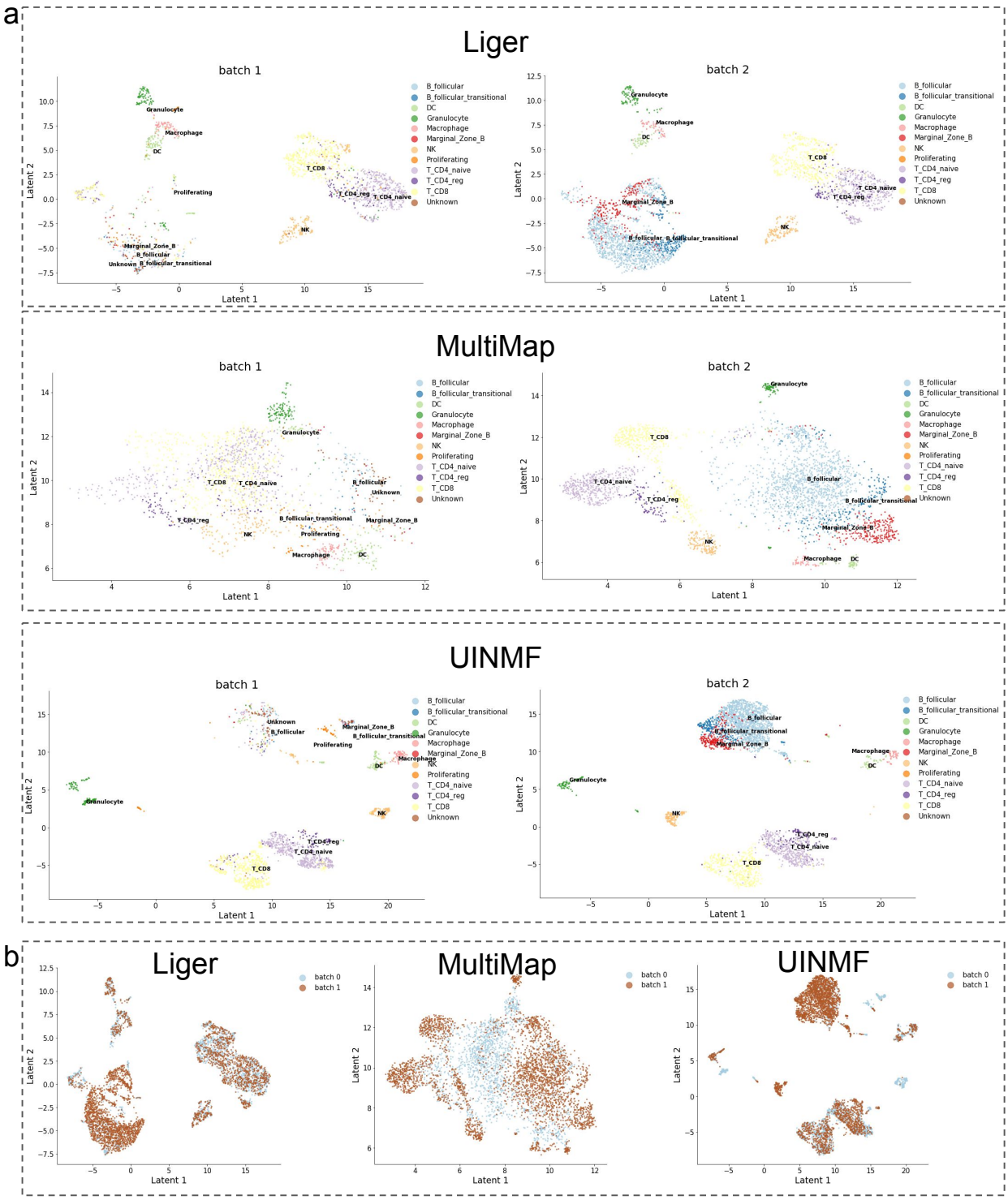



**Figure S5.** Cell embedding of mouse spleen dataset (after sub-sampling), visualized using UMAP and learned by baseline methods including: Liger, MultiMap, and UINMF. Cells are colored by (**a**) ground truth cell type annotation and (**b**) data batches.

## References


1. Stoeckius, M. *et al.* Simultaneous epitope and transcriptome measurement in single cells. *Nat. Methods* **14**, 865–868 (2017).
2. Chen, S., Lake, B. B. & Zhang, K. High-throughput sequencing of the transcriptome and chromatin accessibility in the same cell. *Nat. biotechnology* **37**, 1452–1457 (2019).
3. Mimitou, E. P. *et al.* Scalable, multimodal profiling of chromatin accessibility, gene expression and protein levels in single cells. *Nat. Biotechnol.* (2021).
4. Argelaguet, R., Cuomo, A. S. E., Stegle, O. & Marioni, J. C. Computational principles and challenges in single-cell data integration. *Nat. Biotechnol.* 1–14 (2021).
5. Welch, J. D. *et al.* Single-Cell multi-omic integration compares and contrasts features of brain cell identity. *Cell* **177**, 1873–1887.e17 (2019).
6. Stuart, T. *et al.* Comprehensive integration of Single-Cell data. *Cell* **177**, 1888–1902.e21 (2019).
7. Duren, Z. *et al.* Integrative analysis of single-cell genomics data by coupled nonnegative matrix factorizations. *Proc. Natl. Acad. Sci. U. S. A.* **115**, 7723–7728 (2018).
8. Cao, K., Bai, X., Hong, Y. & Wan, L. Unsupervised topological alignment for single-cell multi-omics integration. *Bioinformatics* **36**, i48–i56 (2020).
9. Singh, R. *et al.* Unsupervised manifold alignment for single-cell multi-omics data. In *Proceedings of the 11th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics*, 1–10 (Association for Computing Machinery, New York, NY, USA, 2020).
10. Zhang, Z., Yang, C. & Zhang, X. Learning latent embedding of multi-modal single cell data and cross-modality relationship simultaneously. *bioRxiv* (2021).
11. Hao, Y. *et al.* Integrated analysis of multimodal single-cell data. *Cell* (2021).
12. Jin, S., Zhang, L. & Nie, Q. scAI: an unsupervised approach for the integrative analysis of parallel single-cell transcriptomic and epigenomic profiles. *Genome Biol.* **21**, 25 (2020).
13. Ashuach, T., Gabitto, M. I., Jordan, M. I. & Yosef, N. Multivi: deep generative model for the integration of multi-modal data. *bioRxiv* (2021).
14. Jain, M. S. *et al.* MultiMAP: dimensionality reduction and integration of multimodal data. *Genome Biol.* **22**, 346 (2021).
15. Hao, Y. *et al.* Dictionary learning for integrative, multimodal, and scalable single-cell analysis. *bioRxiv* (2022).

**16.** Kriebel, A. R. & Welch, J. D. Uinmf performs mosaic integration of single-cell multi-omic datasets using nonnegative matrix factorization. *Nat. communications* **13**, 1–17 (2022).

**17.** Lee, D. D. & Seung, H. S. Learning the parts of objects by non-negative matrix factorization. *Nature* **401**, 788–791 (1999).

**18.** Traag, V. A., Waltman, L. & Van Eck, N. J. From louvain to leiden: guaranteeing well-connected communities. *Sci. reports* **9**, 1–12 (2019).

**19.** Stoeckius, M. *et al.* Simultaneous epitope and transcriptome measurement in single cells. *Nat. methods* **14**, 865–868 (2017).

**20.** Mimitou, E. P. *et al.* Multiplexed detection of proteins, transcriptomes, clonotypes and crispr perturbations in single cells. *Nat. methods* **16**, 409–412 (2019).

**21.** Jain, M. S. *et al.* Multimap: dimensionality reduction and integration of multimodal data. *Genome biology* **22**, 1–26 (2021).

**22.** Yang, C. *et al.* Heterogeneity of human bone marrow and blood natural killer cells defined by single-cell transcriptome. *Nat. communications* **10**, 1–16 (2019).

**23.** Stelzer, G. *et al.* The GeneCards suite: From gene data mining to disease genome sequence analyses. *Curr. Protoc. Bioinforma.* **54**, 1.30.1–1.30.33 (2016).

**24.** Xu-Monette, Z. Y. *et al.* Assessment of cd37 b-cell antigen and cell of origin significantly improves risk prediction in diffuse large b-cell lymphoma. *Blood, The J. Am. Soc. Hematol.* **128**, 3083–3100 (2016).

**25.** Johannisson, A. & Festin, R. Phenotype transition of cd4+ t cells from cd45ra to cd45ro is accompanied by cell activation and proliferation. *Cytom. The J. Int. Soc. for Anal. Cytol.* **19**, 343–352 (1995).

**26.** Caccamo, N., Joosten, S. A., Ottenhoff, T. H. & Dieli, F. Atypical human effector/memory cd4+ t cells with a naive-like phenotype. *Front. Immunol.* 2832 (2018).

**27.** Szabo, P. A. *et al.* Single-cell transcriptomics of human t cells reveals tissue and activation signatures in health and disease. *Nat. communications* **10**, 1–16 (2019).

**28.** Yao, Z. *et al.* A transcriptomic and epigenomic cell atlas of the mouse primary motor cortex. *Nature* **598**, 103–110 (2021).

**29.** Traag, V. A., Waltman, L. & van Eck, N. J. From louvain to leiden: guaranteeing well-connected communities. *Sci. Rep.* **9**, 5233 (2019).

**30.** Bandler, R. C. *et al.* Single-cell delineation of lineage and genetic identity in the mouse brain. *Nature* **601**, 404–409 (2022).

**31.** Schep, A. N., Wu, B., Buenrostro, J. D. & Greenleaf, W. J. chromvar: inferring transcription-factor-associated accessibility from single-cell epigenomic data. *Nat. methods* **14**, 975–978 (2017).

**32.** Li, Y. E. *et al.* An atlas of gene regulatory elements in adult mouse cerebrum. *Nature* **598**, 129–136 (2021).

**33.** Granja, J. M. *et al.* Single-cell multiomic analysis identifies regulatory programs in mixed-phenotype acute leukemia. *Nat. biotechnology* **37**, 1458–1465 (2019).

**34.** Zhao, F. *et al.* S100a9 a new marker for monocytic human myeloid-derived suppressor cells. *Immunology* **136**, 176–183 (2012).

**35.** Bunin, A. *et al.* Protein tyrosine phosphatase ptprs is an inhibitory receptor on human and murine plasmacytoid dendritic cells. *Immunity* **43**, 277–288 (2015).

**36.** Marchwicka, A. & Marcinkowska, E. Regulation of expression of cebp genes by variably expressed vitamin d receptor and retinoic acid receptor $\alpha$ in human acute myeloid leukemia cell lines. *Int. J. Mol. Sci.* **19**, 1918 (2018).

**37.** Kiekens, L. *et al.* T-bet and eomes accelerate and enhance functional differentiation of human natural killer cells. *Front. immunology* **12** (2021).

**38.** Chen, X., Miragaia, R. J., Natarajan, K. N. & Teichmann, S. A. A rapid and robust method for single cell chromatin accessibility profiling. *Nat. communications* **9**, 1–9 (2018).

**39.** Qiu, P. Embracing the dropouts in single-cell rna-seq analysis. *Nat. communications* **11**, 1–9 (2020).

**40.** Bolstad, B. M., Irizarry, R. A., Åstrand, M. & Speed, T. P. A comparison of normalization methods for high density oligonucleotide array data based on variance and bias. *Bioinformatics* **19**, 185–193 (2003).

**41.** Steinley, D. Properties of the hubert-arable adjusted rand index. *Psychol. methods* **9**, 386 (2004).