SIMBA: SIngle-cell eMBedding Along with features

Huidong Chen[1, 2, 3], Jayoung Ryu[1, 2, 4], Michael Vinyard[1, 2, 3, 5], Adam Lerer[6]+, Luca Pinello[1, 2, 3]+

1. Molecular Pathology Unit, Center for Cancer Research, Massachusetts General Hospital, Charlestown, MA, USA
2. Department of Pathology, Harvard Medical School, Boston, MA, USA
3. Broad Institute of Harvard and MIT, Cambridge, MA, USA
4. Department of Biomedical Informatics, Harvard Medical School, Boston, MA, USA
5. Department of Chemistry and Chemical Biology, Harvard University, Cambridge, MA, USA
6. Facebook AI Research

+Corresponding author

Abstract

Recent advances in single cell omics technologies enable the individual or joint profiling of cellular measurements including gene expression, epigenetic features, chromatin structure and DNA sequences. Currently, most single-cell analysis pipelines are cluster-centric, i.e., they first cluster cells into non-overlapping cellular states and then extract their defining genomic features. These approaches assume that discrete clusters correspond to biologically relevant subpopulations and do not explicitly model the interactions between different feature types. However, cellular processes are defined in individual cells and inherently involve multiple genomic features that interact with each other and together provide complementary views on principles of gene regulation. In addition, single-cell methods are generally designed for a particular task as distinct single-cell problems are formulated differently. To address these current shortcomings, we present *SIMBA*, a single-cell embedding method that embeds single cells along with their defining features, such as genes, chromatin accessible regions, and transcription factor binding sequences, into a common latent space. By leveraging the co-embedding of cells and features, SIMBA allows for cellular heterogeneity study, clustering-free marker discovery, gene regulation inference, batch effect removal, and omics data integration. SIMBA has been extensively applied to scRNA-seq, scATAC-seq, and dual-omics data. We show that SIMBA provides a single framework that allows diverse single-cell analysis problems to be formulated in a common way and thus simplifies the development of new analyses and integration of other single-cell modalities.

Introduction

Recent progress in single cell molecular profiling technologies have dramatically advanced our ability to define cell types and states as well as discover key genes and regulatory regions in development and disease. Both the number of cells and the number of cellular modalities that can be profiled has recently expanded rapidly. The emergence of single-cell multi-omics

technologies allows for the measurements of multiple cellular layers, including genomics, epigenomics, transcriptomics, and proteomics. This has opened an avenue to better understand the interplay between these 'omic' layers and cell states based on diverse genomic and molecular features including genes, regulatory elements, transcription factors, and other cellular components. However, as single-cell multi-omics assays quickly evolve towards the incorporation of more modalities and increasing resolution, harnessing their full potential poses significant computational challenges.

In the past few years, numerous computational methods have been developed for single-cell single-modality analysis (e.g., scRNA-seq or scATAC-seq analysis) [1-4]. These methods implement a common workflow with several standard steps including feature selection, dimension reduction, clustering, and differential feature detection. Cluster-centric analysis methods rely on accurately defined clustering solutions to discover meaningful and informative marker features. Unfortunately, clustering solutions may range widely within the space of the user-defined clustering resolution (number of clusters) and the chosen clustering algorithm. These parameters may markedly influence the resulting cluster assignment and clusters may not always correspond to the correct cell populations, thereby leading to inconsistent and potentially misleading biological annotations. Although initial efforts have been made recently to develop clustering-free approaches, they are specifically designed for extracting gene signatures [5, 6] or identifying perturbations between experimental conditions[7] from scRNA-seq data, and are therefore limited to single-modality and single-task analysis.

In addition to single-batch/modality analysis, approaches have also been proposed for multi-batch and cross-modality analysis, such as multimodal analysis (distinct cellular parameters are measured in the same cell)[8], batch correction (the same cellular parameter is measure in different batches) [9-11], and integration of multi-omics datasets (distinct cellular parameters are measured in different cells)[10, 11]. These approaches play a critical role in removing batch effects that confound true biological variation, improving the characterization of cell states by leveraging the unique strengths of each assay, and providing insights into the complex mechanisms of gene regulation.

However, these problems are formulated differently from those in single-batch/modality settings and thus require development of new dedicated analysis techniques. Also, while multiple types of cellular parameters (features) might be present, the relation between features cannot be exploited directly by most current methods. Furthermore, similar to single-batch/modality analysis methods, these methods identify marker features based on groups of cells obtained by clustering and therefore are limited to clustering solutions.

To overcome these limitations, we propose SIMBA (**SIngle-cell eMBedding Along with features**), a versatile single-cell embedding method that co-embeds cells and features into a shared latent space, in which the relation between cells and features or between features (e.g., genes, peaks, or DNA sequences) can be assessed based on their locations. By formulating single-cell analyses as multi-entity graph embedding problems, SIMBA can be used to solve popular single-cell tasks in a single, unified framework that would otherwise require the

development of distinct specialized approaches for each task. For each task, SIMBA constructs a graph, wherein differing entities (i.e., cells and features) are represented as nodes of the graph and relations between these entities are encoded as edges of the graph. Once the graph is constructed, SIMBA then applies a multi-entity graph embedding algorithm leveraged from advances in social networking technologies and knowledge graph embeddings as well as a Softmax-based transformation to embed the nodes/entities of the graph into a common low-dimensional space wherein cells and features can be comparatively analyzed. We show that the SIMBA framework can perform many common and important single-cell analysis tasks, including dimensionality reduction techniques for studying cellular states; clustering-free marker detection based on the similarity between single cells and features; single-cell multimodal analysis and the study of gene regulation; batch correction and omics integration analysis as well as the simultaneous identification of marker features. SIMBA is adapted to these diverse analysis tasks by simply modifying how the input graph is constructed from the relevant single-cell data. We believe that SIMBA will simplify the burden of adapting existing single-cell analyses to new tasks and measurements.

We extensively tested *SIMBA* in multiple scRNA-seq, scATAC-seq and dual-omics datasets covering the popular single-cell tasks including scRNA-seq analysis, scATAC-seq analysis, multimodal analysis, batch correction, and multi-omics integration. We demonstrate that *SIMBA* performs comparably to or better than current state-of-the-art methods specifically developed for each task.

Results

Overview of SIMBA

SIMBA is a single-cell embedding method with support for single- or multi- modality analyses that embeds cells and their associated genomic features into a shared latent space, generating interpretable and comparable embeddings of cells and features. It leverages recent graph embedding techniques that have been successful in modeling complex and hierarchical information present in natural languages, social networks, and other domains, represented as "knowledge graphs". In our case, these graphs encode different components of cellular regulatory circuits and the relations between them.

SIMBA first encodes different types of entities such as cells, genes, open chromatin regions (peaks or bins), transcription factor (TF) motifs, and *k*-mers, into a single graph (Fig. 1, Methods) where nodes represent different entities and edges indicate the relations between them. For example, if a gene is expressed in a cell, an edge is created between the gene and cell. The weight of this edge is determined by the gene expression level. Similarly, an edge is added between a cell and a chromatin region if the region is open in this cell, or between a chromatin region and a TF motif if the TF motif is found in the region.

Once the graph is constructed, the low-dimensional representations of its nodes are then computed using an unsupervised graph embedding method (**Methods**). This graph embedding

procedure leverages the PyTorch-BigGraph framework [12], which allows SIMBA to scale to millions of cells (**Methods**). The resulting joint embedding of cells and features not only reconstructs the heterogeneity of cells but also allows for the discovery of the defining features for each single cell in a clustering-free way, separating cell-type specific features from the non-informative features. In fact, the proximity between the embeddings of entities is informative on the potential importance of a feature to a cell and the discovery of the interplay between features. When multiple types of features (e.g., transcriptomic and epigenetic features) are co-embedded, SIMBA provides an intuitive way to study gene regulation and the regulatory mechanisms underlying cellular differentiation and cell type formation.

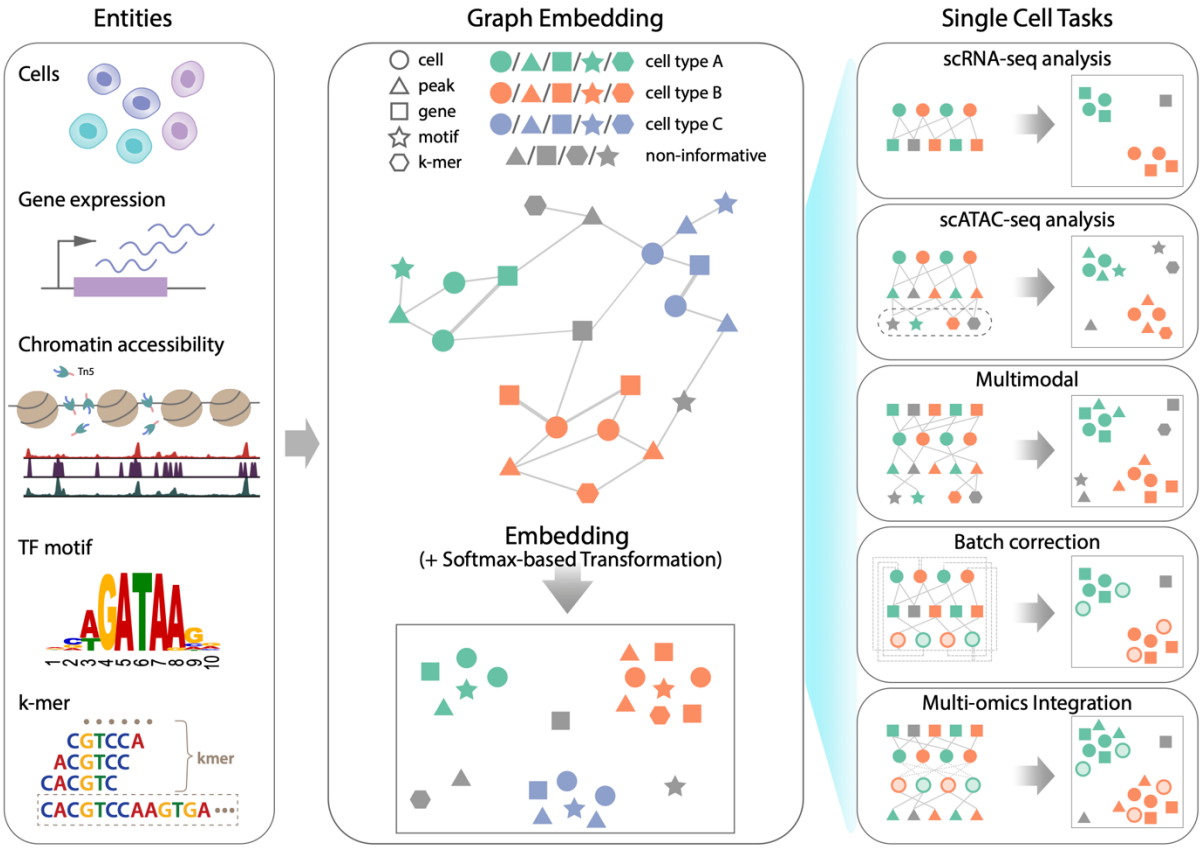


Figure1. SIMBA framework overview. SIMBA co-embeds cells and various features measured during single-cell experiments into a shared latent space to accomplish both common tasks involved in single-cell data analysis as well as tasks, which remain as open problems in single-cell genomics. **(Left)** Examples of possible biological entities may be encoded by SIMBA including cells, gene expression measurements, chromatin accessible regions, TF motifs, and k-mer sequences found in reads. **(Middle)** SIMBA embedding plot with multiple types of entities into a low-dimensional space. All entities represented as shapes (cell = circle, peak = triangle, gene = square, TF motif = star, k-mer = hexagon) are colored by relevant cell type (green, orange, and blue in this example). Non-informative features are colored dark grey. Within the graph, each entity is a node, and an edge indicates a relation between entities (e.g., a gene is expressed in a cell, a chromatin region is accessible in a cell, or a TF motif/k-mer is present within an open

chromatin region, etc.). Once connected in a graph, these entities may be embedded into a shared low-dimensional space, with cell-type specific entities embedded in the same neighborhood and non-informative features embedded elsewhere. **(Right)** Common single-cell analysis tasks that may be accomplished using SIMBA.

Graph construction is inherently flexible, enabling SIMBA to be applied to a wide variety of single-cell tasks. In the following sections, we demonstrate the application of SIMBA to several popular single-cell tasks including scRNA-seq, scATAC-seq, multimodal analysis, batch correction and multi-omics integration **(Fig. 1).**

Single cell RNA-seq analysis with SIMBA

Single-cell RNA sequencing (scRNA-seq) is the most robust and widely used measurement to profile single cells. **Fig. 2a** provides an illustrative overview of the SIMBA graph construction and the resulting low-dimensional embedding matrix of both cells and genes. To clearly demonstrate SIMBA's ability to perform scRNA-seq analysis, we applied SIMBA to a popular PBMCs dataset from 10x Genomics, which is used in the tutorials of both *Scanpy* [2]and *Seurat*[1]. After the standard preprocessing steps including normalization and log-transformation, SIMBA discretizes the gene expression matrix into multiple gene expression levels (five levels, by default). The input graph is then constructed wherein two types of nodes –cells and genes are connected by edges that embody the relation between them and are weighted according to the corresponding multiple levels of gene expression. SIMBA then generates embeddings of these nodes through a graph embedding procedure (**Fig. 2a**, **Methods**).

We first visualized the SIMBA embeddings of cells using UMAP, which is a comparable output to other current single-cell analysis methods. Each of the previously assigned eight cell types, including B cells, megakaryocytes, CD14 monocytes, FCGR3A monocytes, dendritic cells, NK cells, CD4 T, and CD8 T cells, was clearly separated (**Fig. 2b**). We next applied UMAP to visualize the SIMBA embeddings of cells and genes together (**Fig. 2c**). The same set of marker genes used to annotate these cells from *Scanpy* was highlighted on the UMAP plot. In addition, as a control, we also show the locations of two housekeeping genes GAPDH and B2M, which would not be expected to associate with any particular cell type. From the UMAP plot, we can see that SIMBA not only was able to embed major-cell-group specific genes to the correct locations (e.g., IL7R was embedded into CD4T cells and MS4A1 was embedded into B cells), but also was robust to rare-cell-group specific genes (e.g., PPBP was embedded into megakaryocytes). On the contrary, non-informative or non-cell-type specific genes such as the aforementioned housekeeping genes were embedded in the middle of all cell groups.

In addition to visualizing all the entities at once using UMAP, SIMBA also provides a feature-specific 'barcode plot' to visualize the estimated probability of assigning a feature to a cell (**Fig. 2d, Methods**). The barcode plots in **Figure 2d** offer a rank-ordered probability of a given gene being associated with each cell (colored by cell type) where the total probability over all cells

sums to one. An imbalance in probability indicates cell-type-specific association of a gene to a sub-population of cells, whereas a uniform probability distribution indicates a non-cell-type-specific gene. **Figure 2d** displays barcode plots for four genes, which are correspondingly highlighted in red, in **Figure 2c**. Three of these genes are commonly used marker genes for identifying subpopulations within PBMCs datasets (CST3 for monocytes and dendritic cells, MS4A1 for B cells, and NGK7 for NK and CD8T cells). On the contrary, GAPDH is a housekeeping gene expressed in all cell types. With the same threshold, 1e-3 (represented by the dashed line) for marker genes, we observed a clear excess in the probability of assigning each gene to their respective cell types. Conversely, for GAPDH, we observed a more balanced distribution and the probability of associating that gene with any particular subset of cells is much lower than for marker genes. Hence, SIMBA barcode plots serve as an informative way of visualizing gene expression patterns by showing the cell assignment probability distribution.

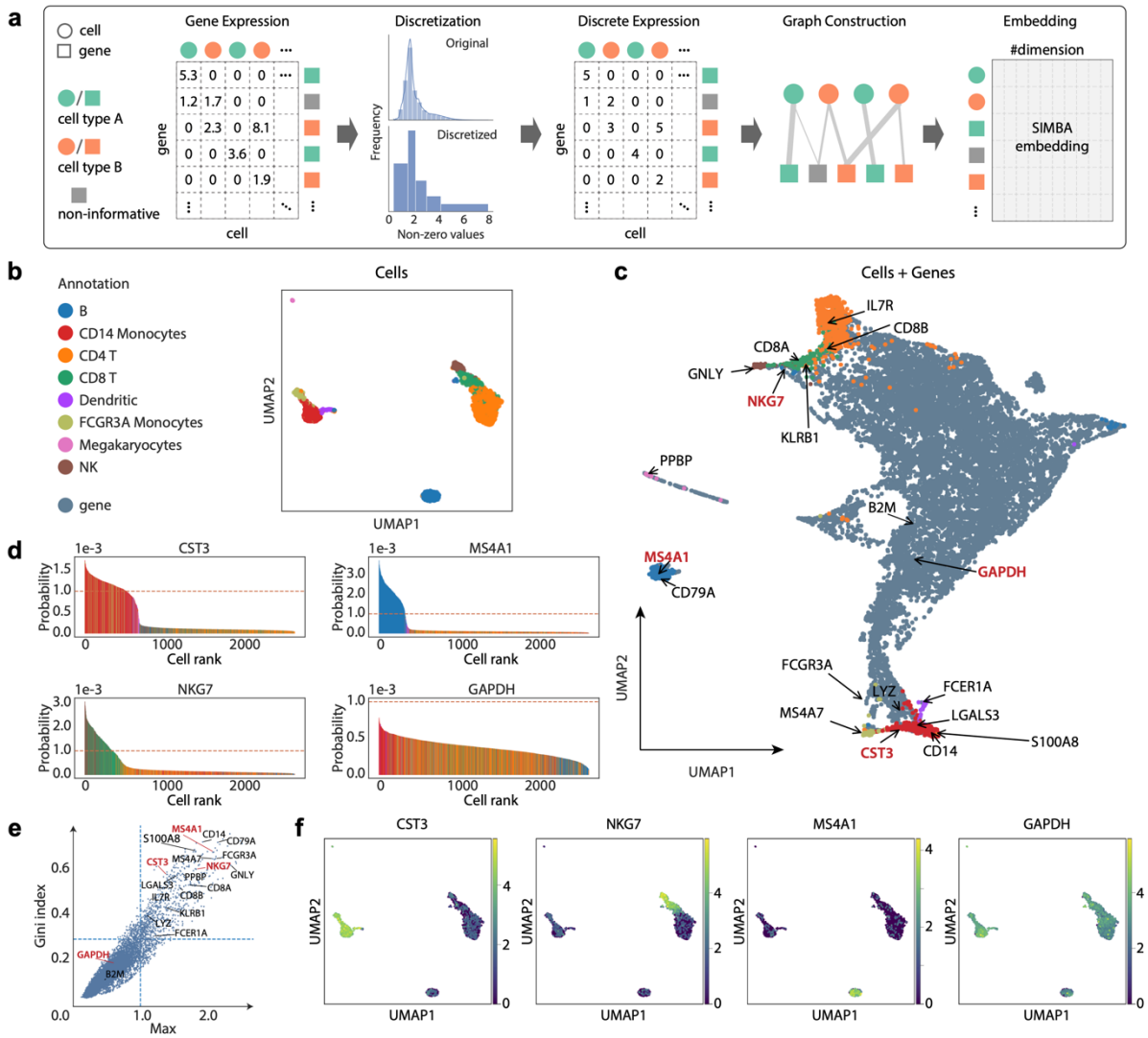


Figure2. ScRNA-seq analysis of the 10x PBMCs dataset using SIMBA. **(a)** SIMBA graph construction and embedding in scRNA-seq analysis. Biological entities including cells and genes are

represented as shapes and colored by relevant cell types (green and orange). Non-informative genes are colored dark grey. Gene expression measurements for each cell are organized into a cell-by-gene matrix. These normalized non-negative observed values undergo discretization into five gene expression levels. Cells and genes are then assembled into a graph with nodes representing cells and genes, and edges between them representing different gene expression levels. This graph may then be embedded into a lower dimensional space resulting in a #entities x # dimension (by default, 50) SIMBA embedding matrix. **(b)** UMAP visualization of SIMBA embeddings of cells colored by cell type. **(c)** UMAP visualization of SIMBA embeddings of cells and genes. Cells are colored according to cell type as defined in b. Genes are colored slate blue. Cell-type-specific marker genes and housekeeping genes collected from Scanpy are indicated with text and arrows. Genes highlighted in red will be shown in d,e,and f. **(d)** SIMBA barcode plots of genes CST3, MS4A1, NKG7, and GAPDH. The x-axis indicates the ordering of a cell as ranked by the probability for each cell to be associated with a given gene.The y-axis describes the probability. The sum of probability over all cells is equal to 1. Each cell is one bar and colored according to cell type as defined in b. **(e)** SIMBA metric plots of genes. All the genes are plotted according to the Gini index against max score. The same set of genes as in **c** are annotated. **(f)** UMAP visualization of SIMBA embeddings of cells colored by gene expression of (left to right): CST3, NKG7, MS4A1, and GAPDH.

In addition, SIMBA also provides several quantitative metrics, including max value, Gini index, standard deviation, and entropy, to assess cell type specificity of various features (**Methods**). As an example, the gene metric plot of max value (a higher value indicates higher cell-type specificity) vs Gini index (a higher value indicates higher cell-type specificity), we see that marker genes (e.g., CST3, NKG7, MS4A1) fall in the upper right corner, as opposed to housekeeping genes (e.g., GAPDH) in the lower left corner (**Fig. 2e**). Similar separation is observed in other metrics (**Supplementary Fig. 1b**). These marker genes were further validated by the visualization of their expression pattern on UMAP plots (**Fig. 1f and Supplementary Fig. 1c**), accompanied by SIMBA barcode plots (**Supplementary Fig. 1d**).

To demonstrate that SIMBA provides a more accurate means of detecting marker genes, which differs from the statistical-testing-based methods implemented by tools such as *Scanpy* and *Seurat*, we compared a similar number of top marker genes identified by SIMBA (based on max value and Gini index) with those identified by Scanpy (based on the Wilcoxon rank-sum test) (**Supplementary Fig. 2a**). Upon comparison, we can see that nearly half of the marker genes discovered by SIMBA overlap with the marker genes identified by *Scanpy* (**Supplementary Fig. 2a**). However, on inspection of the top non-overlapping marker genes, all genes identified by SIMBA are found to be enriched only within certain groups of cells (**Supplementary Figs. 2b and 2c**) while genes identified by *Scanpy* but not by SIMBA include the housekeeping gene B2M and multiple ribosomal protein genes (e.g., RPS3 and RPS6) that are expressed ubiquitously in all cell types (**Supplementary Figs. 2b and 2d**). Specific limitations of cluster-centric approaches to scRNA-seq analysis are highlighted by inconsistencies in the statistical tests for differential expression applied after clustering in *Scanpy*. IL7R (marker gene of CD4 T) was identified only by the t-test method while FCER1A (marker gene of dendritic cells) was identified only by a

Wilcoxon rank-sum test. In contrast, SIMBA successfully identified IL7R and FCER1A as informative genes (**Fig. 2e and Supplementary Fig. 1b**).

Lastly, we showed that SIMBA does not require variable gene selection, which is an essential step in standard scRNA-seq pipelines such as *Seurat* or *Scanpy*. SIMBA produces very similar embeddings for cells and genes with and without variable gene selection (**Supplementary Fig. 2e**), though we observed that variable gene section does improve efficiency of the training procedure.

Single cell ATAC-seq analysis with SIMBA

As one of the most popular single-cell epigenomic techniques, single-cell assay for transposase-accessible chromatin using sequencing (scATAC-seq) has been widely used to profile regions of open chromatin and identify functional *cis*-regulatory elements such as enhancers and active promoters. In scATAC-seq, cells are characterized by different types of features [13], such as regions of accessible chromatin ("peaks" or "bins") and *cis*-regulatory elements (DNA sequences) within these accessible regions including transcription factor (TF) motifs or *k*-mer (short sequences of a specific length, *k*). Unlike existing methods that use only the positional information of peaks/bins or the sequence content found within them, in SIMBA both types of features can be encoded into a single graph. When only positional information is used, SIMBA constructs a graph with nodes representing cells and chromatin regions (peaks or bins) and edges indicating the accessibility of the chromatin regions in cells. When the DNA sequences for chromatin regions are available, SIMBA can also encode DNA sequences including TF motifs and *k*-mers into the graph by adding edges between these entities as nodes and the existing chromatin region nodes. The edges in this case indicate the presence of TF motifs/*k*-mers within these chromatin-accessible regions. Through the embedding procedure, SIMBA generates embeddings of cells along with peaks and DNA sequences (**Fig. 3a**, **Methods**).

To demonstrate the value of SIMBA embeddings for scATAC-seq analysis, we first applied SIMBA to a scATAC-seq data of 2,034 human hematopoietic cells with FACS-characterized cell types[14]. For the embeddings of cells alone, as shown using UMAP (**Figure 3b**), SIMBA accurately embeds cells such that cells belonging to distinct cell types are visually separated. In addition to cells, SIMBA can also embed various types of features. The UMAP plot in **Figure 3c** highlights how distinct features from both positional (peaks) as well as sequence-content (TF motifs and k-mers) information are embedded together based on their biological relations. Notably, these highlighted features that are embedded within the subpopulation of each defined cell type all have high cell-type specificity scores (shown in the upper right part of SIMBA metric plots in **Figure 3d**).

Analysis using SIMBA led to several key findings in human hematopoietic differentiation.

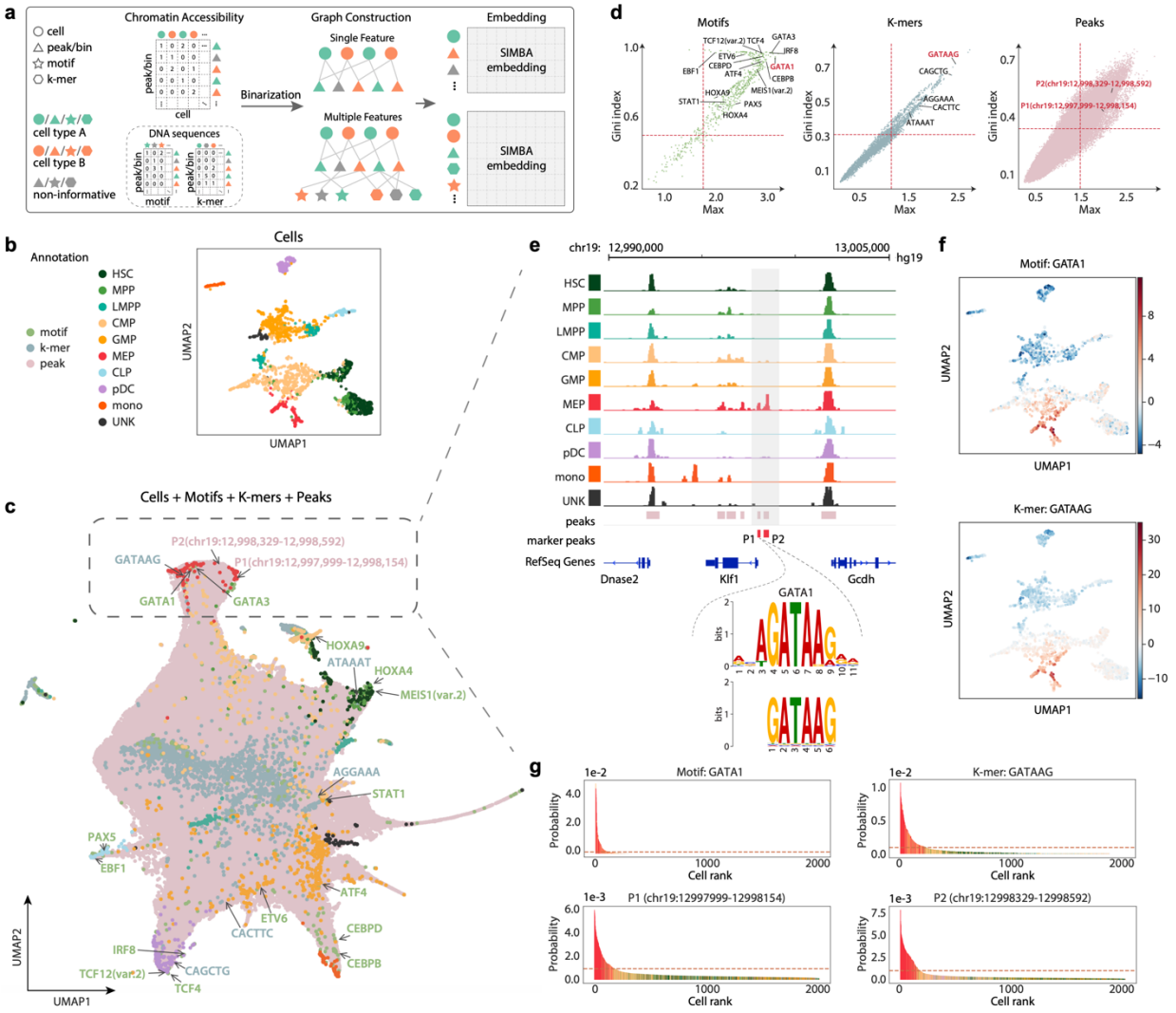


Figure3. ScATAC-seq analysis of the human hematopoiesis dataset *Buenrostro2018* using SIMBA. **(a)** SIMBA graph construction and embedding in scATAC-seq analysis. Biological entities including cells, peaks/bins, TF motifs, k-mers are represented as shapes and colored by relevant cell types (green and orange). Non-informative features are colored dark grey. Cells and chromatin accessible features (peaks / bins) are organized into a cell x peaks / bins matrix. When sequence information (TF motif or k-mer sequence) within these regions is available, they can be organized into two sub-matrices to associate a TF motif or k-mer sequence with each peak/bin. These constructed feature matrices are then binarized and assembled into a graph. When single feature (chromatin accessibility) is used, the graph encodes cells and peaks/bins as nodes. When multiple features (both chromatin accessibility and DNA sequences) are used, this graph may then be extended with the addition of TF motifs and k-mer sequences as nodes connected. Finally, SIMBA embeddings of these entities are generated through a graph embedding procedure. **(b)** UMAP visualization of SIMBA embeddings of cells colored by cell type. **(c)** UMAP visualization of SIMBA embeddings of cells and features including TF motifs, k-mers, and peaks. Cells are colored by cell type while motifs are colored green, k-mer sequences are colored blue, and peaks are colored

pink. Cell type specific features that are embedded nearby their corresponding cell types are indicated through the text (colored according to feature type) with arrows. **(d)** SIMBA metric plots of TF motifs, k-mers, and peaks. All these features are plotted according to the Gini index against max score. Cell-type specific TF motifs, k-mers, and peaks are highlighted. Dashed red lines indicate the cutoffs of cell type specific marker features. **(e)** Genomic tracks of aligned scATAC-seq fragments, separated and colored by cell type. Two marker peaks P1 and P2 in red are shown beneath the alignment as are RefSeq gene annotations. Within the peak P1, k-mer GATAAG and its resembling GATA1 motif are highlighted. **(f)** UMAP visualization of SIMBA embeddings of cells colored by TF activity scores of the GATA1 motif and k-mer GATAAG using chromVAR. **(g)** SIMBA barcode plots of the GATA1 motif, k-mer GATAAG, and the two peaks P1 and P2. Cells are colored according to cell type labels described above. Dotted red line indicates the same cutoff used in all four plots.

First, SIMBA analysis identified key master regulators of hematopoiesis. As highlighted in Figure 3c, we observed that motifs of previously reported TFs were all embedded near their respective cell types in UMAP. For example, the *GATA1* and *GATA3* motifs are close to megakaryocyte-erythroid progenitor (MEP) cells[15]; the *PAX5* and *EBF1* motifs are close to common lymphoid progenitor (CLP) cells[16]; the *CEBPB* and *CEBPD* motifs are close to monocytes (mono)[17].

Second, SIMBA analysis identified an unbiased set of DNA sequences, i.e., *k*-mers, that e important TF binding motifs involved in hematopoiesis, enabling *de novo* motif discovery. We observed that these *k*-mers were embedded near their resembling TF binding motifs and relevant cell subpopulations (Figure 3c and 3e, Supplementary Figure 3b). For example, the DNA sequence *CAGCTG* is embedded in plasmacytoid dendritic cells (pDCs), and this sequence matches the *TCF12* binding motif, which controls dendritic cell lineage specification.

To further illustrate the interpretability of the SIMBA embeddings of TF motifs and k-mers, we calculated TF activity scores (high-variance TF motifs/*k*-mers) with chromVAR[18]. As shown in Figure 3f, the *GATA1* TF motif and k-mer *GATAAG* that were both embedded in MEP cells by SIMBA, also showed high-level activity in MEP cells by chromVAR. The consistency between SIMBA embedding and chromVAR TF activity was observed for most of other TF motifs and *k*-mers as well (**Supplementary Fig. 3a, 3b**). We also noticed that SIMBA was still able to identify cell-type-specific TF motifs even when chromVAR failed to do so (e.g., *PAX5* was embedded in CLP cells by SIMBA but did not show a CLP-specific TF activity pattern using chromVAR). These highlighted features are also accompanied by SIMBA barcode plots, showing the sorted probabilities of each feature being assigned to different cells (Figure 3g and **Supplementary Fig. 3a,3b**). For example, the *GATA1* TF motif and *k*-mer *GATAAG* are both being assigned with much higher probabilities to MEP cells compared to the other cell types.

Third, SIMBA analysis identified differential accessible regions that may mediate cell-type specific gene regulation. For example, the two peaks at chr19:12997999-12998154 (*P1*) and chr19:12998329-12998592 (*P2*) that were embedded within MEP cells were almost exclusively observed in MEP cells on *KLF1* genome track (**Fig. 3e**). Interestingly, P1, upstream of *KLF1*,

contains the *k*-mer *GATAAG* that matches the *GATA1* binding motif, while transcription factor *GATA1* is known to regulate the gene *KLF1*and plays in a pivotal in erythroid cell and megakaryocyte development[19]. Therefore, by embedding these MEP-cell-related regulatory elements into the neighborhood of MEP cells, SIMBA demonstrates a new analytic way for studying cell-type-specific epigenetic landscape of the genome. To further validate the differential accessible regions identified by SIMBA, we randomly picked 100 peaks within each cell type in SIMBA co-embedding space. From the heatmap of chromatin accessibility, we clearly see that the peaks embedded in the neighborhood of each cell type by SIMBA shows strong cell-type specificity and this is robust to the number of cells within each cell type (**Supplementary Fig. 3c**).

Although SIMBA strongly differs from existing computational methods for scATAC-seq analysis in that SIMBA enables the co-embedding of cells and features, we still compared the resulting SIMBA embeddings of cells with these specialized methods by their ability to distinguish cell types. We observed that SIMBA yields very similar embeddings of cells when using either a single feature (peaks) or multiple features (peaks and DNA sequences from within those peaks) across four scATAC-seq datasets of varying technologies and organisms (**Supplementary Fig. 4**). Hence, to be fair we used the same set of features (i.e., peaks) for SIMBA as other methods in comparison against the top three methods, including SnapATAC [4], *Cusanovich*2018[20], and cisTopic[21] recommended by the recent benchmark study[13] (**Supplementary Fig. 5**). We compared SIMBA with these three methods in four scATAC-seq datasets of different technologies and organisms qualitatively based on UMAP visualization and quantitatively based on their clustering performance. As **Supplementary Fig. 5 shows,** SIMBA performs as well as or better than each of the methods evaluated, which are specialized for scATAC-seq only, further demonstrating the wide utility of SIMBA.

Single cell multimodal analysis with SIMBA

scRNA-seq and scATAC-seq are two of the most widely-adopted single-cell sequencing technologies, but they are limited to measuring only a single aspect of cell state at a time. To improve our ability to interrogate cellular states, several single-cell dual-omics technologies have been developed [22-25] to jointly profile transcriptome and chromatin accessibility within the same individual cells, therefore providing the potential to correlate gene expression with accessible regulatory elements and further delineate the yet elusive principles of gene regulation. In this section, we demonstrate how SIMBA may be used to perform multimodal analyses. We applied SIMBA to three recent single-cell dual-omics technologies: SHARE-seq[23], SNARE-seq[22], and a multiome PBMCs dataset from 10x Genomics. **Figure 4a** illustrates the procedure of graph construction and generation of the final SIMBA embedding matrix. Briefly, for scRNA-seq data, the gene expression matrix is discretized to generate different levels of gene expression. For scATAC-seq, both the chromatin accessibility matrix and motif/*k*-mer match matrix are binarized. In this graph, there are five types of entities (nodes), including cells, genes, peaks, motifs, and *k*-mers. For scRNA-seq data, an edge indicates whether a gene is expressed in a cell and its weight indicates the gene expression level (five levels, by default). For

scATAC-seq, an edge indicates whether a peak is present in a cell or if a TF motif/*k*-mer is present within a peak.

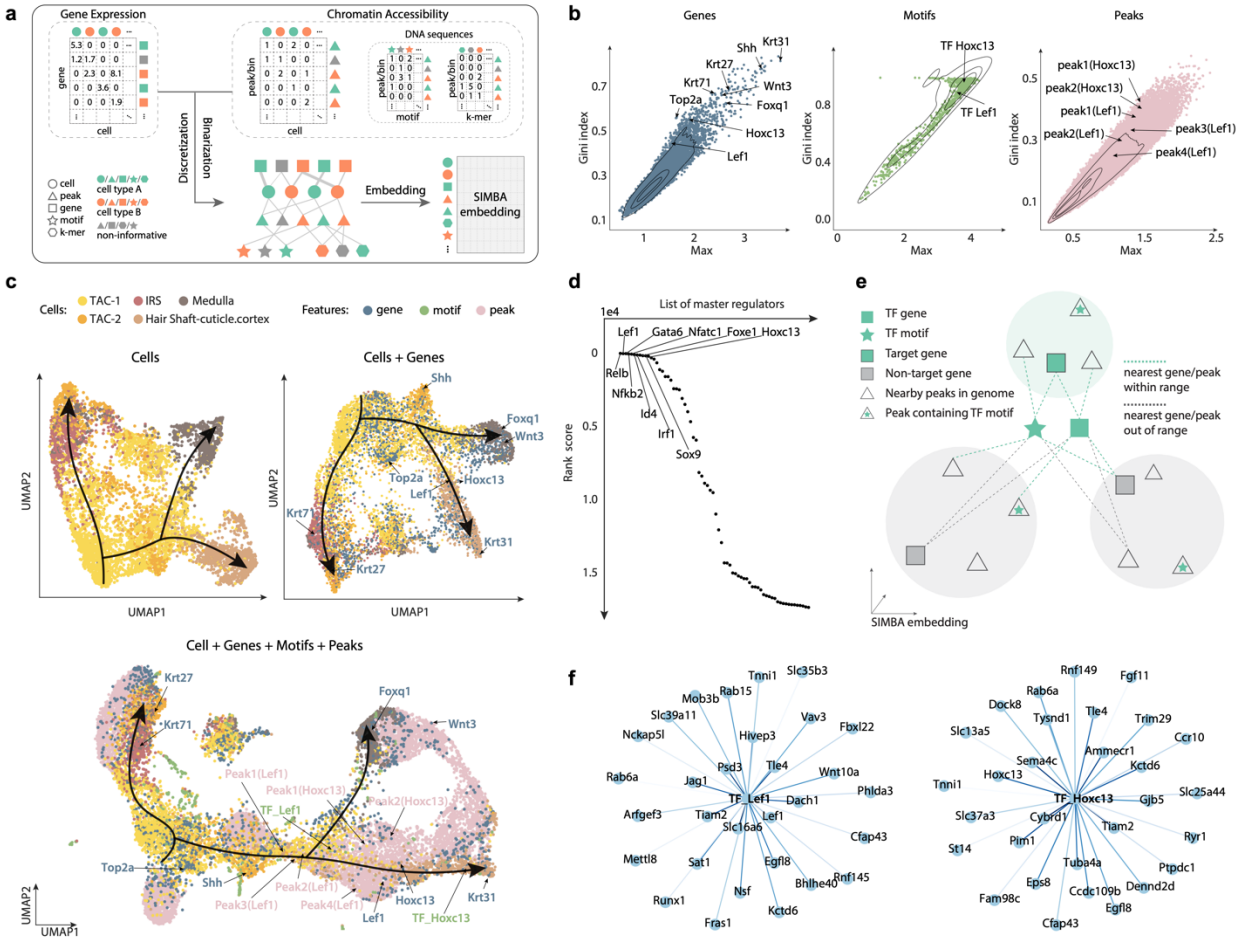


Figure 4. Multimodal analysis of the SHARE-seq hair follicle dataset using SIMBA. **(a)** SIMBA graph construction and embedding in multimodal analysis. Overview of SIMBA's approach to multimodal (scRNA-seq + scATAC-seq) data analysis. **(b)** SIMBA metric plots of genes, TF motifs, and peaks. All these features are plotted according to the Gini index against max score. Cell-type specific genes, TF motifs, and peaks are highlighted. **(c)** UMAP visualization of SIMBA embeddings of cells (Top-left), cells and genes (Top-right), and cells along with genes, TF motifs, and peaks (Bottom). **(d)** Ranked scatter plot of candidate master regulators as identified by SIMBA. **(e)** Schematic description of SIMBA's strategy for identifying target genes given a master regulator. **(f)** Top 30 target genes of transcription factors *Lef1* and *Hoxc13* as inferred by SIMBA.

To demonstrate the usefulness and versatility of the SIMBA embeddings, we analyzed the cell populations undergoing hair follicle differentiation from mouse skin profiled with SHARE-seq.

First, we used SIMBA to assess the cell-type specificity of different types of features, including genes, TF motifs, and peaks (**Fig. 4b, Methods**). As shown in **Figure 4b**, genes (e.g., *Lef1* and *Hoxc13*), which are associated with hair follicles each have relatively higher max values and Gini index scores. Similarly, TF motifs and peaks proximal to the genomic loci of these genes also score in the upper right quadrant of the metric plots. SIMBA's cell-type specificity metrics successfully revealed the key genes and regulatory factors important to the hair follicle differentiation process.

Next, we visualized and interrogated the SIMBA embeddings of 1) cells; 2) cells and genes; and 3) cells together with genes, TF motifs, and peaks. **Figure 4c** shows the UMAP visualization of the SIMBA embeddings of cells and informative features selected based on SIMBA metric plots. The UMAP visualization of SIMBA embeddings of cells and the full set of features was also performed (**Supplementary Fig. 6a**). However, we observed that with the SIMBA embeddings of cells and the informative features selected by SIMBA, UMAP plots show more visible cell subgroups without obstruction by entities that may be present in much larger quantities (e.g., peaks). SIMBA embeddings of cells were able to reveal the three fate decisions from transit-amplifying cells (TACs), including inner root sheath (IRS), medulla, and cuticle/cortex. SIMBA embeddings also uncovered important genes and regulatory factors along the hair follicle differentiation trajectories. For example, the marker genes *Krt71*, *Krt31*, and *Foxq1* were embedded into their corresponding cell types: IRS, cuticle/cortex, and medulla, respectively. The *Lef1* motif was embedded into the beginning of medulla and cuticle/cortex lineages while the *Hoxc13* motif was embedded into the late stage of cuticle/cortex differentiation. Peaks near the *Lef1* and *Hoxc13* loci were also embedded into the nearby regions of these genes and motifs, as expected. To show the robustness of SIMBA embedding, we also performed the single-modality analyses within the SHARE-seq dataset, separating the scRNA-seq and scATAC-seq components. With the consistent embedding results of cells and features as in multimodal analysis, we further demonstrated that SIMBA embedding procedure is robust to the type and number of features encoded in the input graph (**Supplementary Fig. 6b**,**6c**). Each marker gene was further validated using the UMAP plots with cells colored by gene expression as well as using the SIMBA barcode plots. The two aforementioned TF motifs and their respective peak sets were also validated visually using SIMBA barcode plots. As expected, in SIMBA barcode plots these marker features clearly showed an imbalanced distribution with much higher probabilities in the correct cell types (**Supplementary Fig. 7a-d**).

Further, we demonstrated that the SIMBA co-embedding space of cells and features provides the potential to identify master regulators of differentiation and infer their target regulatory genes. To define a master regulator *a priori*, we postulate that both its TF motif and TF gene should be cell-type specific and given that active gene regulation involves the expression of TF and accessibility of its binding sites, TF motif and TF gene should be embedded closely in the shared latent space. Extending this logic to identify putative master regulators, we assessed the cell-type-specificity of TF motifs and genes based on SIMBA metrics and ranked all potential master regulators based on the distance between the TF motif and the respective TF gene in the shared SIMBA embedding space (**Methods**). SIMBA successfully identified previously described master regulators such as *Lef1*, *Gata6*, *Nfatc1*, and *Hoxc13*. as the top master

regulators related to lineage commitment in mouse skin (**Fig. 4d, Supplementary Table 2**). To infer the target genes of a given master regulator, we postulate that in the shared SIMBA embedding space, 1) the target gene is close to both the TF motif and the TF gene; 2) the accessible regions (peaks) near the target gene loci must be close to both the TF motif and the target TF gene. Resting on these assumptions of *cis*-regulatory dynamics, the inference of target genes was performed by calculating the distance between target gene candidates and the respective TF motif and gene. In addition, nearby peaks around the target gene's locus and the presence of TF motif in these nearby peaks are also considered (**Fig. 4e, Methods**). The top 30 target genes of TF *Lef1* and TF *Hoxc13* inferred by SIMBA are shown respectively (**Fig. 4f**). The full list of ranked target genes is provided in Supplementary table 3. Notably, we were able to recover targets genes that were also reported in the original study[23]. For example, genes *Lef1*, *Jag1*, *Hoxc13*, *Gtf2ird1* are regulated by the TF *Lef1*, while genes *Cybrd1*, *Hoxc13*, *St14* are regulated by the TF *Hoxc13*. We also showed the target genes of three additional top master regulators identified by SIMBA including *Relb*, *Gata6*, and *Nfatc1*(**Supplementary Fig. 7e**).

In addition to SHARE-seq, we also applied SIMBA to another two dual-omics datasets, the mouse cerebral cortex dataset profiled by SNARE-seq[22] (**Supplementary Fig. 8**) and the multiome PBMCs dataset from 10x Genomics (**Supplementary Fig.9**). By validating the embeddings of cells and features with given labels (**Supplementary Fig. 8a and Fig.9a**), marker genes from the original study (**Supplementary Fig. 8a,b,d and Fig.9a,b,d**), and differentially accessible chromatin regions (**Supplementary Fig. 8c and Fig.9c**), we further demonstrate the suitability of SIMBA for multimodal analysis.

Single cell batch correction analysis with SIMBA

Efforts to collect data from single cells has grown to the level of consortia that span multiple institutions with the hopes of finely mapping and characterizing specific tissues. However, while the feasibility of generating large cohorts of single-cell data has increased, this has brought with it an increased demand for analysis methods that are capable of negating technical covariates inherent to multi-batch data collection. Covariates including experimental replicate identity, sample preparation, and sequencing platform are capable of confounding biological signal and batch correction that removes the effects of technical covariation while preserving true biological signals is required prior to downstream analysis [26, 27].

We demonstrate that SIMBA readily corrects batch effects and produces joint embeddings of cells and features across multiple datasets with different sequencing platforms and cell type compositions. Thus, while previous methods primarily rely on specialized tools for batch correction, SIMBA works as a stand-alone package obviating the need for prior input data correction when applied to multi-batch scRNA-seq dataset. In SIMBA, batch correction is accomplished by encoding multiple scRNA-seq datasets into a single graph. Cells in different batches are linked to genes as in the previously described scRNA-seq graph construction. Here, the gene nodes are shared between the cell nodes of different batches. In addition to the experimentally measured edges, batch correction is further enhanced through computationally

inferred edges drawn between similar cell nodes across datasets using a truncated randomized singular value decomposition (SVD)-based procedure inspired by *Seurat v3*[11]. SIMBA then generates the embeddings of all nodes including cells of each batch and genes from the resulting graph, which may then be visualized using tools such as UMAP, similar to the analyses presented in the sections above (**Fig. 5a**, **Methods**).

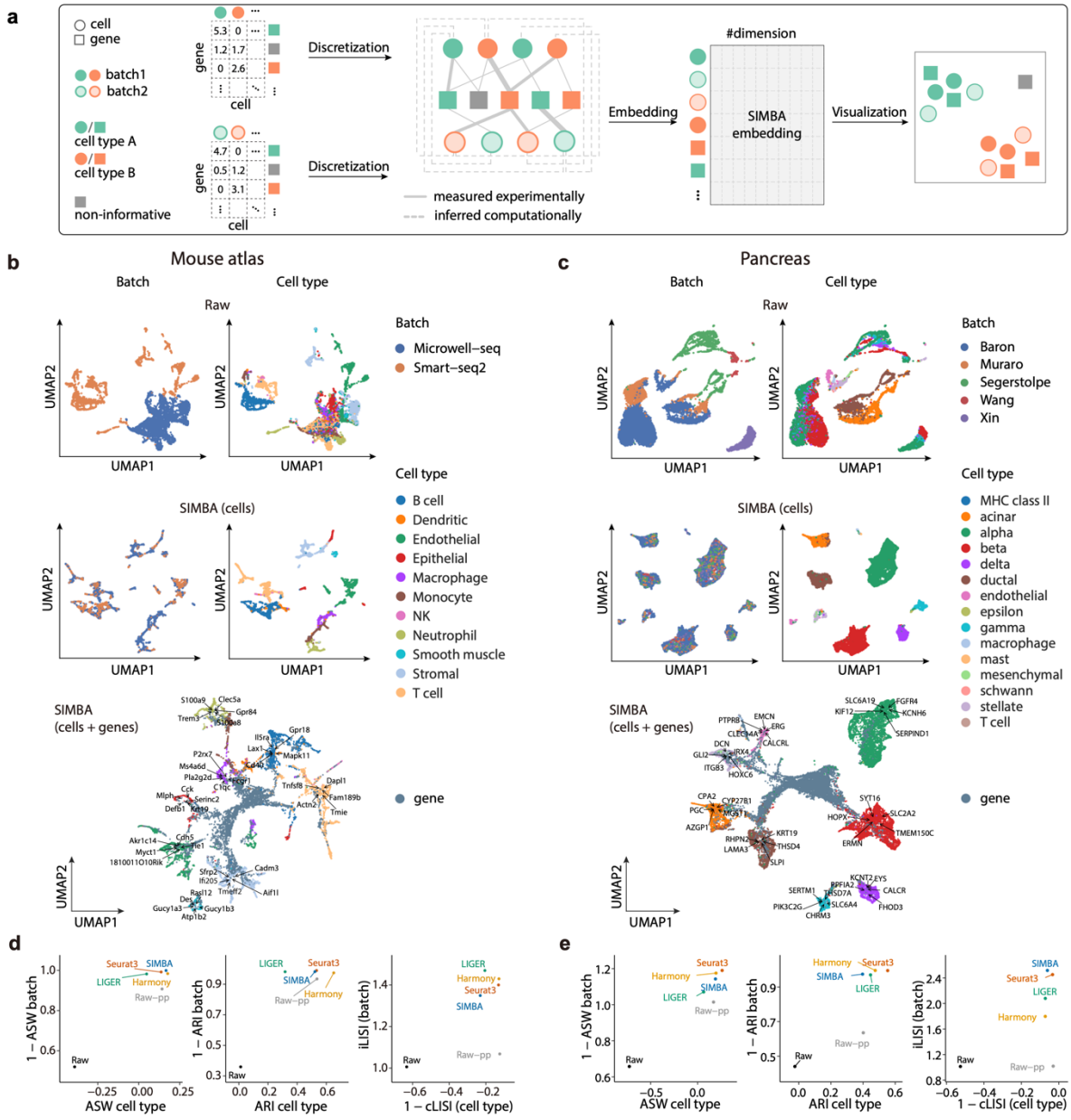


Figure 5. Batch correction analysis of scRNA-seq data using SIMBA. **(a)** SIMBA graph construction and embedding in batch correction analysis. Overview of SIMBA's approach to batch correction across scRNA-seq datasets. Distinct shapes indicate the type of entity (cell or gene). Colors distinguish batches or cell types. **(b)** UMAP visualization of the scRNA-seq mouse atlas dataset with two batches of different technologies (Microwell-seq and Smart-seq2) before and after

batch correction. Cells are colored by scRNA-seq profiling technology and cell type respectively. Top: UMAP visualization before batch correction; Middle: UMAP visualization after batch correction with SIMBA; Bottom: UMAP visualization of SIMBA embeddings of cells and genes, with batch effect removed and known marker genes highlighted. **(c)** UMAP visualization of the scRNA-seq human pancreas dataset with five batches of different studies before and after batch correction. Cells are colored by scRNA-seq data source and cell type respectively. Top: UMAP visualization before batch correction; Middle: UMAP visualization after batch correction with SIMBA; Bottom: UMAP visualization of SIMBA embeddings of cells and genes, with batch effect removed and known marker genes highlighted. **(d-e)** Quantitative comparison of SIMBA with three other batch correction methods including Seurat3, LIGER and Harmony, using, left-to-right: average silhouette width (ASW), adjusted Rand index (ARI), and local inverse Simpson's index (LISI) on the scRNA-seq mouse atlas dataset and human pancreas dataset respectively.

SIMBA was applied to a mouse atlas dataset composed of two batches and a human pancreas dataset that spans several batches used in a recent benchmark study[26]. The mouse atlas dataset contains two scRNA-seq datasets with shared cell types from different sequencing platform. The human pancreas dataset contains five samples pooled from five sources using four different sequencing techniques, in which not all cell types are shared across each sample. For both datasets, SIMBA successfully corrected the batch effects where the resulting embeddings of cells are clustered by the cell types while each batch is mixed evenly, indicating the preservation of biological signal and the simultaneous elimination of confounding technical covariates (**Fig. 5b-c, middle, upper**). It is important to note that the mouse atlas dataset was collected from nine different organ systems, so there exists some expected heterogeneity within the cell type labels. Conversely, the human pancreas datasets are curated from a single organ and SIMBA sufficiently separated cell types into transcriptionally-distinct homogeneous cell clusters (**Fig. 5c**).

In addition to batch effect removal, SIMBA also simultaneously identifies cell-type-specific marker genes (**Fig. 5b-c, bottom**). Having effectively eliminated differences between cells due to technical covariates, marker genes are discoverable across multiple samples by querying according to cells of each cell type within the batch-corrected SIMBA embedding. As shown in the SIMBA co-embedding of cells and variable genes, the known marker genes were correctly placed within each cell type while non-marker genes were embedded away from any cell type (**Supplementary Fig. 10, 11**). The resulting marker genes recapitulated the clustering-based differential expression (DE) analysis results for each datasets[28-33] (e.g. *Cdh5*, *Tie1*, *Myct1* for endothelial cell and *C1qc*, Fcgr1 for macrophage, *S100a8*, *Trem3* for Neutrophil in the mouse atlas dataset and *KIF12* for alpha cell and *KRT19* for ductal cell in the human pancreas dataset) and are shown to be expressed specifically in the queried cell types (**Supplementary Fig. 10, Supplementary Fig. 11**). This distinguishes SIMBA from other batch correction methods, in which clustering is performed first in the batch-corrected space and then marker genes of each batch are identified through DE analysis in the original, uncorrected space of the corresponding batch.

Multiple methods have now been developed to correct for the technical effects of sample preparation and data collection in single cells. While SIMBA is a generalizable graph embedding method capable of several tasks outside of batch correction, we still compared SIMBA with the methods that are particularly designed for batch correction or data integration, including three top-performing batch correction methods, *Seurat3*, *LIGER* and *Harmony*, which are widely adopted and even recommended by a recent benchmark study[26] (**Supplementary Note1**). To qualitatively compare these methods, we visualized cells of each dataset before and after batch-correction in UMAP plots (**Supplementary Figure 12**). For the quantitative evaluation of this batch correction performance, the conservation of biological information and batch effect removal are measured using three different metrics: average silhouette width, adjusted Rand index, and local inverse Simpson's index[9] as in the previously-mentioned benchmark study[26] (**Methods**). Each metric measures the relative mixing of class labels, where optimal performance is associated with maximal mixing in the batch labels and minimal mixing in the cell type labels. We observed that SIMBA achieved comparable batch correction performance both qualitatively and quantitatively for both the mouse atlas dataset and the human pancreas dataset (**Supplementary Fig. 12, Fig. 5d, Fig.5e**).

Single cell multi-omics integration analysis with SIMBA

Single-cell assays are now capable of measuring a broad range of cellular modalities including mRNA, chromatin accessibility, DNA methylation and cell-surface proteins. Thus, data is being generated that describes cells by varying features sets, which has motivated the need for methods that leverage these features to perform multi-omics integration such that a more comprehensive description of cell state may be learned. We demonstrate that SIMBA can be applied to the integration analysis of such multi-omics datasets, especially as it applies to datasets comprised of scRNA-seq and scATAC-seq. Specifically, SIMBA accomplishes this integration by first building one graph for scRNA-seq data and another graph for scATAC-seq data, independently as described in previous sections. To connect these two graphs, SIMBA then calculates gene activity scores by summarizing accessible regions from scATAC-seq data and then infers edges between cells of different assays based on their shared gene expression modules through a similar procedure as in the previously described batch correction section. Finally, SIMBA embeds the graph of cells, genes, and peaks into a common, low-dimensional space. The SIMBA embeddings of these multi-omics entities can be visualized using UMAP or similar visualization tools (**Fig. 6a, Methods**).

To facilitate the evaluation of data integration performance, we created datasets with ground-truth labels by manually splitting the dual-omics datasets into two single-modality datasets (i.e., scRNA-seq and scATAC-seq), in which we know the true matching between cells across the two modalities. We then applied SIMBA to the integration analysis of two case studies where scRNA-seq and scATAC-seq datasets are generated from the SHARE-seq mouse skin dataset and the 10x Genomics multiome human PBMCs dataset, respectively. We observed that SIMBA was able to preserve cellular heterogeneity while evenly mixing the two modalities (**Fig. 6b-c, top**

and middle). For example, within the SHARE-seq mouse skin dataset SIMBA was able to identify the pattern of differentiation from the TAC cell type to the IRS, medulla, and hair shaft cuticle/cortex cell types while maintaining a proper mixture of both scRNA-seq and scATAC-seq modalities (**Fig. 6b, middle**).

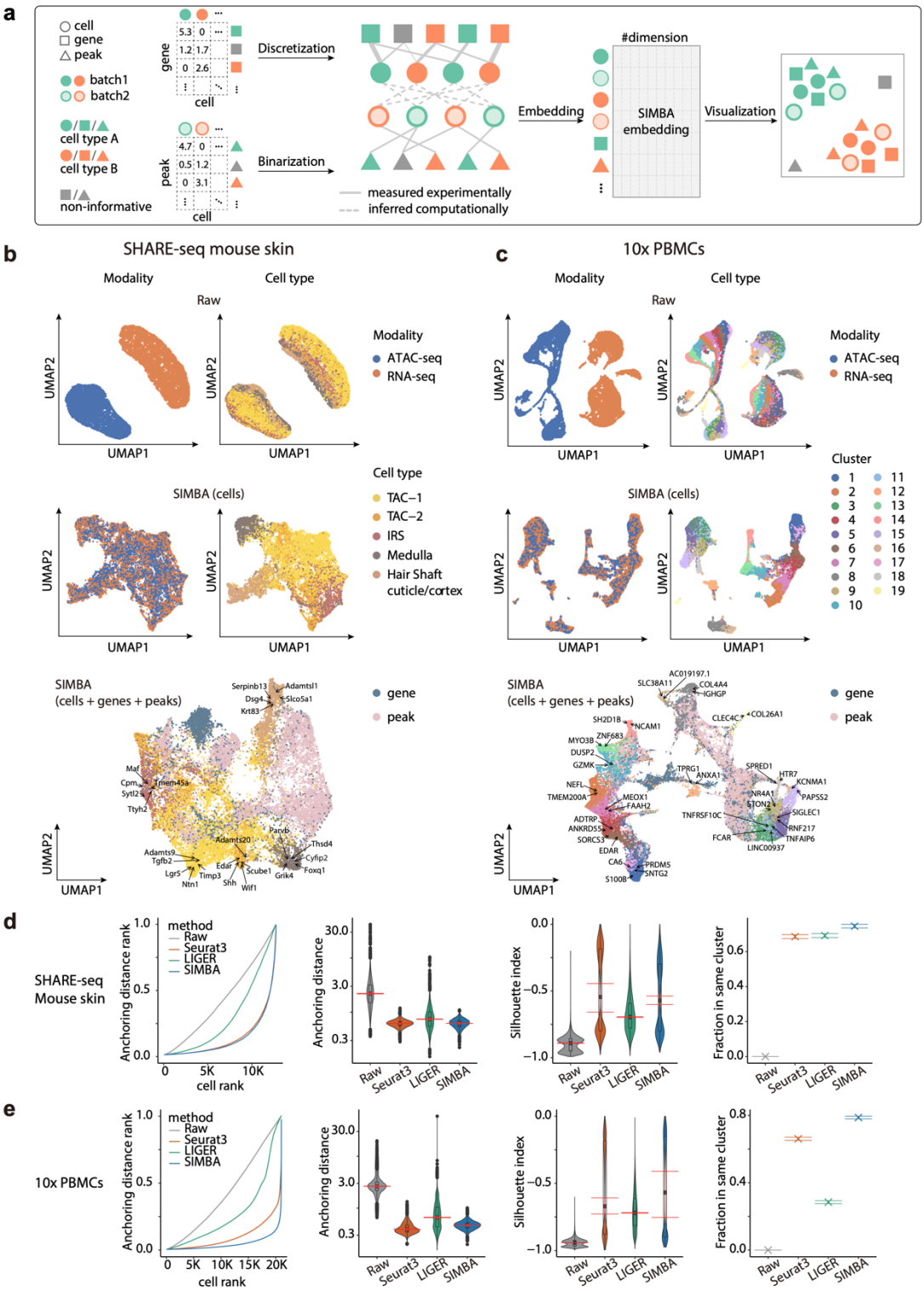


Figure 6. Multi-omics integration of scRNA-seq + scATAC-seq data using SIMBA. **(a)** SIMBA graph construction and embedding in multi-omics integration. Overview of SIMBA's approach to data integration across scRNA-seq and scATAC-seq. Distinct shapes indicate the type of entity (cell, gene, or peak). Colors distinguish batches or cell types. **(b)** UMAP visualization of the integrated scRNA-seq and scATAC-seq data manually created from the SHARE-seq mouse hair follicle dataset before and after data integration. Cells are colored by single-cell modality and cell type respectively. Top: UMAP visualization before integration; Middle: UMAP visualization after integration with SIMBA; Bottom: UMAP visualization of SIMBA embeddings of cells, genes, and peaks with two cell modalities integrated and known marker genes highlighted. **(c)** UMAP visualization of the integrated scRNA-seq and scATAC-seq data manually created from the 10x human PBMCs dataset before and after data integration. Cells are colored by single-cell modality and cell type respectively. Top: UMAP visualization before integration; Middle: UMAP visualization after integration with SIMBA; Bottom: UMAP visualization of SIMBA embeddings of cells, genes, and peaks with two cell modalities integrated and known marker genes highlighted. **(d-e)** Quantitative comparison of SIMBA with two other methods including Seurat3, LIGER for multi-omics integration, using, left-to-right: anchoring distance rank, anchoring distance, silhouette index, and Fraction in the same cluster, on the SHARE-seq mouse hair follicle dataset and 10x human PBMCs dataset, respectively.

Importantly, SIMBA simultaneously identified marker genes and peaks at single-cell resolution. In the co-embedding space, we observed that the neighbor genes of cells (highlighted in UMAP plots), are each exclusively expressed in their corresponding cell types (**Supplementary Fig. 13a-c,e, Fig. 14a-c,**e). For example, in the SHARE-seq mouse skin dataset, *Foxq1* and *Shh* are located within medulla and TAC-2, respectively; in the 10x PBMCs dataset, *PAPSS2* and *KCNMA1*, which are the marker genes of blood monocytes, are embedded close to each other. Similarly, we observed that the neighbor peaks of cells show a clear cell-type-specific accessibility pattern that is robust to the cluster size of a given cell type (**Supplementary Fig. 13a-d, Fig. 14a-d).**

We next demonstrate SIMBA performs similarly or better than two widely-adopted methods for single-cell data integration, *Seurat3* and *LIGER* in these two case studies. Out of the aforementioned, top-performing integration methods, *Seurat3* and *LIGER* were selected because they have explicit documentation for the task of integrating scRNA-seq and scATAC-seq data. We first qualitatively evaluated these methods by inspecting UMAP visualization plots. For the SHARE-seq dataset, we observed that all three methods perform comparably well in mixing cells of two modalities though LIGER generated particularly small and noisy clusters (**Supplementary Fig. 15a)**. For the 10X PBMCs dataset, SIMBA exhibited a clear superiority in mixing cells belonging to each modality (**Supplementary Fig. 15b**). We next quantitatively assessed the integration performance of these methods by four different metrics (**Methods**). Each metric quantifies the distances between matched cells in the integrated space. In addition to the commonly-used metrics including anchoring distance, Silhouette index, and Fraction in the same cluster, we developed an additional metric, *anchoring distance rank* (ADR), which represents the normalized rank of the distance between matching cells. If two matching cells

from scRNA-seq and scATAC-seq are mutually closest to one another, their ADR will be close to 0 (**Methods**). SIMBA performed comparably or better by each metric for both datasets with the best performance in ADR and cluster agreement in both datasets (**Fig. 6d**).

Discussion

Simultaneous multimodal measurements of individual cells offer new and unexplored opportunities to investigate complex interactions between interacting components involved in establishing transcriptional programs. Despite the exciting potential for discovery associated with these datasets, methods to define cell states and important features across modalities remains under development.

As presented in this manuscript, SIMBA models single cells and measured features as nodes encoded in a graph and employs a scalable and efficient graph embedding procedure to embed cells and features into a shared latent space. We demonstrate that direct graph representations of single-cell data are able to capture not only relations between cells and the quantified features of the experiment (e.g., gene expression or chromatin accessibility) but additionally boast the capacity to capture hierarchical relations between features. An example of such a hierarchical relation might include the coordinate-level description of an ATAC-seq peak and the corresponding TF motifs and/or *k*-mer sequences contained within that region. In the resulting joint embedding, proximity-based queries can be performed to discover cell-type-specific regulatory mechanisms and the respective features integral to such mechanisms. Therefore, SIMBA enables unbiased multimodal feature discovery, and complements the current gene regulatory network analyses. SIMBA also circumvents the typical workflow led by cell clustering and followed by differential feature detection, and thus relieves the researcher of relying on user-defined clustering resolution that may lead to artifactual discovery or false negative results.

SIMBA has been extensively benchmarked across single-cell modalities and tasks and obtained comparable or better performance compared to current state-of-the-art methods developed for the respective task. These results suggest a wide applicability of SIMBA's graph-based framework and therefore obviates the need for stitching together workflows over multiple analysis tools. This contrasts with task-specific methods, which have been specially developed for a given data type and modality. A foreseeable extension of SIMBA would include encoding increasingly complex measurements of single-cells such as spatial transcriptomics wherein transcriptomic and (real) proximity data should be considered [34]. We also envision extending this framework to single-cell Hi-C data wherein the interaction between DNA segments should be encoded. SIMBA is particularly apt towards analysis single-cell Hi-C measurements as these DNA fragments, analogous to scATAC-seq data contain hierarchical information such as TF motifs and *k*-mer sequences. Finally, lineage-traced single-cell datasets[35] wherein a subset of cells retain a known lineage are of interest. In this example, lineages might be encoded hierarchically with respect to gene expression. In general, we are interested in the further incorporation of external information and hierarchical relationships between features in the

graph. While the extension to various experimental design is relatively straightforward, we note that the interpretation of the output embedding in each case may vary as a function of the input graph construction and training process thus requiring some level of domain-specific expertise.

Graph embedding methods hold significant promise for the analysis of biological data. Previous applications of graph embedding include functional annotation of genes [36], transcription factor binding to DNA motifs [37] and more recent single-cell RNA-seq analyses [38, 39]. The graph encoding and embedding procedures we have outlined may be potentially improved and extended to better capture or represent biological entities and their respective relations. We envision employing continuous edge weights as well as the consideration of both node attributes and edge information to determine the proximity of node pairs [40]. Considering embedding distances as a function of the node attribute may better equip our approach to extend to time series data as well as to better align cells stratified by positions on a developmental axis. In addition, to improve interpretability it may be possible to provide a metric of uncertainty alongside the resulting embedding and generalization to the unseen nodes[41].

The propensity to generate high-quality, complex datasets that capture and quantify multiple classes of molecules (omics) will undoubtedly continue to increase. Already this progress has outpaced our ability to gain integrative insights from such data, highlighting a need for methods that break through previous limitations as well as extend easily to new tasks. Here we believe SIMBA satisfies these conditions for such a comprehensive yet accessible method for exploring cellular heterogeneity and investigating the regulatory mechanisms that drive the cellular diversity while laying a groundwork for the development of new non-cluster-centric single cell omics computational methods.

Methods

Single-cell data preprocessing

a. Single-cell RNA-seq

Genes expressed in fewer than three cells were filtered. Raw counts were library size-normalized and subsequently log-transformed. Optionally, variable gene selection [11] (a python version is implemented in SIMBA that is inspired by Scanpy[2]) may be performed to remove non-informative genes and accelerate the training procedure. Notable differences in the resulting cell embeddings were not observed upon limiting feature input to those identified by variable gene selection but SIMBA embeddings of non-variable genes will not be generated as they are not encoded in the graph.

b. Single-cell ATAC-seq

Peaks present in fewer than three cells were filtered. Optionally, we implemented a scalable truncated-SVD-based procedure to select variable peaks as a preliminary step to additionally filter non-informative peaks and accelerate the training procedure. First

the top k principal components (PCs) were selected, with k chosen based on the elbow plot of variance ratio. Then for each of the top *k* PCs, peaks were automatically selected based on the loadings using a knee point detection algorithm implemented by 'kneed'[42]. Finally, peaks selected for each PC were combined and denoted as "variable peaks". Similar to the observation made with scRNA-seq data, the optional step of variable peak selection has a negligible effect on the resulting cell embedding. Despite this minimal impact on the resulting embedding, this feature selection step imparts a significant practical advantage in reducing training procedure time.

k-mer and motif scanning was performed using packages 'Biostrings' and 'motifmatchr' with JASPAR2020[43]. Included in the implementation of SIMBA is a convenient R command line script "scan_for_kmers_motifs.R" , which will convert a list of peaks (formatted in a bed file) to a sparse peaks-by-*k*-mers/motifs matrix, which is stored as an hdf5-formated file.

Graph construction (five scenarios)

i. Single-cell RNA-seq analysis

The distribution of non-zero values in the normalized gene expression matrix was first approximated using a *k*-means clustering-based procedure. First, the continuous non-zero values were binned into n intervals (by default n=5). Bin widths were defined using 1-dimensional *k*-means clustering wherein the values in each bin are assigned to the same cluster center. The continuous matrix is then converted into a discrete matrix wherein$1, \ldots, n$ are used to denote n levels of gene expression. Zero values are retained in this matrix. Then the graph was constructed by encoding two types of entities, cells and genes, as nodes and relations with n different weights between them, i.e., n levels of gene expression, as edges. These n relation weights range from 1.0 to 5.0 with a step size of $5/n$ denoting gene expression levels (lowest: 1.0, highest: 5.0), such that edges corresponding to high expression levels affect embeddings more strongly than those with intermediate or low expression levels. This discretization is implemented in the SIMBA package using the function, "si.tl.discretize()".

ii. Single-cell ATAC-seq analysis

Peak-by-cell matrices were binarized, with "1" indicating at least one read within a peak and "0" otherwise. The graph was constructed by encoding two types of entities, cells and peaks, as nodes and the relation between them, denoting the presence of a given peak in a cell, as edges. The single relation type was assigned with a weight of 1.0. When the DNA sequence features were available, they were encoded into the graph using *k*-mer and motif sequence entities as nodes. This was performed by first binarizing the peak-by-*k*-mer/motif matrix then constructing an extension to the original peak/cell graph using the peaks, *k*-mers, and motifs as nodes and the presence of these entities within peaks as edges between these additional nodes and the peak nodes. The relation between k-mers and peaks was assigned a weight of 0.02 while the relation between TF motifs was assigned a weight of 0.2. Of note, *k*-mers and motifs may be used

independently of each other as node inputs to the graph, depending on the specific analysis task.

iii. Multimodal analysis

Combination of the above outlined strategies for graph construction of scRNA-seq and scATAC-seq data was used to construct a multi-omics graph.

iv. Batch correction

A graph for each batch was constructed as described in i). Edges between cells of different batches were inferred through a procedure based on truncated randomized singular value decomposition (SVD) inspired by Seurat v3 [11] to stitch together graphs of different batches. More specifically, in the case of scRNA-seq data, consider two gene expression matrices $X1_{n_1 \times m}$ and $X2_{n_2 \times m}$, where n_1, n_2 denotes the number of cells and m denotes the number of the shared features, i.e., variable genes, between datasets. The matrix $X_{n_1 \times n_2}$ was then computed by multiplying $X1$ and $X2$:

$$X = X1 \times X2^T$$

Truncated randomized SVD was subsequently performed on X:

$$X \approx U \times \Sigma \times V^T$$

where U is an $n_1 \times d$ matrix, Σ is an $d \times d$ matrix, and V is an $n_2 \times d$ matrix (by default $d = 20$).

Both U and V were further $L2$ normalized. For each cell in U, we searched for k nearest neighbors in V and vice versa (by default, $k = 20$). Eventually, only the mutual nearest neighbors between U and V were retained as inferred edges between cells (represented as dashed lines in **Fig. 5a**). The procedure of inferring edges between cells of different batches is implemented in the function "si.tl.infer_edges()" in the SIMBA package.

For multiple batches, SIMBA can flexibly infer edges between any pair of datasets. In practice, however edges are inferred between the largest dataset(s) or the dataset(s) containing the most complete set of expected cell types and other datasets.

v. Multi-omics integration

scRNA-seq and scATAC-seq graphs were constructed following steps i) and ii), respectively. To infer the edges between cells of scRNA-seq and scATAC-seq, gene activity scores were first calculated for scATAC-seq data[3]. More specifically, for each gene, peaks within 100kb upstream and downstream of the TSS were considered. Peaks overlapping gene body region or within 5kb upstream of gene bodies were given the weight of 1.0. Otherwise, peaks were weighted based on their distances to

TSS using the exponential decay function: $e^{\frac{-distance}{5000}}$. Subsequently, the gene score of each gene was computed as a weighted sum of the considered peaks. These gene scores were then scaled to respective gene size. These steps are implemented by the function "si.tl.gene_scores()" in SIMBA. For user convenience, the SIMBA package curates the gene annotations of several commonly used reference genomes, including hg19, hg38, mm9, and mm10. Once gene scores were obtained, the same procedure described in iv) was performed to infer edges between cells profiled by scRNA-seq and scATAC-seq using the function, "si.tl.infer_edges()" in SIMBA.

The procedure of generating constructed graphs is implemented in the function, "si.tl.gen_graph()" in the SIMBA package.

Graph Embeddings with Type Constraints

Following the construction of a multi-relational graph between biological entities, we adapted graph embedding techniques from the knowledge graph and recommendation systems literature to construct unsupervised representations for these entities.

We provide as input a directed graph $G = (V, E)$, where V is a set of entities (vertices) and E is a set of edges, with a generic edge $e = (u, v)$ between a source entity u and destination entity v. We further assume that each entity has a distinct known type (e.g., cell, peak, etc.).

Graph embedding methods learn a D-dimensional embedding vector for each $v \in V$ by optimizing a link prediction objective via stochastic gradient descent, with *D=50* used for our experiments. We will denote the full embedding matrix as $\theta \in R^{|V| \times D}$ and the embedding for an entity v as θ_v.

For an edge $e = (u, v)$, we denote $s_e = \theta_u * \theta_v$ as the score for e, and optimize a multi-class log loss

$$\mathcal{L} = -log \frac{\exp(s_e)}{\sum_{e' \in \mathcal{N}} \exp(s'_e)}$$

Where $\mathcal{N}$ is a set of "negative sampled" candidate edges generated by corrupting e [44]. This log loss objective attempts to maximize the score for all $(u, v) \in E$ and minimize it for $(u, v) \notin E$.

Negative samples are constructed by replacing either the source or target entity in the target edge $e = (u, v)$ with a randomly sampled entity. However, in graphs like ours where only edges between certain entity types are possible, previous work has shown that it is beneficial to optimize the loss only over candidate edges that satisfy the type constraints[45]. Thus, for e.g., a cell-peak edge we only sample negative candidates between cell and peak entities. This modification is crucial in our setting since most

randomly selected edges will be of invalid type (e.g., peak-peak), forcing the embeddings to primarily be optimized for irrelevant tasks (e.g., having low dot product between every pair of peaks).

Furthermore, it has been frequently observed that in graphs with wide distribution of node degrees, it is advantageous to sample negatives proportional to some function of the node degree to produce more informative embeddings that don't merely capture the degree distribution [12, 46]. For each graph edge in the dataset encountered in a training batch, we produce 100 negatives by corrupting the edge with a source or destination sampled uniformly from the nodes with the correct types for this relation and 100 by corrupting the edge with a source or destination node sampled with probability proportional to its degree[12].

As with many ML methods, graph embeddings are prone to overfitting in a low-data regime (i.e., low ratio of edges to parameters). We observed overfitting measurable as a gap between training and validation loss on the link prediction task, which we addressed with $L2$ regularization on the embeddings θ,

$$\mathcal{L}_{reg} = \mathcal{L} + \lambda \sum_{u \in N} \sum_{d=1}^{D} \theta_{ud}^2.$$

with λ =$wd * wd\_interval$. For weight decay parameter (wd), by default it is calculated automatically as $\frac{C}{N_e}$, where N_e is the training sample size (i.e., the total number of edges) and C is a constant. For weight decay interval ($wd\_interval$), we set it to 50 for all experiments.

We use the PyTorch-BigGraph framework, which provides efficient computation of multi-relation graph embeddings over multiple entity types and can scale to graphs with millions or billions of entities[12].

The resulting graph embeddings have two desirable properties that we will take advantage of:

1. First-order similarity: for two entity types T_1, T_2 with a relation between them, edges with high likelihood should have higher dot product; specifically, for any $u \in T_1$, the predicted probability distribution over edges to T_2 originating from u is approximated as $\frac{e^{x_u * x_v}}{\sum_{v' \in T_2} e^{x_u * x_{v'}}}$.
2. Second-order similarity: within a single entity type, entities that have 'similar contexts', i.e., a similar distribution of edge probabilities, should have similar embeddings. Thus, the embeddings of each entity type provide a low-rank latent space that encodes the similarity of those entities' edge distributions.

Evaluation of the model during training

During the PBG training procedure, a small percent of edges is held out (by default, the evaluation fraction is set to 5%) to monitor overfitting and evaluate the final model. Five metrics are computed on the reserved set of edges, including mean reciprocal rank (MRR, the average of the reciprocal of the ranks of all positives), R1 (the fraction of positives that rank better than all their negatives, i.e., have a rank of 1), R10 (the fraction of positives that rank in the top 10 among their negatives), R50 (the fraction of positives that rank in the top 50 among their negatives), and AUC (Area Under the Curve). By default, we show MRR along with training loss and validation loss while other metrics are also available in SIMBA package (Supplementary Fig. 1a). The learning curves for validation loss and these metrics can be used to determine when training has completed. The relative values of training and validation loss along with these evaluation metrics can be used to identify issues with training (underfitting vs overfitting) and tune the hyperparameters weight decay, embedding dimension, and number of training epochs appropriately. For example, in Supplementary Figure 1 training can be stopped once the validation loss plateaus. However, for most datasets we find that the default parameters do not need tuning.

Softmax transformation

PyTorch-BigGraph training provides initial embeddings of all entities (nodes). However, entities of different types (e.g., cells vs peaks, cells of different batches or modalities) have different edge distributions and thus may lie on different manifolds of the latent space. To make the embeddings of entities of different types comparable, we transform the embeddings of features with the Softmax function by utilizing the first-order similarity between cells (reference) and features (query). In the case of batch correction or multi-omics integration, the SoftMax transformation is also performed based on the first-order similarity between cells of different batches or modalities.

Given the initial embeddings of cells (reference) $(v_{c_1}, \dots, v_{c_n})$ and features $(v_{f_1}, \dots, v_{d_m})$, the model-estimated probability of an edge (c_i, f_j) obeys

$$P\left(v_{c_i,f_j}\right) \propto \exp\left(v_{c_i} \cdot v_{f_j}\right)$$

Therefore, if a random edge was sampled from feature f_j to a cell, the model would estimate the distribution over such edges as

$$p_{c_i,f_j} = \frac{\exp(v_{c_i} \cdot v_{f_j})}{\sum_{k=1}^{n} \exp(v_{c_k} \cdot v_{f_j})}$$

i.e., the Softmax weights between all cells $\{c_i\}$ and the feature f_j. We can then compute new embeddings for features as a linear combination of the cell embeddings weighted by the edge probabilities raised to some power.

$$\hat{v}_{f_j} = \frac{\sum_{i=1}^{n} p_{c_i,f_j}^{T^{-1}} v_{c_i}}{\sum_{i=1}^{n} p_{c_i,f_j}^{T^{-1}}}$$

T is a temperature hyperparameter that controls the sharpness of the weighting over cells. At $T = 1$, the cell embeddings are weighted by their estimated edge probabilities; at $T \to 0$, each feature embedding is assigned the cell embedding of its nearest neighbor; at $T \to \infty$, it becomes a discrete uniform distribution, and each query becomes the average of reference embeddings. We set $T = 0.5$ for all the analyses.

These steps are implemented in the function "si.tl.embed()" in the SIMBA package.

Metrics to assess cell-type specificity

Four metrics are proposed to assess the cell type specificity of each feature from different aspects, including max value (a higher value indicates higher cell-type specificity), Gini index (a higher value indicates higher cell-type specificity), standard deviation (a higher value indicates higher cell-type specificity), and entropy (a lower value indicates higher cell-type specificity). We observe these four metrics generally give consistent results. For SIMBA metric plot, by default, Gini index is plotted against max value. For feature f_j :

The max value is defined as the average normalized similarity of top k cells (by default, k=50). The similarity normalization function is defined as:

$$norm(x_i) = x_i - log\frac{\sum_{j=1}^{n} \exp(x_j)}{n}$$

Where $i = 1, \ldots, n$. n is the number of cells and x_i represents the dot product of $\hat{v}_{f_j}$ and the embedding of cell i.

The max value is computed as:

$$\max(f_j) = \frac{\sum_{i=1}^{k} norm(x_i)}{k}$$

The Gini index is computed as:

$$\text{gini}(\text{f}_\text{j}) = \frac{\sum_{i=1}^{n}(2i - n - 1) * p_{c_i,f_j}}{n\sum_{i=1}^{n} p_{c_i,f_j}}$$

The standard deviation is computed as:

$$\mathrm{std}(\mathrm{f_j}) = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(p_{c_i,f_j} - \mu)^2}$$

Where $\mu = \frac{1}{n}\sum_{i=1}^{n} p_{c_i,f_j} i$.

Entropy is computed as:

$$\mathrm{entropy}(\mathrm{f_j}) = -\sum_{i=1}^{n} p_{c_i,f_j} \log(p_{c_i,f_j})$$

Identification of master regulators

To identify master regulators, we take into consideration both the cell type specificity of each pair of TF motif and TF gene and the distance between them. More specifically, for each TF motif, first its distances (Euclidean distance by default) to all the genes are calculated in the SIMBA embedding space. Then the rank of this TF gene among all these genes is computed. In addition, we also assess the cell type specificity of this pair of TF motif and TF gene based on SIMBA metrics (by default, max value and Gini index are used). The same procedure is performed for all TFs. Finally, we identify master regulators by filtering out TFs with low cell-type specificity and scoring them based on TF gene rank. This procedure is implemented in the function "st.tl.find_master_regulators()" in SIMBA package.

Identification of TF target genes

Given a master regulator, its target genes are identified by comparing the locations of the TF gene, TF motif, and the peaks near the genomic loci of candidate target genes in the SIMBA co-embedding space (Fig. 4e). More specifically we first search for k nearest neighbor genes around the motif (TF motif) and the gene (TF gene) of this master regulator, respectively (k = 200 by default). The union of these neighbor genes is the initial set of candidate target genes. These genes are then filtered based on the criterion that open regions (peaks) within 100kb upstream and downstream of the TSS of a putative target gene must contain the TF motif.

Next, for each candidate target gene, we compute four types of distances in SIMBA embedding space: distances between the embeddings of 1) the candidate target gene and TF gene; 2) the candidate target gene and TF motif; 3) peaks near the genomic locus of the candidate target gene and TF motif; 4) peaks near the genomic locus of the candidate target gene and the candidate gene. All the distances (Euclidean distances by default) are converted to ranks out of all genes or all peaks to make the distances comparable across different master regulators.

The final list of target genes is decided using the calculated ranks based on two criteria: 1) at least one of the nearest peaks to TF gene or TF motif is within a predetermined range (top 1,000 by default); 2) the average rank of the candidate target gene is within a predetermined range (top 5,000 by default). This procedure is implemented in the function "st.tl. find_target_genes ()" in SIMBA.

Benchmarking scATAC-seq computational methods

To compare SIMBA to other scATAC-seq computational methods including SnapATAC [4], Cusanovich2018[20], and cisTopic[21], we employed the previously developed benchmarking framework from Chen et al[13]. This framework evaluates different methods based on their ability to distinguish cell types. We applied three clustering algorithms: k-means clustering, hierarchical clustering, and Louvain on the feature matrix derived from each method.

For datasets with ground-truth (FACS-sorted labels or known tissue labels), including simulated bone marrow data, Buenrostro 2018, and sci-ATAC-seq subset, three metrics including Adjusted Rand Index (ARI), Adjusted Mutual Information (AMI), and Homogeneity are applied to evaluate the performance. ARI measures the similarity between two clusters, comparing all pairs of samples assigned to matching or different clusters in the predicted clustering solution vs the true cluster/cell type label. AMI describes an observed frequency of co-occurrence compared to an expected frequency of co-occurrence between two variables, informing the mutual dependence or strength of association of these two variables. Homogeneity measures whether a clustering algorithm preserves cluster assignments towards samples that belong to a single class. A higher metric value indicates a better clustering solution.

For 10x PBMCs dataset with no ground truth, the Residual Average Gini Index (RAGI) proposed in the benchmarking study[13] is used as the clustering evaluation metric. RAGI measures the relative exclusivity of marker genes to their corresponding clusters in comparison to housekeeping genes, which should demonstrate low specificity to any given cluster. In brief, the mean Gini Index is computed for both marker genes and housekeeping genes. The difference between the means is computed to obtain the average residual specificity (i.e., RAGI) of a clustering solution with respect to marker genes. A higher RAGI indicates a better separation of biologically distinct clusters.

Benchmarking single-cell batch correction methods

The batch correction performance of SIMBA was compared to Seurat v3[11], LIGER[10] and Harmony[9] in two benchmark datasets: the mouse atlas dataset and the human pancreas dataset (see Supplementary Table 1). For Seurat3, LIGER and Harmony, the batch correction was done with the same parameters used in a previous benchmark study[26].

To evaluate the batch integration performance, average Silhouette width (ASW), adjusted Rand index (ARI), and local inverse Simpson's index (LISI)[9] were calculated for the batches and cell types using the Euclidean distance as described in a previous benchmark[26]. To make a fair evaluation, only the cell types that are present in all batches were considered. We used the same number of dimensions (50) for these methods and all other parameters were set as in the benchmark.

Average Silhouette width (ASW)

Average Silhouette width is the mean value of Silhouette scores calculated from each cell. Silhouette width measures the relative closeness of cells with the same label compared to the cells with the different label and ranges from -1 to +1. Silhouette score for a data point with a label is calculated as

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}$$

where $a(i)$ is the distance to the closest point with the same label, and $b(i)$ is the distance to the closest point with different labels. A high Silhouette score means the point is located more closely with the same label, where a low Silhouette score closer to -1 means the point is located closer with different labels than that of itself. The ideal batch correction result will give a low ASW score for batch labels as the point is well mixed with other batches and a high ASW score for the cell type labels as the cells of the same cell type should cluster together after the batch correction. The final score is calculated as the median ASW scores from 20 subsets of randomly sampled 80% cells.

Average Rand Index (ARI)

To evaluate the cell type purity, the true cell type labels and the k-means clustering solution were used to calculate the cell type ARI. To evaluate the batch correction performance, the true batch labels and the k-means clustering solution were used to calculate the batch ARI. The final ARI was calculated as the median ARI scores of 20 subsets comprised of randomly sampled 80% cells for batches and cell types, respectively. A superior batch correction will have a high cell type ARI (high agreement between the clustering solution and the true cell type labels), and a low batch ARI (the clustering solution is not mainly driven by batches and clusters contain cells with well-mixed batch labels).

Local Inverse Simpson's Index (LISI)

Local Inverse Simpson's Index (LISI) [9] measures the local batch and cell type mixing. For each data point, it considers the Gaussian kernel weighted distribution of labels in its neighborhood with a perplexity argument. We set perplexity to 50 40 as in the previous benchmark study. Using the weighted neighborhood label distribution, the inverse

Simpson's index is calculated as $\frac{1}{\sum_l p(l)}$ where l is the batch or cell type labels and $p(l)$ is the probability of each label in the local neighborhood obtained with the kernel. For each cell, the LISI is the expected number of cells to be sampled locally before a cell of the same label is sampled. A perfect batch correction will have a cell type LISI (cLISI) of 1 and a batch LISI (integration LISI, iLISI) close to the number of batches. The final LISI score was calculated as the average LISI scores of all cells.

Benchmarking single cell multi-omics integration methods

Two pairs of scRNA-seq and scATAC-seq datasets manually split from the dual-omics SHARE-seq mouse skin dataset and 10X PBMCs dataset respectively were used for the modality integration task. For Seurat and LIGER, the parameters and preprocessing were done as described in their documentations. However for the LIGER analysis of the SHARE-seq mouse skin dataset the parameter 'lambda' was set to 30 and the 'ref_dataset' was set to scATAC-seq to get a better alignment. For the Raw results, the activity matrix of scATAC-seq was constructed using Seurat and the first 20 PCs of the scRNA-seq count matrix and the activity matrix were used for the comparison. The integration results generated by each method were evaluated with four metrics—Anchoring distance, anchoring distance rank, Silhouette index, and cluster agreement—as described below.

Anchoring distance

The Anchoring distance was proposed in Dou et al., 2020[47] and is the normalized distance between the matched cells of two modalities (e.g. RNA and ATAC). Here we considered the Euclidean distance and normalized the distance by the mean of the distances calculated between random pairs of cells. The number of pairs randomly sampled was set to 10% of the total number of cells.

Anchoring distance rank

Given that the anchoring distance does not account for the local density of cells, we propose a new metric entitled *anchoring distance rank* (ADR). The ADR is based on the normalized rank of the distance between the matched cells of two modalities. For each cell x_{ij} with cell identity i and modality j, the distance between the cell and all the other cells of the other modality j', $d(x_{ij}, x_{kj'}), k = 1, \dots, N$ is calculated, where N is the total number of cells. Then the rank of $r_i = d(x_{ij}, x_{ij'})$ within the calculated distances is normalized by the number of pairs $N - 1$ to obtain the final anchoring rank $m_i = \frac{r_i - 1}{N - 1}$. For each cell, an anchoring rank of 0 indicates an ideal modality integration performance as the matched cells are closest to each other in the embedding.

Silhouette index

The silhouette index was calculated as described in 10) based on the cluster assignment wherein each cluster consists of two cells, one cell from a scRNA-seq dataset and one cell from a scATAC-seq dataset.

Fraction in the same cluster

Fraction in the same cluster was calculated as the fraction of the matched cells from two modalities in the same cluster. The clusters of cells were generated using Louvain algorithm and the number of clusters is equal to the number of cell types in the dataset.

Data availability:

All the datasets used in this study (eight scRNA-seq datasets, four scATAC-seq datasets, and three dual-omics datasets) are summarized in Supplementary Table 1. All these datasets are curated in the SIMBA package, and they can be easily downloaded and imported directly to reproduce the analyses presented in this manuscript.

Code availability:

We provide a comprehensive Python package 'simba' available at https://anaconda.org/bioconda/simba and https://github.com/pinellolab/simba. All the proposed procedures are implemented in the "simba" package. 'simba' can be easily installed with conda "*conda install -c bioconda simba*". We also built a website (https://simba-bio.readthedocs.io), providing a detailed introduction of the 'simba' software and several SIMBA tutorials for different types of single-cell analyses presented in this manuscript.

Acknowledgements

The authors thank Ledell (Yu) Wu, Facebook, for the helpful discussions about Starspace; Dr.Sai Ma and Dr. Jason Buenrostro, Broad Institute of MIT and Harvard, for sharing data and metadata. The authors also acknowledge members of the Pinello lab, Mass General Hospital/Harvard Medical School, for helpful comments and feedback. This project has been made possible in part by grant number 2019-202669 from the Chan Zuckerberg Foundation to LP. LP is also partially supported by the National Human Genome Research Institute (NHGRI) Genomic Innovator Award (R35HG010717).

Author contributions

H.C. and L.P. conceived this project. H.C. designed SIMBA, wrote the SIMBA package, and performed SIMBA analysis on all datasets. A.L. contributed to the adaption of PyTorch-BigGraph to single cell analysis. J.R. and H.C. performed the comparison analysis on batch correction and data integration. M.V. and H.C. performed the comparison analysis on scATAC-seq data. L.P.

and A.L. provided guidance and supervised this project. All the authors wrote and approved the final manuscript.

Competing interests

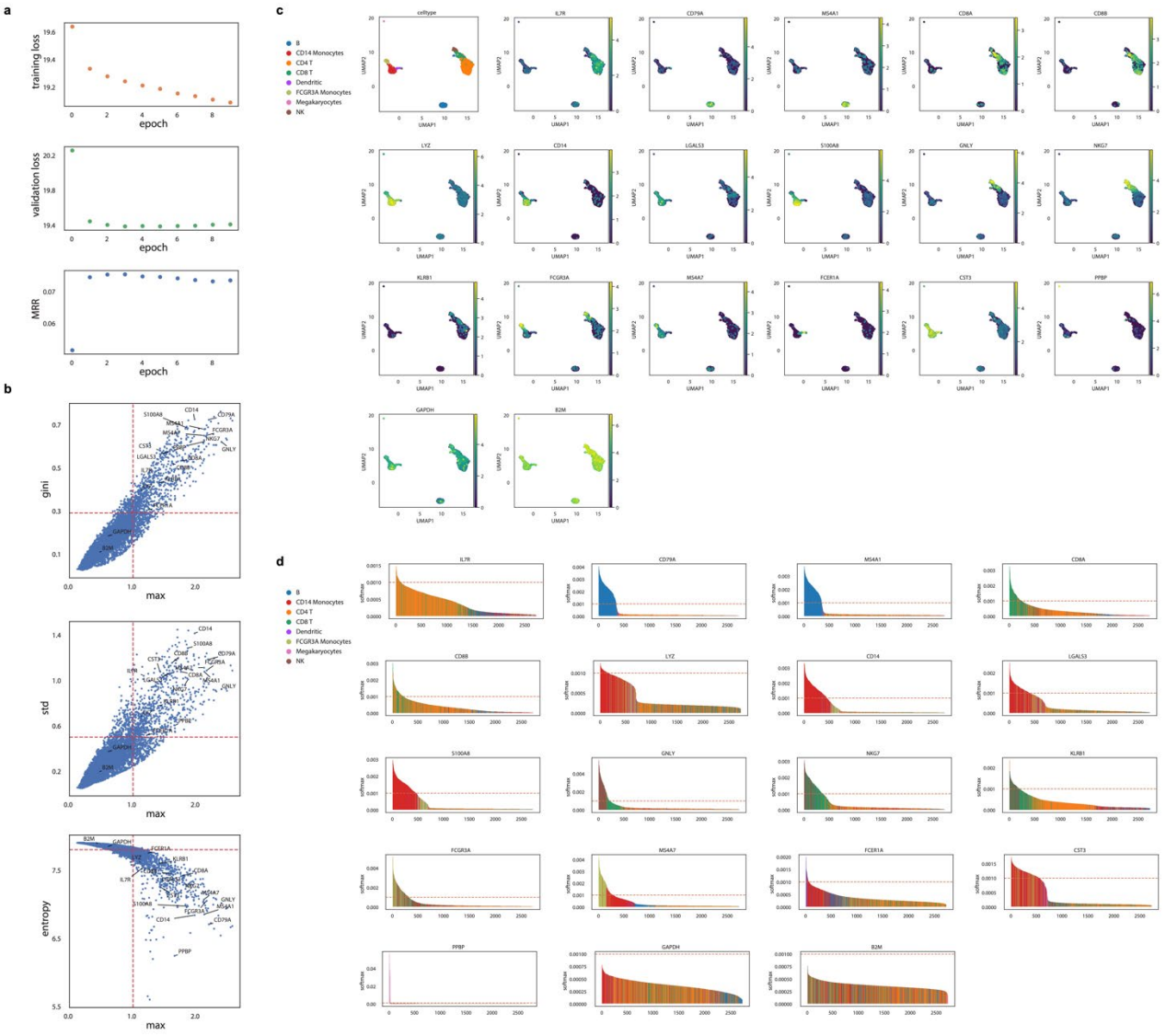
The authors declare that they have no competing interests.

References:

1. Satija, R., Farrell, J.A., Gennert, D., Schier, A.F. & Regev, A. Spatial reconstruction of single-cell gene expression data. *Nat Biotechnol* **33**, 495-502 (2015).
2. Wolf, F.A., Angerer, P. & Theis, F.J. SCANPY: large-scale single-cell gene expression data analysis. *Genome Biol* **19**, 15 (2018).
3. Granja, J.M. et al. ArchR is a scalable software package for integrative single-cell chromatin accessibility analysis. *Nat Genet* **53**, 403-411 (2021).
4. Fang, R. et al. Comprehensive analysis of single cell ATAC-seq data with SnapATAC. *Nat Commun* **12**, 1337 (2021).
5. Cortal, A., Martignetti, L., Six, E. & Rausell, A. Gene signature extraction and cell identity recognition at the single-cell level with Cell-ID. *Nat Biotechnol* (2021).
6. Vandenbon, A. & Diez, D. A clustering-independent method for finding differentially expressed genes in single-cell transcriptome data. *Nat Commun* **11**, 4318 (2020).
7. Dann, E., Henderson, N.C., Teichmann, S.A., Morgan, M.D. & Marioni, J.C. Differential abundance testing on single-cell data using k-nearest neighbor graphs. *Nat Biotechnol* (2021).
8. Hao, Y. et al. Integrated analysis of multimodal single-cell data. *Cell* (2021).
9. Korsunsky, I. et al. Fast, sensitive and accurate integration of single-cell data with Harmony. *Nat Methods* **16**, 1289-1296 (2019).
10. Welch, J.D. et al. Single-Cell Multi-omic Integration Compares and Contrasts Features of Brain Cell Identity. *Cell* **177**, 1873-1887 e1817 (2019).
11. Stuart, T. et al. Comprehensive Integration of Single-Cell Data. *Cell* **177**, 1888-1902 e1821 (2019).
12. Lerer, A. et al. Pytorch-biggraph: A large-scale graph embedding system. *arXiv preprint arXiv:1903.12287* (2019).
13. Chen, H. et al. Assessment of computational methods for the analysis of single-cell ATAC-seq data. *Genome Biology* **20**, 241 (2019).
14. Buenrostro, J.D. et al. Integrated Single-Cell Analysis Maps the Continuous Regulatory Landscape of Human Hematopoietic Differentiation. *Cell* **173**, 1535-1548 e1516 (2018).
15. Ferreira, R., Ohneda, K., Yamamoto, M. & Philipsen, S. GATA1 function, a paradigm for transcription factors in hematopoiesis. *Molecular and cellular biology* **25**, 1215-1227 (2005).
16. Tijchon, E., Havinga, J., Van Leeuwen, F. & Scheijen, B. B-lineage transcription factors and cooperating gene lesions required for leukemia development. *Leukemia* **27**, 541-552 (2013).

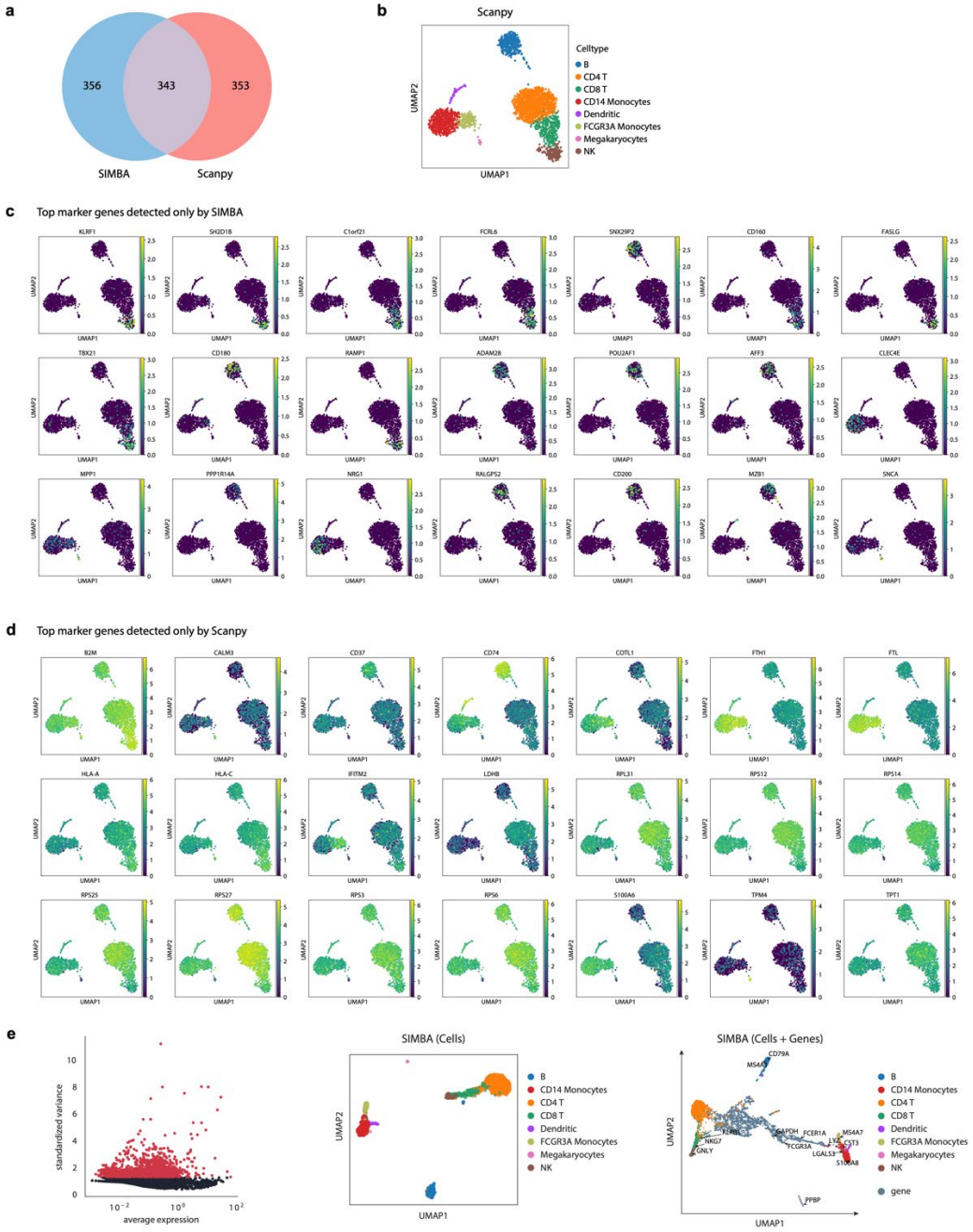
17. Friedman, A. Transcriptional control of granulocyte and monocyte development. *Oncogene* **26**, 6816-6828 (2007).
18. Schep, A.N., Wu, B., Buenrostro, J.D. & Greenleaf, W.J. chromVAR: inferring transcription-factor-associated accessibility from single-cell epigenomic data. *Nat Methods* **14**, 975-978 (2017).
19. Moriguchi, T. & Yamamoto, M. A regulatory network governing Gata1 and Gata2 gene transcription orchestrates erythroid lineage differentiation. *International journal of hematology* **100**, 417-424 (2014).
20. Cusanovich, D.A. et al. The cis-regulatory dynamics of embryonic development at single-cell resolution. *Nature* **555**, 538-542 (2018).
21. Bravo Gonzalez-Blas, C. et al. cisTopic: cis-regulatory topic modeling on single-cell ATAC-seq data. *Nat Methods* **16**, 397-400 (2019).
22. Chen, S., Lake, B.B. & Zhang, K. High-throughput sequencing of the transcriptome and chromatin accessibility in the same cell. *Nat Biotechnol* (2019).
23. Ma, S. et al. Chromatin Potential Identified by Shared Single-Cell Profiling of RNA and Chromatin. *Cell* (2020).
24. Cao, J. et al. Joint profiling of chromatin accessibility and gene expression in thousands of single cells. *Science* **361**, 1380-1385 (2018).
25. Zhu, C. et al. An ultra high-throughput method for single-cell joint analysis of open chromatin and transcriptome. *Nat Struct Mol Biol* (2019).
26. Tran, H.T.N. et al. A benchmark of batch-effect correction methods for single-cell RNA sequencing data. *Genome Biol* **21**, 12 (2020).
27. Leek, J.T. et al. Tackling the widespread and critical impact of batch effects in high-throughput data. *Nature Reviews Genetics* **11**, 733-739 (2010).
28. Han, X. et al. Mapping the mouse cell atlas by microwell-seq. *Cell* **172**, 1091-1107. e1017 (2018).
29. Tabula Muris, C. et al. Single-cell transcriptomics of 20 mouse organs creates a Tabula Muris. *Nature* **562**, 367-372 (2018).
30. Baron, M. et al. A single-cell transcriptomic map of the human and mouse pancreas reveals inter-and intra-cell population structure. *Cell systems* **3**, 346-360. e344 (2016).
31. Muraro, M.J. et al. A single-cell transcriptome atlas of the human pancreas. *Cell systems* **3**, 385-394. e383 (2016).
32. Wang, Y.J. et al. Single-cell transcriptomics of the human endocrine pancreas. *Diabetes* **65**, 3028-3038 (2016).
33. Segerstolpe, Å. et al. Single-cell transcriptome profiling of human pancreatic islets in health and type 2 diabetes. *Cell metabolism* **24**, 593-607 (2016).
34. Palla, G. et al. Squidpy: a scalable framework for spatial single cell analysis. *bioRxiv 2021.02.19.431994v2* (2021).
35. VanHorn, S. & Morris, S.A. Next-Generation Lineage Tracing and Fate Mapping to Interrogate Development. *Dev Cell* **56**, 7-21 (2021).
36. Ietswaart, R., Gyori, B.M., Bachman, J.A., Sorger, P.K. & Churchman, L.S. GeneWalk identifies relevant gene functions for a biological context using network representation learning. *Genome Biol* **22**, 55 (2021).

37. Yuan, H., Kshirsagar, M., Zamparo, L., Lu, Y. & Leslie, C.S. BindSpace decodes transcription factor binding signals by large-scale sequence embedding. *Nat Methods* (2019).
38. Li, H., Xiao, X., Wu, X., Ye, L. & Ji, G. scLINE: A multi-network integration framework based on network embedding for representation of single-cell RNA-seq data. *J Biomed Inform* **122**, 103899 (2021).
39. Buterez, D., Bica, I., Tariq, I., Andrés-Terré, H. & Liò, P. CELLVGAE: AN UNSUPERVISED SCRNA-SEQ ANALYSIS WORKFLOW WITH GRAPH ATTENTION NETWORKS. *bioRxiv 2020.12.20.423645v1* (2020).
40. Sheikh, N., Kefato, Z. & Montresor, A. gat2vec: representation learning for attributed graphs. *Computing* **101**, 187-209 (2018).
41. Bojchevski, A. & Günnemann, S. Deep gaussian embedding of graphs: Unsupervised inductive learning via ranking. *arXiv preprint arXiv:1707.03815* (2017).
42. Satopaa, V., Albrecht, J., Irwin, D. & Raghavan, B. in 2011 31st international conference on distributed computing systems workshops 166-171 (IEEE, 2011).
43. Fornes, O. et al. JASPAR 2020: update of the open-access database of transcription factor binding profiles. *Nucleic acids research* **48**, D87-D92 (2020).
44. Kadlec, R., Bajgar, O. & Kleindienst, J. Knowledge base completion: Baselines strike back. *arXiv preprint arXiv:1705.10744* (2017).
45. Krompaß, D., Baier, S. & Tresp, V. in International semantic web conference 640-655 (Springer, 2015).
46. Mikolov, T., Chen, K., Corrado, G. & Dean, J. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781* (2013).
47. Dou, J. et al. Unbiased integration of single cell multi-omics data. *bioRxiv*, 2020.2012.2011.422014 (2020).



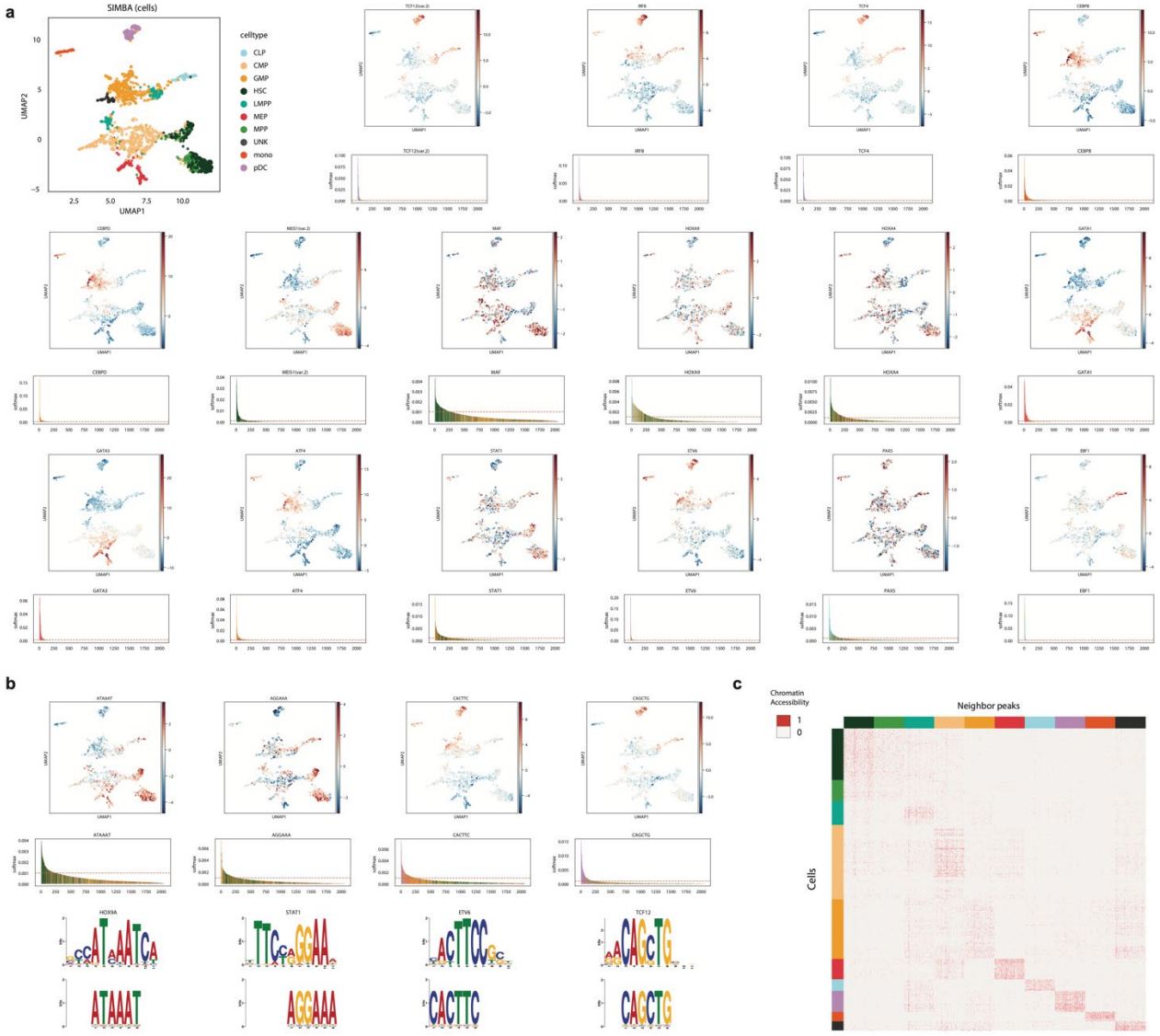
Supplementary Figure 1. SIMBA analysis of the scRNA-seq 10x PBMCs dataset.

- **a.** Three default metrics used to evaluate SIMBA training procedure, including training loss (top), validation loss (middle), mean reciprocal rank (MRR)
- **b.** SIMBA metric plots of genes. All the genes are plotted according to the Gini index against max score, standard deviation (std) against max score, and entropy against max score, respectively. The same set of genes as in Figure 2c are highlighted.
- **c.** UMAP visualization of SIMBA embeddings of cells colored by cell type or gene expression of those genes highlighted in (b).
- **d.** SIMBA barcode plots of the genes highlighted in (b).



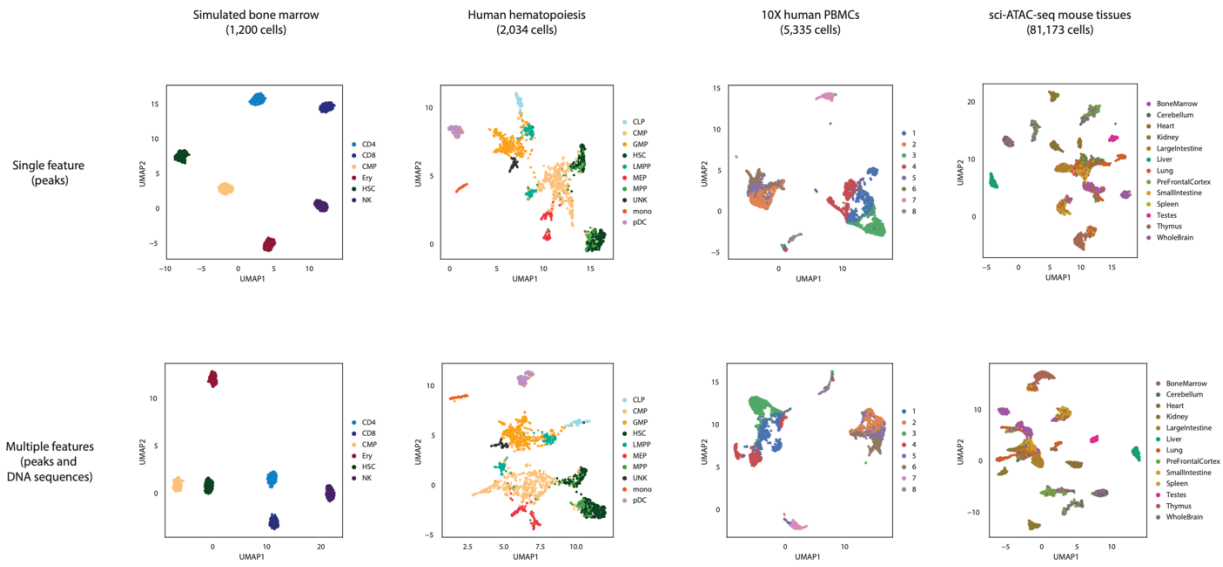
Supplementary Figure 2. Comparison of SIMBA with Scanpy on the scRNA-seq 10x PBMCs dataset.

- **a.** Venn diagram of top marker genes identified by SIMBA and Scanpy
- **b.** Scanpy-derived UMAP visualization of cells colored by cell type
- **c.** Top marker genes detected only by SIMBA. Colored by intensity of gene expression.
- **d.** Top marker genes detected only by Scanpy. Colored by intensity of gene expression.
- **e.** SIMBA embedding result after implementing variable gene selection. Left: variable gene selection step implemented in SIMBA. Middle: UMAP visualization of SIMBA embeddings of cells. Right: UMAP visualization of SIMBA embeddings of cells and variable genes.



Supplementary Figure 3. SIMBA analysis of the *Buenrostro2018* dataset

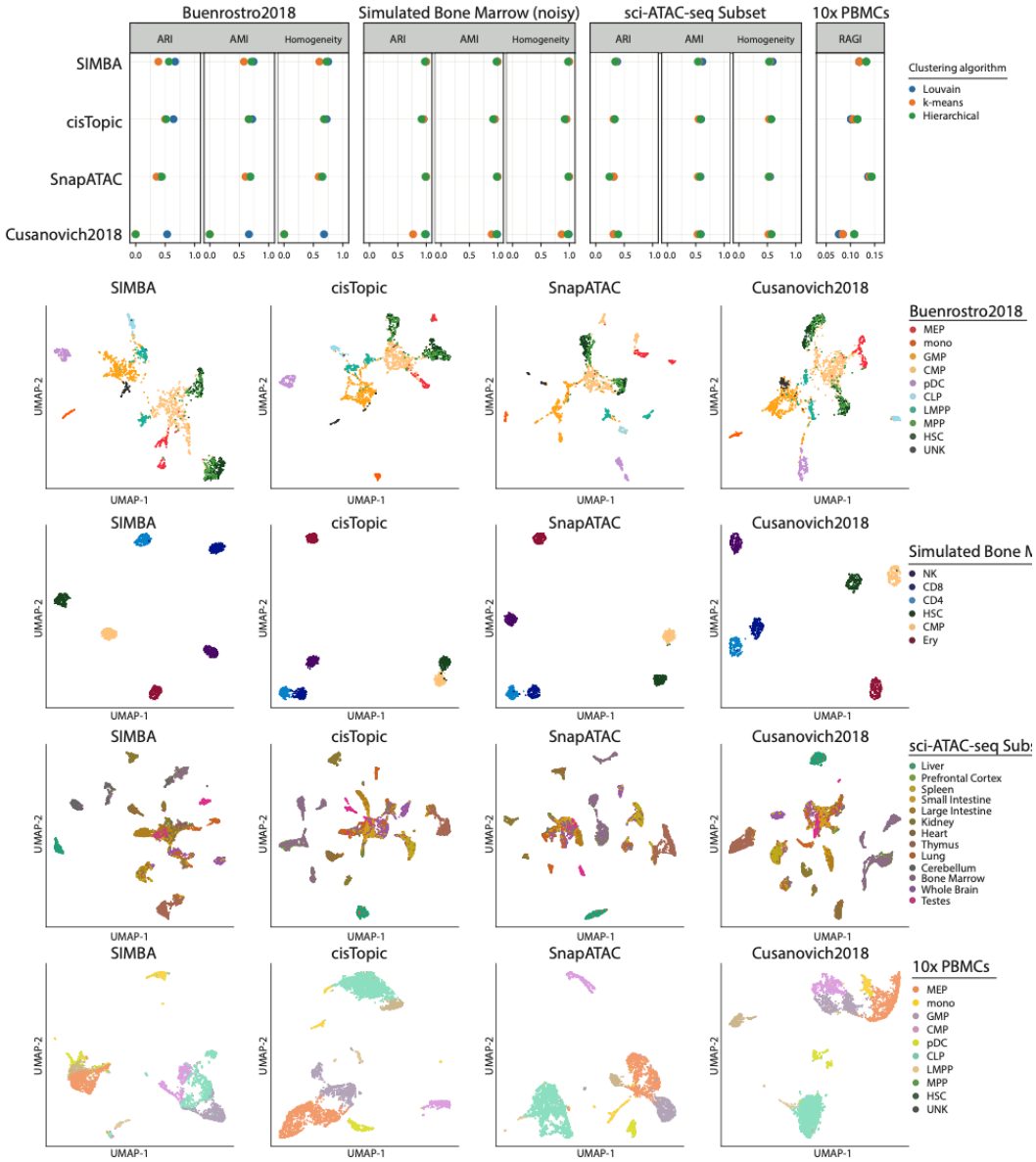
- **a.** UMAP visualization of SIMBA embeddings of cells colored by cell type (top-left), and TF activity scores of TF motifs calculated with chromVAR, respectively. The SIMBA barcode plot of each TF motif is shown below the UMAP plot.
- **b.** Top: UMAP visualization of SIMBA embeddings of cells colored by TF activity scores of k-mers calculated with chromVAR. Middle: SIMBA barcode plots of the corresponding k-mers. Bottom: the matching known motif against the enriched k-mer sequences.
- **c.** Heatmap of cells against neighboring peaks of each cell type that are selected in the SIMBA co-embedding space. Chromatin accessibility is binary and colored accordingly.



Supplementary Figure 4. Comparison of SIMBA performance using scATAC-seq peaks and DNA sequence content vs only scATAC-seq peaks.

Top: UMAP visualization of SIMBA embeddings of cells for each indicated dataset generated from only scATAC-seq peak information.

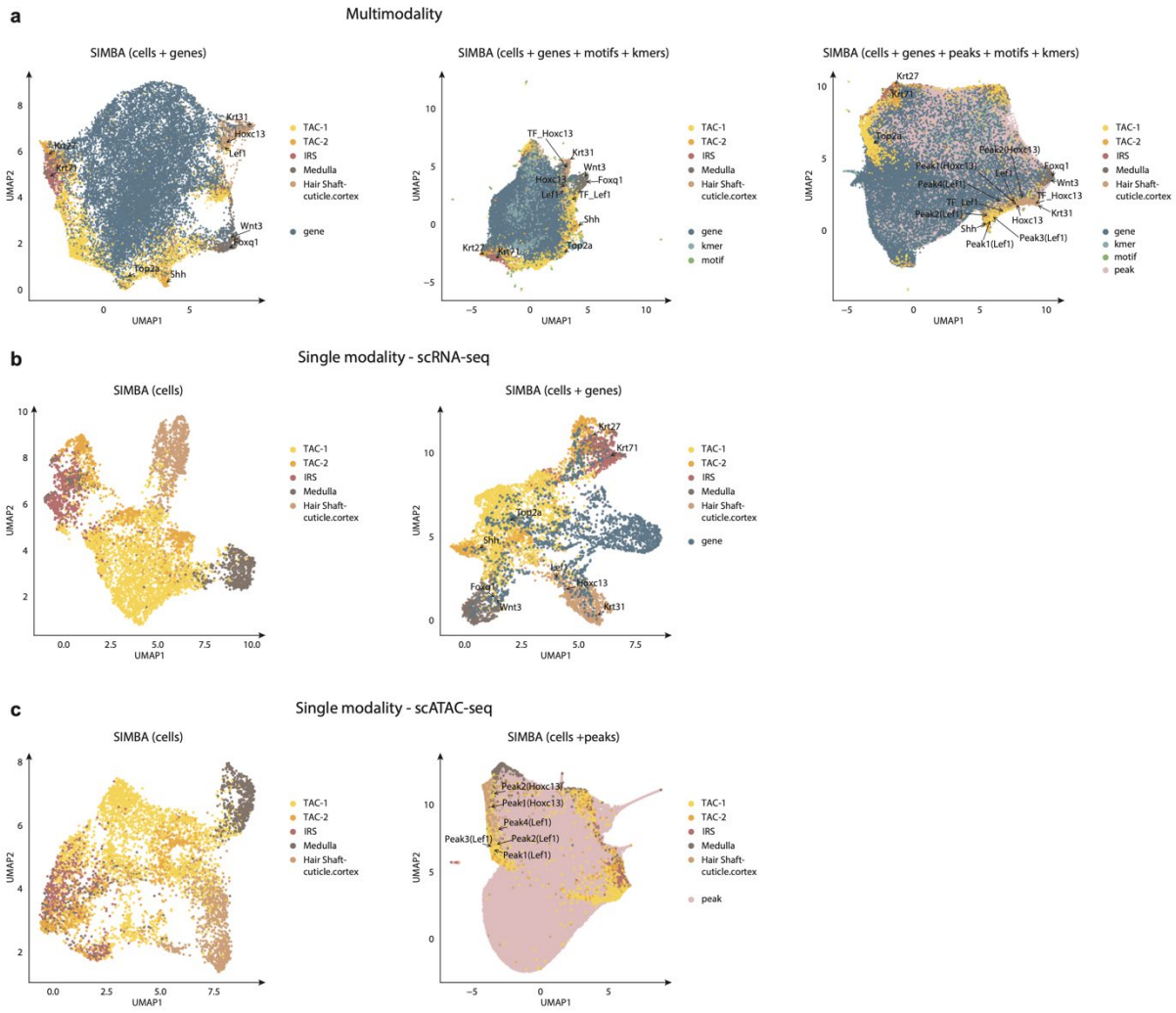
Bottom: UMAP visualization of SIMBA embeddings of cells for each indicated dataset generated from scATAC-seq peak information and DNA sequence content information.



Supplementary Figure 5. Benchmark of SIMBA against top-performing scATAC-seq analysis methods.

Top: Evaluation of SIMBA and other methods including cisTopic, SnapATAC, *Cusanovich2018* for scATAC-seq analysis using metrics 1) ARI, AMI, and Homogeneity for datasets with ground truth cell type labels and 2) Residual Average Gini Index (RAGI) for the 10x PBMCs dataset without ground truth labels.

Bottom: UMAP visualization of feature matrices produced by each method on each dataset colored by cell type annotation or cluster label.

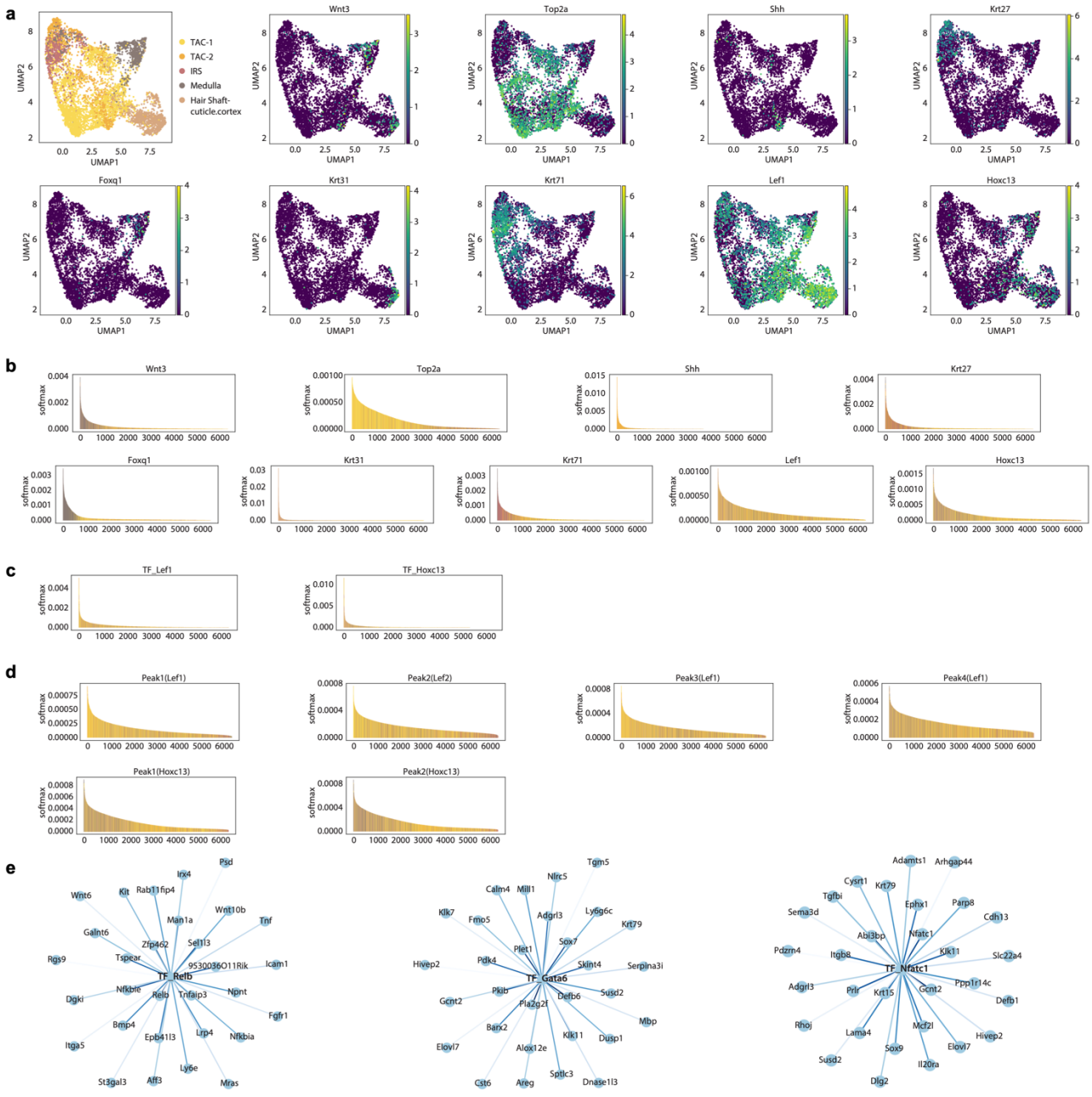


Supplementary Figure 6. SIMBA multimodal analysis of the SHARE-seq hair follicle dataset.

a. SIMBA embedding results when both gene expression and chromatin accessibility are encoded in the graph. Left: UMAP visualization of SIMBA embeddings of cells and genes. Middle: UMAP visualization of SIMBA embeddings of cells along with genes, TF motifs, and k-mers. Right: UMAP visualization of SIMBA embeddings of cells along with genes, peaks, TF motifs, and k-mers.

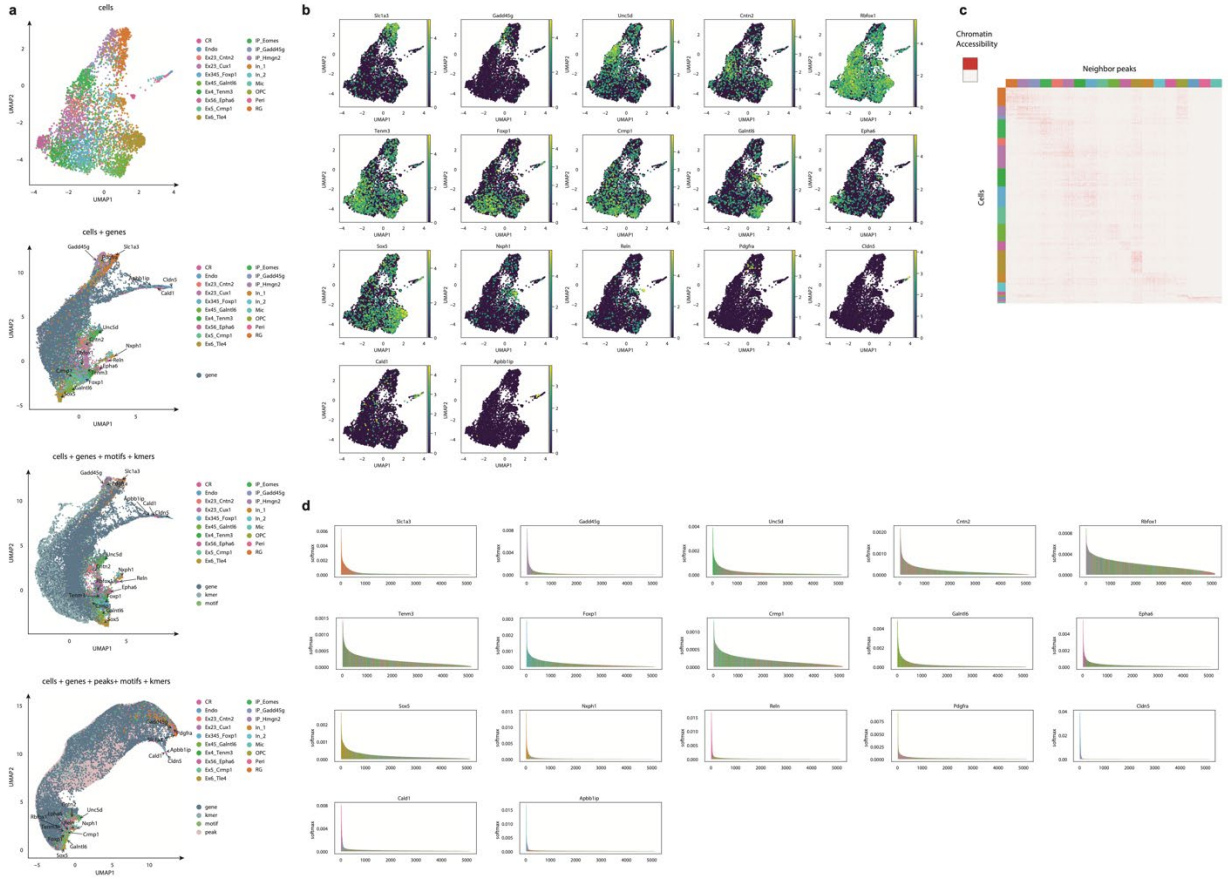
b. SIMBA embedding results when only gene expression is encoded in the graph. Left: UMAP visualization of SIMBA embeddings of cells. Right: UMAP visualization of SIMBA embeddings of cells and variable genes.

c. SIMBA embedding results when only chromatin accessibility is encoded in the graph. Left: UMAP visualization of SIMBA embeddings of cells. Right: UMAP visualization of SIMBA embeddings of cells and peaks.



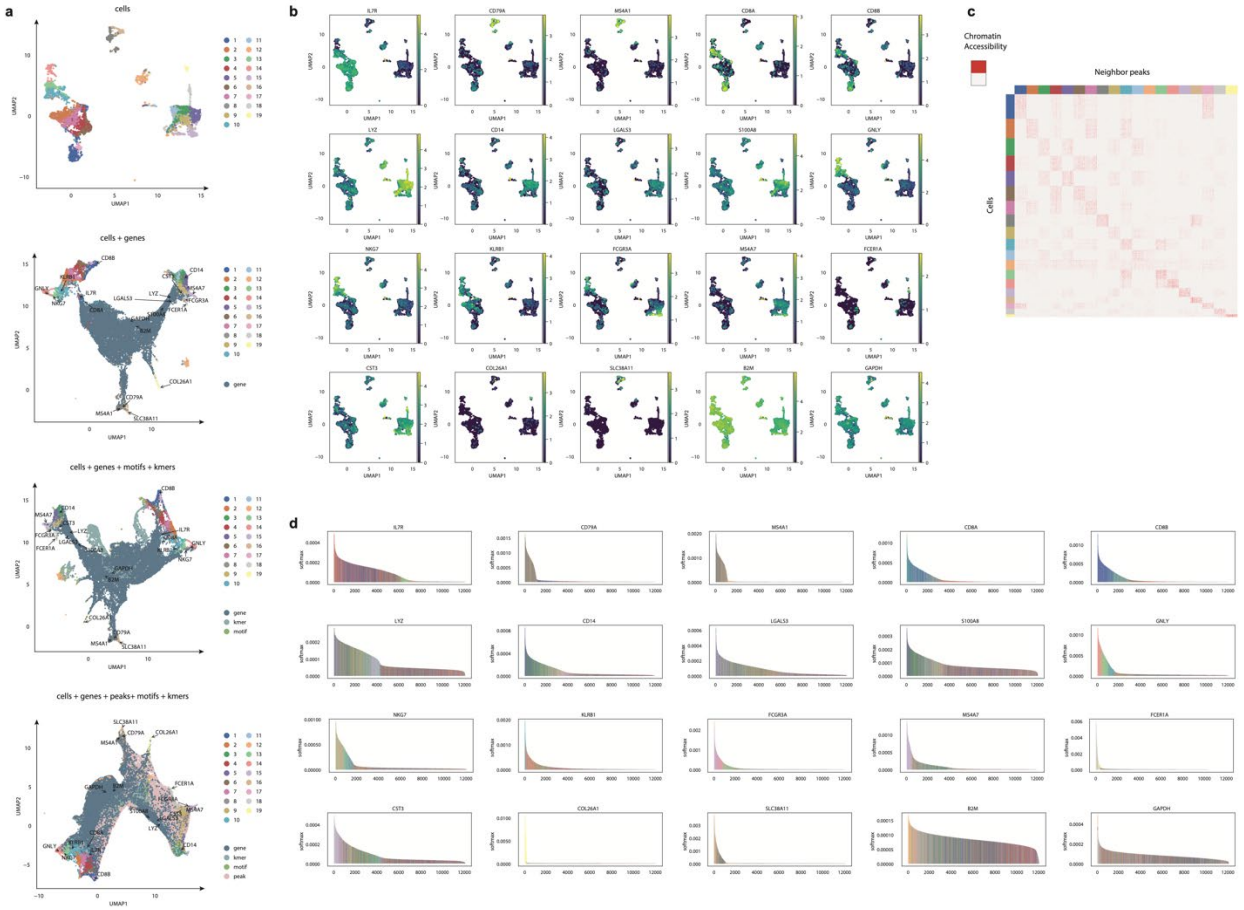
Supplementary Figure 7. Cell type specific marker genes and the target genes of master regulators identified by SIMBA in the SHARE-seq hair follicle subset dataset.

- **a.** UMAP visualization of SIMBA embeddings of cells colored by cell type and gene expression intensity.
- **b.** SIMBA barcode plots of each gene plotted above.
- **c.** SIMBA barcode plots of TF motifs *Lef1* and *Hoxc13.*
- **d.** SIMBA barcode plots of peaks near the loci of *Lef1* and *Hoxc13.*
- **e.** Top 30 target genes of the master regulators *Relb, Gata6,* and *Nfatc1* as inferred by SIMBA.



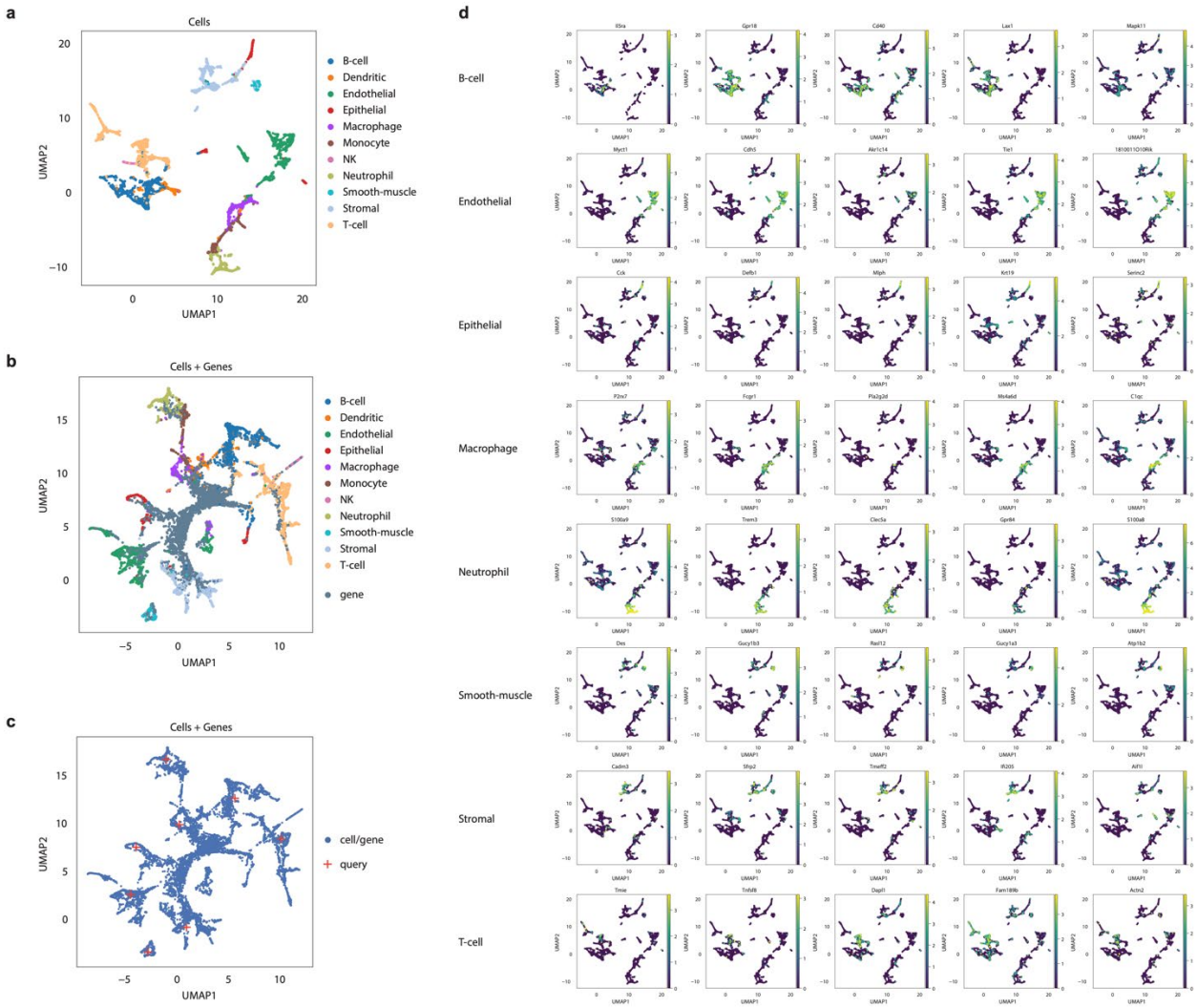
Supplementary Figure 8. SIMBA multimodal analysis of the SNARE-seq mouse cerebral cortex dataset.

- **a.** From top to bottom: UMAP visualization of SIMBA embeddings of (i) cells (ii) genes alongside cells (iii) genes, motifs, and k-mers alongside cells (iv) genes, peaks, motifs, and k-mers alongside cells.
- **b.** UMAP visualization of SIMBA embeddings of cells colored by indicated gene expression intensity.
- **c.** Heatmap of cells against neighboring peaks of each cell type that are selected in the SIMBA co-embedding space. Chromatin accessibility is binary and colored accordingly.
- **d.** SIMBA barcode plots of the genes highlighted in (a).



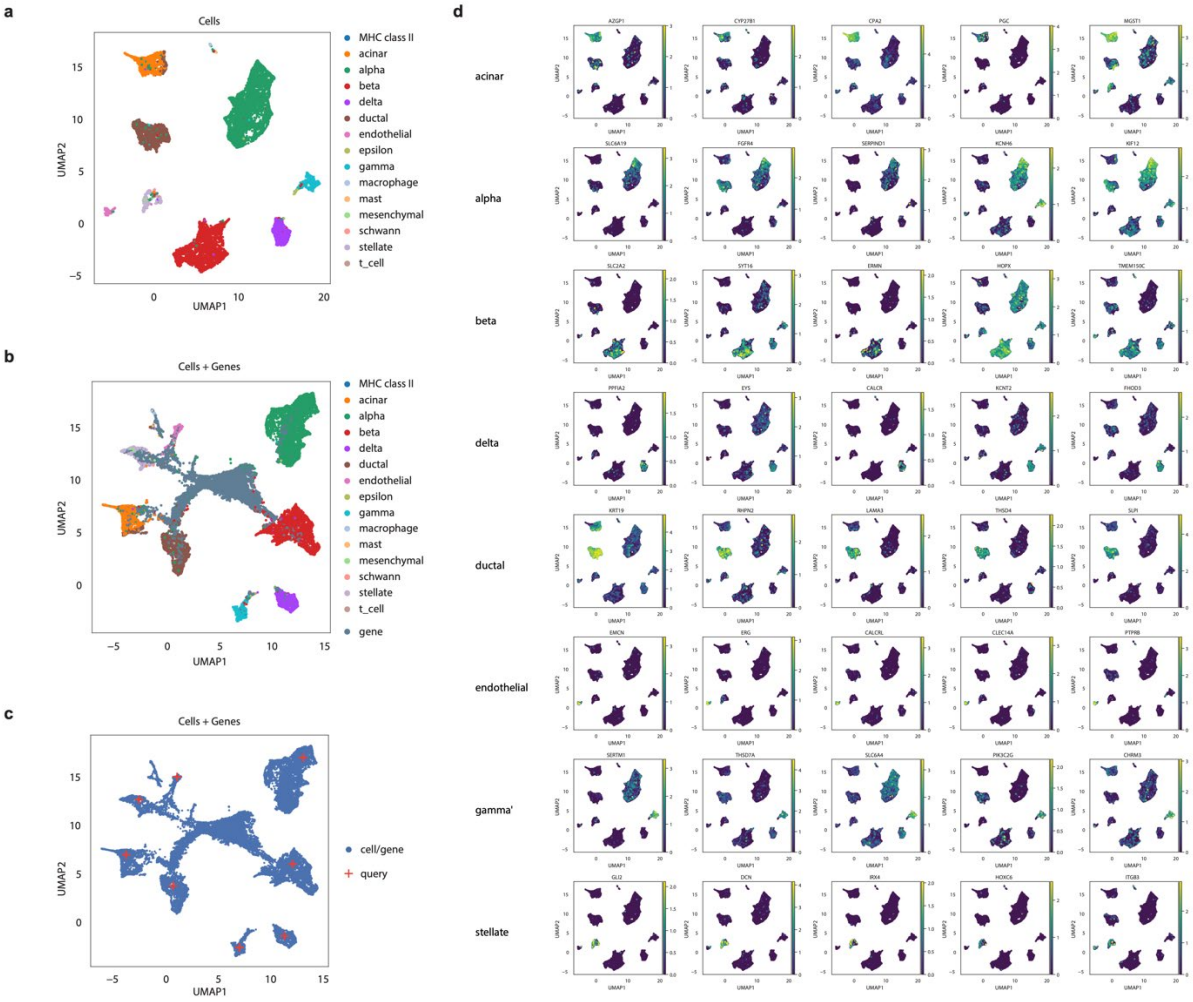
Supplementary Figure 9. SIMBA multimodal analysis of the 10x multiome PBMCs dataset.

- **a.** From top to bottom: UMAP visualization of SIMBA embeddings of (i) cells (ii) genes alongside cells (iii) genes, motifs, and k-mers alongside cells (iv) genes, peaks, motifs, and k-mers alongside cells.
- **b.** UMAP visualization of SIMBA embeddings of cells colored by indicated gene expression intensity.
- **c.** Heatmap of cells against neighboring peaks of each cluster that are selected in the SIMBA co-embedding space. Chromatin accessibility is binary and colored accordingly.
- **d.** SIMBA barcode plots of the genes highlighted in (a).



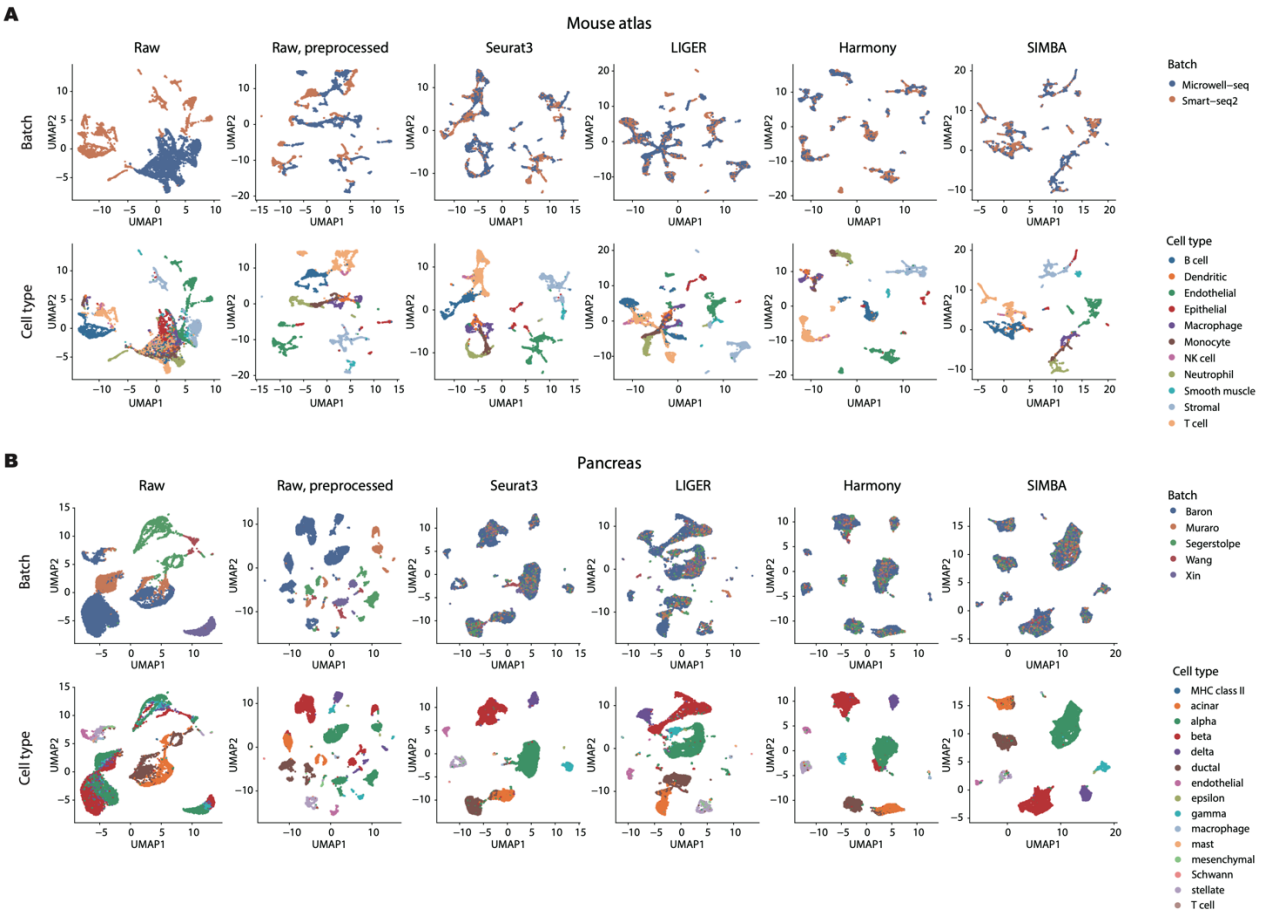
Supplementary Figure 10. SIMBA-inferred marker genes for the scRNA-seq mouse atlas dataset in batch correction analysis.

- **a.** UMAP visualization of SIMBA embeddings of cells colored by cell type.
- **b.** UMAP visualization of SIMBA embeddings of cells and genes.
- **c.** UMAP visualization of SIMBA embeddings of cells and genes. Biological “query” points are highlighted with a red “+”. Nearby informative genes are colored accordingly.
- **d.** UMAP visualization of SIMBA embeddings of cells colored by indicated gene expression intensity, separated by cell type.



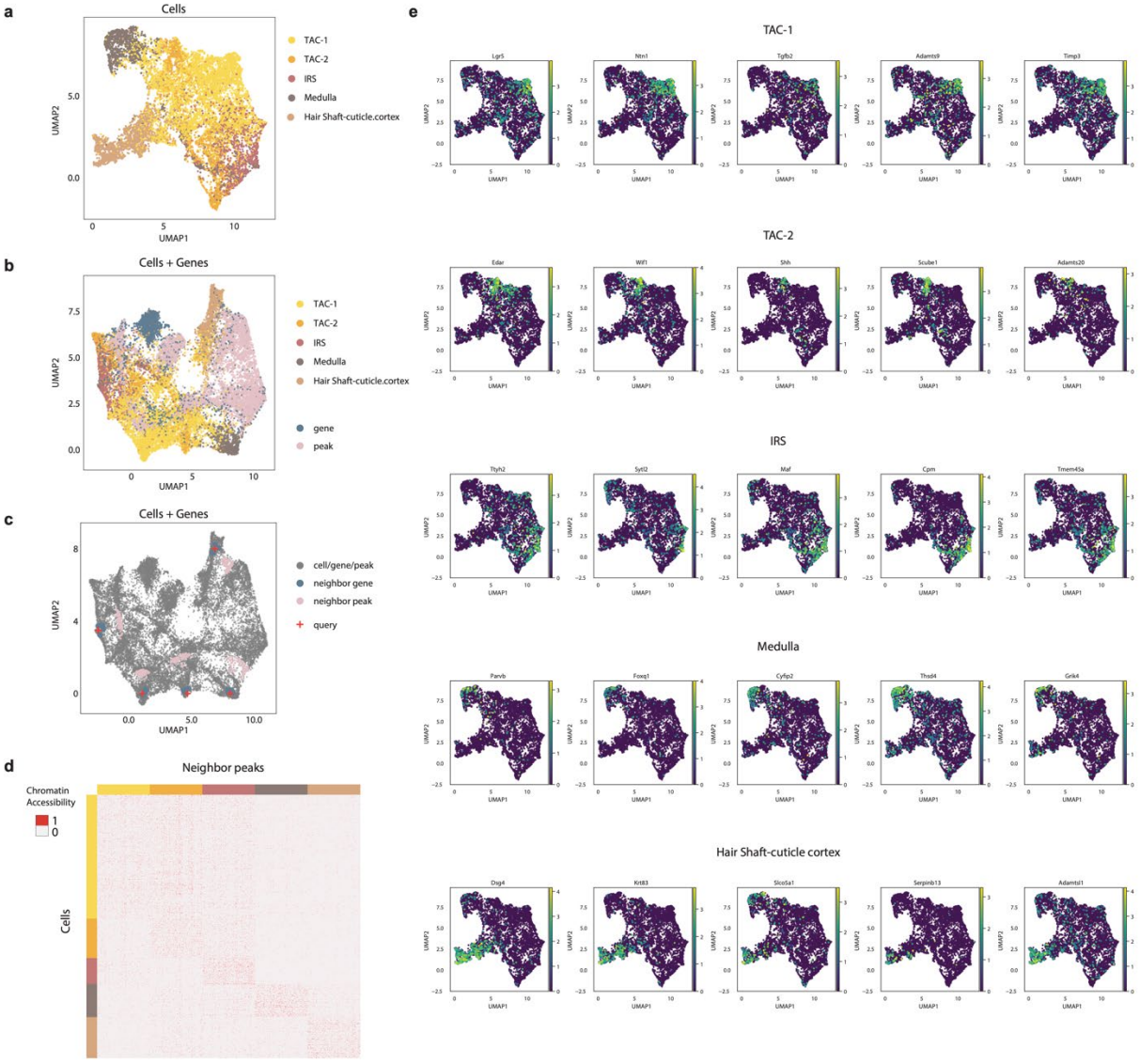
Supplementary Figure 11. SIMBA-inferred marker genes for the scRNA-seq human pancreas dataset in batch correction analysis.

- **a.** UMAP visualization of SIMBA embeddings of cells colored by cell type.
- **b.** UMAP visualization of SIMBA embeddings of cells and genes.
- **c.** UMAP visualization of SIMBA embeddings of cells and genes. Biological “query” points are highlighted with a red “+”. Nearby informative genes are colored accordingly.
- **d.** UMAP visualization of SIMBA embeddings of cells colored by indicated gene expression intensity, separated by cell type.



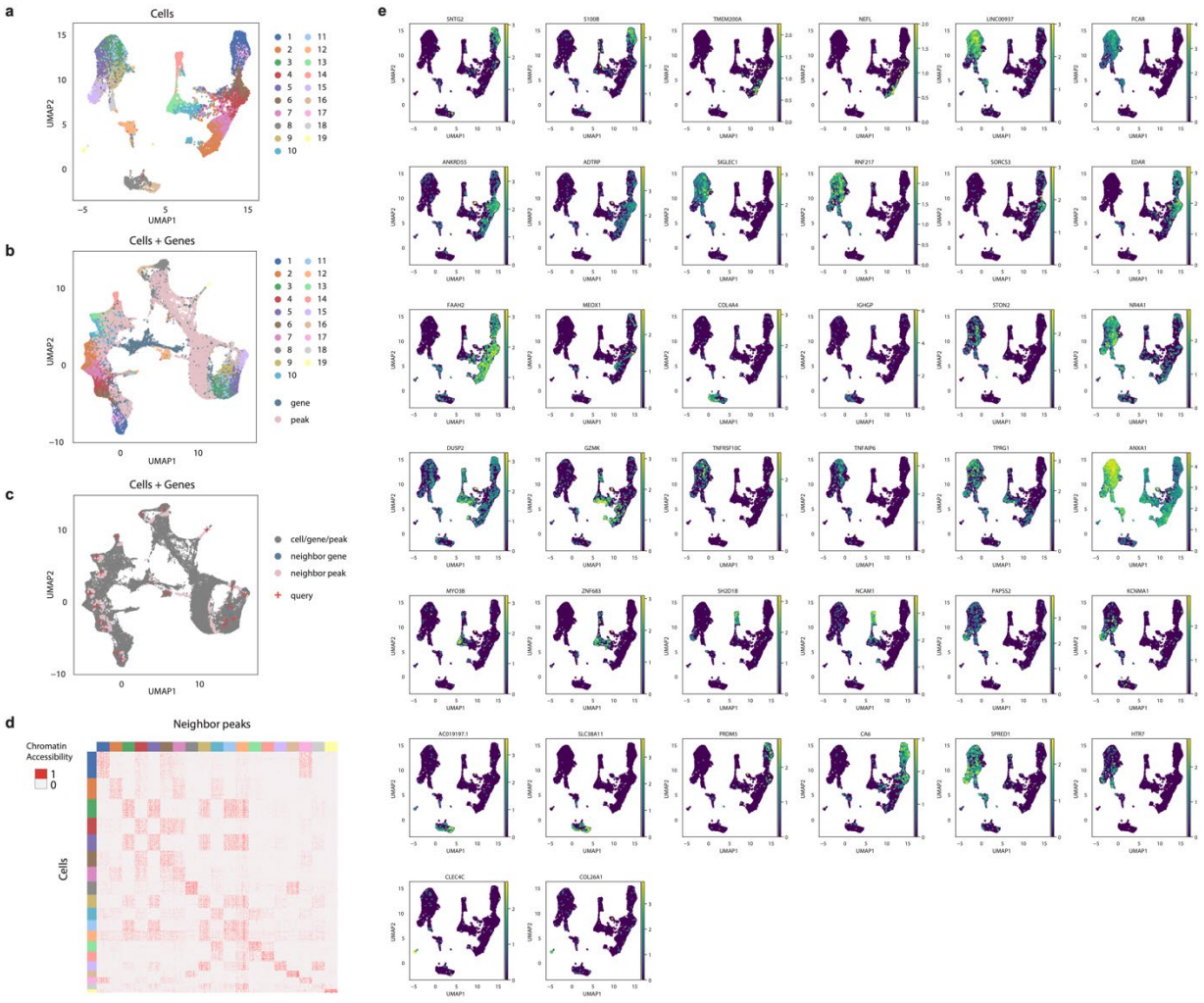
Supplementary Figure 12. Comparison of SIMBA to other methods for batch correction of the mouse atlas and human pancreas scRNA-seq datasets.

a. UMAP visualization of raw and preprocessed mouse atlas data alongside the batch corrected results produced by Seurat3, LIGER, Harmony, and SIMBA. Colored by technology (top) and cell type (bottom).
b. UMAP visualization of raw and preprocessed mouse atlas data alongside the batch corrected results produced by Seurat3, LIGER, Harmony, and SIMBA. Colored by batch origin (top) and cell type (bottom).



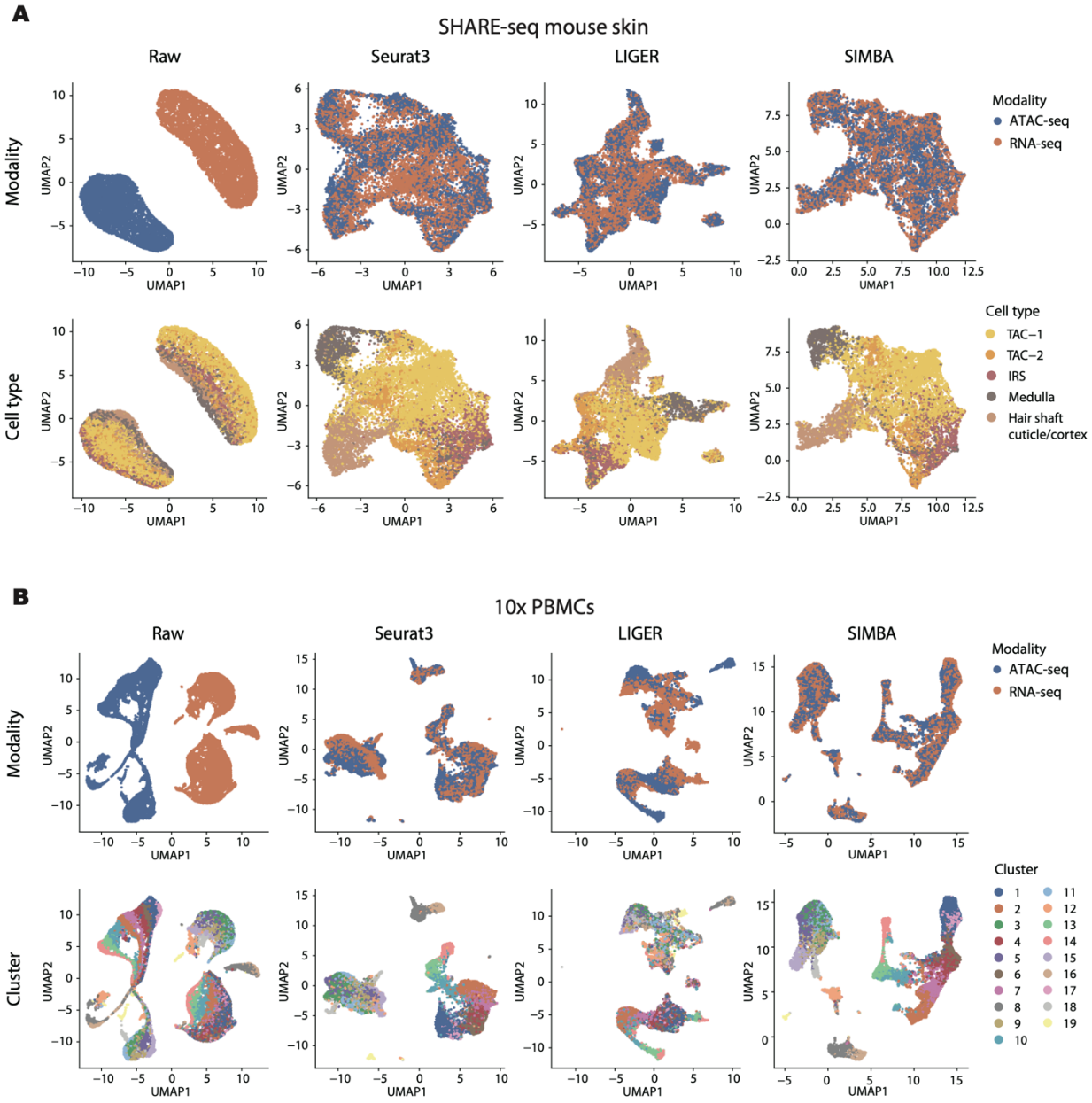
Supplementary Figure 13. SIMBA-inferred marker features for the SHARE-seq mouse skin dataset in multi-omics integration analysis.

- **a.** UMAP visualization of SIMBA embeddings of cells colored by cell type.
- **b.** UMAP visualization of SIMBA embeddings of cells, genes and peaks.
- **c.** UMAP visualization of SIMBA embeddings of cells, genes and peaks. Biological “query” points are highlighted with a red “+”. Nearby informative genes and peaks are colored accordingly.
- **d.** Heatmap of cells against neighboring peaks of each cell type that are selected in the SIMBA co-embedding space. Chromatin accessibility is binary and colored accordingly.
- **e.** UMAP visualization of SIMBA embeddings of cells colored by indicated gene expression intensity, separated by cell type.



Supplementary Figure 14. SIMBA-inferred marker features for the 10x human PBMCs dataset in multi-omics integration analysis.

- **a.** UMAP visualization of SIMBA embeddings of cells colored by cluster assignment.
- **b.** UMAP visualization of SIMBA embeddings of cells, genes and peaks.
- **c.** UMAP visualization of SIMBA embeddings of cells, genes and peaks. Biological “query” points are highlighted with a red “+”. Nearby informative genes and peaks are colored accordingly.
- **d.** Heatmap of cells against neighboring peaks of each cluster that are selected in the SIMBA co-embedding space. Chromatin accessibility is binary and colored accordingly.
- **e.** UMAP visualization of SIMBA embeddings of cells colored by indicated gene expression intensity, separated by cell type.



Supplementary Figure 15. Comparison of SIMBA to other methods for multi-omics integration of the SHARE-seq mouse skin and 10x multiome human PBMCs datasets.

a. UMAP visualization of the raw scRNA-seq and scATAC-seq data from the SHARE-seq mouse skin dataset alongside the integrated results produced by Seurat3, LIGER, and SIMBA. Colored by data modality (top) and cell type (bottom).
b. UMAP visualization of the raw scRNA-seq and scATAC-seq data from the 10x multiome human PBMCs dataset alongside the integrated results produced by Seurat3, LIGER, and SIMBA. Colored by data modality (top) and cluster assignment (bottom).