Simultaneous deep generative modeling and clustering of single cell genomic data

Qiao Liu[1,2], Shengquan Chen[1], Rui Jiang[1,*] and Wing Hung Wong[2,3,*]

[1]MOE key Laboratory of Bioinformatics, Bioinformatics Division, Beijing National Research Center for Information Science and Technology, Department of Automation, Tsinghua University, Beijing 100084, China; [2]Department of Statistics, Stanford University, Stanford, CA 94305, USA; [3]Department of Biomedical Data Science, Bio-X Program, Center for Personal Dynamic Regulomes, Stanford University, Stanford, CA 94305, USA.

* Corresponding authors:

ruijiang@tsinghua.edu.cn; whwong@stanford.edu

Lead Contact: whwong@stanford.edu

Recent advances in single-cell technologies, including single-cell ATAC-seq (scATAC-seq), have enabled large-scale profiling of the chromatin accessibility landscape at the single cell level. However, the characteristics of scATAC-seq data, including high sparsity and high dimensionality, have greatly complicated the computational analysis. Here, we proposed scDEC, a computational tool for single cell ATAC-seq analysis with deep generative neural networks. scDEC is built on a pair of generative adversarial networks (GANs), and is capable of learning the latent representation and inferring the cell labels, simultaneously. In a series of experiments, scDEC demonstrates superior performance over other tools in scATAC-seq analysis across multiple datasets and experimental settings. In the downstream applications, we demonstrated that the generative power of scDEC helps to infer the trajectory and intermediate state of cells during differentiation and the latent features learned by scDEC can potentially reveal both biological cell types and within-cell-type variations.

The organization of chromatin accessibility across the whole genome reflects an epigenetic landscape of gene regulation[1,2]. With the recent development in single-cell technology, it becomes feasible to characterize the epigenetic landscape of individual cells[3]. In particular, single-cell ATAC-seq (scATAC-seq) is an efficient method for the study of variation in chromatin accessibility both between and within populations at single cell level[4,5]. However, the analysis of scATAC-seq presents unique methodological challenges due to the high dimensionality (hundreds of thousands possible peaks) and high data sparsity (only 1-10% peaks are detected per cell) [6].

Several computational approaches have been proposed to tackle the challenges in scATAC-seq analysis. scABC estimated weights of cells based on the number of distinct reads and applied a weighted *k*-medoids clustering to infer cell types[7]. cisTopic applied latent Dirichlet allocation (LDA) as a probabilistic model to identify the *cis*-regulatory topics enriched in different cells by optimizing topic-cell probability and region-topic probability simultaneously[8]. Cusanovich et al. proposed a pipeline which performs the term frequency-inverse document frequency transformation (TF-IDF)

and singular value decomposition (SVD) iteratively to get a low dimensional representation of scATAC-seq data[4,9]. Scasat introduced another pipeline which involved Jaccard similarity measure and multidimensional scaling (MDS) to reduce the high dimensionality in scATAC data[10]. SnapATAC divided genome into bins with equal size and builds a bins-by-cells binary count matrix and then applied principle component analysis (PCA) for a dimension reduction[11]. Recently, deep generative models have emerged as a powerful framework for both representation learning and data generation[12-14]. A newly developed method SCALE utilized a variational autoencoder (VAE) to learn the latent features of scATAC-seq data and then used a *K*-means by default for clustering the latent features[15].

Here, we proposed a new approach for analyzing **sc**ATAC-seq data by simultaneously learning the **D**eep **E**mbedding and **C**lustering of the cells in an unsupervised manner. Our method, named scDEC, was based on learning a pair of generative adversarial networks (GANs) (Fig. 1). Such a symmetrical and paired GAN architecture has been recently successfully applied to image style transfer[16] and density estimation[17]. Here, we adopted this architecture to unsupervised clustering and applied it to the analysis of single cell genomic data. Unlike all current methods discussed above, where an external method (e.g., *K*-means) is typically required for clustering the latent features, in our method the cell clustering process is directly modeled by neural networks. Thus, cell clustering and latent feature representation learning will be jointly optimized during the training process. In other words, scDEC enables simultaneous learning of low dimensional embedding and cell clustering. We demonstrate the advantage of this approach in a series of experiments, where scDEC is shown to outperform other current methods. We also illustrated several downstream applications of scDEC in scATAC-seq analysis, including trajectory inference, donor effect removal and latent feature interpretation.

Results

Overview of scDEC model

scDEC consists of two GAN models, which are utilized for transformations between latent space and data space (Fig. 1). The scATAC-seq data will first be preprocessed through TF-IDF transformation and a PCA dimension reduction before it is fed to the scDEC model. Assume the input scATAC-seq data contains K cell types, a continuous latent variable $\boldsymbol{z}$ and a discrete latent variable $\boldsymbol{c}$ will be introduced, where $\boldsymbol{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and $\mathbf{c} \sim \mathrm{Cat}(K, \boldsymbol{w})$, respectively. We also provided an approach for estimating the number of cell subpopulations if K is unknown (Methods). The forward transformation through the G network can be considered as a process of conditional generation given an encoded style ($\boldsymbol{z}$) and an indicated cluster label ($\boldsymbol{c}$). The backward transformation through the H network aims at encoding a data point $\boldsymbol{x}$ to the latent space and inferring the cluster label, simultaneously. If we assume the last layer of H network contains m nodes ($m>K$), then $\tilde{\boldsymbol{z}}$ denotes the output of the first (m

-K) nodes and $\tilde{\boldsymbol{c}}$ denotes the output of the remaining K nodes with an additional softmax function. $D_{\boldsymbol{x}}$ and $D_{\boldsymbol{z}}$ are two discriminator networks which are used for matching the distributions of data $\tilde{\boldsymbol{x}}$ and $\tilde{\boldsymbol{z}}$ to the empirical distribution of the data and latent variable distribution, respectively. (G, $D_{\boldsymbol{x}}$) and (H, $D_{\boldsymbol{z}}$) can be considered as two GAN models that are jointly trained. The G and H network each contains 10 fully-connected layers while $D_{\boldsymbol{x}}$ and $D_{\boldsymbol{z}}$ each has two fully-connected layers (see detailed hyperparameters in Supplementary Table 1). Note that the weights $\boldsymbol{w}$ in the Category distribution is also learned automatically via an updating scheme according to the feedback of inferred cluster labels

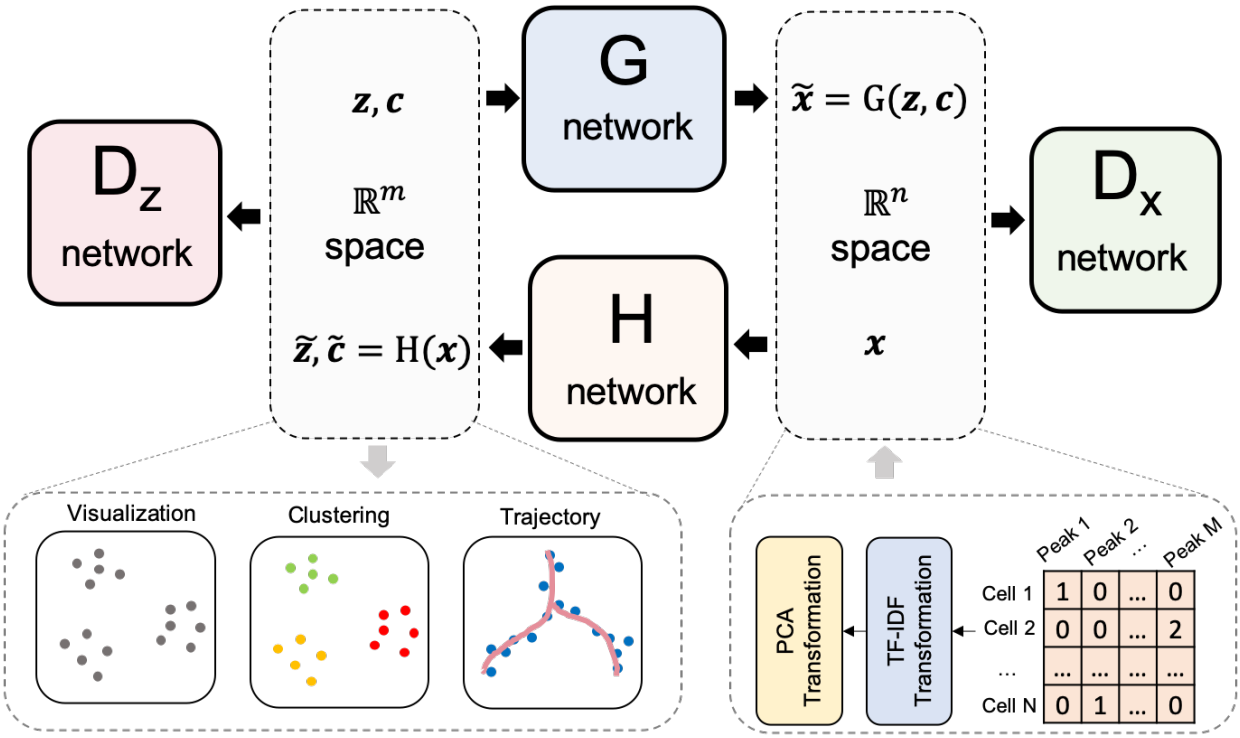

Fig. 1. The illustration of scDEC model. The read count matrix of scATAC-seq will first be preprocessed by a TF-IDF transformation and a PCA dimension reduction (e.g., n=20) before it is fed to the scDEC model. In the latent space, latent variables $\boldsymbol{z}$ and $\boldsymbol{c}$ sampled from a Gaussian distribution and a Category distribution respectively, will be concatenated together before they are fed to the G network. The H network has two outputs of which one corresponds to the latent embedding ($\tilde{\boldsymbol{z}}$) and one corresponds to the estimated cluster label ($\tilde{\boldsymbol{c}}$) through a softmax function. The $D_{\boldsymbol{x}}$ network works as a discriminator for discerning the true scATAC-seq data ($\boldsymbol{x}$) from the generated data ($\tilde{\boldsymbol{x}}$). The $D_{\boldsymbol{z}}$ network is another discriminator for distinguishing the learned continuous latent variable ($\tilde{\boldsymbol{z}}$) from the real continuous latent variable ($\boldsymbol{z}$).

by $\tilde{\boldsymbol{c}}$ (Methods). After model training, the cluster label can be inferred based on $\tilde{\boldsymbol{c}}$ (Methods). The output of the last layer of H network combined with $\tilde{\boldsymbol{z}}$ and $\tilde{\boldsymbol{c}}$ (before softmax) are useful for downstream analysis such as data visualization and trajectory analysis.

scDEC automatically identifies cell types in scATAC-seq data

To demonstrate the ability of scDEC for revealing differences between different cell subpopulations and identifying cell types in an unsupervised manner, we test it on four benchmark scATAC-seq datasets across different number of cells and cell types (Supplementary Figure 1 and Table 2). Specifically, scDEC was benchmarked against six current methods, including scABC[7], SCALE[15], cisTopic[8], Cusanovich2018[4,9], Scasat[10] and SnapATAC[11] (Methods). The performance of a method is evaluated on 1) whether different cell subpopulations can be clearly separated in a low-dimensional space, and 2) whether true cell type labels can be accurately inferred by clustering. To address the first question, we first applied each method to conduct a dimension reduction or to extract the latent features. The latent dimension is set to 15 for the two datasets with relatively smaller number of cells and cell populations, and 20 for the two larger datasets. For each method, we constructed a t-SNE[18] or UMAP[19] plot based on the latent features and then visualized the truth cell labels on the plot to see whether the subpopulations were well separated. To address the second question, for each method we compare its clustering results to the true subpopulations based on three commonly used metrics, namely Normalized Mutual Information (NMI), Adjusted Rand Index (ARI) and Homogeneity score (Homogeneity) (Methods). The results are summarized below. We note that scDEC's performance is not sensitive to the dimension of latent space (Supplementary Figure 2)

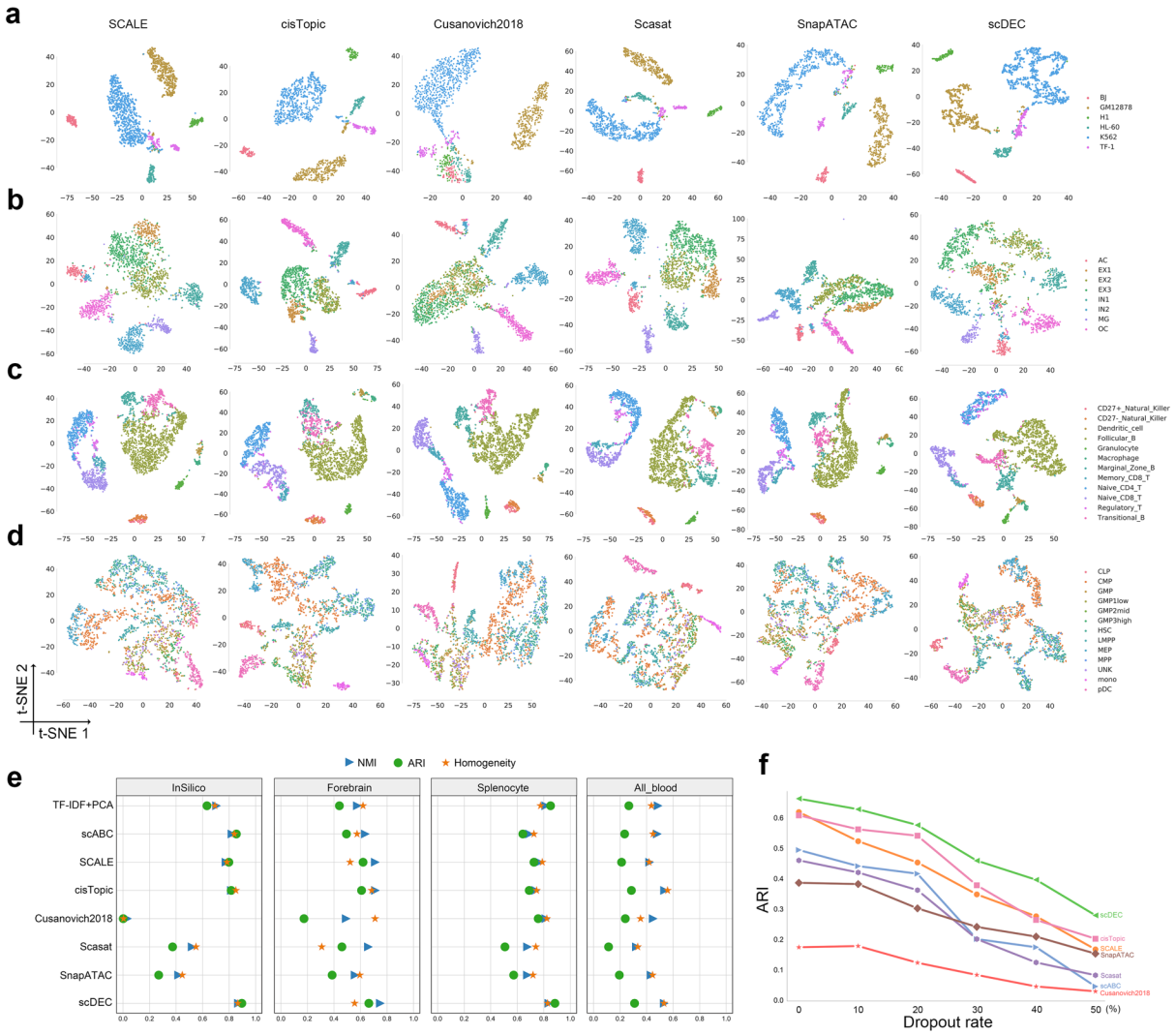

Fig. 2. Evaluation of scDEC compared with other baseline methods. **a**. Visualization of InSilico dataset by different methods. **b**. Visualization of Forebrain dataset by different methods. **c**. Visualization of Splenocyte dataset by different methods. **d**. Visualization of All_blood dataset by different methods. **e**. Clustering results of different methods across four datasets. **f**. Performance of different methods under different dropout rate on the Forebrain dataset.

InSilico dataset[5]. This dataset is an in silico mixture constructed by artificially combining six individual scATAC-seq experiments which were separately conducted on a different cell line. We observed that cells from a minor cell type TF-1 (6.83%, in purple) were dispersed into several clusters by SCALE, Cusanovich2018, Scasat and SnapATAC while cisTopic and scDEC can well maintain

the close distance in the low-dimensional representation (Fig. 2a). scDEC achieves an NMI of 0.871, an ARI of 0.896, and a Homogeneity of 0.866, which outperforms the best baseline method scABC (NMI=0.822, AIR=0.855, and Homogeneity=0.840) by a large margin (Fig. 2e and Supplementary Figure 3).

Forebrain dataset[20]. This dataset was derived from P56 mouse forebrain cells which contained eight different cell groups in adult mouse forebrain. Interestingly, all the baseline methods failed to distinguish three subtypes of excitatory neuron cells (EX1, EX2 and EX3) while scDEC showed a relatively clear separation among these three subpopulations of cells (Fig. 2b). Again, scDEC demonstrates a superior clustering performance by achieving the highest NMI of 0.750 and ARI of 0.663 (Fig. 2e and Supplementary Figure 4).

Splenocyte dataset[21]. This dataset was collected from a mixture of mouse splenocytes after removing red blood cells, which finally resulted in 12 cell subpopulations. A major cell type follicular B cells (42.89%, in brown) and two subtypes of CD8 cells are more or less mixed together with other subpopulations of cells by all baseline methods while scDEC illustrates a clearer separation (Fig. 2c). As the largest dataset (around 3k cells) among the four datasets, scDEC still achieves the highest NMI of 0.839, ARI of 0.884 and Homogeneity of 0.829 (Fig. 2e and Supplementary Figure 5).

All blood dataset[22]. This dataset involves cellular differentiation of multipotent cells during human hematopoiesis, containing 13 subpopulations of cells in total. Three types of cells, including monocyte cells (mono), plasmacytoid dendritic cells (pDC) and CLP cells, can only be separated from other cells by cisTopic, Scasat and scDEC (Fig. 2d). scDEC ranks first in both ARI (0.309) and Homogeneity (0.531) and ranks second in ARI (0.533), which is slightly lower than cisTopic (ARI=0.536) (Fig. 2e and Supplementary Figure 6).

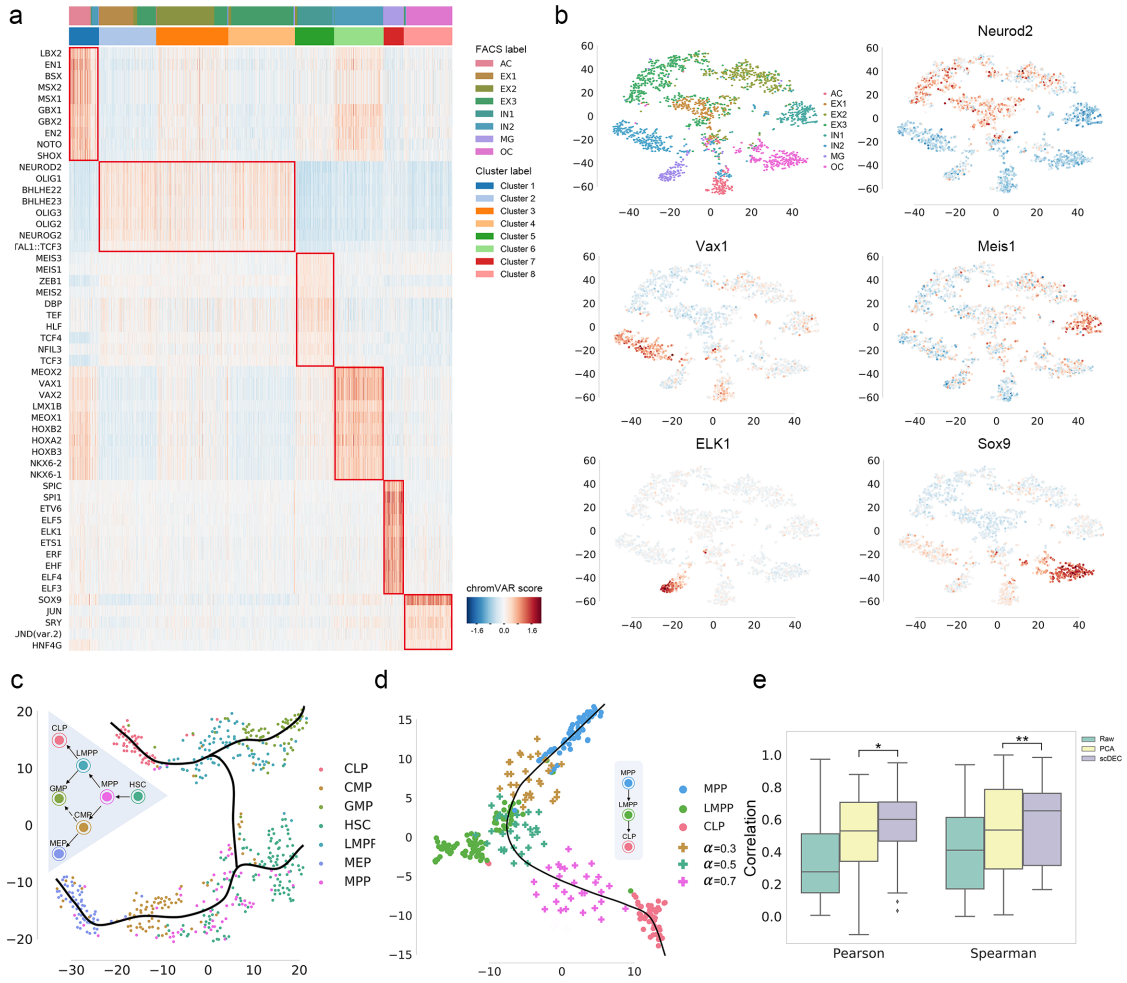

Fig. 3. Cluster-specific motif recovery and trajectory inference. **a**. Heatmap of enriched motifs, each row denotes a motif and each column denotes a cell. Both cluster label and FACS label were provided and aligned. **b**. The t-SNE visualization of several literature-validated motifs. **c**. The hematopoiesis differentiation trajectory inferred by scDEC. **d**. The generated intermediate state between MPP and CLP. 30 data points were generated at different generation coefficient α. **e**. The generated intermediate scATAC data by interpolation on the latent label indicator has a higher correlation with the meta cell (the average profile of ground truth cells) than the scATAC-seq that were directly interpolated on the raw data and PCA reduced data. * p-value$<1.28\times10^{-16}$, ** p-value$< 4.40\times10^{-8}$

Next, we further investigated the performance of different methods at different dropout rate, in order to assess the ability of handing scATAC-seq data with different degree of sparsity. We downsampled the original reads in the Forebrain dataset by randomly dropped out the non-zero

entities in the read count matrix with probability equal to the dropout rate. scDEC consistently demonstrates the best performance *w.r.t* the ARI metric for clustering at different dropout rate ranging from 0 to 50%. At the dropout rate of 50%, scDEC achieves an ARI of 0.279, compared to 0.202 of the best baseline cisTopic (Fig. 2f).

scDEC facilitates cell type-specific motif discovery and trajectory inference

We next explored whether scDEC can help identity cell-type specific motifs, which is essential for understanding the context-specific gene regulation. To achieve this, we first applied scDEC model to the mouse forebrain dataset[20] to infer the cluster label for each individual cell, and used chromVAR[23] to identify cluster-specific enriched motifs from the JASPAR database[24]. We then ranked cluster-specific enriched motifs (Methods) and discovered several significant motif enrichment patterns (Fig. 3a, Supplementary Table 3). We observed both single cluster-specific motifs and the co-occurrence of motifs in two (cluster 1 and 6) or three clusters (cluster 2,3 and 4), which might reveal the co-regulation mechanism underlying the corresponding multiple TFs. For example, En1, which is enriched in cluster 1, is a well-known marker for the brain fate in astrocytes (AC)[25]. It was reported that Neurod2 regulates the cortical projection neuron which constitute the major excitatory neuron (EX) population[26]. Meis1 was proved to reveal crucial functions in neural differentiation from neural progenitors[27]. Vax1 is a novel homeobox-containing gene that regulates the development of the basal forebrain[28]. The impact of Elk1 deficiency was proved to indicate the microglial (MG) activation[29]. The compound loss of Sox9 will lead to a further decrease in oligodendrocyte (OC) progenitors[30]. These literature-validated motifs were demonstrated in the t-SNE visualization according to the enrichment score calculated by chromVAR (Fig. 3b).

Next, we applied scDEC to trajectory inference during the hematopoiesis differentiation. We collected the cells from the donor BM0828 of the All blood dataset, which contains 533 cells across

7 subpopulations at different stage of differentiation. After obtaining the low-dimensional representation and t-SNE projection of scATAC-seq data, we can annotate the smooth curves which represent different cell lineages with the help of Slingshot software[31] (Fig. 3c). The smooth curves with a tree-based structure are largely consistent with the true hematopoietic differentiation tree. Although it has been proved that CMP can differentiate into both GMP and MEP[32], only differentiation path from CMP to MEP was observed in this dataset. We then took the cells from MPP, LMPP and CLP for a further study, where there exists a differentiation path (MPP→LMPP→CLP). To fully exploit the generation power of scDEC, we first left LMPP out as the target cells for imputation and trained scDEC based on the remaining cells composing of only MPP and CLP cells. Then we imputed data by interpolating the latent label indicator (Methods) and visualized the imputed data together with the true data. Interestingly, when the interpolation coefficient α changes from 0 to 1, the imputed data seem to capture the dynamics differentiation path from MPP to CLP. Specifically, the generated scATAC-seq data are similar to the real LMPP data according to t-SNE visualization when $\alpha = 0.5$ (Fig. 3d). Next, we asked whether the interpolation on the latent indicator is a more effective way of data generation than directly interpolating on the raw scATAC-seq. We averaged all the scATAC-seq data of LMPP cells as a meta-cell and calculated the Pearson correlation between generated data and meta-cell. The generated data by scDEC achieves a significantly higher correlation than generated data by direct interpolation and interpolation on PCA reduced data (Fig. 3e and Supplementary Table4). To sum up, the generation power of scDEC shed light on recovering the missing cell types of scATAC data and exploring the intermediate state of two neighboring cell types of scATAC-seq data.

scDEC disentangles donor effect and promotes interpretation of latent features

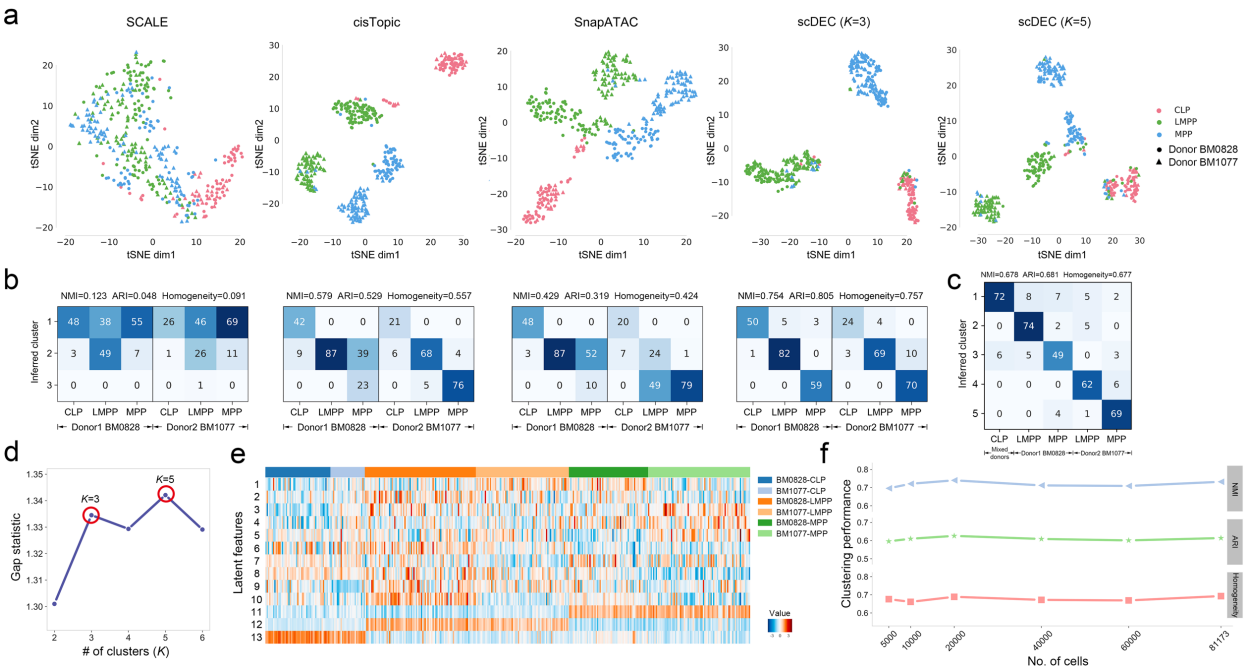

Fig. 4. scDEC alleviates donor effect and is robust to large dataset. **a**. The t-SNE visualization of the latent features learned by different methods. Different colors denote different cell types and different shape (circle or triangle) represents which donor it comes from. For scDEC, different K (3 and 5) results in different latent features visualization. **b**. The confusion matrix of the clustering by scDEC and comparing methods (K=3). The NMI, ARI and Homogeneity are also annotated on the top of the confusion matrix. **c**. The confusion matrix of the clustering by scDEC when K=5. **d**. The gap statistic shows two modes at K=3 and K=5, respectively. **e**. The visualization of the latent features learned by scDEC. The first 10 dimensions correspond to the continuous latent variable $\tilde{\boldsymbol{z}}$ and the last three features correspond to the discrete latent variable $\tilde{\boldsymbol{c}}$. **f**. The clustering performance of scDEC when applying to a large mouse atlas dataset.

Single-cell experiments are often conducted with notable differences in capturing time, equipment and even technology platforms, which may introduce batch effects in the data. To evaluate whether scDEC can automatically correct or alleviate batch effect in the training process. We collected human hematopoietic cells containing three cell types (CLP, LMPP and MPP) from two donors BM0828 and BM1077[22]. We mixed the cells from two donors together and evaluated how well the variation due to cell types and donors are resolved in the embedding (i.e., latent representation) learned by scDEC and alternative methods. Note that the latent dimension of each method was fixed to 13 and

no donor information was revealed to each method. Since the embedding by scDEC depends on the number of clusters K, we varied K from 2 to 6 and examine the gap statistic plot (Fig. 4d), which exhibited two peaks at K=3 and K=5, respectively. The embedding results for scDEC and alternative methods were shown in Fig. 4a and Supplementary Figure 7-8. It is seen that the three cell types as well as the donor effects in two of the cell types are well captured by scDEC (K=5), cisTopics and SnapATAC, but not by SCALE, whereas the donor effect in the third cell type (CLP) is too small to be discernible. It is interesting that at K=3 (the first peak of the gap statistic) the clustering results by scDEC matches the three cell types almost perfectly. Specifically, SCALE is basically unable to separate the three type of cells clearly. cisTopic and SnapATAC cannot alleviate the donor effect in LMPP or MPP cells as the same type of cells from two different donors were separated with a notable distance in the t-SNE plot (Fig. 4a). Considering the first mode where K=3, only 9 cells from donor BM0828 and 17 cells from donor BM1077 were wrongly clustered by scDEC, which illustrates a total error rate of 6.86%. Besides, scDEC also demonstrates an NMI of 0.754, ARI of 0.805 and Homogeneity of 0.757 which outperforms other comparing methods by a large margin (Fig. 4b and Supplementary Figure 8). In this sense our method can be used to adjust for donor- or batch- effects in clustering and visualization.

Next, we carefully analyzed the latent feature learned by scDEC by visualization. We noticed that features corresponding to the latent discrete variable (feature 11-13) were highly correlated to biological cell types while other features more or less revealed within-cell-type variations (Fig. 4e). For example, feature 1 is highly expressed in the donor BM1077 of LMPP and BM0828 of MPP. Feature 10 can be a donor-specific indicator of LMPP. To sum up, the interpretable features in the latent space reveal both biological cell types and within-cell-type variations.

scDEC is capable of analyzing large scATAC-seq data

We further examined that whether scDEC is applicable to extremely large scATAC-seq dataset. We collected a dataset from a mouse atlas study which contains 81,173 single cells from 13 adult mouse tissues using sci-ATAC-seq[9]. The original atlas study applies a computational pipeline to infer 40 cell types, which were regarded as "reference" cell label for the comparison of scDEC and other baselines methods. To investigate the scalability of scDEC, we randomly down-sampled the original dataset to different scale of dataset and scDEC shows a consistently good agreement with the reference cell label (Fig. 4f). For the full scale of the dataset, scDEC achieves an NMI of 0.732, ARI of 0.614 and Homogeneity of 0.693 while most previous methods failed to handle the full dataset according to a benchmark study[6]. We compared scDEC to the deep learning method SCALE and noticed that scDEC achieves a higher consistency with "reference" label but a little slower running time (Supplementary Figure 9). We also noticed that the scDEC successfully identified most of the major reference cell type for each tissue (Supplementary Figure 10).

Discussion

In this study, we proposed scDEC for accurately characterizing cell subpopulations in scATAC-seq data using a deep generative model. Unlike previous studies that take dimension reduction and clustering as two independent tasks. scDEC intrinsically integrates the low-dimensional representation learning and unsupervised clustering together by carefully designing a GAN-based symmetrical architecture. scDEC can serve as a powerful tool for scATAC-seq data analysis, including visualization, clustering and trajectory analysis. In a series experiments, scDEC achieves competitive or superior performance compared to other baseline methods. In the downstream applications, we focused on the generation power of scDEC, which can facilitate the intermediate cell state inference. The latent features learned by scDEC reveals both biological cell types and within-cell-type variations, which shed light on helping better understand the biological mechanism.

We also provide several directions for improving scDEC. First, scDEC serves as a general-purpose unsupervised learning framework, which can also be explored for applying to scRNA-seq data or the joint analysis of scRNA-seq and scATAC-seq data. Second, the way of latent indicator interpolation in the data generation can be further explored, especially in a complicated tree or graph-based trajectory of cell differentiation. One more essential problem will be how to use scDEC to generate missing data at one time point given a time-course scATAC-seq datasets.

To sum up, with scDEC, researchers could perform a scATAC-seq analysis of the cell types or tissues with interests. Then, one can simultaneously infer the cell label and uncover the biological findings underlying the learned latent features. We hope scDEC could help unveil the single-cell regulatory mechanism and contribute to understanding heterogeneous cell populations.

Methods

Data preprocessing

All the scATAC-seq datasets were uniformly preprocessed before fed to scDEC model. To reduce the level of noise, we only kept peaks that have at least one read count in more than 3% of the cells. Next, similar to Cusanovich et al[9], we applied a term frequency-inverse document frequency (TF-IDF) transformation to the raw scATAC-seq count matrix, which is widely used technology in information retrieval and text mining[33,34]. It helps increase proportionally to the number of times a peak appears in the cell, which gives a higher importance weight to the peaks with less frequency. Finally, a principle component analysis[35] (PCA) will be applied to reduce the dimension of the scATAC to 20, which is implemented with “scikit-learn” package[36]. scDEC shows robustness to the dimension of PCA (Supplementary Figure 2). The summary of all scATAC-seq datasets used in this study were provided in Supplementary Table 2.

Visualization

We use t-distributed stochastic neighbor embedding[18] (t-SNE) as the default algorithm for visualization the latent features of scATAC-seq data learned by different methods by setting the visualization dimension to 2. The t-SNE was implemented with "Scikit-learn" package[36]. The uniform manifold approximation and projection (UMAP)[19] was also implemented as an additional visualization tool for latent features.

Adversarial training in scDEC model

The scDEC model consists a pair of two GAN models. For the forward GAN mapping, G network aims at conditionally generating samples $\{\tilde{\boldsymbol{x}}_i\}_{i=1}^N$ that have a similar distribution to the observation data $\{\boldsymbol{x}_i\}_{i=1}^N$ while the discriminator D_x tries to discern observation data (positive) from generated samples (negative). The backward mapping function H and the discriminator D_z aims to transform the data from data space to the latent space. Discriminators can be considered as binary classifiers where an input data point will be asserted to be positive (1) or negative (0). We use WGAN-GP[37] as the architecture for the GAN implementation where the gradient penalty of discriminators will be considered as an additional loss terms. We define the objective loss functions of the above four neural networks ($\mathrm{G}, \mathrm{H}, \mathrm{D}_x$ and D_z) in the training process as the following

$$\begin{cases} \mathcal{L}_{GAN}(\mathrm{G}) = -\mathop{\mathbb{E}}_{\boldsymbol{z}\sim p(\boldsymbol{z}),\mathbf{c}\sim \mathrm{Cat}(K,\boldsymbol{w})}[\mathrm{D}_x(\mathrm{G}(\boldsymbol{z},\boldsymbol{c}))] \\ \mathcal{L}_{GAN}(\mathrm{D}_x) = -\mathop{\mathbb{E}}_{\boldsymbol{x}\sim p(\boldsymbol{x})}[\mathrm{D}_x(\boldsymbol{x})] + \mathop{\mathbb{E}}_{\boldsymbol{z}\sim p(\boldsymbol{z}),\mathbf{c}\sim \mathrm{Cat}(K,\boldsymbol{w})}\big[\mathrm{D}_x\big(\mathrm{G}(\boldsymbol{z},\boldsymbol{c})\big)\big] + \lambda \mathop{\mathbb{E}}_{\hat{\boldsymbol{x}}\sim \hat{p}(\hat{\boldsymbol{x}})}[(||\nabla_{\hat{\boldsymbol{x}}}\mathrm{D}_x(\hat{\boldsymbol{x}})||_2 - 1)^2] \\ \mathcal{L}_{GAN}(\mathrm{H}) = -\mathop{\mathbb{E}}_{\boldsymbol{x}\sim p(\boldsymbol{x})}[\mathrm{D}_z(\mathrm{H}(\boldsymbol{x}))] \\ \mathcal{L}_{GAN}(\mathrm{D}_z) = -\mathop{\mathbb{E}}_{\boldsymbol{z}\sim p(\boldsymbol{z})}[\mathrm{D}_z(\boldsymbol{z})] + \mathop{\mathbb{E}}_{\boldsymbol{x}\sim p(\boldsymbol{x})}\big[\mathrm{D}_z\big(\mathrm{H}(\boldsymbol{x})\big)\big] + \lambda \mathop{\mathbb{E}}_{\bar{\boldsymbol{z}}\sim \bar{p}(\bar{\boldsymbol{z}})}[(||\nabla_{\bar{\boldsymbol{z}}}\mathrm{D}_z(\bar{\boldsymbol{z}})||_2 - 1)^2] \end{cases}$$

where $p(\boldsymbol{z})$ and $\mathrm{Cat}(K,\boldsymbol{w})$ denote the probability distribution of continuous variable and discrete variable in the latent space, respectively. In practice, sampling $\boldsymbol{x}$ from $p(\boldsymbol{x})$ can be regarded as a procedure of randomly sampling from *i.i.d* observations data with replacement. $\hat{p}(\hat{\boldsymbol{x}})$ and $\bar{p}(\bar{\boldsymbol{z}})$ denote uniformly sampling from the straight line between the points sampled from true data and

generated data. Minimizing the loss of a generator (e.g., $\mathcal{L}_{GAN}(\mathrm{G})$) and the corresponding discriminator (e.g., $\mathcal{L}_{GAN}(\mathrm{D}_x)$) are somehow contradictory as the two networks (G and D_x) compete with each other during the training process. λ is a penalty coefficient which is set to 10 in all experiments.

Roundtrip loss

During the training, we also aim to minimize the roundtrip loss which is defined as $\rho((\boldsymbol{z},\boldsymbol{c}),\mathrm{H}(\mathrm{G}(\boldsymbol{z},\boldsymbol{c})))$ and $\rho(\boldsymbol{x},\mathrm{G}(\mathrm{H}(\boldsymbol{x})))$ where $\boldsymbol{z}$ and $\boldsymbol{c}$ are sampled from the distribution of the continuous latent variable $p(\boldsymbol{z})$ and the Category distribution $\mathrm{Cat}(K,\boldsymbol{w})$. The principle is to minimize the distance when a data point goes through a roundtrip transformation between two data domains. In practice, we used l_2 loss as the continuous part in roundtrip loss and used cross entropy loss as the discrete part in roundtrip loss. We further denoted the roundtrip loss as

$$\mathcal{L}_{RT}(\mathrm{G},\mathrm{H}) = \alpha||\boldsymbol{x}-\mathrm{G}(\mathrm{H}(\boldsymbol{x}))||_2^2 + \alpha||\boldsymbol{z}-\mathrm{H}_z(\mathrm{G}(\boldsymbol{z},\boldsymbol{c}))||_2^2 + \beta CE(\boldsymbol{c},\mathrm{H}_c(\mathrm{G}(\boldsymbol{z},\boldsymbol{c})))$$

where α and β are two constant coefficients which are both set to 10. $\mathrm{H}_z(\cdot)$ and $\mathrm{H}_c(\cdot)$ denote the continuous and discrete part of output from $\mathrm{H}(\cdot)$, respectively and $CE(\cdot)$ represents the cross-entropy loss function. The idea of roundtrip loss which exploits transitivity for regularizing structured data has also been used in previous works[16,38].

Full training loss

Combining the adversarial training loss and roundtrip loss together, we can get the full training loss for generator networks and discriminator networks as $\mathcal{L}(\mathrm{G},\mathrm{H}) = \mathcal{L}_{GAN}(\mathrm{G}) + \mathcal{L}_{GAN}(\mathrm{H}) + \mathcal{L}_{RT}(\mathrm{G},\mathrm{H})$ and $\mathcal{L}(\mathrm{D}_x,\mathrm{D}_z) = \mathcal{L}_{GAN}(\mathrm{D}_x) + \mathcal{L}_{GAN}(\mathrm{D}_z)$, respectively. To achieve joint training of the two GAN models, we iteratively updated the parameters in the two generative models (G and H) and the two discriminative models (D_x and D_z), respectively. Thus, the overall iterative optimization problem can be represented as

$$\mathrm{G}^*, \mathrm{D}_x^*, \mathrm{H}^*, \mathrm{D}_z^* = \begin{cases} arg\min\limits_{G,H} \mathcal{L}(\mathrm{G}, \mathrm{H}) \\ arg\min\limits_{\mathrm{D}_x, \mathrm{D}_z} \mathcal{L}(\mathrm{D}_x, \mathrm{D}_z) \end{cases}$$

Table 1. Adaptively updating w in the Category distribution in the latent space of scDEC. r is the ratio coefficient and ϵ denotes the lower bound of the cluster proportion. $\mathrm{U}(\mathrm{p}, \mathrm{q})$ represents a uniform distribution between p and q.

Algorithm Adaptively updating $\boldsymbol{w}$

Input: $\boldsymbol{w}^{(t)}$ and $\{\tilde{\boldsymbol{c}}_i | i = 1, \ldots, N\}$ $\mathrm{r} = 0.2$, $\epsilon = 0.02$;
Output: $\boldsymbol{w}^{(t+1)}$;
For $k \leftarrow 1$ to K **do**
  $\boldsymbol{w}^{esk}(k) = \sum_{i=1}^{N} \mathrm{I}(arg\max(\tilde{\boldsymbol{c}}_i) = k)/N$;
end
$\boldsymbol{w}^{(t+1)} \leftarrow \mathrm{r}\boldsymbol{w}^{(t)} + (1 - r)\boldsymbol{w}^{esk}$
For $k \leftarrow 1$ to K **do**
  if $\boldsymbol{w}^{(t+1)}(k) < \epsilon$ **then**
    $\boldsymbol{w}^{(t+1)}(k) \leftarrow \mathrm{U}\left(\epsilon, \frac{1}{K}\right)$;
  end
end
$\boldsymbol{w}^{(t+1)} \leftarrow \boldsymbol{w}^{(t+1)}/\mathrm{sum}(\boldsymbol{w}^{(t+1)})$;

An Adam optimizer[39] with a learning rate of 2×10^{-4} was used for updating the weights in the neural networks. The training process is illustrated in Supplementary Table 5 in details.

Data generation in scDEC

We generate the state of intermediate cell by interpolating the latent indicator $\boldsymbol{c}$ of two "neighboring" cell types. Assume there are two cell types which correspond to the latent indicator $\boldsymbol{c_1}$ and $\boldsymbol{c_2}$, respectively. The generated data can be represented as $\mathrm{G}(\boldsymbol{z}, \hat{\boldsymbol{c}})$ where $\hat{\boldsymbol{c}} = \alpha\boldsymbol{c_1} + (1 - \alpha)\boldsymbol{c_2}$. Note that the α is the generation coefficient from 0 to 1 and $\boldsymbol{z}$ is still sampled from a standard Gaussian distribution. The interpolation of latent features have already been used for exploring and visualizing the transition from two type of images[40].

Network architecture in scDEC

All the networks in scDEC are made of fully-connected layers. The G network contains 10 fully-connected layers and each hidden layer has 512 nodes while the H network contains 10 fully-

connected layers and each hidden layer has 256 nodes. D_x and D_z both contain 2 fully-connected layers and 256 nodes in the hidden layer. Batch normalization[41] was used in discriminator networks. All hyperparameters were also provided in Supplementary Table 1.

Updating the Category distribution

The probability $\boldsymbol{w}$ in the Category distribution $\mathrm{Cat}(K, \boldsymbol{w})$ is adaptively updated every 100 batches of data based on the inferred cluster label from $\tilde{\boldsymbol{c}}$ of full training data (Table 1).

Evaluation metrics for clustering

We compared different methods for clustering according to three metrics, normalized mutual information (NM)I[42], adjusted Rand index (ARI)[43] and Homogeneity[44]. Assuming U and V are true label assignment and predicted label assignment given n data points, which have C_U and C_V clusters in total, respectively. NMI is then calculated by

$$\mathrm{NMI} = \frac{\sum_{p=1}^{C_U}\sum_{q=1}^{C_V}|U_p \cap V_q|\log\frac{n|U_p \cap V_q|}{|U_p| \times |V_q|}}{\max\left(-\sum_{p=1}^{C_U}|U_p|\log\frac{|U_p|}{n}, -\sum_{q=1}^{C_V}|V_q|\log\frac{|V_q|}{n}\right)}$$

The Rand index[45] is a measure of agreement between two cluster assignments while ARI corrects lacking a constant value when the cluster assignments are selected randomly. We define the following four quantities 1) n_1: number of pairs of two objects in the same groups in both U and V, 2) n_2: number of pairs of two objects in different groups in both U and V, 3) n_3: number of pairs of two objects in the same group of U but different group in V, 4) n_4: number of pairs of two objects in the same group of V but different group in U. Then ARI is calculated by

$$\mathrm{ARI} = \frac{\binom{n}{2}(n_1 + n_4) - [(n_1 + n_2)(n_1 + n_3) + (n_3 + n_4)(n_2 + n_4)]}{\binom{n}{2}^2 - [(n_1 + n_2)(n_1 + n_3) + (n_3 + n_4)(n_2 + n_4)]}$$

Homogeneity is calculated by $\mathrm{Homo} = 1 - \frac{H(U|V)}{H(U)}$, where

$$\begin{cases} H(U|V) = -\sum_{p=1}^{C_U}\sum_{q=1}^{C_V}\frac{|U_p \cap V_q|}{n}\log\frac{|U_p \cap V_q|}{\sum_{q=1}^{C_V}|U_p \cap V_q|} \\ H(U) = -\sum_{p=1}^{C_U}\frac{\sum_{q=1}^{C_V}|U_p \cap V_q|}{C_U}\log\frac{\sum_{q=1}^{C_V}|U_p \cap V_q|}{C_U} \end{cases}$$

Estimating number of clusters *K*

In order to apply scDEC to scATAC-seq where the number of cell types is unknown. We provide an algorithm for estimating the number of clusters *K* using gap statistic[46]. We first compared the average within-cluster distance of the preprocessed scATAC-seq data and a reference dataset, which can be constructed with random matrix with the same size using *K*-means algorithm. The average within-cluster distance on the reference dataset was calculated for 1000 times by Monto Carlo simulation and the average result was used. The optimal choice of *K* is given for which the gap between the single cell data and the reference data is maximum. We note that this estimation of number of clusters *K* well matches the truth clusters numbers with the scATAC-seq used in this study (Supplementary Figure 11).

Identification of cluster-specific motifs

The cluster-specific motifs are identified by Mann-Whitney U test[47] with the alternative hypothesis that the chromVAR scores[23] of cells in one cluster or multiple clusters have a positive shift compared with chromVAR scores of the rest of cells. Then the motifs will be ranked according to the *p*-values and the top-ranked motifs were illustrated.

Baseline methods

We compared scDEC to multiple baseline methods in this study, including scABC[7], SCALE[15], cisTopic[8], Scasat[10], Cusanovich2018[4,9] and SnapATAC[11]. The source code for the implementation of baseline methods was downloaded from a benchmark study[6].

Data availability

InSilico dataset was collected from GEO database with accession number GSE65360. The mouse forebrain dataset was downloaded from GEO database with accession number GSE100033. Splenocyte dataset can be accessed at ArrayExpress database with accession number E-MTAB-6714. All blood dataset can be accessed at GEO database with accession number GSE96772. The mouse atlas data is available at http://atlas.gs.washington.edu/mouse-atac. All the processed data for the input of scDEC can also be downloaded from https://zenodo.org/record/3984189#.XzDpJRNKhTY.

Code availability

scDEC is an open-source software based on the TensorFlow library[48], which can be freely downloaded from https://github.com/kimmo1019/scDEC.

Acknowledgement

We thank Fengling Chen for her helpful discussion. This work was supported by NIH grants R01 HG010359 and P50 HG007735. This work was also supported by the National Key Research and Development Program of China (No. 2018YFC0910404), the National Natural Science Foundation of China (Nos. 61873141, 61721003, 61573207).

Reference

1 Klemm, S. L., Shipony, Z. & Greenleaf, W. J. Chromatin accessibility and the regulatory epigenome. *Nature Reviews Genetics* **20**, 207-220 (2019).

2 Corces, M. R. *et al.* The chromatin accessibility landscape of primary human cancers. *Science* **362** (2018).

3 Stuart, T. & Satija, R. Integrative single-cell analysis. *Nature Reviews Genetics* **20**, 257-272 (2019).

4 Cusanovich, D. A. *et al.* Multiplex single-cell profiling of chromatin accessibility by combinatorial cellular indexing. *Science* **348**, 910-914 (2015).

5 Buenrostro, J. D. *et al.* Single-cell chromatin accessibility reveals principles of regulatory variation. *Nature* **523**, 486-490 (2015).

6 Chen, H. *et al.* Assessment of computational methods for the analysis of single-cell ATAC-seq data. *Genome biology* **20**, 1-25 (2019).

7 Zamanighomi, M. *et al.* Unsupervised clustering and epigenetic classification of single cells. *Nature communications* **9**, 1-8 (2018).

8 González-Blas, C. B. *et al.* cisTopic: cis-regulatory topic modeling on single-cell ATAC-seq data. *Nature methods* **16**, 397-400 (2019).

9 Cusanovich, D. A. *et al.* A single-cell atlas of in vivo mammalian chromatin accessibility. *Cell* **174**, 1309-1324. e1318 (2018).

10 Baker, S. M., Rogerson, C., Hayes, A., Sharrocks, A. D. & Rattray, M. Classifying cells with Scasat, a single-cell ATAC-seq analysis tool. *Nucleic acids research* **47**, e10-e10 (2019).

11 Fang, R. *et al.* Fast and accurate clustering of single cell epigenomes reveals cis-regulatory elements in rare cell types. *bioRxiv*, 615179 (2019).

12 Goodfellow, I. *et al.* in *Advances in neural information processing systems.* 2672-2680.

13 Kingma, D. P. & Welling, M. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114* (2013).

14 Liu, Q., Lv, H. & Jiang, R. hicGAN infers super resolution Hi-C data with generative adversarial networks. *Bioinformatics* **35**, i99-i107 (2019).

15 Xiong, L. *et al.* SCALE method for single-cell ATAC-seq analysis via latent feature extraction. *Nature communications* **10**, 1-10 (2019).

16 Zhu, J.-Y., Park, T., Isola, P. & Efros, A. A. in *Proceedings of the IEEE international conference on computer vision.* 2223-2232.

17 Liu, Q., Xu, J., Jiang, R. & Wong, W. H. Roundtrip: A Deep Generative Neural Density Estimator. *arXiv preprint arXiv:2004.09017* (2020).

18 Maaten, L. v. d. & Hinton, G. Visualizing data using t-SNE. *Journal of machine learning research* **9**, 2579-2605 (2008).

19 McInnes, L., Healy, J. & Melville, J. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426* (2018).

20 Preissl, S. *et al.* Single-nucleus analysis of accessible chromatin in developing mouse forebrain reveals cell-type-specific transcriptional regulation. *Nature neuroscience* **21**, 432-439 (2018).

21 Chen, X., Miragaia, R. J., Natarajan, K. N. & Teichmann, S. A. A rapid and robust method for single cell chromatin accessibility profiling. *Nature Communications* **9**, 1-9 (2018).

22 Buenrostro, J. D. *et al.* Integrated single-cell analysis maps the continuous regulatory landscape of human hematopoietic differentiation. *Cell* **173**, 1535-1548. e1516 (2018).

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

24 Mathelier, A. *et al.* JASPAR 2016: a major expansion and update of the open-access database of transcription factor binding profiles. *Nucleic Acids Res* **44**, D110-115, doi:10.1093/nar/gkv1176 (2016).

25 Shaltouki, A., Peng, J., Liu, Q., Rao, M. S. & Zeng, X. Efficient generation of astrocytes from human pluripotent stem cells in defined conditions. *Stem cells* **31**, 941-952 (2013).

26 Bayam, E. *et al.* Genome-wide target analysis of NEUROD2 provides new insights into regulation of cortical projection neuron migration and differentiation. *BMC genomics* **16**, 681 (2015).

27 Owa, T. *et al.* Meis1 coordinates cerebellar granule cell development by regulating Pax6 transcription, BMP signaling and Atoh1 degradation. *Journal of Neuroscience* **38**, 1277-1294 (2018).

28 Hallonet, M., Hollemann, T., Pieler, T. & Gruss, P. Vax1, a novel homeobox-containing gene, directs development of the basal forebrain and visual system. *Genes & development* **13**, 3106-3114 (1999).

29 Cesari, F. *et al.* Mice deficient for the ets transcription factor elk-1 show normal immune responses and mildly impaired neuronal gene activation. *Molecular and cellular biology* **24**, 294-305 (2004).

30 Stolt, C. C. *et al.* The Sox9 transcription factor determines glial fate choice in the developing spinal cord. *Genes & development* **17**, 1677-1689 (2003).

31 Street, K. *et al.* Slingshot: cell lineage and pseudotime inference for single-cell transcriptomics. *BMC genomics* **19**, 477 (2018).

32 Iwasaki, H. & Akashi, K. Myeloid lineage commitment from the hematopoietic stem cell. *Immunity* **26**, 726-740 (2007).

33 Teller, V. Speech and language processing: an introduction to natural language processing, computational linguistics, and speech recognition. *Computational Linguistics* **26**, 638-641 (2000).

34 Chowdhury, G. G. *Introduction to modern information retrieval*. (Facet publishing, 2010).

35 Halko, N., Martinsson, P.-G. & Tropp, J. A. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM review* **53**, 217-288 (2011).

36 Pedregosa, F. *et al.* Scikit-learn: Machine learning in Python. *the Journal of machine Learning research* **12**, 2825-2830 (2011).

37 Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V. & Courville, A. C. in *Advances in neural information processing systems.* 5767-5777.

38 Yi, Z., Zhang, H., Tan, P. & Gong, M. in *Proceedings of the IEEE international conference on computer vision.* 2849-2857.

39 Kingma, D. P. & Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

40 Mukherjee, S., Asnani, H., Lin, E. & Kannan, S. in *Proceedings of the AAAI Conference on Artificial Intelligence.* 4610-4617.

41 Ioffe, S. & Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167* (2015).

42 Strehl, A. & Ghosh, J. Cluster ensembles---a knowledge reuse framework for combining multiple partitions. *Journal of machine learning research* **3**, 583-617 (2002).
43 Hubert, L. & Arabie, P. Comparing partitions. *Journal of classification* **2**, 193-218 (1985).
44 Rosenberg, A. & Hirschberg, J. in *Proceedings of the 2007 joint conference on empirical methods in natural language processing and computational natural language learning (EMNLP-CoNLL)*. 410-420.
45 Rand, W. M. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical association* **66**, 846-850 (1971).
46 Tibshirani, R., Walther, G. & Hastie, T. Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **63**, 411-423 (2001).
47 Mann, H. B. & Whitney, D. R. On a test of whether one of two random variables is stochastically larger than the other. *The annals of mathematical statistics*, 50-60 (1947).
48 Abadi, M. *et al.* in *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*. 265-283.