Single-cell ATAC-seq clustering and differential analysis by convolution-based approach

Li Lin[1*], Liye Zhang[1*]

[1]School of Life Science and Technology, ShanghaiTech University, Shanghai, China.

[*]Correspondence should be addressed to L.L. (linli@shanghaitech.edu.cn) or L.Z. (zhangly@shanghaitech.edu.cn).

Abstract

Single-cell ATAC-seq is a powerful tool to interrogate the epigenetic heterogeneity of cells. Here, we present a novel method to calculate the pairwise similarities between single cells by directly comparing their Tn5 insertion profiles instead of the binary accessibility matrix using a convolution-based approach. We demonstrate that our method retains the biological heterogeneity of single cells and removes undesirable batch effects, which leads to more accurate results on downstream analyses such as dimension reduction and clustering. Based on the similarity matrix learned from epiConv, we develop an algorithm to infer differentially accessible peaks directly from heterogeneous cell population to overcome the limitations of conventional differential analysis through two-group comparisons.

Introduction

The expression of genes is regulated by a series of transcription factors (TFs) that bind to the regulatory elements of the genome. As the accessible chromatin covers more than 90% TF binding regions, many techniques, such as Assay for Transposase-Accessible Chromatin using sequencing (ATAC-seq), have been developed to detect the accessible states of chromatin[1, 2]. Recent technical advancements in ATAC-seq have made it possible to profile the chromatin states of single cells at a high-throughput manner[3-5]. However, both data processing and interpretation of single-cell ATAC-seq (scATAC-seq) data is more challenging than single-cell RNA-seq (scRNA-seq) data owing to low DNA copy number and complexity of chromatin states[1].

Up to now, most methods cluster single cells based on a peak by cell matrix (e.g. Buenrostro et al. 2015[6]). Unlike well-annotated RNA transcripts in the genome, the exact locus of regulatory elements is largely uncharacterized and must be learned from the data itself. However, learning cell type specific regulatory elements from cell mixtures is problematic[7]. Moreover, given that there are no golden rules to define functional elements across the genome, the strategies to perform such task varied considerably in different studies[6, 8], and its effect on downstream analyses is largely unknown.

Detecting differentially expressed genes (or differentially accessible peaks for ATAC-seq, we call them DE peaks below) is another important task in single cell analysis. In a conventional pipeline, cells are first grouped into several clusters and subsequent differential analysis is performed by comparison between clusters. Thus, the resolution settings (e.g. number of clusters) may have strong effects on the identification of genes or locus accounting for the heterogeneity of cell population. Recently one method incorporated pseudotime as one predictor into the regression model to infer DE peaks, instead of performing two-group comparisons[9]. But it required cells to be properly embedded into one dimensional space (e.g. pseudotime through differentiation process), which greatly limits its application in complex cell population. Moreover, cells still need to be clustered into small groups (50~100 cells). Such processing step overcomes the sparsity of scATAC-seq data but reduces the sample size. In scRNA-seq, an alternative approach is to find highly variable genes instead of differentially expressed genes, which does not require the clustering of cell population to be defined. But this strategy cannot be applied to scATAC-seq as the chromatin state is always binarized. Despite that, several state-of-the-art tools designed for scATAC-seq merge individual peaks into meta features (regulomes, topics, principal components, k-mers, etc.) to overcome the sparsity of data[3, 10, 11]. Subsequent differential analysis is performed on meta features instead of individual peaks. Such strategy may help reveal the epigenetic programs that governs the cell identities but lacks sufficient resolution for the dynamic change of individual peaks.

Here, we introduce a novel tool, named epiConv, for scATAC-seq analysis. EpiConv addresses two important questions in scATAC-seq analysis, cell clustering and differential analysis. Unlike most of existing methods, epiConv learns the similarities (or distances) between single cells from their raw Tn5 insertion profiles by a convolution-based approach, instead of a binary accessibility matrix. We demonstrate that epiConv retains biological heterogeneity of single cells and removes unwanted variations derived from multiple batches or sample preparing protocols. Utilizing the similarities learned by epiConv, we also develop an algorithm to infer DE peaks among single cells that can be directly applied to cell mixtures without resolving the intra population structure.

Results

Infer the similarity from Tn5 insertion profiles

First, we give an overview of the algorithm that calculates the similarity between cells from their Tn5 insertion profiles (**Fig. 1**). Given two cells, A with m insertions and B with n insertions in one genomic region, we collapse the insertions into a continuous distribution across the genome by Gaussian smoothing as follows:

$$f_{Ai}(x) = \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{(x-\mu_{Ai})^2}{2\sigma^2}\right),\ f_A(x) = \sum_i^m f_{Ai}(x)$$

$$f_{Bj}(x) = \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{(x-\mu_{Bj})^2}{2\sigma^2}\right),\ f_B(x) = \sum_j^n f_{Bj}(x)$$

Where μ_{Ai} is the locus of insertion *i* in cell A, μ_{Bj} is the locus of insertion *j* in cell B, $f_A(x)$ and $f_B(x)$ give the overall chromatin states of cell A and cell B in the given region. The similarity between A and B over the given region (S_{AB}) is calculated by the convolution of $f_A(x)$ and $f_B(x)$ and can be solved analytically as follows:

$$s_{AB} = \int f_A(x)f_B(x)dx = \mathrm{C}\cdot\sum_{i,j}\exp\left(-\frac{\left(\mu_{Ai}-\mu_{Bj}\right)^2}{4\sigma^2}\right)$$

Where C is an σ dependent constant. In this study, parameter σ is set to 100 bp. To save running time, long distance (> 4σ) is treated as infinity. Through weighted aggregation of the similarities from all informative regions across the genome and proper normalization with respect to sequencing depth, we can obtain the normalized similarity score between any two cells. Subsequent analyses such as dimension reduction or clustering can be performed on the similarity matrix. We also develop a simplified version of epiConv (epiConv-simp), which can be applied to binary accessibility matrix like existing methods. The simplified version does not perform as well as the full version but always generates similar results and runs much faster. In the benchmarking below, we show the results from both full and simplified versions. Other details of epiConv are provided in Methods section.

EpiConv outperforms other methods in cell lines data

We evaluated the performance of epiConv on several datasets and compared it with cisTopic[11], Latent Semantic Indexing (LSI)[3] and SnapATAC, which show better performance than other methods in one recently published benchmarking study[7]. We first applied epiConv to the data from Buenrostro et al. 2015[6]. Specifically, we mixed the data of four cell lines from hematopoietic lineages (K562, GM12878, HL-60 and TF-1) together and tested whether epiConv could cluster single cells correctly based on their biological identities. Given the apparent difference among cell lines, each method performed well in clustering single cells from the same cell line together (**Fig. 2**). However, we found that LSI could not clearly segregate drug-treated and untreated K562 cells. CisTopic segregated treated and untreated K562 cells into two clusters but cells treated by different drugs were still mixed together. Only epiConv-full and SnapATAC grouped K562 cells treated by different drugs into distinct clusters, while epiConv-full showed higher resolution than SnapATAC, yielding the best results. Notably, untreated K562 cells from three replicates were grouped into one cluster without obvious batch effects. Thus, the segregation of cells treated by different drugs was more likely to be attributed to their biological variations rather than batch effects. EpiConv-simp suggested one extra cluster with mixed cell types, performing worse than other methods (**Fig. 2b**). These results highlighted the superiority of directly comparing the Tn5 insertions when performing clustering on highly similar cells (SnapATAC divided the genome into equal-length bins instead of peak calling, which could also be considered as direct comparison on Tn5 insertions but with decreased resolution than epiConv).However, we found that the worse performance of epiConv-simp was partially due to improper denoising method. With an alternative denoising method, epiConv-simp provided good results but still with lower resolution than the full version (top-right in **Fig. 2b**, see **Supplementary Note 1** for the details of alternative denoising method). These results suggested that the matrix still contained the variations derived from drug treatments but they could be easily overwhelmed by noise.

EpiConv removes batch effects in scATAC-seq data

Next, we applied epiConv to the data generated by droplet-based protocol from Satpathy et al. 2019[4]. The authors reported detectable batch effects from LSI method that confounded downstream analyses. Here we asked whether epiConv could perform better. We tested the performance of epiConv on two datasets, one dataset containing cells from two batches of unsorted peripheral blood mononuclear cells (PBMCs), two batches of sorted CD4+CD45RA+ naïve CD4 T cells and two batches of sorted CD4+CD45RA- memory CD4 T cells (PBMC dataset), and the other dataset containing two batches of sorted CD34+ hematopoietic progenitors (CD34+ dataset). Based on our preliminary analyses, epiConv still suffered from batch effects but was less sensitive compared to other methods (data not shown). So, we developed a simple method to remove detectable effects. Although there are many methods to remove batch effects for scRNA-seq data, few studies examined their performance on scATAC-seq data. So, we just compared the results of epiConv to other methods without any batch correction.

In PBMC dataset, the majority of cells from two replicates of memory CD4 T cells were clustered into one tightly related group by epiConv and were close to a small fraction of unsorted

PBMCs. Two replicates of naive CD4 T cells also showed similar results. Other unsorted PBMCs formed several groups without strong batch effects (**Fig. 3a**). On the contrary, cells were mostly clustered by batches for cisTopic, LSI and SnapATAC (**Fig. 3b, Fig. S1a,b**). These results demonstrated that epiConv successfully removed batch effects. To verify whether epiConv clustered single cells based on their biological identities, we marked single cells according to their annotations from Satpathy et al. 2019[4]. The results of epiConv were also largely consistent with the annotations and revealed all major lineages of PBMCs (T cells, NK cells, B cells and Monocytes) and several subpopulations of T cells (**Fig. S2a-d**). In CD34+ dataset, epiConv still performed better in removing batch effects compared to cisTopic, LSI and SnapATAC (**Fig. 3c,d, Fig. S1c,d**). Based on the annotations from Satpathy et al. 2019[4], the results of epiConv were also consistent with our knowledge on hematopoietic differentiation (**Fig. S2e-h**). Moreover, only epiConv and cisTopic clearly revealed the trajectory of hematopoietic differentiation in unsupervised manner, while the results of epiConv were with higher resolution and less noise than cisTopic.

To demonstrate that the power of epiConv was not restricted to specific cell lineages or sample-preparing protocols, we combined scATAC-seq data of adult mouse brain from three experimental protocols, mouse cortex from 10x Genomics, whole mouse brain from droplet single-cell assay for transposase-accessible chromatin using sequencing (dscATAC-seq)[5] and sci-protocols for chromatin accessibility (sci-ATAC-seq)[8]. The dataset contained single cells from 5 batches, one from 10x Genomics, two from dscATAC-seq and two from sci-ATAC-seq. Consistent with previous results, epiConv performed better than cisTopic, LSI and SnapATAC in removing batch effects (**Fig. 3e,f, Fig. S1e,f**) and agreed with the annotations from Cusanovich et al. 2018[8] and Lareau et al. 2019[5] by clustering cells with the same identity together (**Fig. S2i,m**). Although cisTopic suffered from batch effects, it largely agreed with the annotations from original articles within each batch (**Fig. S2j,n**). However, LSI and snapATAC performed worse when comparing them with the annotations from original articles (**Fig. S2k,l,o,p**). Although we lacked direct evidence to evaluate which method performed best in clustering cells according to their cell identities, the results of epiConv and cisTopic largely agreed with each other and could be supported by the annotations from original article. Besides that, only epiConv was capable of clustering cells in a batch-independent manner. Finally, we compared the results between full and simplified versions of epiConv. Simplified version was highly consistent with full version on the three datasets described above and also performed better than other methods (**Fig. S3**).

EpiConv is scalable with large datasets

As the full version of epiConv do pairwise comparisons between single cells, the step of insertions counting is slower than other methods but can be split into small jobs and run in parallel. Based on our tests, it requires 75 CPU hours for 50 million fragments from 5,000 cells (after removing low quality cells and fragments outside informative regions) and 2,400 CPU hours for 270 million fragments from 20,000 cells. The simplified version runs much faster and can be applied to large datasets. Based on our tests, the simplified version requires 17 hours and 520 GB RAM for the Mouse Cell Atlas dataset[8] (81,173 cells and 436,206 peaks) with single thread, faster than cisTopic (48 hours) but slower than LSI (1 hour). SnapATAC failed to run on the full dataset of Mouse Cell Atlas dataset in the step of calculating Jaccard distance due to the memory limitation for single object in R (This error may depend on the system as Chen et al.[7] reported that SnapATAC could run on the full dataset. Actually, we also encountered the same error for epiConv but we modified our scripts to avoid it). The results of Mouse Cell Atlas dataset by epiConv-simp also largely agreed with the annotations from Cusanovich et al. 2018[8] (**Fig. S4**).

Notably, a large proportion of cells were marked as unknown in the Mouse Cell Atlas dataset (**Fig. 4a-d**). In the results of cisTopic , LSI and SnapATAC (we randomly sampled 25% cells in Mouse Cell Altas dataset for SnapATAC), these cells formed a large cluster of their own, showed close relationships with several clusters with known identities but did not overlap with them (**Fig. 4a-c**). However, unknown cells did not form a single cluster but were mixed with other known cell types in the results of epiConv-simp (mainly associated with 6 clusters with more than 10% cells marked as unknown, **Fig. 4d**). This might suggest a large improvement of epiConv over existing methods. In order to validate our findings, we aggregated the cells with known and unknown cell identities respectively for each cluster. Then we calculated the spearman correlation between the 12 aggregated samples over a set of highly accessible peaks (accessible in at least 1% cells from these 6 clusters). We found that all unknown samples showed highest correlations with corresponding known samples within the same clusters (**Fig. 4e**). Individual unknown samples showed low correlations with each other, suggesting that epiConv successfully “demultiplexed” unknown cells by their biological identities. By these results, we confirmed that epiConv showed significant improvements over existing methods on the Mouse Cell Atlas dataset. Combined with other results of epiConv-simp mentioned above, we concluded that in most cases epiConv-simp also proved to be a reliable tool for the investigation of large datasets.

EpiConv detects differentially accessible peaks in cell mixtures

In the section below, we aim to develop an algorithm to infer DE peaks directly from cell mixtures. Our algorithm compares the number of accessible cells among each cell's neighbors with the proportion of accessible cells in cell mixture for each peak and turns the binary chromatin states into normalized z-scores, which show the enrichment of accessible cells among neighbors (we call it z-scores below). If the number of cells showing high z-scores for one peak exceeds the threshold, we then consider the peak to be differentially accessible. Details of our algorithm can be found in Methods.

In order to test whether the algorithm could detect DE peaks in cell mixture, we first applied our method to one dataset of myoblast differentiation[9]. We found that although epiConv could reconstruct the differentiation process of myoblasts, where cells were roughly ordered by harvesting times (**Fig. 5a,b**), it was difficult to cluster cells due to the continuous differentiation process. Using our algorithm, we detected 37,107 peaks to be differentially accessible during the differentiation process. To show the dynamics of DE peaks, we plotted heatmap of z-scores, where cells and DE peaks were embedded into one-dimensional (1D) space based on the similarity matrix and the spearman correlation of z-scores between peaks (**Fig. 5b**). The results showed approximately half peaks to be more accessible in the early stage of differentiation and others to be more accessible in the later stage. The dynamic changes of z-scores along differentiation was consistent with merged scATAC-seq profiles by harvesting times, demonstrating the reliability of our algorithm (**Fig. 5c**).

Next, we want to test the sensitivity of our algorithm. We applied our algorithm to the HSC-MPP-LMPP cluster in the CD34+ dataset. We chose the HSC-MPP-LMPP cluster because up to now, few methods could distinguish MPPs from HSCs in scATAC-seq data duo to the high similarity between them (see the benchmarking study of Chen et al.[7]). Given epiConv already removed most of batch effects, we could include cells from both replicates of CD34+ dataset to increase the statistical power. Through our algorithm, we detected 27,612 DE peaks within the HSC-MPP-LMPP cluster. The dynamic changes of z-scores were highly consistent with the bulk ATAC-seq profiles of FACS-sorted HSCs, MPPs and LMPPs[12](**Fig. 5d,e**). All DE peaks were properly ordered through the 1D embedding and agreed with their accessibility dynamics in both single-cell and bulk samples, suggesting that the co-accessible pattern between peaks could be revealed by z-scores. (**Fig. 5d,e**). Moreover, our results also showed gradual gain or loss of accessibility in a wide range of peaks in HSCs, revealing the continuous cell state transition within HSCs. Although we lacked direct evidence to evaluate whether epiConv clustered HSCs and MPPs into two groups, we could still extract HSC and MPP unique signatures through DE analysis. As scaled heatmap could not reveal the fold change of peaks, we also examined the log2 Fold Change between MPP, HSC and LMPP bulk samples for all detected DE peaks (**Fig. 5f**). Most peaks showed strong difference between MPP/LMPP or HSC/LMPP bulk samples, while MPP or HSC unique peaks just showed weak difference between MPP/HSC bulk samples. As is shown by many single-cell studies, FACS-sorted cells may still be the mixtures of similar cell types. We thought that this could partially explain the weak difference between MPP/HSC bulk samples. Unexpectedly, we also found that LMPPs could be further divided into three groups based on their unique signatures and bulk LMPPs seemed to be the mixture of these three groups. By comparing the z-scores of single cells with bulk samples, we found that they might represent different stages of LMPPs during differentiation (early undifferentiated stage, later stage to GMP and later stage to CLP, see heatmap in the right in **Fig. 5e**). These results demonstrated that inferring DE peaks directly from cell mixtures helped reveal the intra-population structure and intermediate cell states in a signature-driven manner instead of statistical ways.

We also applied our algorithm to all cells in CD34+ dataset to test the scalability of our algorithm. Similar with previous results on HSC-MPP-LMPP cluster, we also found a series of peaks that gradually gained or lost accessibility through differentiation (e.g. in MDPs, **Fig. S5a**). The z-scores did not fully capture the chromatin states of bulk samples for a few peaks (**Fig. S5b**). We found that it could be explained by the difference between single-cell and bulk samples (data not shown), probably because there might be some batch effects between them. In fact, it was not difficult to infer DE peaks from distinct clusters. But we demonstrated that our algorithm could also perform such task like conventional methods.

We found that sometimes the z-scores did not agree with the binary accessibility profiles of single cells. It was because z-scores were normalized by the library size of single cells. However, we thought that the library size of single cells in droplet-based protocols could reveal the difference of global chromatin states between different cell types. By comparing the chromatin states of neighbors with the background, our algorithm already removed the variation of library size for individual cells. So, we designed another normalization strategy, where the scaling factors of all cells were set to 1. We tried this normalization strategy in cells from replicate 1, where all cells were processed in parallel during experiment and their library size could reflect the global chromatin states. The z-scores were consistent with binary accessibility profiles under this normalization strategy (**Fig. S5c,d**) but did not agree with corresponding bulk samples (**Fig. S5e**).

Discussions

In this study, we developed a novel method to directly compare the Tn5 insertions between single cells and compared it with three existing methods, cisTopic, LSI and SnapATAC. Results demonstrated that our method had several advantages over existing methods. The most significant difference between our algorithm and others is that we calculated the distance between single cells using a convolution-based approach instead of commonly used Euclidean-distance. Although the Jaccard similarity used by SnapATAC is similar to epiConv (Assuming two binary vector A and B, Jaccard similarity is calculated by $\frac{|A \cap B|}{|A \cup B|}$, while epiConv uses $\frac{|A \cap B|}{|A| \cdot |B|}$. Moreover, epiConv assigns weights to different loci), the distance is calculated by Euclidean-distance on principal components. Interestingly, we also found a way to make epiConv mimic the behavior of other methods, making it easy to compare the difference between two forms of distance. Given the similarity matrix S before denoising step, we used Eigen value decomposition to obtain a series of latent features from S. Given $Q^T S\ Q = \Lambda$, where Q is the matrix containing Eigen vectors of S and Λ is the diagonal matrix containing Eigen values of S, the columns of $Q\Lambda^{\frac{1}{2}}$ can be considered as latent features. Here we used top 50 latent features. By calculating the Euclidean-distance on these latent features, the behavior of epiConv was highly similar with existing methods (we showed the results of PBMC dataset and Mouse Cell Atlas dataset in **Fig. S6**). In PBMC dataset, epiConv became sensitive to batch effects and the batch effects could not be removed by our algorithm (compare **Fig. S6a** with **Fig. 3a,b** and **Fig. S1a,b**). However, within each batch, major cell types could also be distinguished like other methods (**Fig. S6b**). In Mouse Cell Atlas dataset, epiConv clustered “unknown” cells into single cluster like other methods (compare **Fig. S6c** with **Fig. 4a-d**). These results clearly demonstrated that different denoising process could have significant effects on our understanding of the cell heterogeneity even when the raw data is identical. We hypothesized that methods trying to capture latent features may suffer from common biases. By using convolution-based approach to define the similarities between cells, epiConv provides a new angle of view in the analysis of sparse epigenetic data. Moreover, epiConv also provides DE analysis in single-cell resolution and in unbiased manner, while no existing methods could perform such task. Thus, we believe that epiConv will have wide applications and improve our understanding on the epigenetic dynamics of single cells.

Methods

Informative region calling for epiConv. EpiConv takes processed fragments as input file. To call informative regions for epiConv, we first extended Tn5 insertions from both directions using the pileup command in MACS2[13] (-B --extsize 100). Then, we sorted all sites of the genome by their density in decreasing order and selected regions with cumulative density less than 70% of total insertions. These regions were extended from both directions by 100 bp and merged together if having any overlap. Tn5 insertions overlapping with these informative regions (~70% of total reads) were used for downstream analysis. We used such strategy instead of MACS2 because the proportion of reads used in downstream analyses could be easily specified through the threshold of cumulative density. Moreover, this strategy can always obtain some peaks, while MACS2 may fail when the number of cells is low (e.g. < 200, reported by Satpathy et al. 2019[4]). The threshold of cumulative density is determined by the distribution of insertion length. Based on our preliminary analysis, fragments spanning one or more nucleosomes are nosier than fragments from nucleosome-free regions. Thus, the threshold should be close to the proportion of fragments from nucleosome-free regions. For the myoblast and mouse brain datasets, we set the threshold to 50% as they had higher proportion of fragments spanning one or more nucleosomes (data not shown). The major purpose of informative region calling is to calculate the weights for different genomic regions (see below). Additionally, it could remove some background noise. Although it is possible to compare the Tn5 insertions of the whole genome, which might help detect rare cell types, we find that it just increases running time but does not improve the results.

epiConv algorithm. In the results section, we described the algorithm to calculate the similarity between two cells over one region. Here assume that we have N cells and K regions, with the similarities between any two cells *i* and *j* over region *k* (s_{ijk}) being known. First, we weight each region as follows:

$$freq_k = \sqrt{\frac{2}{N(N-1)}\sum\nolimits_{ij} s_{ijk}}$$

$$w_k = log10(1 + {freq_k}^{-1})$$

The form of weight is similar to that used in LSI but the frequency is replaced by a pseudo-frequency estimated from our convolution-based approach. We use such form of weight to increase the contribution of low-density regions to the similarity score. The similarity between cell *i* and *j* is calculated using a bootstrap approach. Assuming we perform L replicates (L = 30 in this study) and in each replicate we randomly sample some regions (12.5% of total informative regions in this study). The similarity of s_{ij} is calculated as follows:

$$s_{ij} = \frac{\sum_l log10\left(\sum_{k\in rep_l} s_{ijk} \cdot {w_k}^2\right)}{L} - \log10\left(lib_i \cdot lib_j\right)$$

where lib_i and lib_j is the library size of cell *i* and *j*. We normalize the aggregated similarity by $lib_i \cdot lib_j$ because $\sum_{k\in rep_l} s_{ijk} \cdot {w_k}^2$ can be considered as the sum of $lib_i \cdot lib_j$ random variables with identical distribution given the analytical form of similarity described above.

Averaging the similarities from replicates helps reduce the noise compared to simple aggregation of similarities from all regions. But for deep sequencing data, we find that simple aggregation also generates similar results (data not shown).

In the simplified version, matrix is first binarized and TF-IDF transformed like LSI[3] (In epiConv-simp, normalization with respect to sequencing depth and peak weighting are identical as LSI). Given TF-IDF matrix *M* and L bootstrap matrices M_{rep_l} by randomly sampling peaks from *M*, the similarity matrix *S* can be calculated as follows:

$$S = \frac{\sum_l log10\left({M_{rep_l}}^T \cdot M_{rep_l}\right)}{L}$$

where ${M_{rep_l}}^T \cdot M_{rep_l}$ is the matrix product. Unlike LSI implemented in Cusanovich et al. 2015[3] and Cusanovich et al. 2018[8], we do not filter any peaks. By adopting the formula above, the distance between two insertions $\mu_{Ai} - \mu_{Bj}$ is considered as zero if they are in the same peak or infinite otherwise. Further steps are identical for full and simplified versions.

Next, we denoise the similarities between cells by borrowing the information from their neighbors. The denoised similarities are calculated by the number of shared nearest neighbors between two cells. The number of nearest neighbors for each cell is set to 50 in this study. If the dataset contains cells from multiple batches, we force cells to select equal number of nearest neighbors from each batch to remove batch effects. The distance matrix D is calculated by $\mathrm{D} = -S_{denoise}$. Although the batch removal strategy can be applied to the similarity or distance matrix generated by various methods, we find that it only works well with epiConv. As mentioned above, it is because that epiConv is less sensitive to batch effects even without any correction.

The denoising method above changes the unit of similarity matrix (from continuous values to integer values). Occasionally we find that it may make the results worse (see the results of epiConv-simp for cell lines data in **Fig. 2b**). We also developed an alternative denoising method that keeps the unit of similarity matrix unchanged (**Supplementary Note 1**). Generally, it is noisier than the first method and cannot remove batch effects. But it may perform better when the first method fails (see top-right in **Fig. 2b**).

Pre-processing of ATAC-seq data. We took the processed fragment file or peak by cell matrix as inputs if available. For the unprocessed data from Buenrostro et al. 2015[6] and bulk samples from Corces et al. 2016[12], we aligned raw reads to the hg19 genome using Bowtie2[14] (-X 2000 --no-mixed --no-discordant) and removed reads with mapping quality <10 and duplicates using Picard tools. The start and end of the fragments were adjusted (+5 for forward strand and −4 for reverse strand). We called peaks using MACS2[13] (--nomodel --nolambda --keep-dup all --shift -200 --extsize 400) and generated the count matrix by counting the number of Tn5 insertions falling in peaks.

For the mouse brain dataset, we randomly sampled 2,000 cells from Channel 1 and Channel 2 in Lareau et al. 2019 (dscATAC-seq)[5], 1,000 cells from the mouse cortex data from 10x Genomics and 2,000 cells from two replicates of whole mouse brain in Cusanovich et al. 2018 (sciATAC-seq)[8]. The dataset contains 5,000 cells in total. Data from Cusanovich et al. 2018 were converted from mm9 to mm10 using liftOver[15]. Data from 10x Genomics and Cusanovich et al. 2018 were re-counted against the peaks called by Lareau et al. 2019 for data integration.

For the myoblast dataset, few outlier cells that did not cluster together with the majority of cells were excluded in differential analysis (**Fig. 5a**).

Implement of cisTopic, LSI and SnapATAC. In cisTopic, the number of topics is set to 20, 30, 40 and 50 and automatically decided by cisTopic. For the analysis of cell lines data from Buenrostro et al. 2015[6], in order to explore whether increased number of topics could provide higher resolution for K562 cells, we increase the number of topics from 20 to 100 with a step of 10 but the optimal number of topics is still decided by cisTopic. In LSI, we use the scripts from Cusanovich et al. 2018[8], filter out peaks with frequency < 0.01 and use the top 50 components of singular value decomposition for dimension reduction. In SnapATAC, the bin size was set to 5000. We fixed the number of principal components used for dimension reduction to 30 instead of manually examining the distribution of each component to avoid ambiguity.

Differential analysis algorithm. The input data is a binarized peak by cell matrix and a similarity matrix between cells. Here we use the peak by cell matrix from previous steps. The similarity matrix is calculated by $S_{denoise} + S/100$ (The similarity matrix is mainly determined by $S_{denoise}$. When two cells have equal number of common neighbors to another cell, the similarities are further determined by the original similarity matrix). For each single cell, we define *k* cells with highest similarities as its neighbors (including itself). Then for each peak, we test whether it is more likely to be accessible in the cell's neighbors. This problem can be resolved using hypergeometric test, with cells accessible as black balls, cells inaccessible as white balls. The sampling times ($\hat{k}$, the adjusted number of neighbors) is calculated by the total scaling factors of all neighbors divided by the average scaling factors of all cells. The scaling factors of cells were equal to their library sizes or set to 1 for all cells (DE analysis on replicate 1 of CD34+ dataset, see Results). The z-scores are calculated by the number of cells accessible among neighbors and z-normalized by corresponding mean and variance of the null distribution.

In differential analyses in this study, the number of neighbors *k* is set to 5% of total cells. The number of neighbors *k* defines the size of potential clusters, which serves similar function as the number of clusters in conventional pipeline. However, the results demonstrated that our algorithm with fixed *k* could still detect DE peaks in clusters with a wide range of size. Here, *k* is set to 5% in order to make our algorithm more sensitive to DE peaks of small clusters. After obtaining the z-scores, we select peaks with z-score > 2 in at least 10% cells as DE peaks. When we applied our algorithm to all CD34+ cells (whole dataset or replicate 1), we select peaks with z-score > 2 in at least 30% cells as we only want to detect DE peaks between major clusters and the criterion of 10% cells suggested most peaks to be differentially accessible, which was reasonable but not desired.

In fact, it is not straightforward to choose a proper threshold for z-score. We find that peaks that do not satisfy the threshold described above may also show weak DE pattern. Here, we use the threshold of 10% cells with z-score >2 because selected peaks can be easily validated by bulk samples. For general purpose, users can set the threshold manually to obtain appropriate number of DE peaks.

Dimension reduction. We perform dimension reduction of single cells using the uniform manifold projection (UMAP) algorithm[16] by feeding umap with the distance matrix learned by epiConv, cisTopic, LSI and SnapATAC using default settings. The number of reduced components was set to 1 for heatmaps and 2 for scatterplot of cells. We also embed DE peaks into 1D space by feeding umap with the distance matrix that is calculated by one minus spearman correlation of z-scores between peaks.

Bulk sample processing. For bulk samples of hematopoietic cells from Corces et al. 2016[12], we count the Tn5 insertions against the peaks called from Satpathy et al. 2019[5], normalize the counts by library size and average the normalized counts across all replicates for each cell type. For the myoblast dataset, we de-multiplex the reads, count the Tn5 insertions and normalize the counts by harvesting times.

Data availability. The cell lines data of Buenrostro et al. 2015[6] is obtained from Gene Expression Omnibus (GEO) accession GSE65360. The data of Satpathy et al. 2019[4] is obtained from GEO accession GSE129785. The data of Lareau et al. 2019[5] is obtained from GEO accession GSE123581. The data of Cusanovich et al. 2018[8] is obtained from Mouse Cell Atlas (http://atlas.gs.washington.edu/mouse-atac/). The data of adult mouse cortex is obtained from 10X Genomics website (https://support.10xgenomics.com/single-cell-atac/datasets/1.2.0/atac_v1_adult_brain_fresh_5k). Myoblasts data[9] is obtained from GEO accession GSE109828. EpiConv is available at Github (https://github.com/LiLin-biosoft/epiConv).

Acknowledgements

This project was funded by the National Key Research and Development Program of China (2018YFC1004602), National Natural Science Foundation of China (NSF 31871332) and a startup fund to L.Z. from ShanghaiTech University. We would like to thank Xiaojing Zhao for testing the reproducibility of the study. We would like to thank Yingdong Zhang on his technical support on the HPC platform of ShanghaiTech University.

Author contributions

L.L. conceived the study, developed the methods and performed the analyses. L.L. and L.Z. wrote the manuscript. L.Z supervised the study.

Competing interests

The authors declare no competing interests.

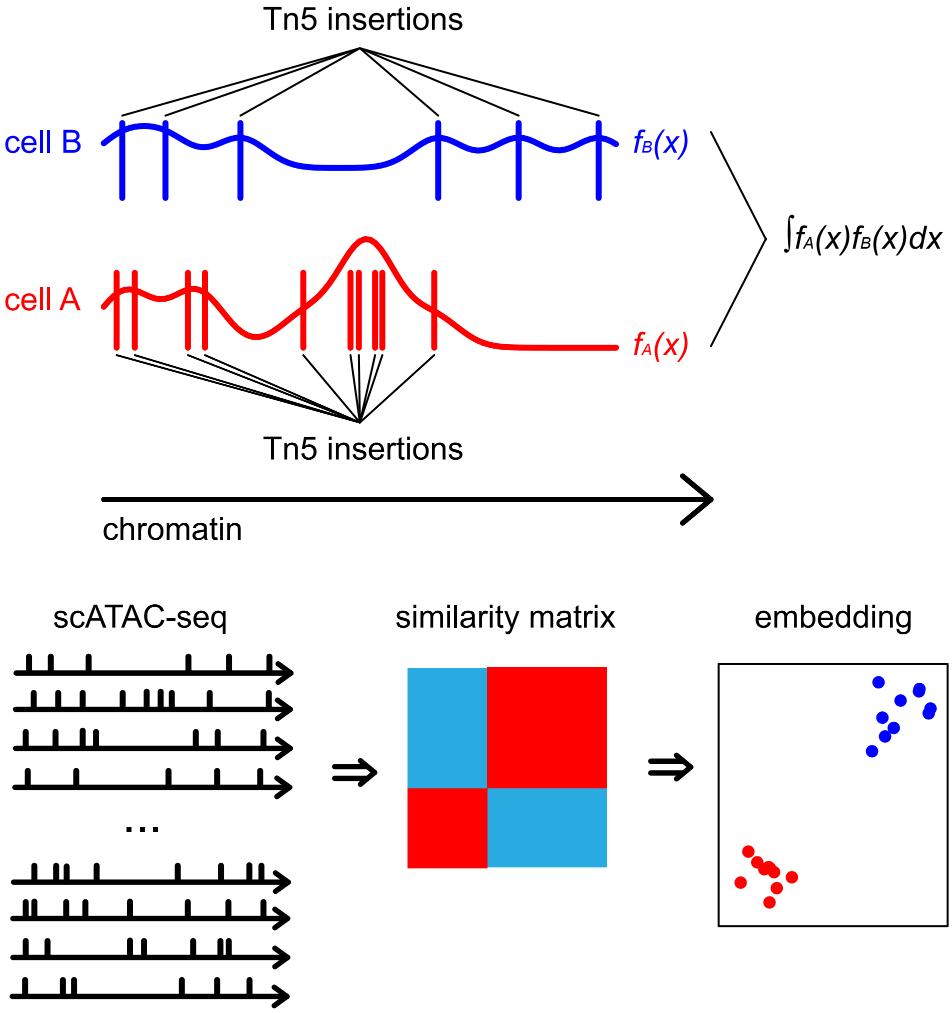
Tn5 insertions
cell B
$f_B(x)$
$\int f_A(x) f_B(x) dx$
cell A
$f_A(x)$
Tn5 insertions
chromatin
scATAC-seq
...
⇒
similarity matrix
⇒
embedding

Figure 1. An overview of the epiConv algorithm.

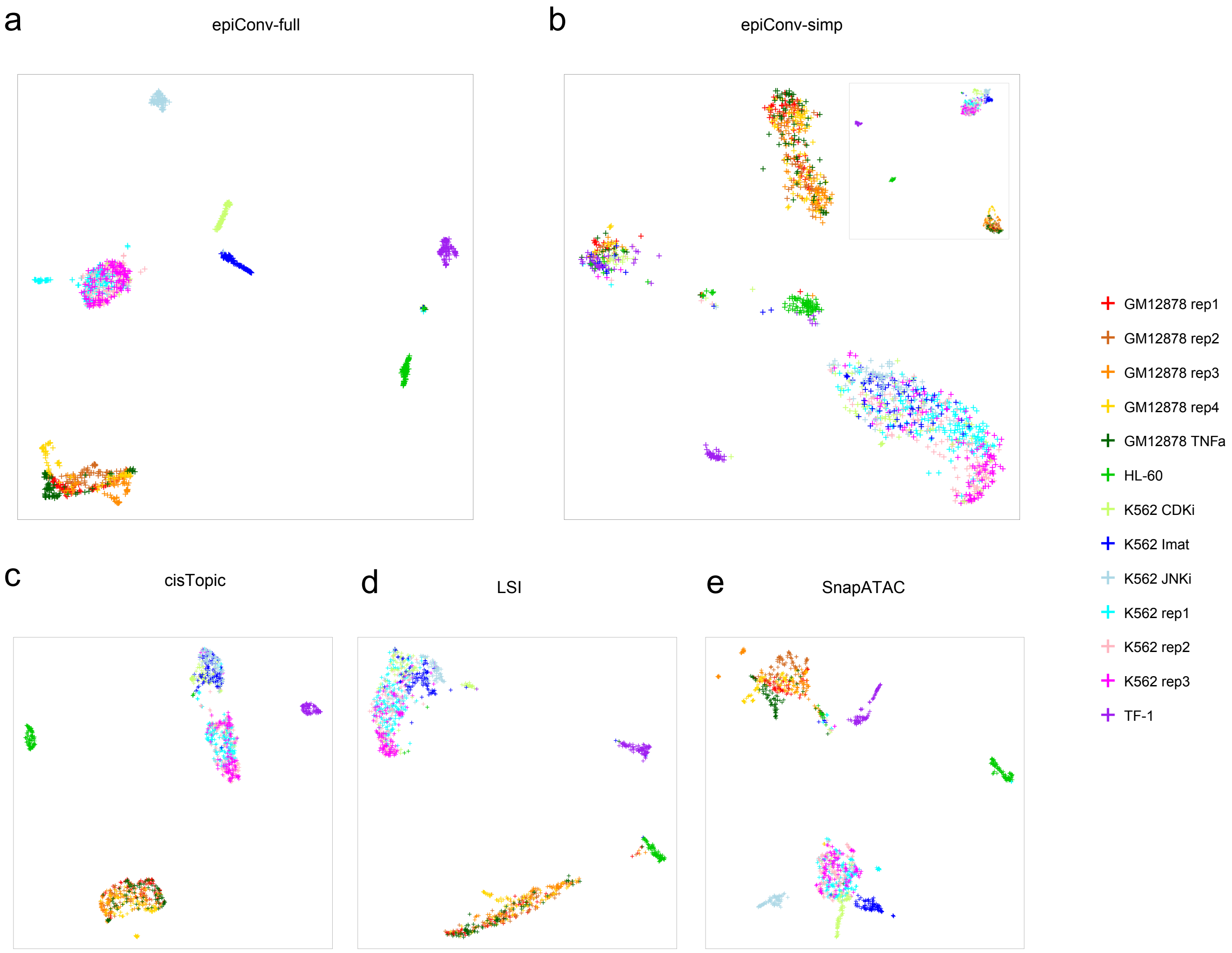
a
epiConv-full
b
epiConv-simp
c
cisTopic
d
LSI
e
SnapATAC
GM12878 rep1
GM12878 rep2
GM12878 rep3
GM12878 rep4
GM12878 TNFa
HL-60
K562 CDKi
K562 Imat
K562 JNKi
K562 rep1
K562 rep2
K562 rep3
TF-1

Figure 2. EpiConv performs better than other methods on cell lines data. (**a**) Embedding by epiConv-full. (**b**) Embedding by epiConv-simp. (**c**) Embedding by cisTopic. (**d**) Embedding by LSI (**e**) Embedding by snapATAC.

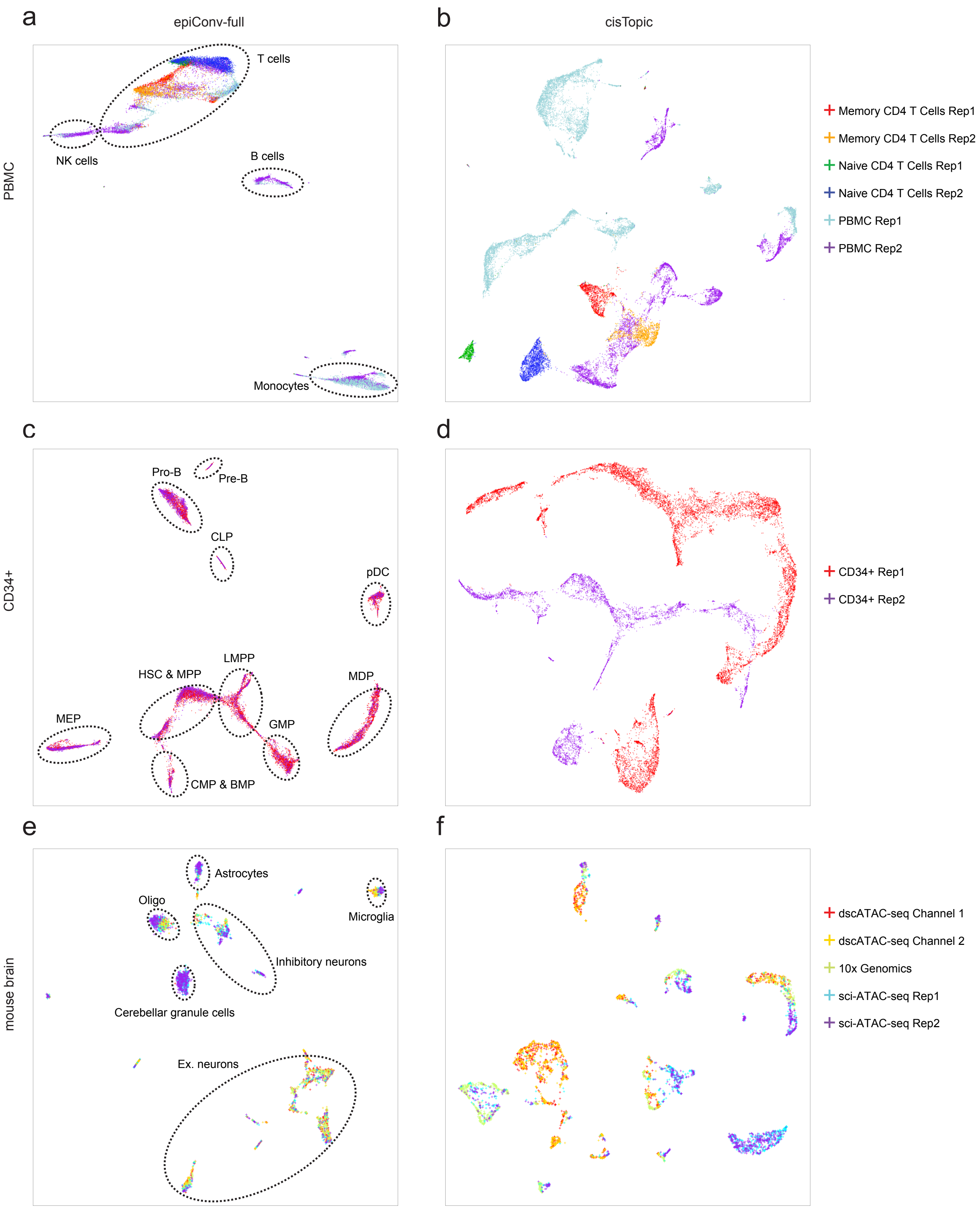
a
epiConv-full
b
cisTopic
PBMC
T cells
NK cells
B cells
Monocytes
+ Memory CD4 T Cells Rep1
+ Memory CD4 T Cells Rep2
+ Naive CD4 T Cells Rep1
+ Naive CD4 T Cells Rep2
+ PBMC Rep1
+ PBMC Rep2
c
d
CD34+
Pro-B
Pre-B
CLP
pDC
LMPP
HSC & MPP
MDP
MEP
GMP
CMP & BMP
+ CD34+ Rep1
+ CD34+ Rep2
e
f
mouse brain
Astrocytes
Oligo
Microglia
Inhibitory neurons
Cerebellar granule cells
Ex. neurons
+ dscATAC-seq Channel 1
+ dscATAC-seq Channel 2
+ 10x Genomics
+ sci-ATAC-seq Rep1
+ sci-ATAC-seq Rep2

Figure 3. EpiConv removes batch effects. (**a,b**) Embeddings of PBMC dataset by epiConv-full and cisTopic. (**c,d**) Embeddings of CD34+ dataset by epiConv-full and cisTopic. (**e,f**) Embeddings of the integration of mouse brain data from dscATAC-seq, 10x Genomics and sci-ATAC-seq by epiConv-full and cisTopic. Major cell types in the embeddings of epiConv are circled. Embeddings by LSI and SnapATAC can be found in **Fig. S1**. HSC, hematopoietic stem cells; MPP, multipotent progenitors; LMPP, lymphoid-primed multipotent progenitors; CMP, common myeloid progenitors; BMP, basophil-mast cell progenitors; GMP, granulocyte-macrophage progenitors; MDP, monocyte-dendritic cell progenitors; pDC, plasmacytoid dendritic cells; MEP, megakaryocyte-erythroid progenitors; CLP, common lymphoid progenitors; Oligo, oligodendrocytes; Ex. neurons, excitatory neurons.

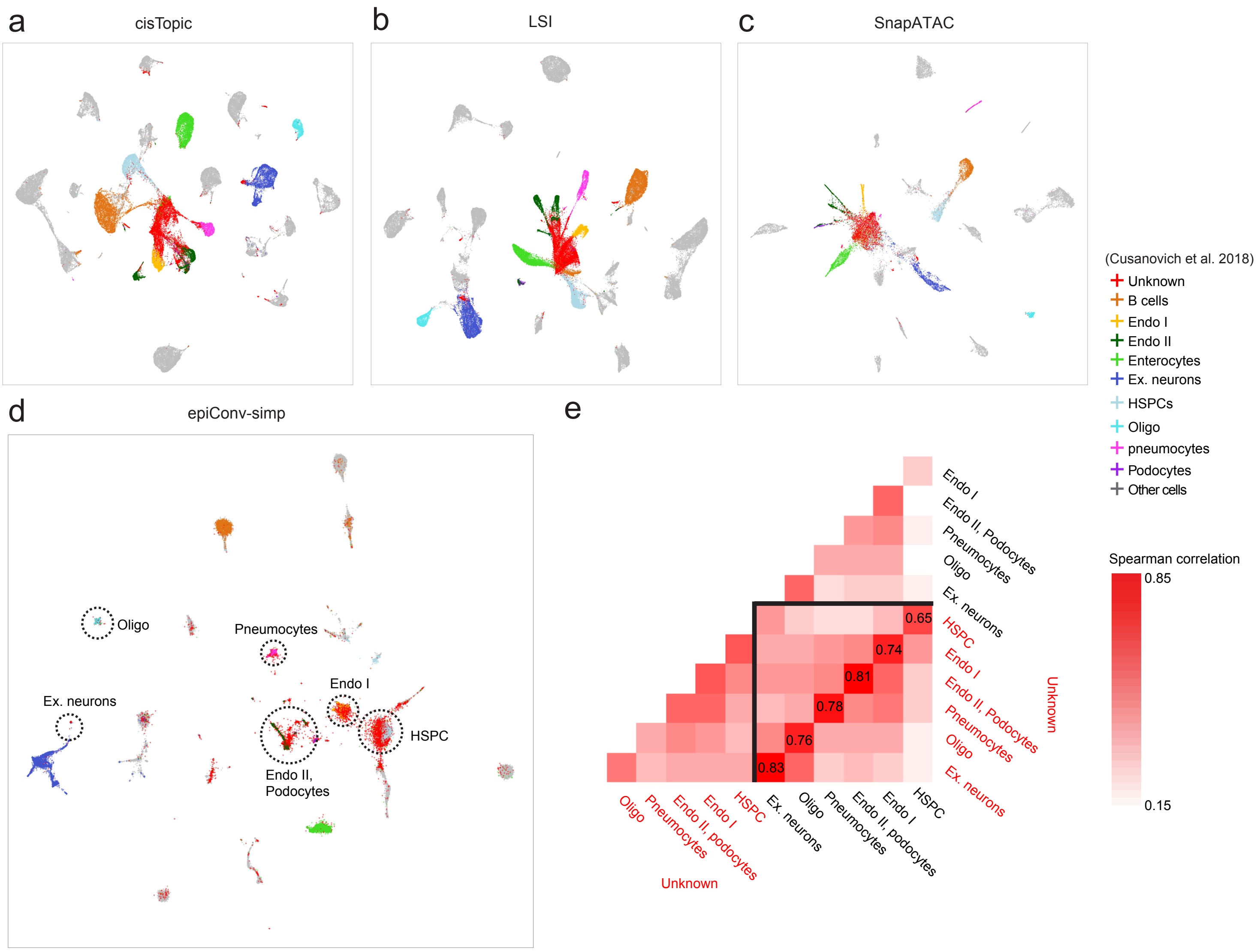
a
cisTopic
b
LSI
c
SnapATAC
(Cusanovich et al. 2018)
Unknown
B cells
Endo I
Endo II
Enterocytes
Ex. neurons
HSPCs
Oligo
pneumocytes
Podocytes
Other cells
d
epiConv-simp
Oligo
Pneumocytes
Endo I
Ex. neurons
HSPC
Endo II,
Podocytes
e
Endo I
Endo II, Podocytes
Pneumocytes
Oligo
Ex. neurons
HSPC
Endo I
Endo II, Podocytes
Pneumocytes
Oligo
Ex. neurons
Unknown
0.65
0.74
0.81
0.78
0.76
0.83
Oligo
Pneumocytes
Endo II, podocytes
Endo I
HSPC
Ex. neurons
Oligo
Pneumocytes
Endo II, podocytes
Endo I
HSPC
Unknown
Spearman correlation
0.85
0.15

Figure 4. EpiConv reveals the identities of unknown cells in Mouse Cell Atlas dataset. (**a**) Embedding by cisTopic. (**b**) Embedding by LSI. (**c**) Embedding by SnapATAC. (**d**) Embedding by epiConv-simp. In (**a-d**), unknown cells and cells showing close relationships with them are colored according to the annotations from Cusanovich et al. 2018. Other irrelevant cells are colored in grey. Six major clusters in (**d**) that contain high proportion of unknown cells are circled. (**e**) Spearman correlations between aggregated samples with known and unknown identities from 6 major clusters marked in (**d**). Labels of unknown samples are colored in red. Numbers in the diagonal elements show the correlations between unknown samples and corresponding known samples. Endo I, endothelial I cells; Endo II, endothelial II cells; Ex. neurons, excitatory neurons; HSPC, hematopoietic progenitors; Oligo, oligodendrocytes.

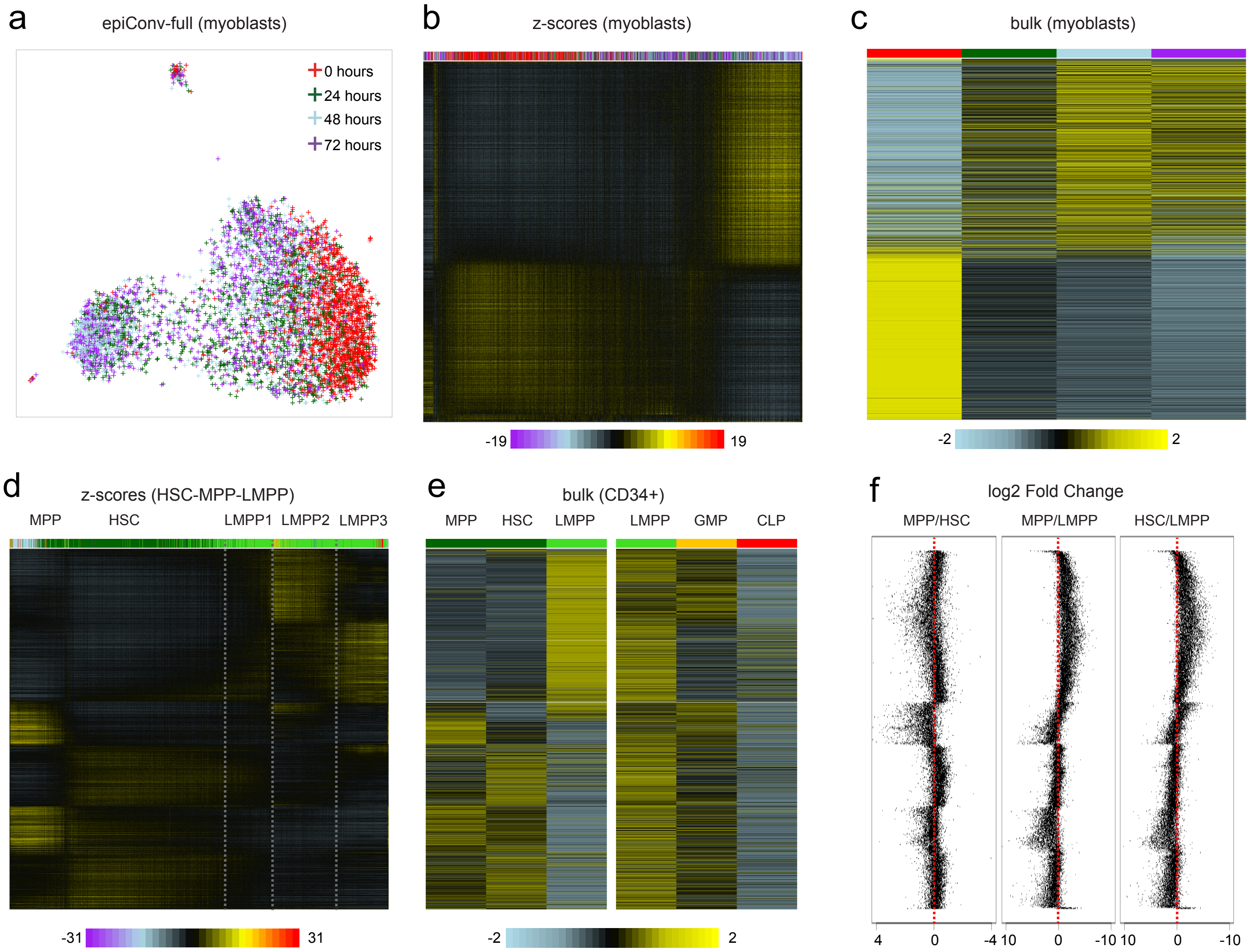
a
epiConv-full (myoblasts)
0 hours
24 hours
48 hours
72 hours
b
z-scores (myoblasts)
-19
19
c
bulk (myoblasts)
-2
2
d
z-scores (HSC-MPP-LMPP)
MPP
HSC
LMPP1
LMPP2
LMPP3
-31
31
e
bulk (CD34+)
MPP
HSC
LMPP
LMPP
GMP
CLP
-2
2
f
log2 Fold Change
MPP/HSC
MPP/LMPP
HSC/LMPP
4
0
-4
10
0
-10
10
0
-10

Figure 5. EpiConv detects differentially accessible peaks in cell mixtures. (**a**) Embedding of myoblast single cells by epiConv-full. (**b**) Accessibility z-scores of myoblast single cells inferred by epiConv. (**c**) Accessibility profiles of aggregated myoblast samples by harvesting times. Cells or aggregated samples in (**b,c**) are colored by harvesting times according to (**a**). (**d**) Accessibility z-scores of HSC-MPP-LMPP single cells inferred by epiConv. (**e**) Accessibility profiles of HSC, MPP and LMPP bulk samples and LMPP, GMP and CLP bulk samples. (**f**) Log2 Fold Change of peaks between HSC, MPP and LMPP bulk samples. Points represent the corresponding peaks in (**d,e**)

Reference:

1. Buenrostro, J.D., Giresi, P.G., Zaba, L.C., Chang, H.Y. & Greenleaf, W.J. Transposition of native chromatin for fast and sensitive epigenomic profiling of open chromatin, DNA-binding proteins and nucleosome position. *Nat Methods* **10**, 1213-1218 (2013).
2. Klemm, S.L., Shipony, Z. & Greenleaf, W.J. Chromatin accessibility and the regulatory epigenome. *Nat Rev Genet* **20**, 207-220 (2019).
3. Cusanovich, D.A. et al. Multiplex single cell profiling of chromatin accessibility by combinatorial cellular indexing. *Science* **348**, 910-914 (2015).
4. Satpathy, A.T. et al. Massively parallel single-cell chromatin landscapes of human immune cell development and intratumoral T cell exhaustion. *Nat Biotechnol* **37**, 925-936 (2019).
5. Lareau, C.A. et al. Droplet-based combinatorial indexing for massive-scale single-cell chromatin accessibility. *Nat Biotechnol* **37**, 916-924 (2019).
6. Buenrostro, J.D. et al. Single-cell chromatin accessibility reveals principles of regulatory variation. *Nature* **523**, 486-490 (2015).
7. Chen, H. et al. Assessment of computational methods for the analysis of single-cell ATAC-seq data. *Genome Biol* **20**, 241 (2019).
8. Cusanovich, D.A. et al. A Single-Cell Atlas of In Vivo Mammalian Chromatin Accessibility. *Cell* **174**, 1309-1324 e1318 (2018).
9. Pliner, H.A. et al. Cicero Predicts cis-Regulatory DNA Interactions from Single-Cell Chromatin Accessibility Data. *Mol Cell* **71**, 858-871 e858 (2018).
10. 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).
11. Bravo Gonzalez-Blas, C. et al. cisTopic: cis-regulatory topic modeling on single-cell ATAC-seq data. *Nat Methods* **16**, 397-400 (2019).
12. Corces, M.R. et al. Lineage-specific and single-cell chromatin accessibility charts human hematopoiesis and leukemia evolution. *Nat Genet* **48**, 1193-1203 (2016).
13. Zhang, Y. et al. Model-based analysis of ChIP-Seq (MACS). *Genome Biol* **9**, R137 (2008).
14. Langmead, B. & Salzberg, S.L. Fast gapped-read alignment with Bowtie 2. *Nat Methods* **9**, 357-359 (2012).
15. Kent, W.J. et al. The human genome browser at UCSC. *Genome Res* **12**, 996-1006 (2002).
16. McInnes, L., Healy, J., Saul, N. & Großberger, L. UMAP: Uniform Manifold Approximation and Projection. *Journal of Open Source Software* **3** (2018).

Supplementary materials

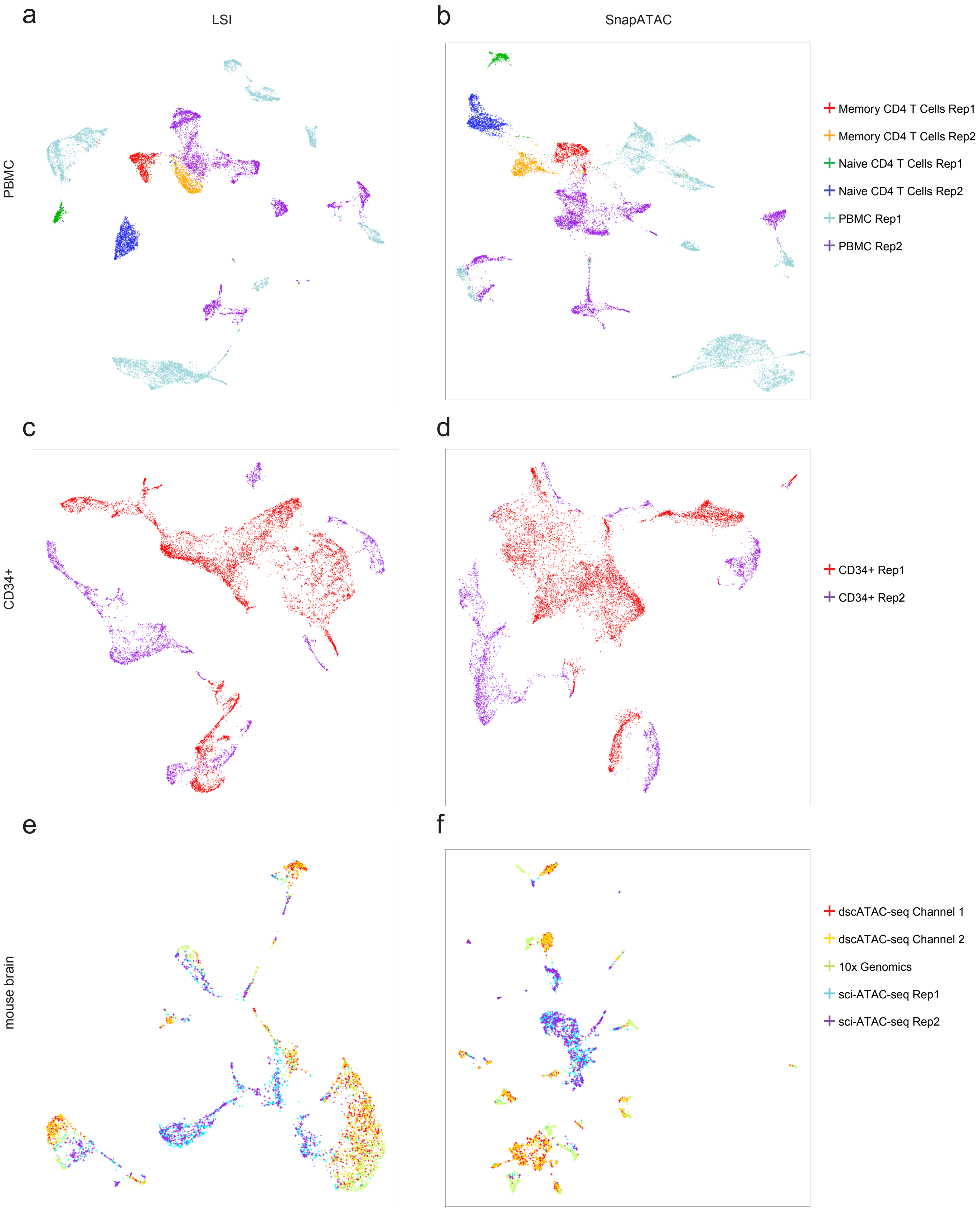
a
LSI
b
SnapATAC
PBMC
+ Memory CD4 T Cells Rep1
+ Memory CD4 T Cells Rep2
+ Naive CD4 T Cells Rep1
+ Naive CD4 T Cells Rep2
+ PBMC Rep1
+ PBMC Rep2
c
d
CD34+
+ CD34+ Rep1
+ CD34+ Rep2
e
f
mouse brain
+ dscATAC-seq Channel 1
+ dscATAC-seq Channel 2
+ 10x Genomics
+ sci-ATAC-seq Rep1
+ sci-ATAC-seq Rep2

Figure S1. Embeddings of PBMC (**a,b**), CD34+ (**c,d**) and mouse brain (**e,f**) datasets by LSI and SnapATAC .

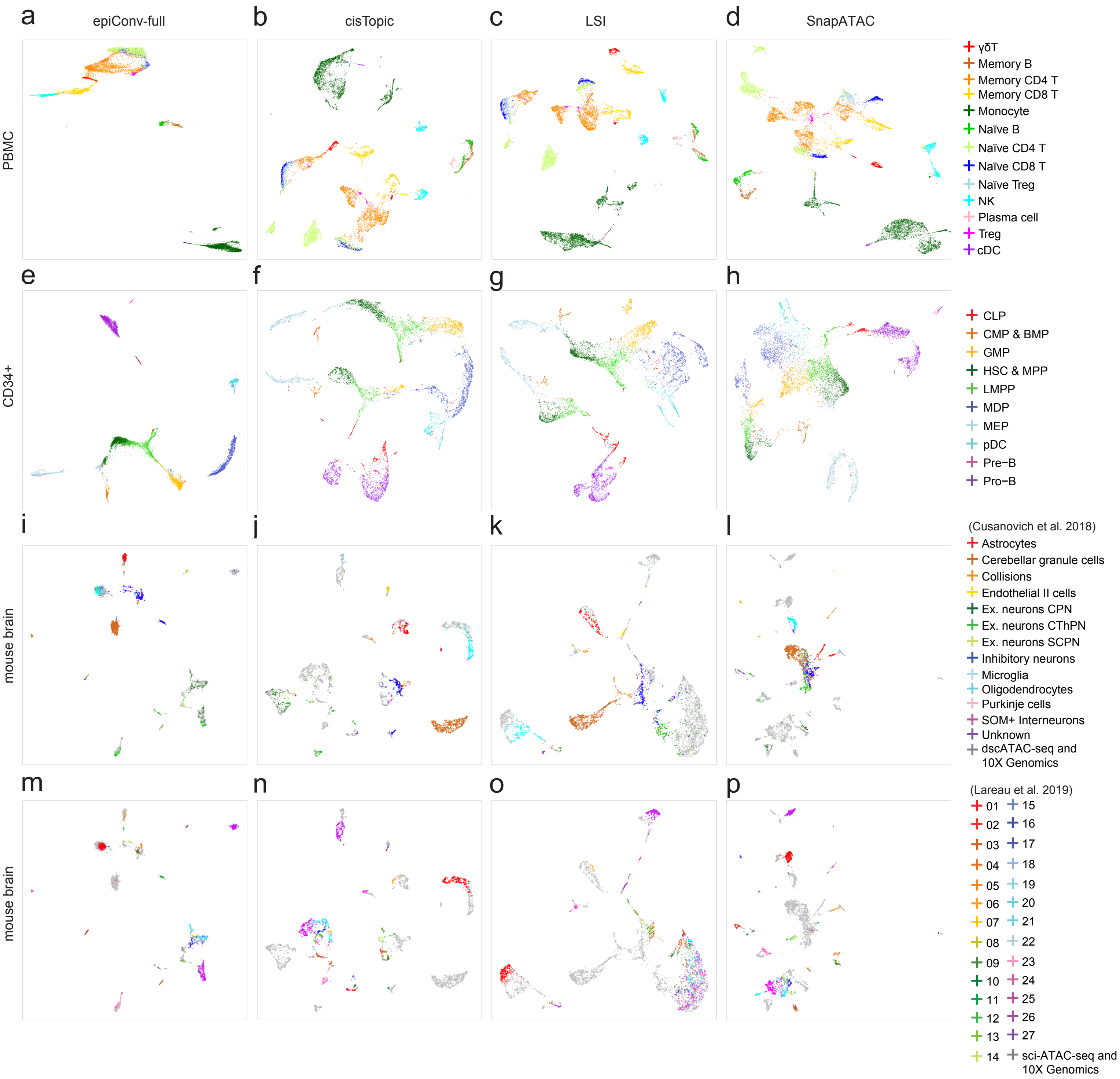
a
epiConv-full
b
cisTopic
c
LSI
d
SnapATAC
PBMC
γδT
Memory B
Memory CD4 T
Memory CD8 T
Monocyte
Naïve B
Naïve CD4 T
Naïve CD8 T
Naïve Treg
NK
Plasma cell
Treg
cDC
e
f
g
h
CD34+
CLP
CMP & BMP
GMP
HSC & MPP
LMPP
MDP
MEP
pDC
Pre−B
Pro−B
i
j
k
l
mouse brain
(Cusanovich et al. 2018)
Astrocytes
Cerebellar granule cells
Collisions
Endothelial II cells
Ex. neurons CPN
Ex. neurons CThPN
Ex. neurons SCPN
Inhibitory neurons
Microglia
Oligodendrocytes
Purkinje cells
SOM+ Interneurons
Unknown
dscATAC-seq and 10X Genomics
m
n
o
p
mouse brain
(Lareau et al. 2019)
01
02
03
04
05
06
07
08
09
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
sci-ATAC-seq and 10X Genomics

Figure S2. Comparison of embeddings by epiConv-full, cisTopic, LSI and SnapATAC with cell annotations from original articles. (**a-d**) Embeddings of PBMC dataset by epiConv-full, cisTopic, LSI and SnapATAC, colored by annotations from Satpathy et al. 2019. (**e-h**) Embeddings of CD34+ dataset by epiConv-full, cisTopic, LSI and SnapATAC, colored by annotations from Satpathy et al. 2019. (**i-p**) Embeddings of mouse brain dataset by epiConv-full, cisTopic, LSI and SnapATAC, colored by annotations from Cusanovich et al. 2018 (**i-l**) and Lareau et al. 2019 (**m-p**).

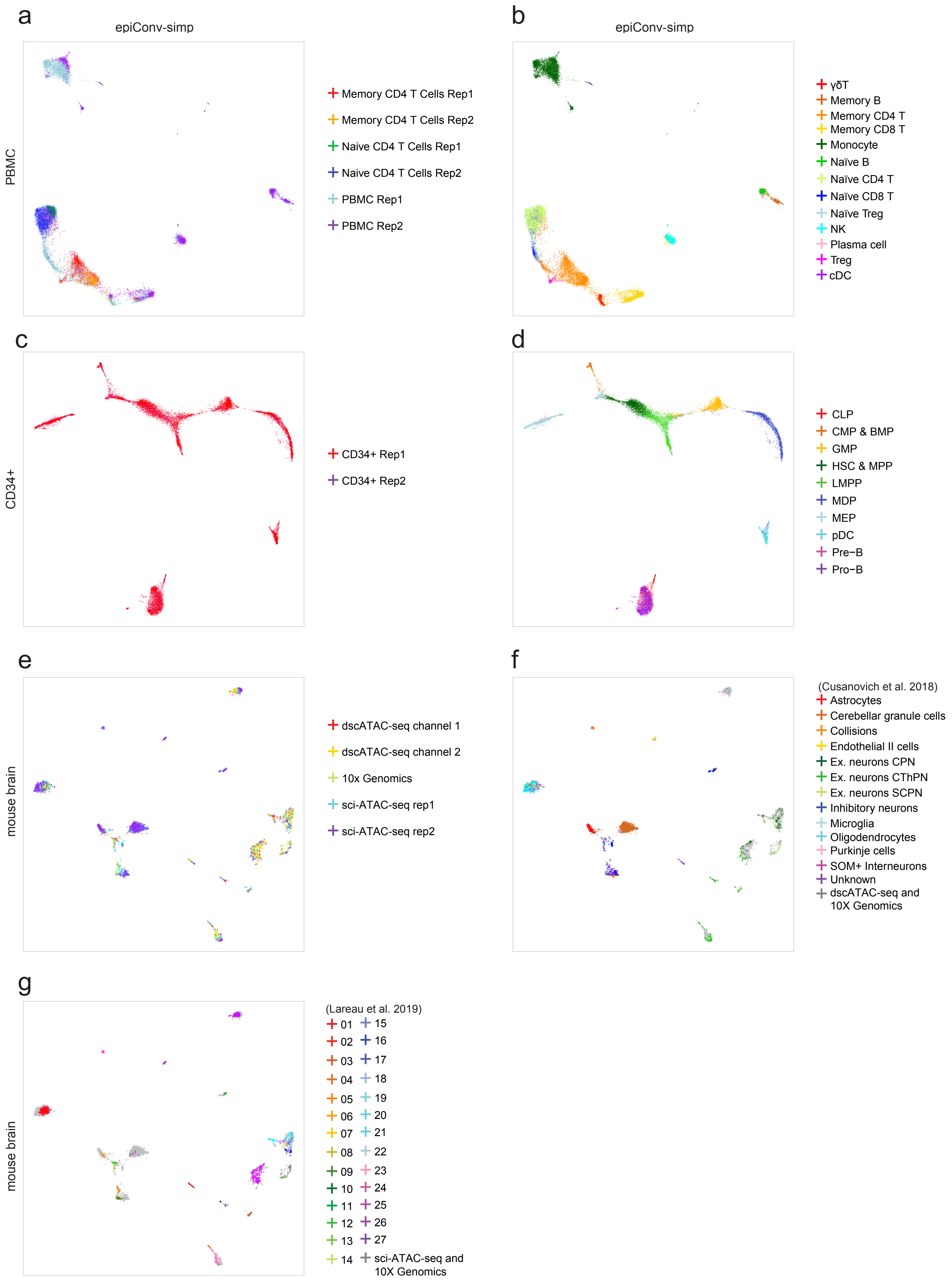
a
epiConv-simp
PBMC
Memory CD4 T Cells Rep1
Memory CD4 T Cells Rep2
Naive CD4 T Cells Rep1
Naive CD4 T Cells Rep2
PBMC Rep1
PBMC Rep2
b
epiConv-simp
γδT
Memory B
Memory CD4 T
Memory CD8 T
Monocyte
Naïve B
Naïve CD4 T
Naïve CD8 T
Naïve Treg
NK
Plasma cell
Treg
cDC
c
CD34+
CD34+ Rep1
CD34+ Rep2
d
CLP
CMP & BMP
GMP
HSC & MPP
LMPP
MDP
MEP
pDC
Pre−B
Pro−B
e
mouse brain
dscATAC-seq channel 1
dscATAC-seq channel 2
10x Genomics
sci-ATAC-seq rep1
sci-ATAC-seq rep2
f
(Cusanovich et al. 2018)
Astrocytes
Cerebellar granule cells
Collisions
Endothelial II cells
Ex. neurons CPN
Ex. neurons CThPN
Ex. neurons SCPN
Inhibitory neurons
Microglia
Oligodendrocytes
Purkinje cells
SOM+ Interneurons
Unknown
dscATAC-seq and 10X Genomics
g
mouse brain
(Lareau et al. 2019)
01
02
03
04
05
06
07
08
09
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
sci-ATAC-seq and 10X Genomics

Figure S3. Embeddings of PBMC, CD34+ and mouse brain datasets by epiConv-simp. (**a,b**) Embeddings of PBMC dataset by epiConv-simp, colored by batch (**a**) and annotations from Satpathy et al. 2019 (**b**). (**c,d**) Embeddings of CD34+ dataset by epiConv-simp, colored by batch (**c**) and annotations from Satpathy et al. 2019 (**d**). (**e-g**) Embeddings of mouse brain dataset by epiConv-simp, colored by batch (**e**), annotations from Cusanovich et al. 2018 (**f**) and annotations from Lareau et al. 2019 (**g**).

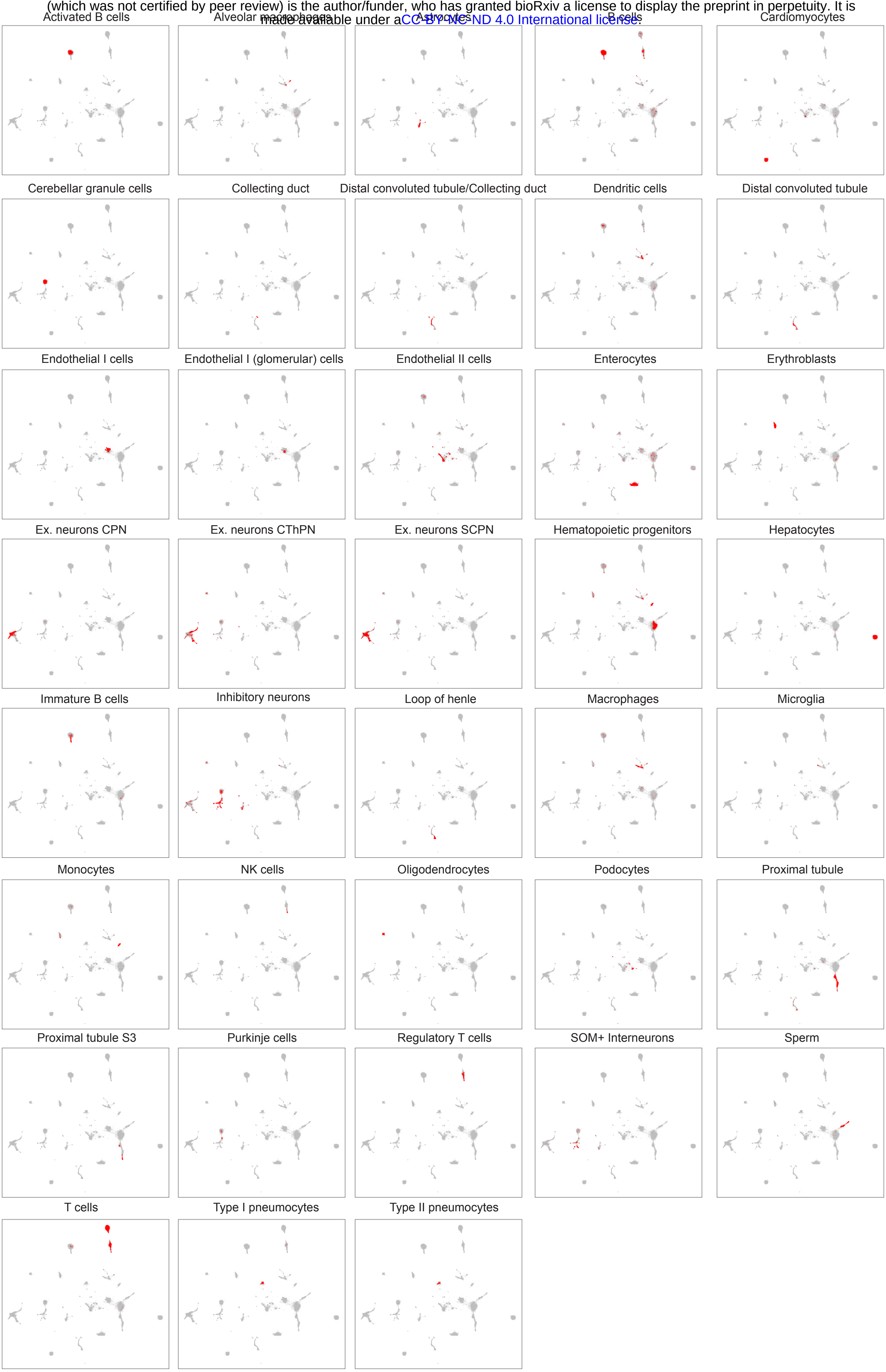
Activated B cells
Alveolar macrophages
Astrocytes
B cells
Cardiomyocytes
Cerebellar granule cells
Collecting duct
Distal convoluted tubule/Collecting duct
Dendritic cells
Distal convoluted tubule
Endothelial I cells
Endothelial I (glomerular) cells
Endothelial II cells
Enterocytes
Erythroblasts
Ex. neurons CPN
Ex. neurons CThPN
Ex. neurons SCPN
Hematopoietic progenitors
Hepatocytes
Immature B cells
Inhibitory neurons
Loop of henle
Macrophages
Microglia
Monocytes
NK cells
Oligodendrocytes
Podocytes
Proximal tubule
Proximal tubule S3
Purkinje cells
Regulatory T cells
SOM+ Interneurons
Sperm
T cells
Type I pneumocytes
Type II pneumocytes

Figure S4. Embeddings of Mouse Cell Atlas dataset by epiConv-simp. The corresponding cell types are colored in red and other cells are colored in grey.

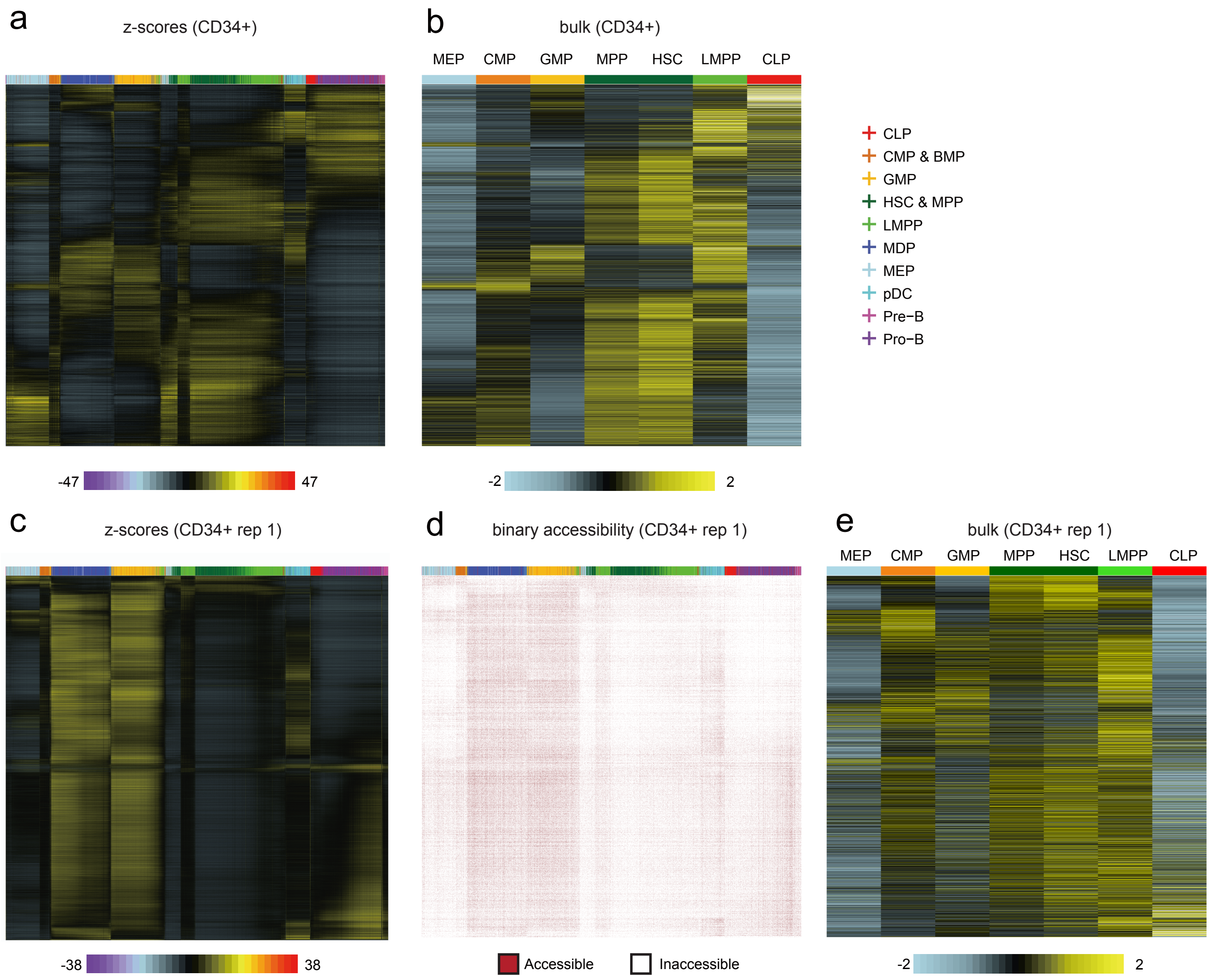
a
z-scores (CD34+)
-47
47
b
bulk (CD34+)
MEP
CMP
GMP
MPP
HSC
LMPP
CLP
-2
2
CLP
CMP & BMP
GMP
HSC & MPP
LMPP
MDP
MEP
pDC
Pre−B
Pro−B
c
z-scores (CD34+ rep 1)
-38
38
d
binary accessibility (CD34+ rep 1)
Accessible
Inaccessible
e
bulk (CD34+ rep 1)
MEP
CMP
GMP
MPP
HSC
LMPP
CLP
-2
2

Figure S5. Accessibility z-scores of CD34+ dataset. (**a**) Accessibility z-scores of CD34+ single cells (**b**) Accessibility profiles of bulk CD34+ samples. (**c**) Accessibility z-scores of CD34+ single cells in replicate 1. (**d**) Binary accessibility profiles of CD34+ single cells in replicate 1. (**e**) Accessibility profiles of bulk CD34+ samples. DE peaks in (**a,b**) are selected from whole CD34+ dataset by setting the scaling factors of cells to their library sizes and DE peaks in (**c-e**) are selected from replicate 1 in CD34+ dataset by setting the scaling factors of cells to 1.

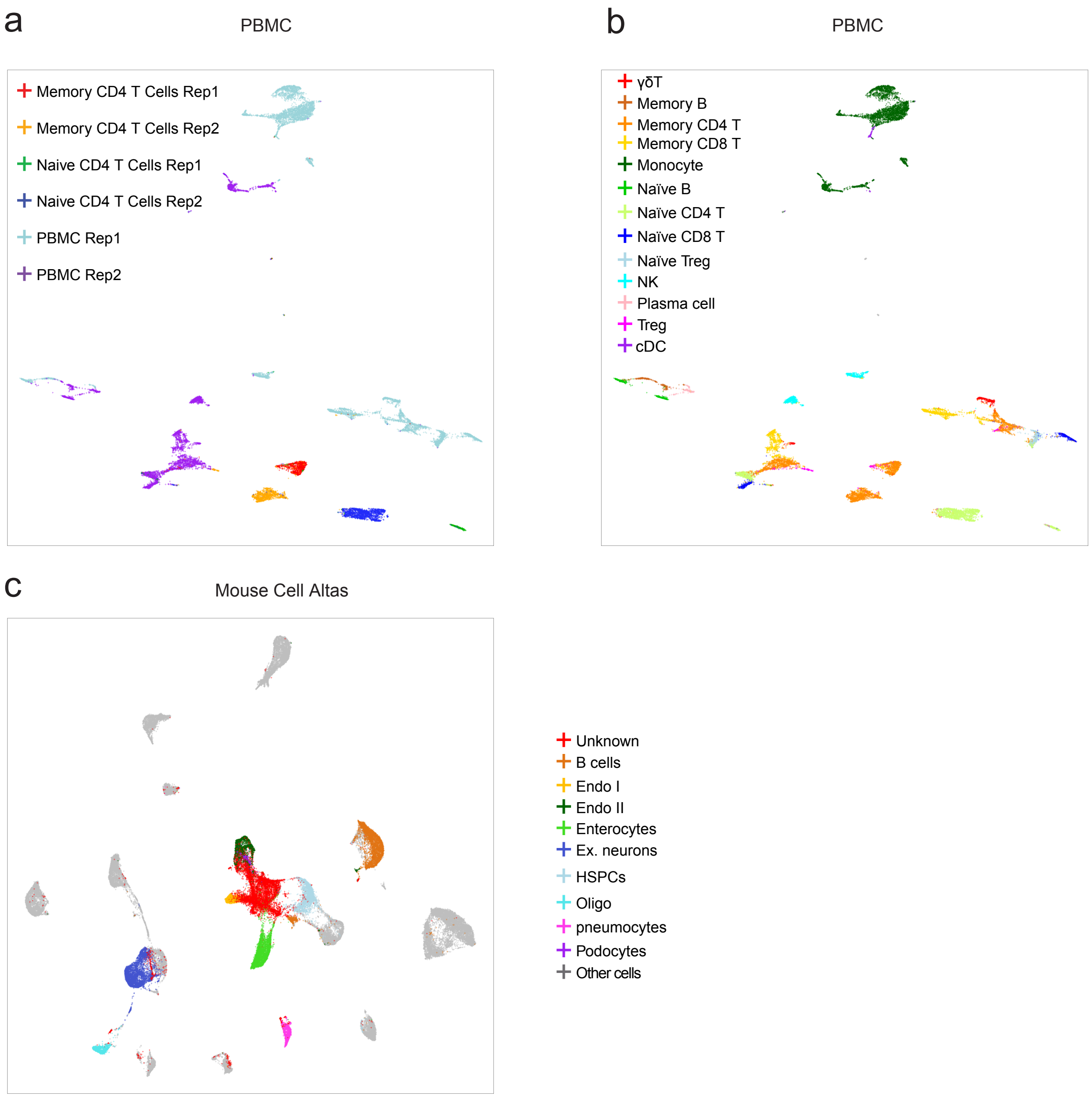
a
PBMC
Memory CD4 T Cells Rep1
Memory CD4 T Cells Rep2
Naive CD4 T Cells Rep1
Naive CD4 T Cells Rep2
PBMC Rep1
PBMC Rep2
b
PBMC
γδT
Memory B
Memory CD4 T
Memory CD8 T
Monocyte
Naïve B
Naïve CD4 T
Naïve CD8 T
Naïve Treg
NK
Plasma cell
Treg
cDC
c
Mouse Cell Altas
Unknown
B cells
Endo I
Endo II
Enterocytes
Ex. neurons
HSPCs
Oligo
pneumocytes
Podocytes
Other cells

Figure S6. Embeddings of PBMC and Mouse Cell Atlas dataset by inferring latent features from the similarity matrix of epiConv. (**a**) Embedding of PBMC dataset, colored by batch. (**b**) Embedding of PBMC dataset, colored by annotations from Satpathy et al. 2019. (**c**) Embedding of Mouse Cell Atlas dataset, colored by annotations from Cusanovich et al. 2018.

Supplementary Note 1

Here, we describe an alternative denoising method that keeps the unit of similarity matrix unchanged. Given N cells and their similarity matrix S where s_{ij} is the similarity between cell *i* and *j*, we first transform S to a weight matrix W as follows:

$$w_{ij} = \begin{cases} 10^{s_{ij}} \cdot \log10(lib_i), i \in j's\ neighbors \\ 0, \qquad\qquad\qquad\quad i \notin j's\ neighbors \end{cases}$$

Where *j*'s neighbors are the top 20 cells with highest similarities to *j*. For each column *j*, we scale the sum of column (excluding the diagonal elements) to a fraction parameter θ between 0 and 1 and the diagonal elements of W are set to $1 - \theta$. Then the sum of each column is equal to 1. The matrix W defines how to mix the information from the cell itself and its neighbors, where θ proportion of information comes from its neighbors and the weight of each neighbor is determined by its similarity to cell *j* multiplied by its log10 library size, and $1 - \theta$ proportion of information comes from cell *j* itself. In this study, we set θ to 0.25. We create a similarity matrix S' where its elements are equal to S except for the diagonal elements (the similarity of each cell to itself, which is not defined for S). The diagonal element s'_{jj} is set to the 99th percentile of column *j*, which can be used to approximate the similarity of cell *j* to itself. The denoised similarity matrix $S_{denoise}$ is calculated by matrix product of S' and W as follows:

$$S_{denoise} = \frac{S' \cdot W + (S' \cdot W)^T}{2}$$

Given $S' \cdot W$ is not a symmetrical matrix, we average $S' \cdot W$ and $(S' \cdot W)^T$ to obtain the denoised matrix. As a proof of the reliability of our algorithm, the upper triangle and lower triangle of $S' \cdot W$ are always close to each other. The distance matrix D is calculated by $\mathrm{D} = -S_{denoise}$. Compared to the denoising method described in Methods, the alternative method denoises the data and largely keeps the information of original matrix (including variations from both batch effects and biological heterogeneity).