Gene Expression

Learning latent embedding of multi-modal single cell data and cross-modality relationship simultaneously

Ziqi Zhang [1], Chengkai Yang [1] and Xiuwei Zhang [1,*]

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

* To whom correspondence should be addressed.

Abstract

Motivation: Single cell multi-omics studies allow researchers to understand cell differentiation and development mechanisms in a more comprehensive manner. Single cell ATAC-sequencing (scATAC-seq) measures the chromatin accessibility of cells, and computational methods have been proposed to integrate scATAC-seq with single cell RNA-sequencing (scRNA-seq) data of cells from the same cell types. This computational task is particularly challenging when the two modalities are not profiled from the same cells. Some existing methods first transform the scATAC-seq data into scRNA-seq data and integrate two scRNA-seq datasets, but how to perform the transformation is still a difficult problem. In addition, most of existing methods to integrate scRNA-seq and scATAC-seq data focus on preserving distinct cell clusters before and after the integration, and it is not clear whether these methods can preserve the continuous trajectories for cells from continuous development or differentiation processes.

Results: We propose `scDART` (**s**ingle **c**ell **D**eep learning model for **A**TAC-Seq and **R**NA-Seq **T**rajectory integration), a scalable deep learning framework that embeds the two data modalities of single cells, scRNA-seq and scATAC-seq data, into a shared low-dimensional latent space while preserving cell trajectory structures. Furthermore, `scDART` learns a nonlinear function represented by a neural network encoding the cross-modality relationship simultaneously when learning the latent space representations of the integrated dataset. We test `scDART` on both real and simulated datasets, and compare it with the state-of-the-art methods. We show that `scDART` is able to integrate scRNA-seq and scATAC-seq data well while preserving the continuous cell trajectories. `scDART` also predicts scRNA-seq data accurately from the scATAC-seq data using the neural network module that represents cross-modality relationship.

Availability: https://github.com/PeterZZQ/scDART

Contact: xiuwei.zhang@gatech.edu

1 Introduction

The availability of single cell multimodal omics data provides a comprehensive view of each single cell. Single cell RNA-sequencing (scRNA-seq) and single cell ATAC-sequencing (scATAC-seq) respectively measures the gene-expression and chromatin accessibility profiles of cells, each being considered as an important aspect of a cell. Recently, techniques which can measure both gene-expression and chromatin accessibility in the same cells have been proposed (Chen *et al.*, 2019; Cao *et al.*, 2018; Ma *et al.*, 2020), but these technologies are still not widely used, and they can suffer from low sensitivity of one of the data modalities. To make use of the enormous amount of existing data, computational methods have been proposed to integrate scRNA-seq and scATAC-seq data obtained separately for the same cell types in different batches (Stuart *et al.*, 2019; Welch *et al.*, 2019; Duren *et al.*, 2018; Korsunsky *et al.*, 2019), with the aim of building larger datasets and potentially learning the relationship between chromatin accessibility and gene-expression. An integration method is expected to be able to "merge" the two datasets, that is, to remove the batch effect, as well as to preserve the biological identity of each cell.

Most of the existing methods benchmark whether the cells preserve the biological identity of cells in terms of discrete cell clusters. When the cells form continuous trajectories instead of discrete clusters, the identity of a cell can be thought of as where the cell falls on the trajectory. For example, if the trajectory has a structure of a rooted tree, the identity of a cell is reflected by which branch it belongs to and its distance from the root, which corresponds to the pseudotime of the cell. Maintaining the continuous trajectory and the relative positions of cells on the trajectory during data integration can be a more challenging task compared to maintaining the separation of distinct clusters.

Existing methods which can integrate scRNA-seq and scATAC-seq data from different cells use various optimization objectives. Some methods aim to learn the integrated dataset and its latent embedding such

picture(0,0)(-35,0)(1,0)30 (0,35)(0,-1)30 picture picture(0,0)(35,0)(-1,0)30 (0,35)(0,-1)30 picture

that the latent embedding can reconstruct the original dataset (Welch *et al.*, 2019; Duren *et al.*, 2018; Argelaguet *et al.*, 2020). Some use manifold alignment (Cui *et al.*, 2014) and aim to learn the latent embedding by enforcing the latent embedding to preserve the pairwise distances of cells in the original high-dimensional space (Singh *et al.*, 2020; Cao *et al.*, 2020). `Seurat` maps a query dataset to a reference dataset, and obtain a new data matrix for the query dataset based on the reference dataset.

A majority of the existing methods (Stuart *et al.*, 2019; Welch *et al.*, 2019; Duren *et al.*, 2018) require a pre-defined *gene activity matrix* (GAM, also called a region-gene association matrix), representing which genomic regions regulate the expression of which genes, to transform the scATAC-seq data into scRNA-seq data. A common way to obtain a pre-defined GAM is to consider the relative positions between the potential open regions and the gene bodies on the genome. However, the region-gene relationships can be cell-type specific, and is best learned from a well-integrated dataset, especially along a continuous trajectory which provides higher resolutions of dynamic changes in cells compared to discrete cell populations. Therefore, the integration task and learning the GAM become chicken-and-egg problems. Another challenge is that, even if an accurate GAM is available, simply multiplying it to the scATAC-seq data to obtain scRNA-seq data makes an assumption of linear relationships between chromatin accessibility and gene-expression in cells, which is often not true in biological systems.

Hereby we propose `scDART` (**s**ingle **c**ell **D**eep learning model for **A**TAC-Seq and **R**NA-Seq **T**rajectory integration), a scalable deep learning framework that embed data modalities into a shared low-dimensional latent space while preserving continuous cell trajectories. The design of `scDART` allows one to learn not only the latent space representations of cells, but also the relationships between chromatin accessibility and gene-expression, thus provides a solution to the chicken-and-egg problem. Compared to methods which preserve the pairwise distance between cells (Cao *et al.*, 2020; Singh *et al.*, 2020), `scDART` uses a different distance measure, the *diffusion distance* which has been successfully used by trajectory inference methods (Haghverdi *et al.*, 2016; Moon *et al.*, 2019), and this distance can directly be translated into the pseudotime of cells which facilitates downstream trajectory inference using the integrated datasets.

The schematics of `scDART` are shown in Fig. 1a. `scDART` takes as input the scRNA-seq and scATAC-seq data matrices, and a pre-defined gene activity matrix (GAM), and learns a refined nonlinear gene activity function and the latent embedding of both datasets. In `scDART`, we construct the pre-defined binary GAM by considering the regions that lie within 3000 base-pairs upstream of the gene body on the genome to be the regulatory regions of that gene. The neural net framework of `scDART` (Fig.1c) consists of two modules: gene activity function module and projection module. The gene activity function module can encode nonlinear regulatory relationships between regions and genes (Fig.1b). It takes in the scATAC-seq matrix, transforms its chromatin accessibility data into the gene expression level data, and generates a "pseudo-scRNA-seq" count matrix. The projection module takes in both both scRNA-seq count matrix and the pseudo-scRNA-seq count matrix generated from scATAC-seq count matrix, and projects them into the same latent space.

We test `scDART` on three real datasets: two "unpaired" datasets where the scRNA-seq and scATAC-seq data were not jointly profiled, and a "paired" dataset where both chromatin accessibility and gene-expression daa are measured simultaneously in the same cells. We also test `scDART` on simulated datasets and measure the quality of integration and the learned gene activity function. The results show that `scDART` learns a joint latent space for both data modalities that well preserve the cell developmental trajectories, and that `scDART` can accurately predict scRNA-seq data from scATAC-seq data with the gene activity function module it learns.

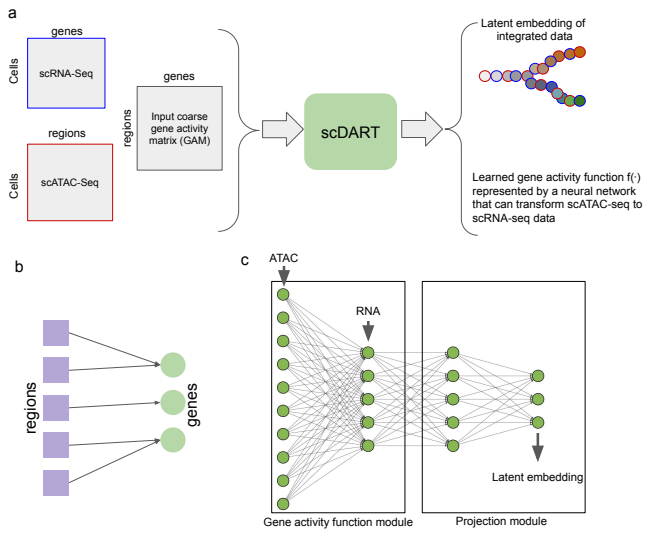

Fig. 1. (a) The schematic of `scDART`. Users input scRNA-seq data matrix, scATAC-seq data matrix and a coarse gene activity matrix (constructed using the relative positions of chromatin regions and genes on the genome). `scDART` learns a nonlinear gene activity function and the latent embedding of the two datasets. (b) The regulatory relationship between scATAC-seq and scRNA-seq data can be represented by a bipartite graph. The regulatory relationship between regions and genes are usually nonlinear, which can be modeled using a deep neural network (DNN). (c) The DNN structure of `scDART`. `scDART` includes two functional modules: 1) the gene activity function module is a fully-connected neural network. This module encodes the nonlinear regulatory relationship between regions and genes. 2) the projection module takes in the scRNA-seq count matrix and the pseudo-scRNA-seq matrix, and generates the latent embedding of both modalities.

2 Methods

When training `scDART`, we consider the following objectives: (1) The latent space representation of each data modality (scRNA-seq or scATAC-seq) should preserve the trajectory and cell-cell distance in the original dataset; (2) The batch effect between the two modalities should be removed such that the cells from the two modalities should merge in a shared latent space; (3) A pre-defined GAM should be used as a constraint when learning the gene activity function using `scDART`. These objectives lead to different terms in our overall loss function. Each objective is discussed as follows.

2.1 Preserving cell trajectories

For each data modality, we aim to learn the latent space representation that preserves the pairwise distance between cells in the original dimensional space. We use diffusion-based distance as the distance measure between cells in the original space. Diffusion-based distances are shown to better measure the manifold distance between data points within the high-dimensional dataset compared to other distance metrics (Nadler *et al.*, 2005). Applications of diffusion distance in single-cell RNA-seq data in Phate (Moon *et al.*, 2019) and diffusion pseudotime (DPT) (Haghverdi *et al.*, 2016) show that diffusion distance can be a good metrics of measuring developmental differences between cells within the cell developmental trajectory.

Given a data matrix (can be either the scRNA-seq or scATAC-seq data matrix) in the original dimensional space $\mathbf{X}$, the calculation of the diffusion matrix $\mathbf{D}_X$ in (Moon *et al.*, 2019) (where $\mathbf{X}$ is the scRNA-seq matrix) consists of multiple steps. First, a kernelized pairwise distance matrix between every two cells (i, j) is calculated based on their low-dimensional features $\mathbf{x}_i$ and $\mathbf{x}_j$ obtained by performing PCA on $\mathbf{X}$:

$$\mathbf{K}(i,j) = \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|_2^2}{2\sigma^2}\right) \quad (1)$$

where σ of each cell is set to be proportional to the distance between each cell and its kth-nearest neighbor (k is set to 5). Second, the cell transition

picture(0,0)(-35,0)(1,0)30 (0,-35)(0,1)30 picture picture(0,0)(35,0)(-1,0)30 (0,-35)(0,1)30 picture

picture(0,0)(-35,0)(1,0)30 (0,35)(0,-1)30 picture picture(0,0)(35,0)(-1,0)30 (0,35)(0,-1)30 picture

matrix $\mathbf{P}$ is constructed by normalizing the similarity matrix $\mathbf{K}$ such that values in each row sum up to 1: $\mathbf{P} = \mathbf{K}/(\sum_j \mathbf{K}_{ij})$.

Then, the diffusion process is performed by powering the transition matrix $\mathbf{P}$ to t times to obtain $\mathbf{P}_t = \mathbf{P}^t$. Finally, considering each row of $\mathbf{P}_t$ to be the feature vector of the corresponding cell, the diffusion distance matrix $\mathbf{D}_X$ can be calculated as the pairwise euclidean distance between cells in this feature space.

Our process of calculating the diffusion distance distinguishes from what is used in (Moon *et al.*, 2019) in three ways: (1) Diffusion step t is an important parameter. Moon *et al* used the Von Neumann Entropy to pick t for each dataset. Inspired by Haghverdi *et al* who summed up P_t of all t values from one to infinity to eliminate ts, we calculate the P_t with multiple t values ($t = 10, 30, 50$), normalize them to the same scale, and then take the average. (2) When calculating the diffusion matrix $\mathbf{P}_t$ for the scATAC-seq data, we use Latent Semantic Indexing (LSI) (Hofmann, 1999) instead of PCA for dimensionality reduction. (3) The above process to calculate $\mathbf{P}_t$ requires $O(N^3)$ time complexity (N represents the number of cells). We design a fast heuristic for approximate diffusion matrix calculation which works as follows.

We first select m anchor cells randomly within the dataset and calculate the diffusion matrix between these anchor cells $\mathbf{P}_t^{\text{anchor}}$ using the process above. Then we construct a k-nearest neighbor graph (k = 30) between the anchor cells and the remaining $N - m$ cells, which is a bipartite graph. The weights of this graph are calculated using the same Gaussian kernel as shown in Eq. 1. Denoting the adjacency matrix with the calculated weights by $\mathbf{W}$, the cell transition matrix $\mathbf{W}'$ is calculated as $\mathbf{W}' = \mathbf{W}/(\sum_j \mathbf{W}_{ij})$, and the diffusion matrix of the rest of the cells are calculated as $\mathbf{P}_t^{\text{others}} = \mathbf{W}' \cdot \mathbf{P}_t^{\text{anchor}}$. The entire diffusion matrix $\mathbf{P}_t$ is then:

$$\mathbf{P}_t = \begin{bmatrix} \mathbf{P}_t^{\text{anchor}} & \mathbf{P}_t^{\text{others}} \end{bmatrix}^{\mathrm{T}} \tag{2}$$

The time complexity of the fast implementation is $O(mN^2)$. In practice, it is sufficient to set $m \approx 500$, which can be much smaller than N in large datasets thus this implementation can be much faster than the standard procedure.

With the latent embedding learned by `scDART`, we calculate the pairwise Euclidean distance between cells denoted by $\mathbf{D}_Z$. In order to enforce the latent embedding to preserve the original trajectory structure, we constrain $\mathbf{D}_Z$ to be similar to $\mathbf{D}_X$ calculated from the original datasets. Inspired by tSNE (Van der Maaten and Hinton, 2008), we first normalize $\mathbf{D}_X$ and $\mathbf{D}_Z$ into $\mathbf{Q}_X$ and $\mathbf{Q}_Z$ (Eq. 3):

$$\mathbf{Q}_X = \frac{\mathbf{D}_X}{\sum_{ij} \mathbf{D}_X(i,j)}; \quad \mathbf{Q}_Z = \frac{\mathbf{D}_Z}{\sum_{ij} \mathbf{D}_Z(i,j)} \tag{3}$$

then use Kullback–Leibler divergence (KL-divergence) loss (Eq. 4) to measure the difference between the two distance matrices:

$$L_{\text{dist}}(\mathbf{Z}, \mathbf{X}) = KL(\mathbf{Q}_Z \| \mathbf{Q}_X) = \sum_{ij} \mathbf{Q}_Z(i,j) \log \frac{\mathbf{Q}_Z(i,j)}{\mathbf{Q}_X(i,j)} \tag{4}$$

Compared to other loss functions, for example, mean square error loss, inner product loss, the asymmetric formulation of KL-divergence loss (Eq. 4) has a larger penalty when $\mathbf{Q}_X(i,j)$ is small and $\mathbf{Q}_Z(i,j)$ is large, and this will force the latent embedding to better preserve the local manifold structure.

2.2 Integrating modalities and batch removal

Even though the scATAC-seq and scRNA-seq count matrix measures different features of different cell batches, they follows the same underlying developmental process. Following existing work (Cao *et al.*, 2020; Welch *et al.*, 2017), we assume the underlying trajectory structures of these two datasets are similar. In order to project scRNA-seq data and scATAC-seq data into the same latent space where the trajectory topology of both dataset merge, we incorporate the maximum mean discrepancy (MMD) loss (Eq. 5) (Dziugaite *et al.*, 2015). MMD provides a statistical measures of the difference between the distributions of the latent embedding of scRNA-seq and scATAC-seq data. Denoting the latent embedding of scRNA-seq and scATAC-seq data respectively as $\mathbf{Z}_{\text{RNA}}$ and $\mathbf{Z}_{\text{ATAC}}$, the MMD loss function takes the following form:

$$\begin{aligned} L_{\text{mmd}}(\mathbf{Z}_{\text{RNA}}, \mathbf{Z}_{\text{ATAC}}, \gamma) =& \mathbb{E}[K(\mathbf{Z}_{\text{RNA}}, \mathbf{Z}_{\text{RNA}})] + \mathbb{E}[K(\mathbf{Z}_{\text{ATAC}}, \mathbf{Z}_{\text{ATAC}})] \\ & -2\mathbb{E}[K(\mathbf{Z}_{\text{RNA}}, \mathbf{Z}_{\text{ATAC}})] \end{aligned} \tag{5}$$

Where K is the Gaussian kernel function as shown in Eq. 1, $\mathbb{E}$ means expectation; γ is a key parameter of the Gaussian kernel function. Following (Amodio *et al.*, 2019), we sum up the MMD loss with different γ values to improve the robustness of the loss term:

$$L_{\text{mmd}}(\mathbf{Z}_{\text{RNA}}, \mathbf{Z}_{\text{ATAC}}) = \sum_{\gamma \in \Gamma} L_{\text{mmd}}(\mathbf{Z}_{\text{RNA}}, \mathbf{Z}_{\text{ATAC}}, \gamma) \tag{6}$$

where $\Gamma = \{10^u\}$ and u is an integer ranging from -6 to 6.

2.3 Using prior gene activity matrix

The expression of a gene requires that its gene regulatory elements (GRE, eg. promoter) are accessible to molecules like RNA polymerase and transcription factors (TFs) such that the molecules can bind to the GRE regions to trigger the transcription. The dependency between genes and potential GRE regions can be modeled as a sparse bipartite graph (Fig. 1b). A gene activity matrix (GAM) usually refers to the adjacency matrix of that bipartite graph. A coarse version of the matrix can be obtained using solely the locations of regions and genes on the genome, where the regions within a certain distance (e.g., 3000bp) at the upstream of a gene body are considered to regulate the expression of the gene. We denote a GAM obtained in this manner by $\mathbf{A}$. Some existing methods which integrate scRNA-seq and scATAC-seq data multiply $\mathbf{A}$ to the scATAC-seq data matrix and obtain another scRNA-seq data matrix (Welch *et al.*, 2019; Stuart *et al.*, 2019; Duren *et al.*, 2018), which is a linear transformation process. However, GAMs obtained with only genomic positions are inaccurate. How the accessibility of a genomic region affects the expression level of a gene is a complex mechanism, which can be both nonlinear and cell-type specific.

We utilize a three-layer fully connected neural network, termed the "gene activity module", to learn a gene activity function that can transform scATAC-seq data into scRNA-seq data. We call the scRNA-seq derived from the scATAC-seq data *pseudo-scRNA-seq* data. The gene activity module thus represents the data-specific relationship between scATAC-seq and scRNA-seq data, and it can encode nonlinearity in this relationship.

The network has an input dimension equal to the number of regions in scATAC-seq data, and an output dimension equal to the number of genes in scRNA-seq data. We use rectified linear unit (ReLU) as the activation function between the layers and remove the bias term of each layer. Taking the region accessibility of a cell ($\mathbf{x}_{\text{ATAC}}$) as the input, the corresponding gene-expression data of that cell ($\mathbf{x}'_{\text{RNA}}$) can be predicted as

$$\mathbf{x}_{\text{ATAC}} = \mathbf{W}_3 \cdot (\text{Relu}(\mathbf{W}_2 \cdot (\text{Relu}(\mathbf{W}_1 \cdot \mathbf{x}_{\text{RNA}})))) \tag{7}$$

where $\mathbf{W}_i$ represents the weights of the ith layer. When learning the weights, we use the coarse GAM $\mathbf{A}$ (which is a binary matrix) as prior information to constrain the training procedure. We assume that $\mathbf{A}$ includes all the potential regulations between regions and genes, that is, the 0s in $\mathbf{A}$ are correct information but 1s in $\mathbf{A}$ can be false positives. Given this assumption, we construct a regularization term L_{GAM} to penalize the non-zero regulation strength from a region to a gene in the learned gene activity module that should be zero according to $\mathbf{A}$:

picture(0,0)(-35,0)(1,0)30 (0,-35)(0,1)30 picture picture(0,0)(35,0)(-1,0)30 (0,-35)(0,1)30 picture

picture(0,0)(-35,0)(1,0)30 (0,35)(0,-1)30 picture picture(0,0)(35,0)(-1,0)30 (0,35)(0,-1)30 picture

$$L_{\text{GAM}} = \|(\prod_{i=1}^{\ell} \mathbf{W}_i) \odot \widehat{\mathbf{A}}\|_1 \tag{8}$$

$\odot$ denotes the element-wise multiplication between two matrices and $\widehat{\mathbf{A}}$ is the element-wise reversion of $\mathbf{A}$. We use ℓ_1 norm to enforce the sparsity of the learned regulation strength.

After finishing the training procedure and having learned $\mathbf{W}_1$, $\mathbf{W}_2$ and $\mathbf{W}_3$, we then obtain a trained gene activity module which represents the complex nonlinear gene activity function between the scATAC-seq and the scRNA-seq data.

2.4 Overall loss function formulation and training procedure

We train the gene activity module and the projection module (Fig. 1c) jointly using both scRNA-seq and scATAC-seq data matrices as input. The training processes are different for different modalities: when training with the scATAC-seq count matrix, we feed the data into the gene activity function module, and take the transformed pseudo-scRNA-seq data into the projection module. When training with scRNA-seq count matrix, we directly feed the data into the projection module. The projection module generates the latent embedding for both datasets. We construct loss function summarizing the loss terms discussed above in Eq. 9 and use stochastic gradient descent (Bottou, 2012) to minimize the loss function.

$$\begin{aligned} L_1 = & L_{\text{dist}}(\mathbf{Z}_{\text{RNA}}, \mathbf{X}_{\text{RNA}}) + L_{\text{dist}}(\mathbf{Z}_{\text{ATAC}}, \mathbf{X}_{\text{ATAC}}) \\ & + \lambda_{\text{mmd}} \cdot L_{\text{mmd}}(\mathbf{Z}_{\text{ATAC}}, \mathbf{Z}_{\text{RNA}}) + \lambda_g \cdot L_{\text{GAM}} \end{aligned} \tag{9}$$

The loss function includes four terms. The first two terms, $L_{\text{dist}}(\mathbf{Z}_{\text{RNA}}, \mathbf{X}_{\text{RNA}})$ and $L_{\text{dist}}(\mathbf{Z}_{\text{ATAC}}, \mathbf{X}_{\text{ATAC}})$, measure the trajectory structure preservation of the latent embedding of scRNA-seq and scATAC-seq datasets (Eq. 4, Section 2.1). The third term, $L_{\text{mmd}}(\mathbf{Z}_{\text{ATAC}}, \mathbf{Z}_{\text{RNA}})$ measures how well the two modalities are integrated together (Section 2.2, Eq. 6). The last term enforces that the gene activity module follows our prior knowledge from the input GAM (Section 2.3, Eq. 8).

2.5 Post-processing and trajectory inference

After obtaining the latent embedding $\mathbf{Z}_{\text{ATAC}}$ and $\mathbf{Z}_{\text{RNA}}$, we apply a post-processing step to further refine the latent embedding to form cleaner trajectory structure. We construct k mutual nearest neighbor graph (Haghverdi *et al.*, 2018) on the cells from $\mathbf{Z}_{\text{ATAC}}$ and $\mathbf{Z}_{\text{RNA}}$: for each cell in $\mathbf{Z}_{\text{RNA}}$, we find its k nearest cells in $\mathbf{Z}_{\text{ATAC}}$, and vice versa. After constructing the graph, we calculate weights on the graph. For cell i in $\mathbf{Z}_{\text{RNA}}$ and cell j in $\mathbf{Z}_{\text{ATAC}}$, the weight is:.

$$\mathbf{B}_{ij} = \exp\left(-\|\mathbf{z}_{\text{RNA}}(i) - \mathbf{z}_{\text{ATAC}}(j)\|_2^2\right) \tag{10}$$

then we update the latent embedding of each cell by the embedding of its neighbors. For example, for each cell i in the scRNA-seq data, the new latent embedding $\mathbf{z}'_{\text{RNA}}(i)$ is calculated as:

$$\mathbf{z}_{\text{RNA}}(i)' = \frac{\sum_{j \in \text{neigh}(i)} \mathbf{B}_{ij} \mathbf{z}_{\text{ATAC}}(j)}{\sum_{j \in \text{neigh}(i)} \mathbf{B}_{ij}} \tag{11}$$

The post-processing step makes different trajectory lineages more distinguishable in the latent space, thus helps trajectory inference methods to detect more accurate trajectories in complex trajectory structures.

We then perform trajectory inference on the joint latent embedding of the scRNA-seq and scATAC-seq data to learn cell trajectories based on both types of data. We apply DPT (Haghverdi *et al.*, 2016) on the latent embedding to infer the pseudotime for cells from both modalities jointly, and we use Leiden clustering (Traag *et al.*, 2019) and minimum spanning tree to infer the trajectory backbone similar to PAGA (Wolf *et al.*, 2019).

2.6 Data simulation

The simulated scRNA-seq and scATAC-seq data are generated with an extended version of SymSim (Zhang *et al.*, 2019) which simulates scRNA-seq data. In SymSim, a kinetic model is used to model the mRNA counts in cells, where a gene is considered to be either in an *on* state or in an *off* state (Munsky *et al.*, 2012). When a gene is in the *on* state, its transcripts are synthesized with rate s, and synthesized mRNAs degrade with a rate d. A parameter k_{on} represents the rate at which a gene enters the *on* state, and k_{off} represents the rate of the gene entering the *off* state. To generate multiple discrete or continuous cell types, SymSim defines an "identity vector" for each cell, and the identity vectors can evolve along a user-provided tree which represents the trajectory backbone.

In this work, we extended SymSim so that it also generates scATAC-seq data. Fig. 2 shows the process of generating N cells which have both scRNA-seq and scATAC-seq data. Denote the number of genes by G and the number of regions by R. A binary $R \times G$ GAM is provided to represent which regions affect which genes. As the scRNA-seq data depends on the scATAC-seq data, we first generate the scATAC-seq data. Similar to how SymSim generates scRNA-seq data along a continuous trajectory, we start with a "cell chromatin accessibility identity vector" of length v for the root cell and let it evolve along the given trajectory structure through a Brownian motion process to generate the "cell chromatin accessibility identity vectors" of cells along the tree. Each region has a "region identity vector" which is of the same length v. Multiplying the "cell chromatin accessibility identity matrix" and the "region identity vector matrix" we obtain an $N \times R$ matrix, where entries with larger values correspond to higher chromatin accessibility. We call this matrix "non-realistic scATAC-seq data" as its distribution is not the same as the distribution in real data. We then map the data in this matrix to a distribution obtained from a real scATAC-seq dataset (Chen *et al.*, 2018) to get the realistic scATAC-seq data.

The scRNA-seq data is affected by both the input trajectory tree and the scATAC-seq data. We first generate the kinetic parameters for generating scRNA-seq data in the same way as in SymSim, and obtain the "realistic kinetic parameter matrix" shown in Fig. 2. We now use the scATAC-seq data and the GAM to adjust k_{on}, as we consider that the accessibility of the associated regions of a gene affects the rate that the gene is switched on, which is what k_{on} corresponds to. Now among the three kinetic parameters of scRNA-seq data, k_{on} is affected by scATAC-seq data, and k_{off} and s are affected by the input trajectory, thus we have combined both the effects of chromatin accessibility and cell differentiation process into the final scRNA-seq data. We then add technical noise to the scRNA-seq data, and divide all cells into two batches while adding batch effects. To mimic the unpaired data, for one batch we keep only the scRNA-seq data and for the other batch we keep only the scATAC-seq data.

2.7 Evaluation metrics

When ground truth information is available, we use the following metrics to evaluate the latent embedding learned by `scDART` and the trajectories inferred based on it: neighborhood overlap score, lineage separation score and the Kendall rank correlation coefficient on inferred pseudotime. With simulated data, we also evaluate the gene activity module learned by `scDART`. Given the scATAC-seq data, we use the gene activity module of `scDART` to predict the scRNA-seq data and calculate the mean square error (MSE) with the ground truth paired scRNA-seq data.

Neighborhood overlap score (Cao *et al.*, 2020; Korsunsky *et al.*, 2019) can be used to measure how well datasets are integrated when there exist cell-cell correspondence across data modalities. Given a neighborhood size k, it constructs a k-nearest neighbor graph on the latent embedding of cells from both scRNA-seq and scATAC-seq data, and calculates the proportion of cells that have its corresponding cell in the other modality included within its neighborhood.

picture(0,0)(-35,0)(1,0)30 (0,-35)(0,1)30 picture picture(0,0)(35,0)(-1,0)30 (0,-35)(0,1)30 picture

picture(0,0)(-35,0)(1,0)30 (0,35)(0,-1)30 picture picture(0,0)(35,0)(-1,0)30 (0,35)(0,-1)30 picture

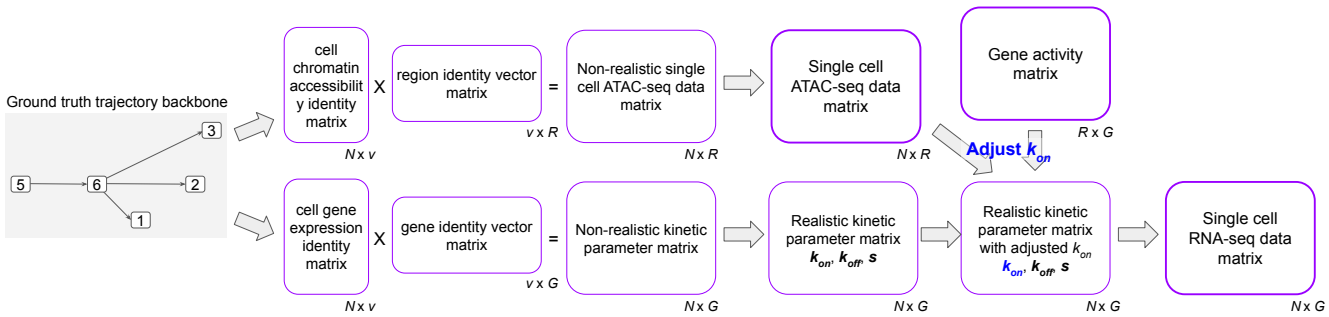

Fig. 2. Illustration of the data simulation process. Given the ground truth trajectory and the gene activity matrix, the simulation procedure generates both scRNA-seq and scATAC-seq data.

We design the *lineage separation score* to measure how well different trajectory lineages are separated in the inferred trajectory based on the latent embedding learned by `scDART`. This score is adapted from the *alignment score* used in (Butler *et al.*, 2018) in the case of discrete clusters. For any two lineages, we first subsample the lineage with larger number of cells such that it has the same number of cells as the smaller lineage. Then we construct a k-nearest neighbor graph on the latent embedding of both lineages (k is set to be 1% of the total number of cells N). Then we measure the composition of the neighborhood of each cell, and use r to denote the proportions of cells from each data modality. We average the composition value over all cells to obtain $\bar{r}$. Then the lineage separation score is

$$s_{\text{sep}} = 1 - (\bar{r} - \frac{k}{N})/(k - \frac{k}{N})$$

3 Experiments

3.1 Results on real data

We test `scDART` on two unpaired datasets: a mouse endothelial cell development dataset (Zhu *et al.*, 2020) and a human hematopoiesis dataset that we collected from two papers (Buenrostro *et al.*, 2018; Pellin *et al.*, 2019), and one paired dataset on a mouse neonatal brain cortex (Chen *et al.*, 2019). In the unpaired dataset, scATAC-seq and scRNA-seq were performed separately on different cell batches. There is no one-to-one pairing between cells from the scATAC-seq data and cells from the scRNA-seq data. In the paired dataset, scATAC-seq and scRNA-seq were performed in the same cells, and cells are paired between the two data modalities. We compare the performance of `scDART` with that of existing methods including `UnionCom` (Cao *et al.*, 2020), `Seurat` (Stuart *et al.*, 2019) and MMD-MA (Singh *et al.*, 2020) on these datasets. We do not show the results of MMD-MA because its overall performance is the poorest of all in our results. We apply PCA to the learned latent embedding $\mathbf{Z}_{\text{RNA}}$ and $\mathbf{Z}_{\text{ATAC}}$ to visualize them in 2D. Here we choose PCA instead of UMAP or TSNE as both of the latter methods focus only on the local structure between samples and may destruct the global trajectory structure that we aim to preserve during the training.

3.1.1 Results on the mouse endothelial cell development data

The mouse endothelial cell development dataset profiles the chromatin accessibility and gene-expression data of single cells that undergo a differentiation path from endothelial cells (Endo) to Hematopoietic stem and progenitor cells (HSPCs) that accumulate in intra-arterial clusters (IAC). The cell trajectory in this dataset overall is a linear-like path, with a small bifurcating-converging trajectory topology structure (Fig. 3a), where endothelial cells separate into two close streams of Endothelial sub-celltypes "Arterial Endo 1" and "Arterial Endo 2" and then conflux into "Pre-BN1" cells. "Pre-BN1" cell then undergoes a maturation path to "ICA" through "Pre-HE" and "HE" stage.

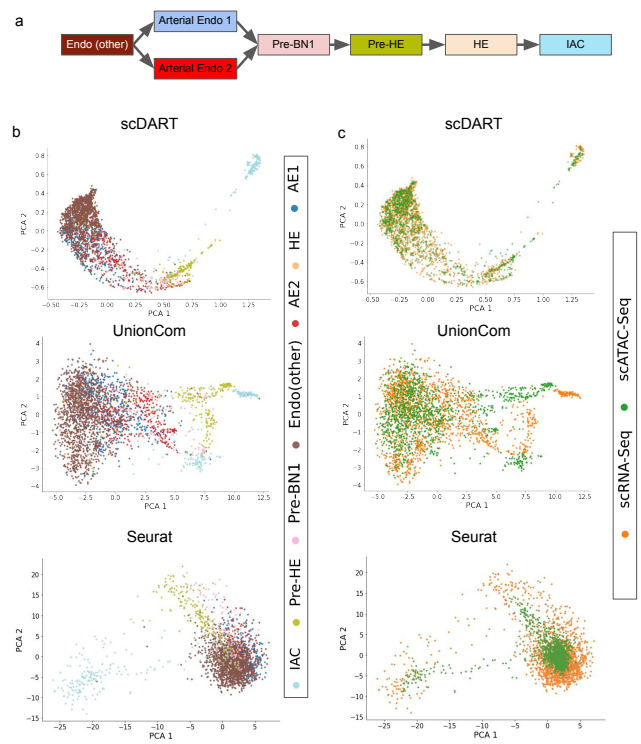

Fig. 3. (a) Expected backbone trajectory of the mouse endothelial cell development dataset. (b) The latent embedding of `scDART`, `UnionCom` and `Seurat` visualized with PCA. The cells are colored with their ground truth cell type annotation from the original publication (Zhu et al., 2020). (c) The latent embeddding of `scDART`, `UnionCom` and `Seurat`, where cells are colored with their corresponding data modalities.

Fig. 3b-c shows the latent embedding visualization of the datasets, obtained by `scDART`, `UnionCom` and `Seurat` respectively. Cell are colored with ground-truth cell type annotation (Fig. 3b, annotation from the original paper (Zhu *et al.*, 2020)) or the data modality information (Fig. 3c). The latent embedding of `scDART` corresponds the best to the expected backbone trajectory shown in Fig. 3a. The expected developmental path can be clearly observed as a curved line in `scDART` latent embedding. The change in the density of cells, that the cells are distributed densely at the beginning of the lineage and then becomes very sparse at the end of the path corresponds to the developmental "bottleneck" between "Pre-HE" and "HE" discussed in (Zhu *et al.*, 2020). This also shows that there may be a "speeding up" of cell differentiation towards the end of the trajectory. The results of `Seurat` can show the overall trajectory structure (Fig. 3b) but cells from the two data modalities do not merge well (Fig. 3c). In the case of `UnionCom`, there is no clear trajectory topology in the latent space

picture(0,0)(-35,0)(1,0)30 (0,-35)(0,1)30 picture picture(0,0)(35,0)(-1,0)30 (0,-35)(0,1)30 picture

picture(0,0)(-35,0)(1,0)30 (0,35)(0,-1)30 picture

picture(0,0)(35,0)(-1,0)30 (0,35)(0,-1)30 picture

embedding. The two modalities are overall mixed but some cell types are not well aligned in the latent space, such as the "Pre-HE" and "IAC" cells.

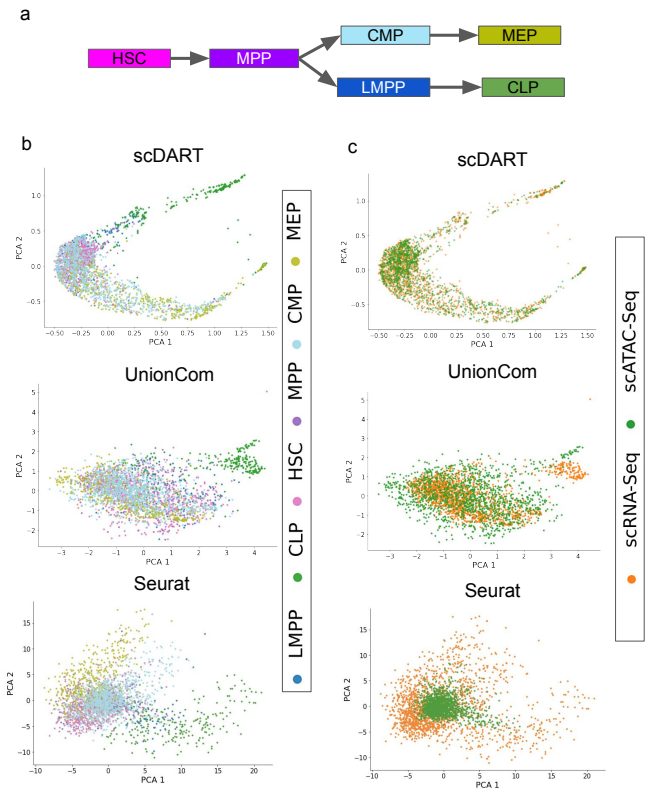

Fig. 4. (a) Expected trajectory backbone of the human hematopoiesis dataset. (b) The corresponding latent embedding of `scDART`, `UnionCom` and `Seurat` visualized with PCA. Cells are colored with their ground truth cell type annotations. (c) The same latent embedding as in (b) but cells are colored with their original modalities.

3.1.2 Results on the human hematopoiesis data

We test `scDART` on a human hematopoiesis dataset, which is a dataset with bifurcating differentiation trajectory (Fig. 4a). In this dataset, hematopoietic stem and progenitor cells (HSC) first become multipotent progenitors (MPP), where they undergo two potential differentiation branches until maturity: the Lymphoid-committed branch and the Erythroid-committed branch. Cells in Lymphoid-committed branch first transit into Lymphoid multipotent progenitors (LMPP) and mature into common lymphoid progenitor cell (CLP), whereas cells in Erythroid-committed branch undergoes a differentiation path to megakaryocyte-erythroid progenitor (MEP) cells through common myeloid progenitor (CMP) cells (Fig. 4a). Fig. 4b shows the latent embedding of `scDART` along with `UnionCom` and `Seurat`, where cells are colored by cell type. In the latent embedding of `scDART`, we can see clearly two different commitments, and Fig. 4c shows that `scDART` is able to learn latent space embedding where the cell types in the scRNA-seq and scATAC-seq data are well aligned. `UnionCom` again merges the two modalities reasonably well but does not preserve the trajectory structure. `Seurat` does a better job at preserving the trajectory compared to `UnionCom` but is worse with integrating the modalities (Fig. 4).

3.1.3 Results on the mouse neonatal brain cortex dataset

We further applied `scDART` along with `UnionCom` and `Seurat` to a mouse neonatal brain cortex dataset (Chen *et al.*, 2019). The Mouse neonatal brain cortex dataset includes 1469 cells, which are expected to

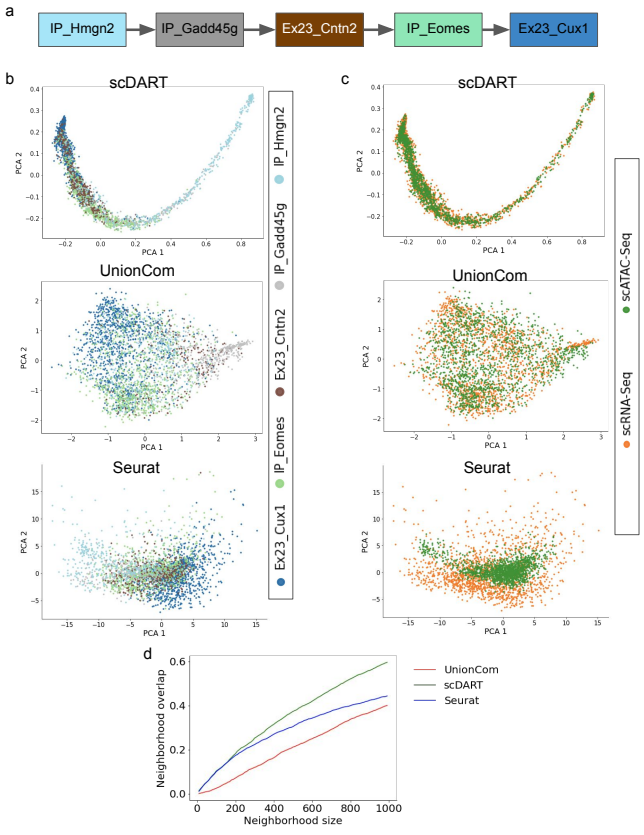

Fig. 5. (a) Expected trajectory backbone of cells in the mouse neonatal brain cortex dataset. (b) The latent embedding learned by `scDART`, `UnionCom` and `Seurat` visualized via PCA. Cells are colored with their corresponding cell type annotations. (c) The latent embedding of `scDART`, `UnionCom` and `Seurat`. Cells are colored according to their original data modalities. (d) The neighborhood overlap score of `scDART`, `UnionCom` and `Seurat` using different neighborhood size k.

follow a linear structure differentiation trajectory (Fig. 5a). This dataset was profiled with the SNARE-Seq technology, where the gene-expression and chromatin accessibility data were measured jointly in the same cells. Therefore, the input gene-expression matrix and chromatin accessibility data matrix share the same cells. When applying these methods, we assume that the cells in the two data matrices are different cells, that is, the two data matrices correspond to two different cell batches, and then use the one-to-one correspondence of cells between the two data matrices to test how well the data are integrated using the neighborhood overlap score.

Fig. 5b shows that the joint latent embedding learned through `scDART`, from the rightmost population to the leftmost one, preserves the differentiation trajectory best among all three methods. Meanwhile, `scDART` aligns the two batches of cells well (Fig. 5c). `UnionCom` does not preserve the cell differentiation trajectory (Fig. 5b) and `Seurat` does not merge the two batches as well as the other two methods.

In this paired dataset, since we have ground truth cell correspondence information across the two modalities, we can quantitatively measure how well the cells in the two modalities are aligned in the latent embedding. We calculate the neighborhood overlap score as described in Section 2.7. The results (Fig. 5d) show that `scDART` performs best in integrating the two datasets out of all three methods.

3.2 Results on simulated data

With simulated data we can measure the quality of the gene activity module learned by `scDART`. We also measure the accuracy of the learned latent embedding by performing trajectory inference algorithms (Section 2.5) on

picture(0,0)(-35,0)(1,0)30 (0,-35)(0,1)30 picture

picture(0,0)(35,0)(-1,0)30 (0,-35)(0,1)30 picture

picture(0,0)(-35,0)(1,0)30 (0,35)(0,-1)30 picture picture(0,0)(35,0)(-1,0)30 (0,35)(0,-1)30 picture

the latent embedding, and calculating the pseudotime accuracy and lineage separation score (Section 2.7) on the inferred trajectories.

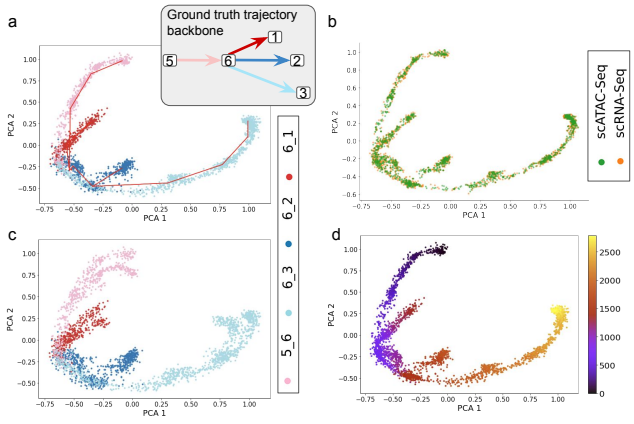

Fig. 6. Latent embedding visualization of a simulated dataset learned by `scDART`. The data has a trifurcating trajectory topology, and the latent embedding is visualized using PCA. (a) The latent embedding is colored using the ground truth lineage label. (b) The latent embedding is colored using different data modalities. (c) Pseudotime inference on the latent embedding, diffusion pseudotime algorithm is used. (d) Inferred trajectory backbone from the latent embedding, the backbone is shown in red lines, and the cells are colored with cluster labels.

3.2.1 Latent embedding quality

We simulated 18 datasets with different trajectory topologies: 6 with linear topology, 6 with bifurcating topology and 6 with trifurcating topology using the procedure described in Section 2.6. We benchmark the quality of the learned latent embedding by performing the trajectory inference algorithm on the latent embedding and quantifying the resulting trajectories, as we know the ground truth trajectory backbone. First, we show the latent embedding of `scDART` on a simulated dataset with trifurcating trajectory topology in Fig. 6. Figs. 6a-b show the visualization of the cells colored respectively with ground truth lineage and with original modality. Fig. 6a also includes our inferred trajectory backbone (in red lines), which aligns with the true trajectory. Fig. 6c shows the latent embedding without applying the post-processing step (Section 2.5). Without the post-processing step we can also recover the trajectories and the two modalities merge fairly well, but the post-processing step makes the trajectories more clear. Fig. 6d shows the same latent embedding as in Figs. 6a-b, but cells are colored with inferred pseudotime.

We then compare `scDART` with `UnionCom` in terms of the lineage separation score and the Kendall rank correlation (Kentall-τ). Given a simulated dataset, we first run `scDART` (or `UnionCom`) and obtain the latent embedding. Then we infer the trajectory and the cell pseudotime on the latent embedding as described in Section 2.5. For each dataset, each method is run 3 times and we include all results in the boxplots. Fig. 7a shows the comparison of the two methods in terms of correctly assigning cells to the right lineage. Fig. 7b compares the accuracy of inferred pseudotime from the two methods. Figs. 7a-b together show that the latent embedding learned through `scDART` leads to more accurate inferred trajectories than that of `UnionCom`.

3.2.2 Gene activity module accuracy

The gene activity module in `scDART` aims to learn the relationship between scATAC-seq data and scRNA-seq data, and here we test how well this learned gene activity module can predict scRNA-seq data from the scATAC-seq data. Using our simulation procedure, we generate N cells with jointly profiled scATAC-seq and scRNA-seq data. We then divide these N cells into 2 batches with respectively N_1 and N_2 cells. The input

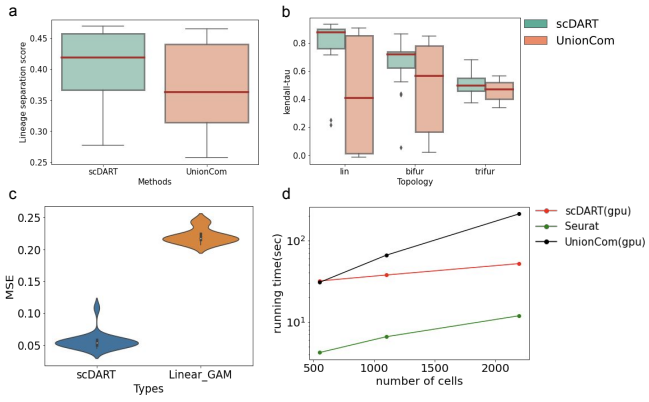

Fig. 7. (a) Boxplots comparing the lineage separation scores. Only datasets with bifurcating and trifurcating backbone structures are included. (b) Boxplots comparing Kendall rank correlation scores (Kendall-τ) of diffusion pseudotime run on the latent embedding of `scDART` and `UnionCom`. The scores are compared separately for different trajectory topologies (`lin`: linear, `bifur`: bifurcating, `trifur`: trifurcating). (c) Mean square error (MSE) of predicted scRNA-seq data by two different methods: using the learned gene activity module from `scDART` and applying a linear GAM. The `scDART` violin plot includes 18 data points, corresponding to 6 simulated datasets and 3 runs of `scDART` on each dataset. (d) Running time of `scDART` and other methods on datasets of different size.

data for `scDART` are the scRNA-seq data from batch 1, $\mathbf{X}_{\text{RNA}}^{\text{batch1}}$, and the scATAC-seq data from batch 2, $\mathbf{X}_{\text{ATAC}}^{\text{batch2}}$. After training `scDART`, we use the gene activity module to predict a scRNA-seq dataset from $\mathbf{X}_{\text{ATAC}}^{\text{batch2}}$, denoted by $\mathbf{X}_{\text{RNA}}^{\text{batch2_pred}}$. We then calculate the MSE between $\mathbf{X}_{\text{RNA}}^{\text{batch2_pred}}$ and the simulated scRNA-seq data of batch 2, $\mathbf{X}_{\text{RNA}}^{\text{batch2_true}}$ (after log-transforming the scRNA-seq matrices), to evaluate the performance of the learned gene activity module.

The baseline method we compare with is to simply multiply the GAM with $\mathbf{X}_{\text{ATAC}}^{\text{batch2}}$ to obtain $\mathbf{X}_{\text{RNA}}^{\text{batch2_pred}}$ as used in existing work (Welch *et al.*, 2019; Stuart *et al.*, 2019). As the procedure represents a linear transformation from the scATAC-seq to the scRNA-seq data, we denote this method as "linear_GAM". Here we apply the true GAM used in the simulation. Fig. 7c shows the MSE of each method to obtain predicted scRNA-seq data. The MSE of the `scDART` gene activity module which encodes the complex nonlinear relationship between the two data modalities predicts the scRNA-seq data much more accurately than a linear transformation, even when the true GAM is used.

3.3 Running time measurement

We test the scalability of `scDART`, `UnionCom` and `Seurat` by running those methods on datasets with different numbers of cells. The computation is performed with Nividia Tesla V100-16GB GPU and Xeon 6226 CPU. The running time on each dataset is shown in Fig.7d. Seurat runs the fastest for small datasets, but its running time increases fast with the number of cells. `scDART` comes in the second position with small datasets and its running time remains relatively stable with the increase of the data size. `UnionCom` takes the longest run time even with GPU acceleration.

4 Discussion and Conclusion

We have proposed `scDART`, a deep learning framework which learns a joint latent space of scRNA-seq and scATAC-seq data from different batches, while learning the relationship between genomic regions and genes, represented by a gene activity module neural network, which can encode complex nonlinear relationships between chromatin accessibility and gene-expression. `scDART` performs particularly well in preserving the continuous trajectory of cells when mapping data to the low-dimensional latent space, thanks to the usage of the diffusion distance between cells and the penalty on the difference of this distance between the low- and

picture(0,0)(-35,0)(1,0)30 (0,-35)(0,1)30 picture picture(0,0)(35,0)(-1,0)30 (0,-35)(0,1)30 picture

picture(0,0)(-35,0)(1,0)30 (0,35)(0,-1)30 picture picture(0,0)(35,0)(-1,0)30 (0,35)(0,-1)30 picture

high-dimensional space. On both real data and simulated data, `scDART` shows superior performance compared to state-of-the-art methods.

To our knowledge, `scDART` is the first method that performs the two important tasks, integrating two data modalities which are not jointly profiled and learning the cross-modality relationships, simultaneously in the case of scATAC-seq and scRNA-seq data. In the era of single cell multi-omics, the goal of data integration should be not only removing batch effects and compiling larger datasets, but also learning the relationship between different data modalities. We expect that more methods which can learn relationship across modalities will be developed in the future to take advantage of the multi-modal omics data.

Acknowledgement

The authors would like to thank Dr. Xi Chen (from SUSTech) for helpful discussions.

This work was supported in part by the US National Science Foundation DBI-2019771. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of NSF.

References

Amodio, M., van Dijk, D., Srinivasan, K., Chen, W. S., Mohsen, H., Moon, K. R., Campbell, A., Zhao, Y., Wang, X., Venkataswamy, M., Desai, A., Ravi, V., Kumar, P., Montgomery, R., Wolf, G., and Krishnaswamy, S. (2019). Exploring single-cell data with deep multitasking neural networks. *Nat. Methods*, **16**(11), 1139–1145.

Argelaguet, R., Arnol, D., Bredikhin, D., Deloro, Y., Velten, B., Marioni, J. C., and Stegle, O. (2020). MOFA+: a statistical framework for comprehensive integration of multi-modal single-cell data. *Genome Biol.*, **21**(1), 111.

Bottou, L. (2012). Stochastic gradient descent tricks. In G. Montavon, G. B. Orr, and K.-R. Müller, editors, *Neural Networks: Tricks of the Trade: Second Edition*, pages 421–436. Springer Berlin Heidelberg, Berlin, Heidelberg.

Buenrostro, J. D., Corces, M. R., Lareau, C. A., Wu, B., Schep, A. N., Aryee, M. J., Majeti, R., Chang, H. Y., and Greenleaf, W. J. (2018). Integrated Single-Cell analysis maps the continuous regulatory landscape of human hematopoietic differentiation. *Cell*, **173**(6), 1535–1548.e16.

Butler, A., Hoffman, P., Smibert, P., Papalexi, E., and Satija, R. (2018). Integrating single-cell transcriptomic data across different conditions, technologies, and species. *Nat. Biotechnol.*, **36**(5), 411–420.

Cao, J., Cusanovich, D. A., Ramani, V., Aghamirzaie, D., Pliner, H. A., Hill, A. J., Daza, R. M., McFaline-Figueroa, J. L., Packer, J. S., Christiansen, L., Steemers, F. J., Adey, A. C., Trapnell, C., and Shendure, J. (2018). Joint profiling of chromatin accessibility and gene expression in thousands of single cells. *Science*, **361**(6409), 1380–1385.

Cao, K., Bai, X., Hong, Y., and Wan, L. (2020). Unsupervised topological alignment for single-cell multi-omics integration. *Bioinformatics*, **36**(Supplement_1), i48–i56.

Chen, S., Lake, B. B., and Zhang, K. (2019). High-throughput sequencing of the transcriptome and chromatin accessibility in the same cell. *Nat. Biotechnol.*, **37**(12), 1452–1457.

Chen, X., Miragaia, R. J., Natarajan, K. N., and Teichmann, S. A. (2018). A rapid and robust method for single cell chromatin accessibility profiling. *Nat. Commun.*, **9**(1), 5345.

Cui, Z., Chang, H., Shan, S., and Chen, X. (2014). Generalized unsupervised manifold alignment. *Adv. Neural Inf. Process. Syst.*, **27**, 2429–2437.

Duren, Z., Chen, X., Zamanighomi, M., Zeng, W., Satpathy, A. T., Chang, H. Y., Wang, Y., and Wong, W. H. (2018). Integrative analysis of single-cell genomics data by coupled nonnegative matrix factorizations. *Proc. Natl. Acad. Sci. U. S. A.*, **115**(30), 7723–7728.

Dziugaite, G. K., Roy, D. M., and Ghahramani, Z. (2015). Training generative neural networks via maximum mean discrepancy optimization.

Haghverdi, L., Büttner, M., Wolf, F. A., Buettner, F., and Theis, F. J. (2016). Diffusion pseudotime robustly reconstructs lineage branching. *Nat. Methods*, **13**(10), 845–848.

Haghverdi, L., Lun, A. T. L., Morgan, M. D., and Marioni, J. C. (2018). Batch effects in single-cell RNA-sequencing data are corrected by matching mutual nearest neighbors. *Nat. Biotechnol.*, **36**(5), 421–427.

Hofmann, T. (1999). Probabilistic latent semantic indexing. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '99, pages 50–57, New York, NY, USA. Association for Computing Machinery.

Korsunsky, I., Millard, N., Fan, J., Slowikowski, K., Zhang, F., Wei, K., Baglaenko, Y., Brenner, M., Loh, P.-R., and Raychaudhuri, S. (2019). Fast, sensitive and accurate integration of single-cell data with harmony. *Nat. Methods*, **16**(12), 1289–1296.

Ma, S., Zhang, B., LaFave, L. M., Earl, A. S., Chiang, Z., Hu, Y., Ding, J., Brack, A., Kartha, V. K., Tay, T., Law, T., Lareau, C., Hsu, Y.-C., Regev, A., and Buenrostro, J. D. (2020). Chromatin potential identified by shared Single-Cell profiling of RNA and chromatin. *Cell*, **183**(4), 1103–1116.e20.

Moon, K. R., van Dijk, D., Wang, Z., Gigante, S., Burkhardt, D. B., Chen, W. S., Yim, K., van den Elzen, A., Hirn, M. J., Coifman, R. R., Ivanova, N. B., Wolf, G., and Krishnaswamy, S. (2019). Visualizing structure and transitions in high-dimensional biological data. *Nat. Biotechnol.*, **37**(12), 1482–1492.

Munsky, B., Neuert, G., and van Oudenaarden, A. (2012). Using gene expression noise to understand gene regulation. *Science*, **336**(6078), 183–187.

Nadler, B., Lafon, S., Coifman, R. R., and Kevrekidis, I. G. (2005). Diffusion maps, spectral clustering and eigenfunctions of Fokker-Planck operators.

Pellin, D., Loperfido, M., Baricordi, C., Wolock, S. L., Montepeloso, A., Weinberg, O. K., Biffi, A., Klein, A. M., and Biasco, L. (2019). A comprehensive single cell transcriptional landscape of human hematopoietic progenitors. *Nat. Commun.*, **10**(1), 2395.

Singh, R., Demetci, P., Bonora, G., Ramani, V., Lee, C., Fang, H., Duan, Z., Deng, X., Shendure, J., Disteche, C., and Noble, W. S. (2020). Unsupervised manifold alignment for single-cell multi-omics data. In *Proceedings of the 11th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics*, pages 1–10. Association for Computing Machinery, New York, NY, USA.

Stuart, T., Butler, A., Hoffman, P., Hafemeister, C., Papalexi, E., Mauck, 3rd, W. M., Hao, Y., Stoeckius, M., Smibert, P., and Satija, R. (2019). Comprehensive integration of Single-Cell data. *Cell*, **177**(7), 1888–1902.e21.

Traag, V. A., Waltman, L., and van Eck, N. J. (2019). From louvain to leiden: guaranteeing well-connected communities. *Sci. Rep.*, **9**(1), 5233.

Van der Maaten, L. and Hinton, G. (2008). Visualizing data using t-sne. *Journal of machine learning research*, **9**(11).

Welch, J. D., Hartemink, A. J., and Prins, J. F. (2017). MATCHER: manifold alignment reveals correspondence between single cell transcriptome and epigenome dynamics. *Genome Biol.*, **18**(1), 138.

Welch, J. D., Kozareva, V., Ferreira, A., Vanderburg, C., Martin, C., and Macosko, E. Z. (2019). Single-Cell multi-omic integration compares and contrasts features of brain cell identity. *Cell*, **177**(7), 1873–1887.e17.

Wolf, F. A., Hamey, F. K., Plass, M., Solana, J., Dahlin, J. S., Göttgens, B., Rajewsky, N., Simon, L., and Theis, F. J. (2019). PAGA: graph abstraction reconciles clustering with trajectory inference through a topology preserving map of single cells. *Genome Biol.*, **20**(1), 59.

Zhang, X., Xu, C., and Yosef, N. (2019). Simulating multiple faceted variability in single cell RNA sequencing. *Nat. Commun.*, **10**(1), 2611.

Zhu, Q., Gao, P., Tober, J., Bennett, L., Chen, C., Uzun, Y., Li, Y., Howell, E. D., Mumau, M., Yu, W., He, B., Speck, N. A., and Tan, K. (2020). Developmental trajectory of prehematopoietic stem cell formation from endothelium. *Blood*, **136**(7), 845–856.

picture(0,0)(-35,0)(1,0)30 (0,-35)(0,1)30 picture picture(0,0)(35,0)(-1,0)30 (0,-35)(0,1)30 picture