Check for updates

Modeling gene regulatory networks using neural network architectures

Hantao Shu [1], **Jingtian Zhou**[2,3], **Qiuyu Lian**[4,5], **Han Li** [1], **Dan Zhao** [1], **Jianyang Zeng** [1] ✉ and **Jianzhu Ma** [6] ✉

Gene regulatory networks (GRNs) encode the complex molecular interactions that govern cell identity. Here we propose DeepSEM, a deep generative model that can jointly infer GRNs and biologically meaningful representation of single-cell RNA sequencing (scRNA-seq) data. In particular, we developed a neural network version of the structural equation model (SEM) to explicitly model the regulatory relationships among genes. Benchmark results show that DeepSEM achieves comparable or better performance on a variety of single-cell computational tasks, such as GRN inference, scRNA-seq data visualization, clustering and simulation, compared with the state-of-the-art methods. In addition, the gene regulations predicted by DeepSEM on cell-type marker genes in the mouse cortex can be validated by epigenetic data, which further demonstrates the accuracy and efficiency of our method. DeepSEM can provide a useful and powerful tool to analyze scRNA-seq data and infer a GRN.

The rapid development of single-cell sequencing technologies[1–3] provides unprecedented opportunities for biologists to investigate cellular states. However, it also poses new challenges in the form of experimental noise not found in bulk sequencing data[4,5], which might significantly decrease the accuracy of downstream bioinformatics analysis by introducing biases in the gene expression. To address these problems, recently, there has been great interest in applying deep learning models to filter the noise in single-cell transcriptome data by modeling the complicated interaction patterns among the genes[6,7]. Inherently, the transcriptome of a cell is governed by its gene regulation process in a cell-specific manner. Hence, we expect that the deep learning-based methods are able to model such gene interactions to reveal a more clear landscape of cell heterogeneity, capturing both transcriptomic similarities between cells of the same cell type and differences across different cell types[8–10]. However, so far, deep learning-based single-cell analysis framework[6,7,11–13] are usually black boxes and it is hard to evaluate to what extent gene regulatory network (GRN) structure or any other internal structure of the data is learned.

A number of computational models[14–19] have attempted to incorporate GRN inference into their single-cell data analysis models. One class of these methods relies on side measurements such as single-cell chromatin accessibility or transcription factor (TF) binding motifs[19]. However, these measurements often require more complicated experimental designs and could also introduce additional noise as these data could come from different experiments. Current methods solely based on single-cell RNA sequencing (scRNA-seq) data also have explicit limitations. For example, it is common for GRN inference algorithms to use statistics algorithms that focus on the co-expression networks instead of decoding the casual relationships among TFs and their corresponding target genes[15,20]. In addition, most algorithms that incorporate gene interactions are linear models[16,17] or tree-based models[14,18] and it is generally hard to directly generalize these approaches to more comprehensive

nonlinear frameworks and benefit from the computational power that the deep learning model brought to us.

To address the above problems, we present DeepSEM, a deep generative model that can jointly embed the gene expression data and simultaneously construct a GRN that reflects the inner structure of gene interactions in single cells without relying on any additional information such as TF binding motifs or single-cell ATAC sequencing (scATAC-seq) data. To implement such an idea, we took inspiration from the work of Yu et al.[21], which generalized a popular approach, called the structural equation model (SEM), that infers the causality using a linear model. By adding proper mathematical constraints, part of the neural network architecture could be used to predict the GRN of the scRNA-seq data. A previous study by Lin et al.[22] showed that more accurate cell representations could be achieved by guiding the neural network architecture with a GRN structure derived from the literature and databases. In this Article, we show that the neural network architecture can reflect GRN structure by properly designing the neural network layer without relying on any prior knowledge. The neural network architecture can be inferred jointly with the training of the weights of the neural network in an end-to-end manner. The overall framework of DeepSEM is a beta-variational autoencoder (beta-VAE)[23] in which the weights of both the encoder and decoder functions represent the adjacency matrix of the GRN. Our model does not require any extra experimental data such as open chromatin information, ChIP sequencing (ChIP-seq) data or TF binding motifs to infer the GRN structure. The nonlinear neural networks in DeepSEM are employed to address the challenges in single-cell data analysis, such as experimental noise, high dimensionality and scalability. In addition, by explicitly modeling the GRN, DeepSEM is more 'transparent' than the conventional neural network models and can reduce the overfitting problem of deep learning models by greatly restricting the parameter space. By inspecting the architecture of the model that represents the inner workings of a cell, we can observe how

[1]Institute for Interdisciplinary Information Sciences, Tsinghua University, Beijing, China. [2]Genomic Analysis Laboratory, The Salk Institute for Biological Studies, La Jolla, CA, USA. [3]Bioinformatics and Systems Biology Program, University of California, San Diego, La Jolla, CA, USA. [4]UM-SJTU Joint Institute, Shanghai Jiao Tong University, Shanghai, China. [5]Department of Automation, Shanghai Jiao Tong University, Shanghai, China. [6]Institute for Artificial Intelligence, Peking University, Beijing, China. ✉e-mail: zengjy321@mail.tsinghua.edu.cn; majianzhu@pku.edu.cn

multiple genes interact with each other to determine the expression levels of individual genes.

We evaluate the performance of DeepSEM for various single-cell tasks such as GRN inference, scRNA-seq data visualization, cell-type identification and cell simulations on several benchmark datasets. We first show that DeepSEM is able to achieve better performance on the GRN inference task compared with the state-of-the-art algorithms on several popular benchmark datasets. We also apply DeepSEM to another single-cell dataset without the ground-truth GRN measured, and provide extensive evidence extracted from the single-cell DNA methylation and open chromatin data to demonstrate the accuracy and efficiency of our algorithm. Moreover, we also evaluate the quality of the single-cell representation regularized by the GRN structure. We find that DeepSEM can achieve comparable or better performance compared with current state-of-the-art methods on the tasks of visualization and cell-type identification on various benchmark datasets.

Another important functional component of DeepSEM is to simulate scRNA-seq data by perturbing the values of its hidden neurons. In silico data simulations have already achieved tremendous success in computer vision for data augmentation, especially when the number of training samples is limited[24]. In single-cell biology, the same types of simulation algorithm have also been applied to scRNA-seq data to predict the single-cell perturbation response out of sample[19,25], identify marker genes[26] and augment the sparse cell populations to improve the accuracy of cell-type classification[27]. The state-of-the-art simulation algorithms[27] aim to generate 'realistic' scRNA-seq data based on the generative adversarial networks (GANs) to make the low-dimensional projection of the simulated data indistinguishable from the data distributions of the real cells. In this Article, we show that DeepSEM is able to achieve more realistic simulations compared with other GAN-based models through guiding the information flow using the GRN layer and mirroring the in vivo generation process of RNA governed by multiple TFs. In addition, we propose another concept, called the GRN consistency, to measure the quality of the simulated single-cell data. In particular, the GRN consistency measures the difference of the predicted GRN between the simulated and real scRNA-seq data, which accesses how much the computational model captures both the marginal and conditional independence from the original distribution, and, more importantly, quantifies how likely it is that one method can generate realistic scRNA-seq data satisfying the biological constraints. Tests on several benchmark datasets show that DeepSEM is able to achieve realistic scRNA-seq profiles and higher GRN consistency compared with the state-of-the-art single-cell simulators.

Results

Overview of the DeepSEM framework. Given an scRNA-seq dataset as input, DeepSEM jointly models the GRN and the transcriptome by generating the SEM with a beta-VAE framework (Fig. 1). We designed two neural network layers, named the GRN layer and the inverse GRN layer, to explicitly model the GRN structure (Methods). Different from conventional deep learning models that embed the expressions of all the genes together into a latent space[6,7], the encoder function of DeepSEM takes the expression of only one gene as the input feature of the neural network. The neural networks for different genes share their weights or it could be viewed as using one neural network to scan all the genes. At this step, there are no interactions among different genes in the model. Later, another two fully connected neural networks transform the output of these small neural networks to the posterior mean and standard deviation of a multivariate Gaussian distribution. Decoupling the nonlinear operation and the gene interaction is the key for DeepSEM to achieve more robust and interpretable hidden representations at the same time. Next, a decoder function equipped with the inverse GRN layer transforms the hidden representations back to the gene
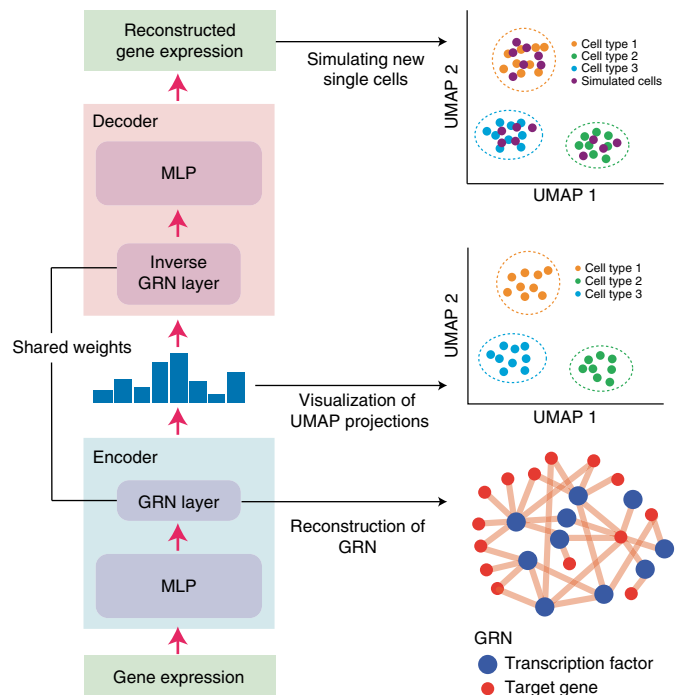

Fig. 1 | Overview of DeepSEM. Left: DeepSEM is a generative model including two main modules: an encoder (bottom left) and a decoder (top left). Right: DeepSEM performs three major functions by leveraging different modules: (1) GRN prediction (bottom right), (2) scRNA-seq data embedding and visualization (middle right), and (3) scRNA-seq simulation (top right).

expression values, which makes the entire framework an autoencoder (Fig. 2 and Methods). By jointly modeling GRN and single-cell transcriptome data, DeepSEM acts as a multipurpose tool that could serve for various tasks in single-cell data analysis by analyzing different modules.

Performance of GRN inference. To evaluate the performance of DeepSEM on GRN inference, we followed the BEELINE framework[28], which collected four different kinds of ground-truth network (Supplementary Table 1) and seven scRNA-seq datasets including five cell lines from mouse and two cell lines from human (Supplementary Table 2). For each dataset, as recommended by Pratapa et al.[28], we considered only highly variable TFs and top N ($N = 500$ and 1,000) most-varying genes (details in Supplementary Section 1). The performance was evaluated by the early precision ratio (EPR) (Fig. 3) and area under the precision–recall curve ratio (AUPRC ratio) (Supplementary Fig. 1) as used in the BEELINE framework[28], which are defined as the odds ratio of the true positives among the top K predicted edges and the AUPRC between the model and the random predictions. Here K denotes the number of edges in ground-truth GRN. We compared DeepSEM with six baseline algorithms, including GENIE3[14], PIDC[15], GRNBoost2[18], SCODE[16], ppcor[20] and SINCERITIES[17], which had been proved to achieve state-of-the-art performance on the benchmark datasets based on the evaluation of BEELINE[28]. To achieve stable predictions from deep learning models, we use the ensemble strategy to generate the final predictions (Methods). We provide a brief introduction of these baseline methods for their functionalities and their running details can be found in Supplementary Section 2.

Overall, DeepSEM outperforms all the other baseline methods on scRNA-seq datasets in terms of both EPR and AUPRC ratio metrics (Fig. 3 and Supplementary Fig. 1). DeepSEM achieves the best
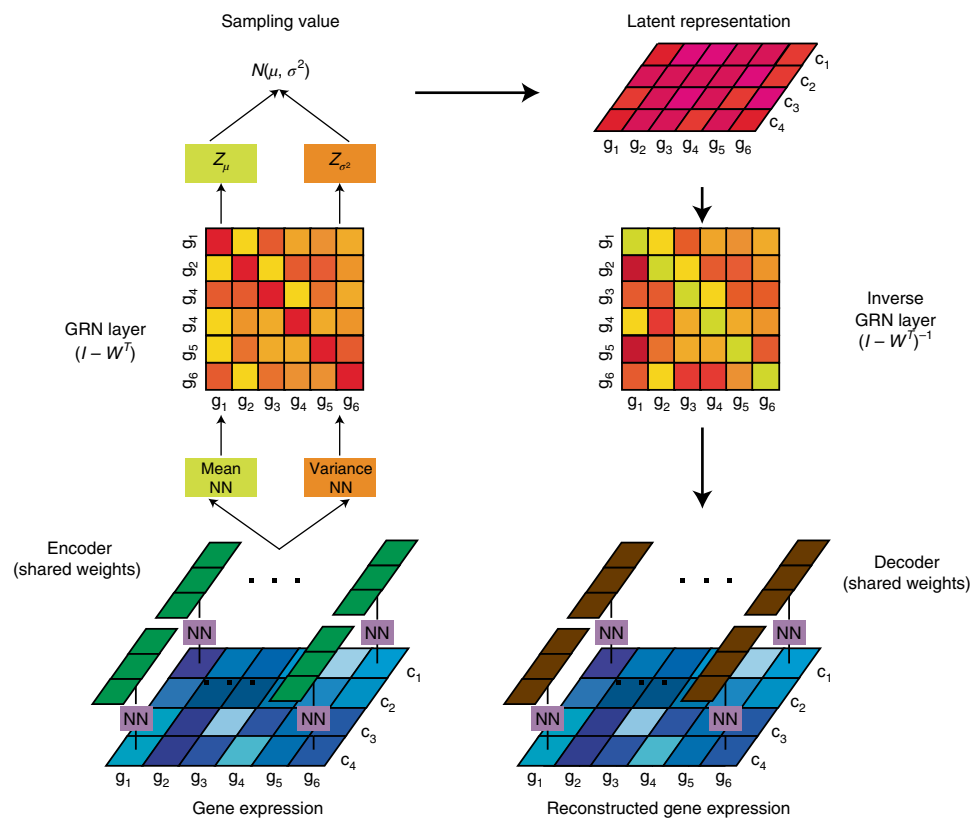
Fig. 2 | The neural network architecture of DeepSEM. The VAE of DeepSEM contains four modules: encoder, GRN layer, inverse GRN layer and decoder. The encoder and decoder are both MLPs taking one gene as input, and the weights of encoder and decoder are shared between different genes. The GRN layer and inverse GRN layer are both gene interaction matrices, which explicitly model the GRN network and guide the information flow of the neural networks. g_1–g_6, the name of genes in gene expression data; c_1–c_4, the name of the cells in gene expression data; NN, neural network.
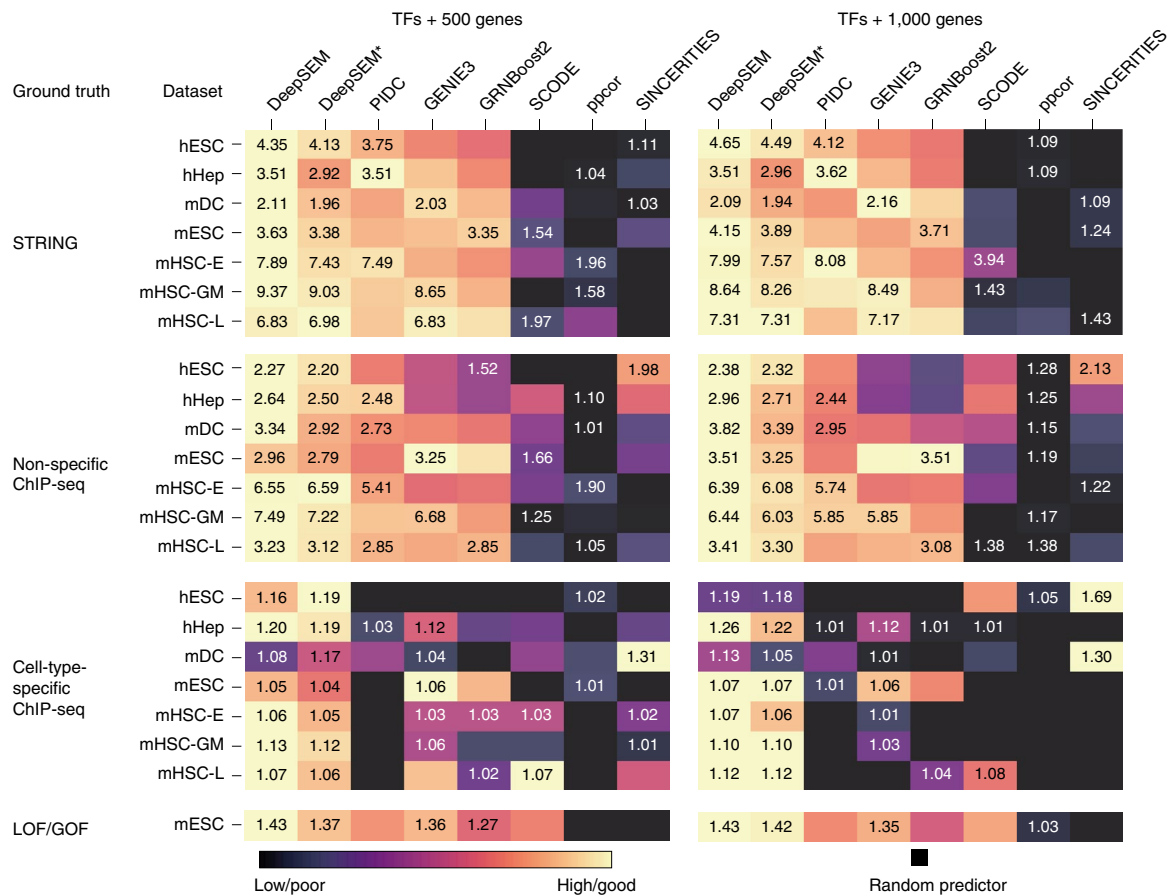
prediction performance on 81.82% (36/44) of the benchmarks and at least 10% improvement compared with the second-best approach (GENIE3) on 54.54% (24/44) of the benchmarks when evaluated using EPR. When considering the AUPRC ratio metric, DeepSEM achieves the best prediction performance on 86.36% (38/44) of the benchmarks and at least 10% improvement compared with the second-best approach (PIDC) on at least 54.54% (24/44) of benchmarks. In addition, DeepSEM substantially outperforms SCODE, ppcor and SINCERITIES on most benchmarks. In particular, among all the benchmarks, DeepSEM achieves an average of improvement of 12.46%, 14.38% and 20.70% compared with PIDC, GENIE3 and GRNBoost2, respectively, and performs equally or better than these approaches on 90.91% (40/44), 93.18% (41/44) and 95.45% (42/44) on all the benchmarks in both evaluation metrics.

Next, we investigate how the performance of DeepSEM is influenced by the number of cells and whether DeepSEM could work with limited training data (Supplementary Fig. 2a,b). We first constructed five datasets by subsampling 400, 300, 200, 100 and 50 single cells from the BEELINE benchmark[28] and evaluated the accuracy of the GRN prediction of DeepSEM on these five datasets. First, we found that the performance was relatively stable between 200 and 400 cells. The performance drops about 20% when only 50 single cells are provided to our model (Supplementary Fig. 2a). Second, we analyzed the performance with respect to different kinds of ground truth. As shown in Supplementary Fig. 2b, the performance drops significantly on cell-type non-specific ground truth (STRING and cell-type non-specific ChIP-seq) with 31.47% (STRING) and 28.35% (cell-type non-specific ChIP-seq), respectively, when only 50 cells are used as training data. In contrast, the

performance drops only 3.39% when using cell-type specific ChIP-seq as the ground truth. In the end, we found that the ensemble strategy provided greater performance improvement when the training cell was extremely limited, as shown in Supplementary Fig. 2c. The ensemble strategy provides additional performance improvement in the range of 3.80% to 12.80% (on average) when the size of the data is extremely small. In addition to performance comparison, we also studied the scalability of DeepSEM for large-scale datasets (Supplementary Section 3 and Supplementary Figs. 3 and 4).

Validation of GRN using epigenetic data. DNA methylation and chromatin accessibility can affect the binding of transcription factors to *cis*-regulatory elements and therefore influence the expression of downstream target genes[29,30]. To further explore the biological significance of the GRN identified by DeepSEM, we also examined the concordance between the gene regulations predicted by DeepSEM with the ones inferred from cell-type specific epigenetic data. Previous studies have reported that integrating TF binding motif information with epigenetic data can accurately predict TF binding sites in a cell-type-specific manner[31,32]. Therefore, we hypothesize that if one TF is regulating a gene in a given cell type, it should be more likely to associate the TF motifs with hypo CG methylation and open chromatins at the flanking regions of the target gene in the corresponding cell type.

To test this hypothesis, we applied the DeepSEM framework to an scRNA-seq dataset from the mouse cortex[33], and compared the results with the single-nucleus methyl-cytosine sequencing (snmC-seq)[34] and scATAC-seq data[35]. To search for epigenetic evidence supporting regulations of the marker genes, we used the

Ground truth	Dataset	TFs + 500 genes								TFs + 1,000 genes							
		DeepSEM	DeepSEM*	PIDC	GENIE3	GRNBoost2	SCODE	ppcor	SINCERITIES	DeepSEM	DeepSEM*	PIDC	GENIE3	GRNBoost2	SCODE	ppcor	SINCERITIES
STRING	hESC	4.35	4.13	3.75					1.11	4.65	4.49	4.12				1.09	
	hHep	3.51	2.92	3.51				1.04		3.51	2.96	3.62				1.09	
	mDC	2.11	1.96		2.03				1.03	2.09	1.94		2.16				1.09
	mESC	3.63	3.38			3.35	1.54			4.15	3.89			3.71			1.24
	mHSC-E	7.89	7.43	7.49				1.96		7.99	7.57	8.08				3.94	
	mHSC-GM	9.37	9.03		8.65			1.58		8.64	8.26		8.49		1.43		
	mHSC-L	6.83	6.98		6.83		1.97			7.31	7.31		7.17				1.43
Non-specific ChIP-seq	hESC	2.27	2.20		1.52				1.98	2.38	2.32					1.28	2.13
	hHep	2.64	2.50	2.48				1.10		2.96	2.71	2.44				1.25	
	mDC	3.34	2.92	2.73				1.01		3.82	3.39	2.95				1.15	
	mESC	2.96	2.79		3.25		1.66			3.51	3.25		3.51			1.19	
	mHSC-E	6.55	6.59	5.41				1.90		6.39	6.08	5.74					1.22
	mHSC-GM	7.49	7.22		6.68		1.25			6.44	6.03	5.85	5.85			1.17	
	mHSC-L	3.23	3.12	2.85		2.85		1.05		3.41	3.30		3.08		1.38	1.38	
Cell-type-specific ChIP-seq	hESC	1.16	1.19					1.02		1.19	1.18					1.05	1.69
	hHep	1.20	1.19	1.03	1.12					1.26	1.22	1.01	1.12	1.01	1.01		
	mDC	1.08	1.17		1.04				1.31	1.13	1.05		1.01				1.30
	mESC	1.05	1.04		1.06		1.01			1.07	1.07	1.01	1.06				
	mHSC-E	1.06	1.05		1.03	1.03	1.03		1.02	1.07	1.06		1.01				
	mHSC-GM	1.13	1.12		1.06				1.01	1.10	1.10		1.03				
	mHSC-L	1.07	1.06			1.02	1.07			1.12	1.12				1.04	1.08	
LOF/GOF	mESC	1.43	1.37		1.36	1.27				1.43	1.42		1.35			1.03	

Low/poor — High/good ▪ Random predictor

Fig. 3 | Summary of the GRN prediction performance in terms of EPR. The performance of GRN inference on seven different datasets with four distinct ground-truth sources using 500 (left) and 1,000 (right) most-varying genes and all varying TFs evaluated by the median EPR value over ten repeats. For each dataset, the color is scaled between 0 and 1 using the minimum–maximum scale on EPR value. The black squares denote performance worse than random predictors. DeepSEM* denotes the DeepSEM without using ensemble learning. EPR is defined as the odds ratio of the true positives among the top K predicted edges between the model and the random predictions where K denotes the number of edges in ground-truth GRN.

scATAC-seq peaks and differentially methylated regions (DMRs)[34] as potential regulatory elements for each cell-type-specific gene ($\pm$200 kb of transcription start sites), and then for each gene calculated the proportion of its predicted regulators whose motifs are located in the regulatory elements near the gene. Consistent with our hypothesis, we found substantial enrichment of different types of regulatory region containing the motifs of predicted TFs (Fig. 4a and Supplementary Fig. 5), suggesting high accuracy of the GRN identified by DeepSEM from the epigenetic aspect. We provide the summary statistics for the predicted GRN in Supplementary Section 4 and Supplementary Table 3.

More concretely, we examined the predictions related to *Rorb*, which is a gene encoding a nuclear hormone receptor highly expressed in L4 cells. We applied DeepSEM to predict the regulators of *Rorb* in L4, and our model identified 12 putative upstream TFs regulating *Rorb*, including genes *Mef2c*, *Nr2f1* and *Pknox2* (Supplementary Table 4). We found that the binding motifs of these TFs are located in the first intron of *Rorb* where the cytosines within the motifs were specifically hypomethylated in L4, indicating the cell-type-specific regulation relationship between these TFs and *Rorb* (Fig. 4b). As another example, *Syt6* encodes a transmembrane protein that involves synaptic vesicle exocytosis, whose expression is restricted to L6 CT cells. We predicted that *Syt6* could be regulated by *Nf1a*, *Stat1* and *Sp3* in L6 CT (Supplementary Table 4). Accordingly, by comparing with scATAC-seq data, we observed that the regions associated with the binding motifs of these TFs along the *Syt6* gene body were specifically open in L6 CT cells (Fig. 4c).

Notably, all these six upstream TFs were consistently expressed in both the target and non-target cell types (Fig. 4d). Together, these studies provide orthogonal evidence to support our predicted GRN and indicate the potential utility of DeepSEM to study the cell-type-specific gene interaction networks, especially on the wide range of housekeeping TFs that have large functional impact on the cell dynamics.

Cell representation. Previous studies have indicated that more biologically meaningful representations for scRNA-seq could be generated by considering the interactions among different genes, such as protein–protein interaction networks[36], GRNs[37], co-expression networks from bulk RNA-seq data and annotated pathways[38]. In particular, linking regulatory relationships to gene expression has been proved to be able to effectively overcome dropout and other technical variations in both single-cell and bulk sequencing experiments[39,40]. Since the cell representation of DeepSEM is a nonlinear mapping from the expression to GRN activities, we hypothesized that the hidden representation can also effectively define cell states and cell types by explicitly modeling the GRN structure. In the encoder function (Fig. 2 and Methods), the nonlinear function can be viewed as a denoising function to amplify or suppress the values of the expression of certain genes; whereas the GRN layer can be viewed as a scoring function of calculating the activity of each regulon. To evaluate the quality of these representations, we applied DeepSEM to identify different cell types on nine scRNA-seq datasets, including
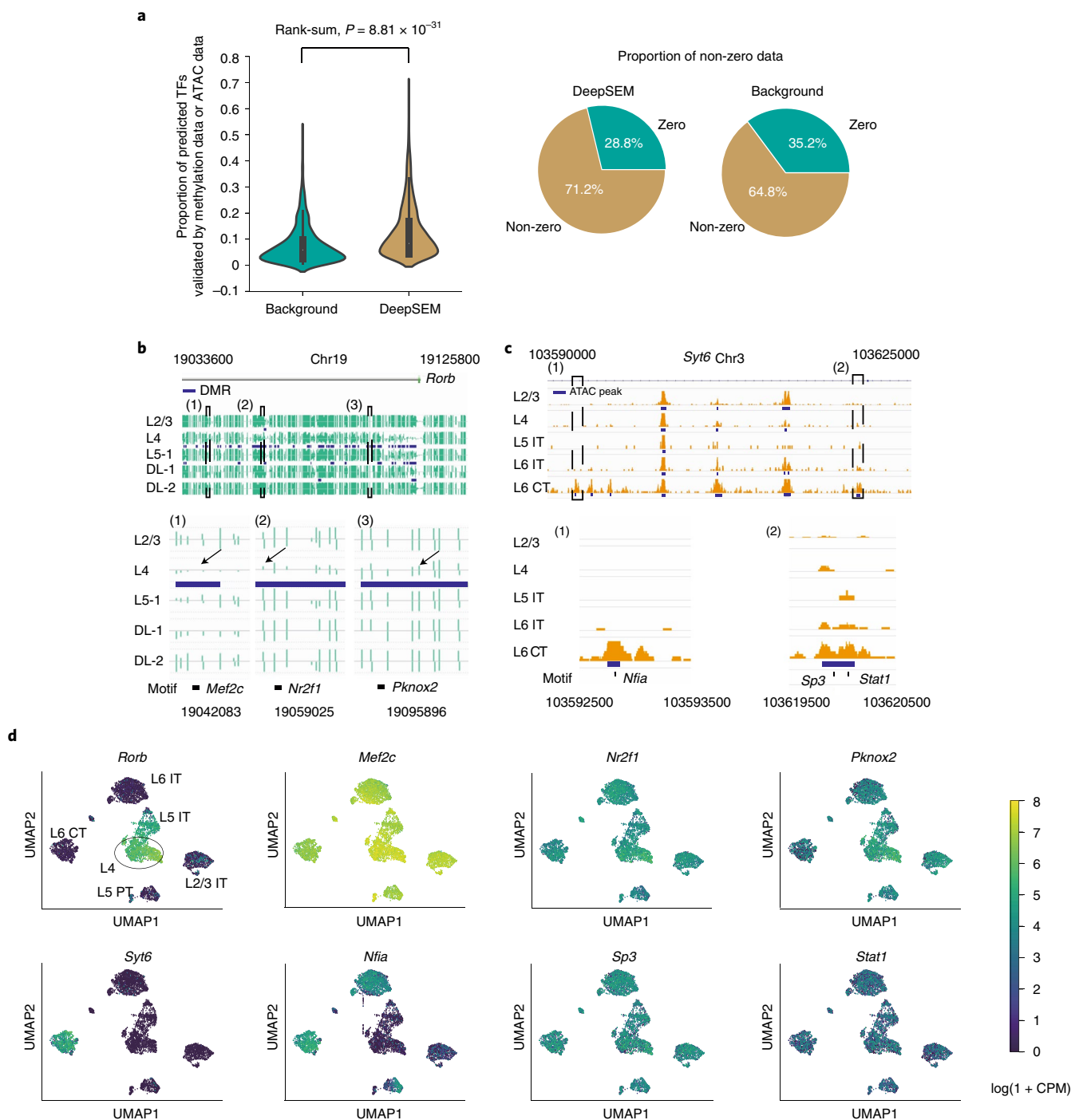
Fig. 4 | Validating GRN prediction using epigenetic data. a, For each target gene ($n = 1{,}822$ genes), the proportion of the DeepSEM predicted TFs (top 5% among all TFs) whose motifs were in the ATAC peak or DMR region at ± 200 kb of its transcription start sites compared with background (random selections). Genes whose motifs were in the ATAC peak or DMR region at ± 200 kb of its transcription start sites but without any regulators were removed. We ran the experiments for five times and reported the average P values calculated by the one-sided Wilcoxon test. Left: The violin plot of non-zero data. Right: the proportion of non-zero data. **b**, The genome browser view of the L4 marker gene *Rorb*. The green bar shows the methylation level of each CpG site, and bars on different sides of the x axis represent different strands. Black arrows signify the differentially methylated CpG sites (DMS) within/next to each motif, and the coordinates of these DMSs are shown on the bottom. The cluster labels from Luo et al.[34] were used, where L5-1 represents L5 IT, and DL-1 and DL-2 are subtypes of L6 IT. **c**, The genome browser view of L6 CT maker gene *Syt6*. Orange bars show the ATAC-seq signals. Only exemplified cell types related to the comparison are shown. **d**, UMAP plots of the six cell types used in our study ($n = 6{,}456$ cells) colored by expression levels of *Rorb*, *Syt6* or their predicted regulators in the corresponding cell types.

a widely used mouse brain dataset (the Zeisel dataset[41]), a mouse embryo dataset[42] and a mouse peripheral blood mononuclear cell (PBMC) dataset[43]. A complete list of these datasets can be found

in Supplementary Table 5. To benchmark DeepSEM, we also compared its low-dimensional embeddings to four other methods: scVI[6], DCA[7], ZIFA[44] and factor analysis (FA)[45], following the
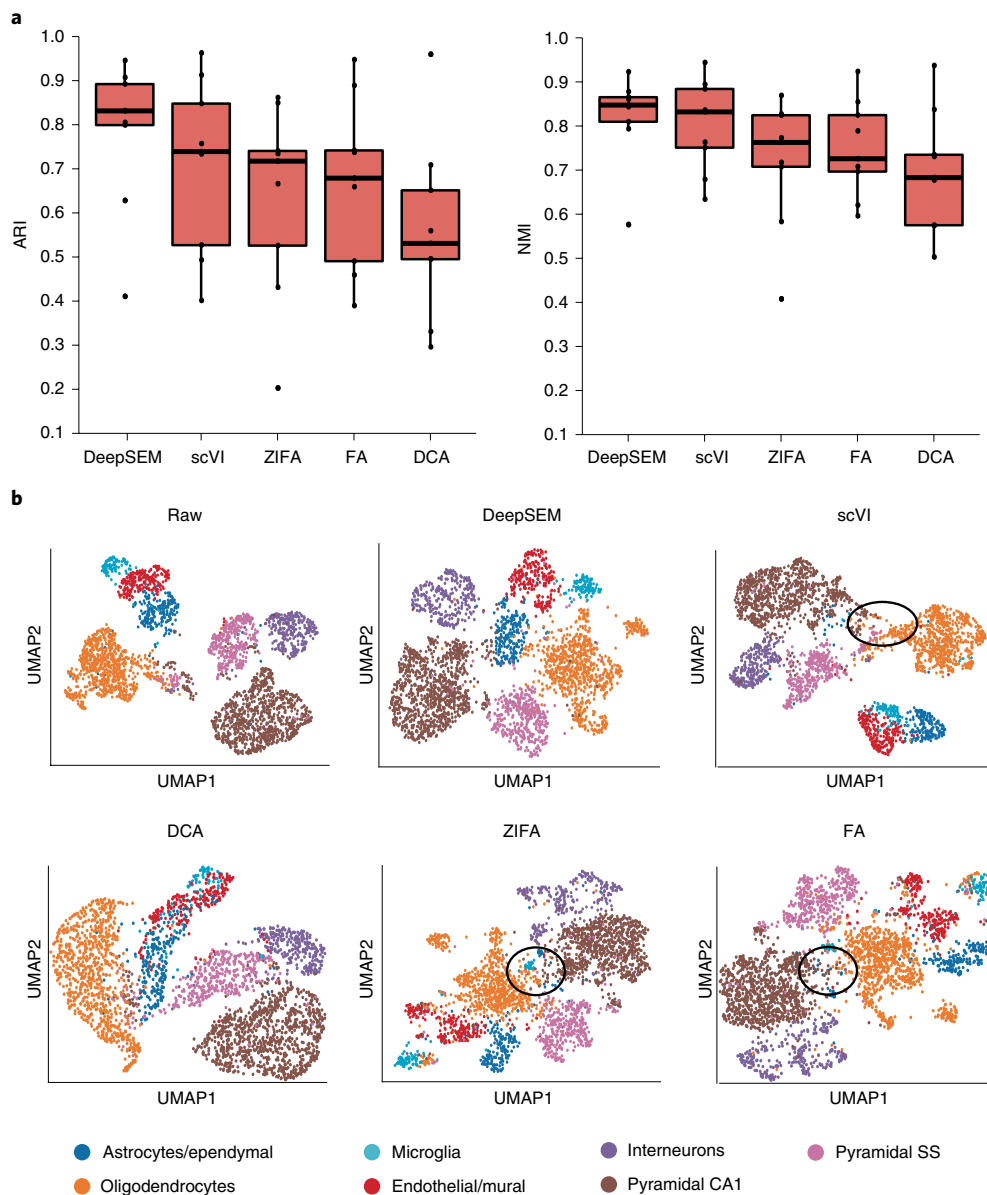
Fig. 5 | Single-cell clustering and embedding. a, The clustering performance of DeepSEM on nine single-cell datasets compared with four baseline methods. The box plots show the median (line), the interquartile range (box) and the whiskers (extending 1.5 times the interquartile range). **b**, UMAP visualization of hidden representations on the Zeisel dataset for different methods.

Louvain algorithm[46] to cluster all the single cells into the same number of clusters. We provide a brief introduction and running details of these baseline methods in Supplementary Section 5. To quantify the clustering accuracy based on the reference labels, we used the adjusted Rand index (ARI) and normalized mutual information (NMI), both of which range from 0 (random clusters) to 1 (identical clusters). In general, DeepSEM performs better than all the four baseline methods on five of the nine benchmark datasets including the Zeisel dataset[41], which are used in scVI[6], and also achieved comparable performance on the other datasets (within a range of 5% on average) (Fig. 5a and Supplementary Fig. 6). We also showed that DeepSEM can maintain the hierarchical structure among different cell types by evaluating the clustering performance on the Zeisel dataset[41] with the ground-truth hierarchical structure provided by the origin study[41] (Supplementary Fig. 6a). In particular, DeepSEM outperforms another probabilistic scRNA-seq model scVI[6], which is also based on the VAE, suggesting the necessity of explicitly modeling the GRN structure.

Next, we evaluated to what extent the latent space generated by DeepSEM could reflect the biological variability among different cells on the Zeisel dataset[41]. Visualizing using the uniform manifold approximation and projection (UMAP)[47], we found that DeepSEM was able to provide a more biologically meaningful representation. As shown in Fig. 5b, DeepSEM organizes oligodendrocytes and pyramidal CA1 cells into two clear clusters, while methods including scVI, ZIFA and FA failed to do so (highlight in the black circles). Quantitative comparison further highlights the advantage of DeepSEM. Membership weights (Supplementary Table 6) show that DeepSEM has an average closer distance between cell and cell-type center. We further compare DeepSEM with the baseline methods on R, earth-mover's distance (EMD) and K-nearest neighbor (KNN) preservation metrics[48], which all measure the distribution difference between embedding data and origin data[48]. The results show that DeepSEM can maintain better KNN consistency compared with scVI (Supplementary Table 7). Besides preserving good clustering, it had been expected that single cells typically follow a
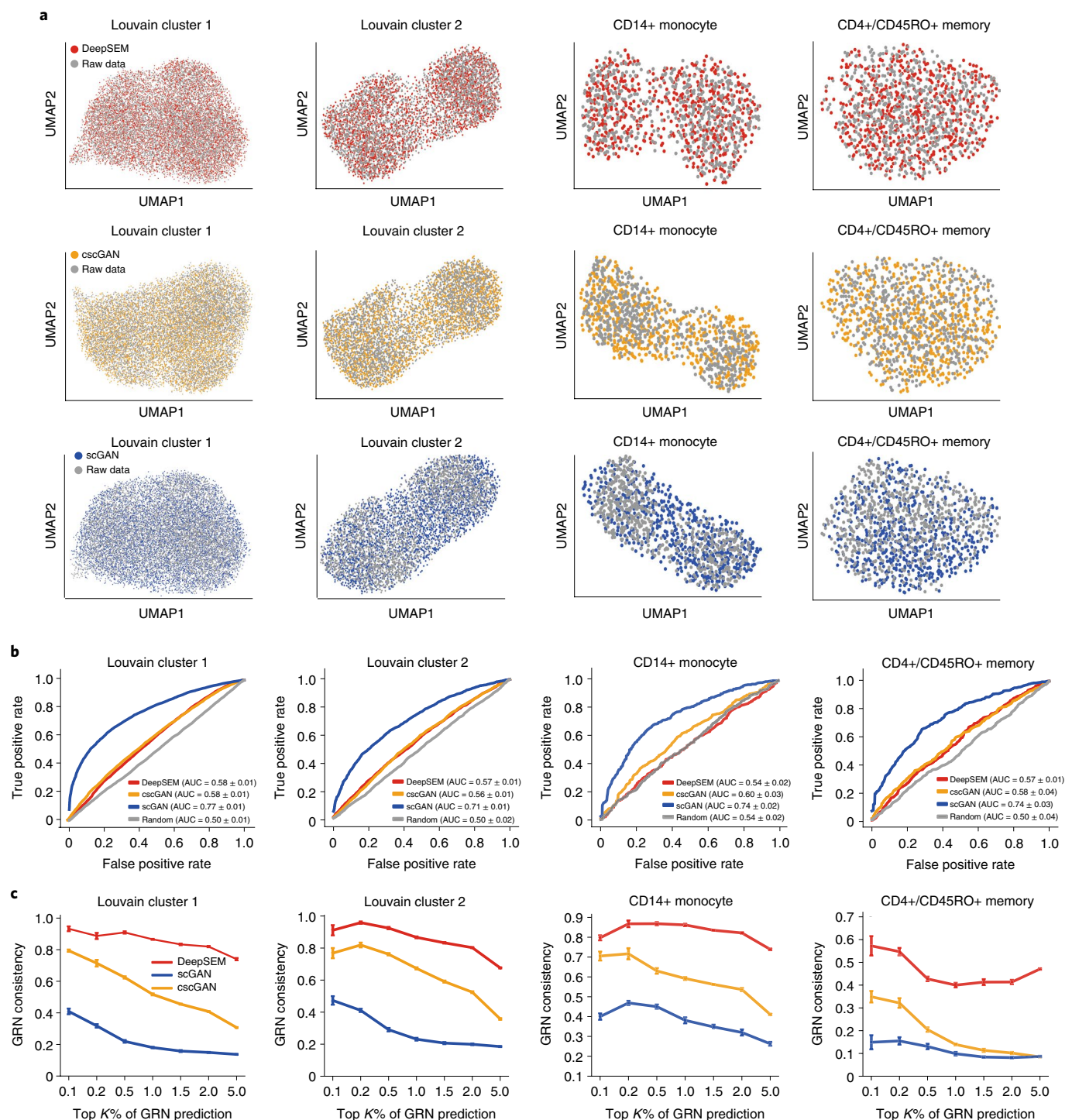
Fig. 6 | Simulation performance of DeepSEM compared with cscGAN and scGAN. a, The embedding plot layout was determined by UMAP visualization. **b**, Receiver operator characteristic (ROC) curves and area under the ROC Curve (AUC) scores (means ± s.d., lower is better) using RF to classify real and simulation cells. **c**, GRN consistency (means ± 95% CI, higher is better) between the simulated and real data.

temporal progression process, and are more apt to be represented with a continuous differentiation trajectory. We then embedded only the oligodendrocyte cells in a separate low-dimensional space and checked the consistency between the embeddings of cell subtypes and the differentiation trajectory defined in Zeisel et al.[41] (Supplementary Fig. 7). DeepSEM, ZIFA and FA were able to produce tight cell clusters for subtypes Oligo 3. This finding indicated that Oligo 3 presents a distinct cellular state compared with other

cell types, which was consistent with the finding in Zeisel et al.[41]. We also found that the embedding produced by DeepSEM can represent the same differentiation trajectory as defined in Zeisel et al.[41], whereas methods including scVI and DCA failed to capture this information (Supplementary Fig. 7).

scRNA-seq simulation. In this study, we compared the simulation performance of DeepSEM with two other GAN-based approaches,

cscGAN and scGAN[27], on the PBMC dataset[43]. A brief introduction of cscGAN and scGAN including their running details could be found in Supplementary Section 6. The original study[27] classified cells through the Louvain algorithm[46], but we found that the Louvain clusters were inconsistent with cell types identified by Zheng et al.[43]. For instance, 47.0% of the CD56+ natural killer (NK) cells were misclustered to a CD8+ cytotoxic T cluster (Supplementary Fig. 8). Therefore, we used two different annotation strategies as introduced in Methods. We found that all three methods could generate almost indistinguishable low-dimensional embeddings as in the original data, when clustered using both computational approaches and cell types labeled by the experts (Fig. 6a and Supplementary Figs. 9 and 10). We trained a random forest (RF) classifier to distinguish the simulated data from the real data for each Louvain cluster and each cell type that contains more than 2,500 cells. Our hypothesis is that the prediction performance of the RF model should be close to random if the simulated data generation was 'realistic'. We observed that the classification performance of RF dropped with an average of 1.21% and 19.77% to distinguish simulated data generated by DeepSEM from the real test data compared with cscGAN and scGAN, respectively (Fig. 6b and Supplementary Figs. 11 and 12). Note that the objective functions of cscGAN and scGAN are to train a generator function that can fool another discriminator function, so it was expected that they perform well on this task. On the other side, DeepSEM achieved realistic simulation, suggesting that integrating GRN may serve as a crucial step for modeling scRNA-seq data.

We also proposed another concept, called GRN consistency, to measure the quality of the simulated single cells. GRN consistency measures the difference of the predicted GRN between the real and simulated scRNA-seq data, which quantifies how much the single-cell model captures both the marginal and conditional independence in the original distribution. That is, if a scRNA-seq simulation is realistic, the GRN predictions obtained using the simulated dataset should match the predictions from the real dataset. Different from recent work[49] that used marginal independence to measure the consistency, our GRN consistency considers the conditional independence between TF and target genes, which is harder to maintain for simulators. We found that DeepSEM was able to achieve much higher GRN consistency compared with cscGAN and scGAN (Fig. 6c and Supplementary Figs. 13 and 14). For instance, for the top 0.1% predicted GRN edges, there is only 61.68% and 32.04% (on average) predicted GRN edge overlap between the real data and the simulated one by cscGAN and scGAN, respectively. On the other hand, DeepSEM can improve the GRN consistency by 33.07% and 156.14% compared with cscGAN and scGAN, respectively (Fig. 6c and Supplementary Figs. 13 and 14). This result demonstrates that DeepSEM is able to generate more realistic scRNA-seq data satisfying biological constraints.

We further investigated how the simulated single-cell data could be used to improve the quality of the downstream cell-type classification task, when only a limited number of single cells were provided. To investigate this, we downsampled single-cell data from two clusters (Louvain cluster 2 and CD56+ NK), and trained an RF model to distinguish these cells from others. We observed that for the approaches to be compared, the performance could be improved by simply upsampling the downsampled clusters, which was equivalent to putting higher weights on some samples to address the class imbalance problem for the RF model (Supplementary Fig. 15). The performances of DeepSEM and scGAN are about the same and only slightly lower than the performance of cscGAN by 1.02% (Supplementary Fig. 15).

Discussion

In this Article, we introduce a general computational framework that can jointly model the GRN and single-cell transcriptomic data.

The structure of the GRN is explicitly modeled as special layers of the neural network, which act as biological constraints to restrict the parameter space. One of the limitations of our study is that the running time increases with the number of genes involved due to the 'inverse' operation in the inverse GRN layer. Empirically, DeepSEM is relatively slower than other VAE models such as scVI[6]. To address the potential limitation during the training process, we recommend users to select highly variable genes instead of using the whole transcriptome as input features.

Models like DeepSEM may have other potential applications in single-cell biology. For instance, since all cells share the identical genome, GRN can be shared among different modalities, such as transcriptomic and epigenomic data. Therefore, DeepSEM can be adopted to integrate different single-cell modalities by leveraging GRN as a 'bridge' to construct a common latent space. A second potential application is to use the DeepSEM framework to integrate other molecular interaction networks, such as a protein–protein interaction network, open chromatin data, DNA binding motifs and a genetic interaction network to further infer a GRN and achieve higher accuracy.

Methods

The DeepSEM framework. Structural equation modeling is a multivariate statistical model to analyze structural relationships among different random variables. The basic SEM was first developed to model the covariance matrix for random variables[50]. Later, the SEM was found to be very powerful in modeling the relationship between observed features and hidden latent variables and was widely used in econometrics and sociology for causal inference[51,52]. More importantly, the SEM can be adopted to detect the conditional dependency among random variables and therefore also used to predict the graph structure of Bayesian networks and Markov random fields[53–56]. DeepSEM generalizes the SEM, which models the conditional dependencies among random variables and is formulated as a self-regression problem,

$$\mathbf{X} = \mathbf{W}^T\mathbf{X} + \mathbf{Z}, \tag{1}$$

we can modify equation (1) to the following form

$$\mathbf{X} = \left(\mathbf{I} - \mathbf{W}^T\right)^{-1}\mathbf{Z}$$
$$\mathbf{Z} = \left(\mathbf{I} - \mathbf{W}^T\right)\mathbf{X}, \tag{2}$$

where $\mathbf{I} \in \mathbb{R}^{m \times m}$ denotes the identity matrix, $\mathbf{X} \in \mathbb{R}^{n \times m}$ denotes the gene expression matrix with n cells and m genes, $\mathbf{W} \in \mathbb{R}^{m \times m}$ represents the adjacency matrix of the GRN that captures the conditional dependencies among different genes, and $\mathbf{Z} \in \mathbb{R}^{n \times m}$ stands for a noise matrix following a Gaussian distribution. Here we modify equation (2) to a nonlinear version of the SEM, which was originally proposed by Yu et al.[21], as follows

$$\mathbf{X} = f_1((\mathbf{I} - \mathbf{W}^T)^{-1}\mathbf{Z}), \tag{3}$$

$$\mathbf{Z} = \left(\mathbf{I} - \mathbf{W}^T\right)f_2(\mathbf{X}), \tag{4}$$

where f_1 and f_2 stand for multilayer neural networks. In particular, equation (3) can be decomposed into the following formulas:

$$\mathbf{H_Z} = (\mathbf{I} - \mathbf{W})^{-1}\mathbf{Z},$$

where the $\mathbf{H_Z}$ can be further rewritten in column-wise vectors:

$$\mathbf{H_Z} = [\mathbf{h}_0; \mathbf{h}_1; \ldots; \mathbf{h}_m], \mathbf{h}_i \in \mathbb{R}^{n \times 1},$$
$$f_1(\mathbf{h}_i) = \tanh\left(\tanh\left(\tanh\left(\mathbf{h}_i\mathbf{W}_1^T\right)\mathbf{W}_2^T\right)\mathbf{W}_3^T\right),$$
$$\mathbf{X} = [f_1(\mathbf{h}_0); f_1(\mathbf{h}_1); \ldots; f_1(\mathbf{h}_m)], \tag{5}$$
$$\mathbf{W}_1 \in \mathbb{R}^{d \times 1}, \mathbf{W}_2 \in \mathbb{R}^{d \times d}, \mathbf{W}_3 \in \mathbb{R}^{1 \times d},$$

where d denotes the number of hidden neurons for each layer and $\mathbf{h}_i$ denotes to the hidden latent for gene i. Equation (5) can be viewed as a nonlinear decoder function from random variable $\mathbf{Z}$. $\mathbf{W}_1, \mathbf{W}_2$ and $\mathbf{W}_3$ are linear weights of different layers of the neural network. Different from the conventional neural network

used in scRNA-seq modeling, f_1 takes only one feature as input. We then define a corresponding encoder function with a similar form, that is

$$\mathbf{X} = [\mathbf{x_0};\mathbf{x_1};\ldots;\mathbf{x_m}], \ \mathbf{x_i} \in \mathbb{R}^{n \times 1},$$

$$\mathbf{H_X} = [f_2(\mathbf{x_0}); f_2(\mathbf{x_1});\ldots;f_2(\mathbf{x_m})],$$

$$f_2(\mathbf{x_i}) = \tanh\left(\tanh\left(\tanh\left(\mathbf{x_i}\mathbf{W_4^T}\right)\mathbf{W_5^T}\right)\mathbf{W_6^T}\right), \quad (6)$$

$$\mathbf{Z} = \text{Reparameter}\left(\left(\mathbf{I} - \mathbf{W}^T\right)\mathbf{H_X}\right),$$

$$\mathbf{W_4} \in \mathbb{R}^{d \times 1}, \mathbf{W_5} \in \mathbb{R}^{d \times d}, \mathbf{W_6} \in \mathbb{R}^{2 \times d},$$

where $\mathbf{x_i}$ denotes the expression for gene i and f_2 stands for another multilayer neural network to model the noise in $\mathbf{X}$, which takes the expression of each gene as input feature and 'Reparameter' stands for the reparameterization trick[57]. $\mathbf{W_4}$, $\mathbf{W_5}$ and $\mathbf{W_6}$ are linear weights of different layers of the neural network. We name $\mathbf{I} - \mathbf{W}^T$ the GRN layer and $(\mathbf{I} - \mathbf{W}^T)^{-1}$ the inverse GRN layer. To integrate both equation (5) and equation (6), we use a beta-VAE model[23] with an additional L_1 norm to regularize the adjacent matrix $\mathbf{W}$. More specifically, the loss function of the DeepSEM is defined as follows

$$L = -E_{q(\mathbf{X})}\left[\log p(\mathbf{X}|\mathbf{Z})\right] + \beta KL(q(\mathbf{Z}|\mathbf{X})||p(\mathbf{Z})) + \alpha||\mathbf{W}||_1, \quad (7)$$

where E denotes the expected value function, p and q denote the distribution of $\mathbf{X}$ and $\mathbf{Z}$, KL denotes to KL-divergence function and α and β denote to hyperparameters. That is, instead of directly transforming $\mathbf{X}$ to $\mathbf{Z}$, the beta-VAE framework models $\mathbf{Z}$ as a Gaussian distribution in which the mean and variance are the outputs of neural networks taking $\mathbf{X}$ as the input features.

Similar to other autoencoder models, beta-VAE also recovers the input $\mathbf{X}$ in a probabilistic model. For cell-type-specific tasks, such as GRN prediction, single-cell embedding and simulation, we only considered the non-zero elements of $\mathbf{X}$ because it is typically hard to denoise (impute) all the non-zero values using only a limited number of samples for each cell type. For the prior distribution $p(\mathbf{Z})$, we chose Gaussian distribution $N(\mu, \sigma^2)$ in which the mean (μ) and variance (σ^2) were also estimated by minimizing equation (7). A Gaussian distribution-based VAE model has been widely used in other domains such as images, videos and natural languages. Empirically, we also tried the Gaussian mixture model from one to five Gaussian components as the prior distribution but found that the performance did not significantly outperform the original Gaussian distribution with a single component (Supplementary Fig. 16). Note that f_1 and f_2 are multilayer perceptrons (MLPs) with unique architectures. For each cell, we used f_2 to scan all the m genes and output m hidden variables. Similarly, f_1 scans m hidden variables and outputs m recovered gene expression values. In this way, we made sure that all the genes can only interact at the GRN layer and Inverse GRN layer.

Optimization of DeepSEM. We adopted the gradient descent and the reparameterization trick[57] to train our DeepSEM model. Different from conventional beta-VAE, there is an 'inverse' operation in equation (3), which makes the optimization problem harder because the inverse operation is generally sensitive to noise. We found that directly using the Adam algorithm[58] sometimes yielded unstable results. Inspired by the coordinate descent in graphical lasso[59] in which the matrix inverse is required, we optimized the weight matrix $\mathbf{W}$ in the GRN layer and the weights in the normal linear layers in an alternative way. More specifically, after using the RMSprop algorithm[60] to optimize the weights of MLPs for one epoch, we then optimized $\mathbf{W}$ using the same optimization algorithm for another two epochs with different learning rates (details in Supplementary Table 8).

GRN inference. The core element of DeepSEM is its ability to infer the GRN structure through a probabilistic modeling of scRNA-seq data. In this study, similar to the causal inference in SEM, we take the adjacent matrix $\mathbf{W}$ defined in equation (3) to indicate the GRN learned by DeepSEM. The absolute value of each element of $\mathbf{W}$ is used to rank the possibility of the regulatory relationships between genes. To obtain stable prediction, we run the training process with ten different random initializations of the models. The final GRN prediction is the average of the absolute adjacent matrices $\mathbf{W}$ derived over the ten different models.

Simulation of scRNA-seq data. Another important component of DeepSEM is that it can simulate scRNA-seq data by perturbing the values of hidden neurons. DeepSEM simulates 'realistic' scRNA-seq in a unique manner by guiding the information flow through the GRN layer, which mirrors the in vivo generation process of messenger RNA dynamics governed by multiple TFs. In particular, DeepSEM first perturbs the hidden vector $\mathbf{Z}$ as defined in equation (3). Note that $\mathbf{Z}$ follows a Gaussian distribution in which mean and variance are calculated from two separate neural networks. Let $\mathbf{n}$ denote the white noise variable following $N(0, \mathbf{I})$, where $\mathbf{I}$ stands for the identity matrix. Then a perturbation $\hat{\mathbf{Z}}$ is defined as $\mu + \mathbf{n}\sigma$, where μ and σ stand for the mean and standard values of the posterior probability of $\mathbf{Z}$. The simulated gene expression can be generated by decoding $\hat{\mathbf{Z}}$ by multiplying with the GRN layer. In this study, we only focus on simulating the expression of those non-zero input expression values.

Implementation of DeepSEM. In DeepSEM, the log-transformed scRNA-seq expression data after Z-normalizing is fed into the neural network. We initialized MLPs by using the 'kaiming_uniform'[61] and initialized $\mathbf{W}$ by setting the matrix diagonal as zeros and the others following a Gaussian distribution $N(1/(m-1), \varepsilon^2)$, in which m stands for number of genes and ε denotes a small value to avoid being trapped in the local optimal. The values on the diagonal are fixed as zeroes in the whole training process to guarantee that $\mathbf{W}$ is able to learn the regulatory network between genes. We determined the key hyperparameters (α, β and number of epochs) using a grid search strategy and used common default values for others. We summarize all the hyperparameters used in this study in Supplementary Table 8 and discuss the influence of the key hyperparameters in Supplementary Section 7 and Supplementary Fig. 17.

Datasets and data processing. *Datasets used to evaluate GRN inference.* We evaluated the performance of GRN inference on seven datasets (Supplementary Table 2) as in the BEELINE framework[28], where ground-truth GRNs are all available from: (1) mouse embryonic stem cells (mESC)[62], (2) mouse dendritic cells (mDC)[63], (3) three lineages of mouse hematopoietic stem cells[64], including erythroid lineage (mHSC-E), granulocyte-macrophage lineage (mHSC-GM) and lymphoid lineage (mHSC-L), (4) human mature hepatocytes (hHep)[65] and (5) human embryonic stem cells (hESC)[66]. To preprocess the raw gene expression data, we adopted the same strategy as in the BEELINE framework[28] (Supplementary Section 1).

For each dataset, there are three kinds of ground-truth GRN according to their information sources: cell-type-specific ChIP-seq[67–70], non-specific ChIP-seq[71–73] and functional interaction networks collected from the STRING database[74]. For the mouse embryonic stem cells (mESC), the loss-/gain-of-function data (LOF/GOF)[70] were also collected as a ground-truth GRN. Following Pratapa et al.[28], we excluded all edges that were not outgoing from TFs in the prediction during the evaluation. In addition, we also included another mouse cortex dataset[33] for which the GRN at the single-cell level was not measured. We focused on six major neuronal cell types from the primary visual cortex, including layer 2/3 (L2/3), L4, L5 and L6 intra-telencephalic (IT) neurons, L5 pyramidal tract (PT) neurons and L6 cortical-thalamic (CT) neurons. Following the same data-processing procedures as in the original data paper, we excluded those cells that annotated as low quality and those genes that expressed in less than 10% of cells. Then we normalized the count library sizes using count per million mapped reads (CPM). For each cell type, we selected 500 cell-type-specific genes using the 'rank_genes_group' function provided by the scanpy package[75] and selected TF genes from the JASPAR 2018 CORE vertebrates non-redundant motif database[76]. The motifs were scanned again with the mm10 genome using FIMO[77] with P value thresholds of 1×10^{-5}; after that, 487 TFs were under consideration in the following analysis. After removing duplicated genes, 2,762 genes including 487 TFs were considered in this study. We included the downloading addresses of the GRN data and the snmC-seq and scATAC-seq data for the mouse cortex dataset in Supplementary Table 1.

Datasets used to evaluate cell embeddings. We evaluated the single-cell embedding on the Zeisel dataset[41], which was also used to assess the performance of scVI[6]. We also collected eight other datasets including the mouse embryo stem dataset[42], human pancreas dataset[78] and the PBMC dataset[43], all with known cell types annotated. For each dataset, we discarded all cells without annotations and whose cell type contained less than 10 cells. We also discarded those genes that expressed in less than 1% of cells. For each dataset, only the top 1,000 highly variable genes were considered in evaluation. We present the summarized statistics for each dataset in Supplementary Table 5.

Datasets used to evaluate the performance of simulation. We evaluated the performance of scRNA-seq simulation following the experimental setting in Marouf et al.[27]. In particular, we trained and evaluated our model on a published human PBMC (healthy donor A) dataset[43]. Then we used two different strategies to annotate the cell types: (1) annotations from the original authors[43], including 11 different cell types (Supplementary Table 9) and (2) annotations derived from the Louvain algorithm with 'resolution' parameter 0.15 (Supplementary Table 10) following the same procedure as in Marouf et al.[27]. After discarding those genes that expressed in less than 1% of cells and cells that expressed less than 10 genes, we normalized the read counts to 20,000. We selected the top 1,000 variable genes based on the log-transformed normalized data and further normalized the gene expression using the 'normalize_total' function provided in the scanpy package[75], with the input formats of cscGAN and scGAN[27].

Latent representation visualization and clustering. For both DeepSEM and all other approaches to be evaluated, we first extracted the top 50 principal components (PCs) using principal component analysis if the size of hidden embeddings was larger than 50 and calculated the cell neighborhood graph by setting the 'n_neighbors' parameter as 30. Then we visualized the results of these datasets in two dimensions using the UMAP algorithm with default parameters. We used the Louvain algorithm[46] to cluster cells and selected parameter 'resolution' by binary search to generate the same number of clusters with cell-type annotation.

Visualization of simulated and real data. In this study, for each Louvain cluster and cell type containing more than 2,500 cells, we randomly selected 80% of cells as the training data and the remaining 20% of cells as the test data. DeepSEM and other baseline methods were trained on the training data, and for each method we simulated the same amount of data as the number of cells in the test data and used UMAP with default hyperparameters to visualize simulated and real test data. Following Marouf et al.[27], we further trained an RF model with 1,000 trees to classify the real test data and the simulated data on the first 50 PCs. Fivefold cross-validation was performed to evaluate the prediction performance in terms of the averaged area under the receiver operator characteristic (ROC) curve. We also conducted a positive control by discriminating training data from test data. We also consider the membership weight of the clustering to quantitatively compare the visualization performance. Membership weight is calculated by the distribution of the inverse Euclidean distance between each cell and the centers of all the clusters

$$\text{Membership weights} = \text{softmax}\left(\frac{1}{\text{dist}\left(x_{ij}, \text{center}(x_{i,:})\right)}\right), \quad (8)$$

where x_{ij} denotes the UMAP embedding of cell j in cell type i.

The GRN consistency between the simulated and real data. In this study, we investigated whether the generated scRNA-seq data have the same GRN with scRNA-seq from real cells. For each cluster or cell type that contains more than 2,500 cells, we used DeepSEM, scGAN and cscGAN to generate n gene expression profiles, where n was equal to the number of cells. We used GRNBoost2[18] to infer the GRNs for both real and simulated single-cell data. Similar to the GRN inference benchmark BEELINE[28], only GRNs outgoing from TFs were considered. For each cluster, we selected the top $K = \{0.1\%, 0.2\%, 0.5\%, 1.0\%, 1.5\%, 2.0\%, 2.5\%\}$ GRNs and used them to evaluate the consistency between the real and simulated cells. The GRN consistency is calculated as follows:

$$\text{GRN consistency} =$$
$$\frac{\text{Number of overlap edges in top } N \text{ predicted edges between real and simulated cells}}{N} \quad (9)$$

where $N = K\% \times$ number of predicted GRN in real cells. We reported the average GRN consistency within a ±95% confidence interval (CI).

Effects of data downsampling and augmentation on cell-type classification. In this study, we evaluated the data augmentation performance by discriminating cells annotated by the selected cluster or cell type from the other cells with limited training data. More specifically, we selected cells annotated as cluster 2 obtained from the Louvain algorithm[46], also as in Marouf et al.[27], and CD56+ NK cells annotated by Zheng et al.[43]. First, we randomly sampled 80% of the data as the training dataset and used the remaining 20% of the data as the test dataset. Then, the cells were randomly downsampled with eight different percentages {50%, 25%, 10%, 5%, 3%, 2%, 1%, 0.5%} on the training dataset. For each downsampling rate, we simulated 2,000 cells from the selected Louvain cluster or cell type by DeepSEM, cscGAN and scGAN. We also randomly sampled 2,000 cells with replacement from the selected Louvain cluster or cell type and annotated the above procedure as upsampling. An RF model with default hyperparameters was trained and the top 50 PCs of each cell were selected as features to discriminate cells from the others. We trained the RF model five times and reported the mean AUPR ± 95% CI.

Data availability

We provide all datasets generated or analyzed during this study. The gene experimental scRNA-seq datasets were downloaded from Gene Expression Omnibus with the accession numbers GSE81252 (hHEP dataset[65]), GSE75748 (hESC dataset[66]), GSE98664 (mESC dataset[62]), GSE48968 (mDC dataset[63]), GSE81682 (mHSC dataset[64]), GSE115746 (mouse cortex dataset[33]), GSE60361 (Zeisel dataset[41]), GSE85241 (Muraro dataset[78]), GSE81861 (Li dataset[79]), and GSE45719 (Deng dataset[80]). The other experimental scRNA-seq dataset were downloaded from ArrayExpress with the accession number E-MTAB-5061 (Segerstolpe dataset[81]), NCBI Sequence Read Archive (SRA) with accession number SRP041736 (Pollen dataset[42]), GitHub repositories (https://github.com/LuyiTian/sc_mixology) (CellBench dataset[82]) and the website for x10genomics (https://support.10xgenomics.com/single-cell-gene-expression/datasets/) (PBMC dataset[43]). The scATAC-seq and snmC-seq for mouse cortex were downloaded from Gene Expression Omnibus with the accession numbers GSE126724 (scATAC-seq[35]) and GSE97179 (snmC-seq[34]). More information for these datasets could be found in Methods. We also summarize the accession and download links in Supplementary Tables 1, 2, 5 and 9. Source Data for Figs. 3–6 are available with this manuscript.

Code availability

The codes generated during this study are available on GitHub (https://github.com/HantaoShu/DeepSEM) and in Zenodo[83].

References

1. Picelli, S. et al. Smart-seq2 for sensitive full-length transcriptome profiling in single cells. *Nat. Methods* **10**, 1096–1098 (2013).
2. Macosko, E. Z. et al. Highly parallel genome-wide expression profiling of individual cells using nanoliter droplets. *Cell* **161**, 1202–1214 (2015).
3. Hashimshony, T. et al. CEL-Seq2: sensitive highly-multiplexed single-cell RNA-seq. *Genome Biol.* **17**, 77 (2016).
4. Wagner, A., Regev, A. & Yosef, N. Revealing the vectors of cellular identity with single-cell genomics. *Nat. Biotechnol.* **34**, 1145–1160 (2016).
5. Kharchenko, P. V., Silberstein, L. & Scadden, D. T. Bayesian approach to single-cell differential expression analysis. *Nat. Methods* **11**, 740–742 (2014).
6. Lopez, R., Regier, J., Cole, M. B., Jordan, M. I. & Yosef, N. Deep generative modeling for single-cell transcriptomics. *Nat. Methods* **15**, 1053–1058 (2018).
7. Eraslan, G., Simon, L. M., Mircea, M., Mueller, N. S. & Theis, F. J. Single-cell RNA-seq denoising using a deep count autoencoder. *Nat. Commun.* **10**, 390 (2019).
8. Cuomo, A. S. E. et al. Single-cell RNA-sequencing of differentiating iPS cells reveals dynamic genetic effects on gene expression. *Nat. Commun.* **11**, 810 (2020).
9. Olsson, A. et al. Single-cell analysis of mixed-lineage states leading to a binary cell fate choice. *Nature* **537**, 698–702 (2016).
10. Sharma, A. et al. Onco-fetal reprogramming of endothelial cells drives immunosuppressive macrophages in hepatocellular carcinoma. *Cell* **183**, 377–394.e21 (2020).
11. Arisdakessian, C., Poirion, O., Yunits, B., Zhu, X. & Garmire, L. X. DeepImpute: an accurate, fast, and scalable deep neural network method to impute single-cell RNA-seq data. *Genome Biol.* **20**, 211 (2019).
12. Wang, T. et al. BERMUDA: a novel deep transfer learning method for single-cell RNA sequencing batch correction reveals hidden high-resolution cellular subtypes. *Genome Biol.* **20**, 165 (2019).
13. Li, X. et al. Deep learning enables accurate clustering with batch effect removal in single-cell RNA-seq analysis. *Nat. Commun.* **11**, 2338 (2020).
14. Huynh-Thu, V. A., Irrthum, A., Wehenkel, L. & Geurts, P. Inferring regulatory networks from expression data using tree-based methods. *PLoS ONE* **5**, e12776 (2010).
15. Chan, T. E., Stumpf, M. P. H. & Babtie, A. C. Gene regulatory network inference from single-cell data using multivariate information measures. *Cell Syst.* **5**, 251–267.e3 (2017).
16. Matsumoto, H. et al. SCODE: an efficient regulatory network inference algorithm from single-cell RNA-seq during differentiation. *Bioinformatics* **33**, 2314–2321 (2017).
17. Papili Gao, N., Ud-Dean, S. M. M., Gandrillon, O. & Gunawan, R. SINCERITIES: inferring gene regulatory networks from time-stamped single cell transcriptional expression profiles. *Bioinformatics* **34**, 258–266 (2018).
18. Moerman, T. et al. GRNBoost2 and Arboreto: efficient and scalable inference of gene regulatory networks. *Bioinformatics* **35**, 2159–2161 (2019).
19. Kamimoto, K., Hoffmann, C. M. & Morris, S. A. CellOracle: dissecting cell identity via network inference and in silico gene perturbation. Preprint at *bioRxiv* https://doi.org/10.1101/2020.02.17.947416 (2020).
20. Kim, S. ppcor: an R package for a fast calculation to semi-partial correlation coefficients. *Commun. Stat. Appl. Methods* **22**, 665–674 (2015).
21. Yu, Y., Jie, C., Tian, G. & Mo, Y. DAG-GNN: DAG structure learning with graph neural networks. In *Proceedings of the 36th International Conference on Machine Learning* 7154–7163 (ICML, 2019).
22. Lin, C., Jain, S., Kim, H. & Bar-Joseph, Z. Using neural networks for reducing the dimensions of single-cell RNA-seq data. *Nucleic Acids Res.* **45**, e156 (2017).
23. Higgins, I. et al. beta-VAE: learning basic visual concepts with a constrained variational framework. In *Proceedings of the 5th International Conference on Learning Representations* (ICLR, 2017).
24. Zhao, A., Balakrishnan, G., Durand, F., Guttag, J. V. & Dalca, A. V. Data augmentation using learned transformations for one-shot medical image segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* 8543–8553 (IEEE, 2019).
25. Lotfollahi, M., Wolf, F. A. & Theis, F. J. scGen predicts single-cell perturbation responses. *Nat. Methods* **16**, 715–721 (2019).
26. Wang, X., Ghasedi Dizaji, K. & Huang, H. Conditional generative adversarial network for gene expression inference. *Bioinformatics* **34**, i603–i611 (2018).
27. Marouf, M. et al. Realistic in silico generation and augmentation of single-cell RNA-seq data using generative adversarial networks. *Nat. Commun.* **11**, 166 (2020).
28. Pratapa, A., Jalihal, A. P., Law, J. N., Bharadwaj, A. & Murali, T. M. Benchmarking algorithms for gene regulatory network inference from single-cell transcriptomic data. *Nat. Methods* **17**, 147–154 (2020).
29. Moore, L. D., Le, T. & Fan, G. DNA methylation and its basic function. *Neuropsychopharmacology* (2013).
30. Thurman, R. E. et al. The accessible chromatin landscape of the human genome. *Nature* **489**, 75–82 (2012).
31. Keilwagen, J., Posch, S. & Grau, J. Accurate prediction of cell type-specific transcription factor binding. *Genome Biol.* **20**, 9 (2019).

32. Funk, C. C. et al. Atlas of transcription factor binding sites from ENCODE DNase hypersensitivity data across 27 tissue types. *Cell Rep.* **32**, 108029 (2020).

33. Tasic, B. et al. Shared and distinct transcriptomic cell types across neocortical areas. *Nature* **563**, 72–78 (2018).

34. Luo, C. et al. Single-cell methylomes identify neuronal subtypes and regulatory elements in mammalian cortex. *Science* **357**, 600–604 (2017).

35. Fang, R. et al. Comprehensive analysis of single cell ATAC-seq data with SnapATAC. *Nat. Commun.* **12**, 1337 (2021).

36. Dong, J. et al. Enhancing single-cell cellular state inference by incorporating molecular network features. Preprint at *bioRxiv* (2019).

37. Aibar, S. et al. SCENIC: single-cell regulatory network inference and clustering. *Nat. Methods* **14**, 1083–1086 (2017).

38. Li, X. et al. Network embedding-based representation learning for single cell RNA-seq data. *Nucleic Acids Res.* **45**, e166–e166 (2017).

39. Cahan, P. et al. CellNet: network biology applied to stem cell engineering. *Cell* **158**, 903–915 (2014).

40. Morris, S. A. et al. Dissecting engineered cell types and enhancing cell fate conversion via CellNet. *Cell* **158**, 889–902 (2014).

41. Zeisel, A. et al. Brain structure. Cell types in the mouse cortex and hippocampus revealed by single-cell RNA-seq. *Science* **347**, 1138–1142 (2015).

42. Pollen, A. A. et al. Low-coverage single-cell mRNA sequencing reveals cellular heterogeneity and activated signaling pathways in developing cerebral cortex. *Nat. Biotechnol.* **32**, 1053–1058 (2014).

43. Zheng, G. X. Y. et al. Massively parallel digital transcriptional profiling of single cells. *Nat. Commun.* **8**, 14049 (2017).

44. Pierson, E. & Yau, C. ZIFA: dimensionality reduction for zero-inflated single-cell gene expression analysis. *Genome Biol.* **16**, 241 (2015).

45. Jolliffe, I. T. in *Principal Component Analysis* (ed. Jolliffe, I. T.) 115–128 (Springer, 1986).

46. Blondel, V. D., Guillaume, J.-L., Lambiotte, R. & Lefebvre, E. Fast unfolding of communities in large networks. *J. Stat. Mech.* **2008**, P10008 (2008).

47. Becht, E. et al. Dimensionality reduction for visualizing single-cell data using UMAP. *Nat. Biotechnol.* https://doi.org/10.1038/nbt.4314 (2019).

48. Heiser, C. N. & Lau, K. S. A quantitative framework for evaluating single-cell data structure preservation by dimensionality reduction techniques. *Cell Rep.* **31**, 107576 (2020).

49. Viñas, R., Andrés-Terré, H., Liò, P. & Bryson, K. Adversarial generation of gene expression data. *Bioinformatics* https://doi.org/10.1093/bioinformatics/btab035 (2021).

50. Bollen, K. A. *Structural Equations with Latent Variables* (John Wiley & Sons, 1989).

51. Haavelmo, T. The statistical implications of a system of simultaneous equations. *Econometrica* **11**, 1–12 (1943).

52. King, M., Goldberger, A. S. & Duncan, O. D. Structural equation models in the social sciences. *Econ. J.* **84**, 212–214 (1974).

53. Duarte, C. W., Klimentidis, Y. C., Harris, J. J., Cardel, M. & Fernández, J. R. A hybrid Bayesian network/structural equation (BN/SEM) modeling approach for detecting physiological networks for obesity-related genetic variants. In *Proceedings of the IEEE International Conference on Bioinformatics and Biomedicine* 696–702 (IEEE, 2012).

54. Yoo, C. & Oh, S. Combining structure equation model with Bayesian networks for predicting with high accuracy of recommending surgery for better survival in Benign prostatic hyperplasia patients. In *20th International Congress on Modelling and Simulation-Adapting to Change* 2029–2033 (Modelling and Simulation Society of Australia and New Zealand, 2013).

55. Zheng, X., Aragam, B., Ravikumar, P. & Xing, E. P. DAGs with NO TEARS: continuous optimization for structure learning. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems* 9492–9503 (IEEE, 2018).

56. Luo, Y., Peng, J. & Ma, J. When causal inference meets deep learning. *Nat. Mach. Intell.* **2**, 426–427 (2020).

57. Kingma, D. P. & Welling, M. An introduction to variational autoencoders. *Found Trends Mach. Learn.* **12**, 307–392 (2019).

58. Kingma, D. P. & Ba, J. Adam: a method for stochastic optimization. In *Proceedings of the 3rd International Conference on Learning Representations* (ICLR, 2015).

59. Friedman, J., Hastie, T. & Tibshirani, R. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics* **9**, 432–441 (2008).

60. Tieleman, T. & Hinton, G. *Lecture 6.5-rmsprop, Coursera: Neural Networks for Machine Learning* Technical Report (Univ. Toronto, 2012).

61. He, K., Zhang, X., Ren, S. & Sun, J. Delving deep into rectifiers: surpassing human-level performance on ImageNet classification. In *Proceedings of the IEEE International Conference on Computer Vision* 1026–1034 (IEEE, 2015).

62. Hayashi, T. et al. Single-cell full-length total RNA sequencing uncovers dynamics of recursive splicing and enhancer RNAs. *Nat. Commun.* **9**, 619 (2018).

63. Shalek, A. K. et al. Single-cell RNA-seq reveals dynamic paracrine control of cellular variation. *Nature* **510**, 363–369 (2014).

64. Nestorowa, S. et al. A single-cell resolution map of mouse hematopoietic stem and progenitor cell differentiation. *Blood* **128**, e20–e31 (2016).

65. Camp, J. G. et al. Multilineage communication regulates human liver bud development from pluripotency. *Nature* **546**, 533–538 (2017).

66. Chu, L.-F. et al. Single-cell RNA-seq reveals novel regulators of human embryonic stem cell differentiation to definitive endoderm. *Genome Biol.* **17**, 173 (2016).

67. ENCODE Project Consortium An integrated encyclopedia of DNA elements in the human genome. *Nature* **489**, 57–74 (2012).

68. Davis, C. A. et al. The encyclopedia of DNA elements (ENCODE): data portal update. *Nucleic Acids Res.* **46**, D794–D801 (2018).

69. Oki, S. et al. ChIP-Atlas: a data-mining suite powered by full integration of public ChIP-seq data. *EMBO Rep.* **19**, e46255 (2018).

70. Xu, H. et al. ESCAPE: database for integrating high-content published data collected from human and mouse embryonic stem cells. *Database* **2013**, bat045 (2013).

71. Garcia-Alonso, L., Holland, C. H., Ibrahim, M. M., Turei, D. & Saez-Rodriguez, J. Benchmark and integration of resources for the estimation of human transcription factor activities. *Genome Res.* **29**, 1363–1375 (2019).

72. Liu, Z.-P., Wu, C., Miao, H. & Wu, H. RegNetwork: an integrated database of transcriptional and post-transcriptional regulatory networks in human and mouse. *Database* **2015**, bav095 (2015).

73. Han, H. et al. TRRUST: a reference database of human transcriptional regulatory interactions. *Sci. Rep.* **5**, 11432 (2015).

74. Szklarczyk, D. et al. STRING v11: protein–protein association networks with increased coverage, supporting functional discovery in genome-wide experimental datasets. *Nucleic Acids Res.* **47**, D607–D613 (2019).

75. Wolf, F. A., Angerer, P. & Theis, F. J. SCANPY: large-scale single-cell gene expression data analysis. *Genome Biol.* **19**, 15 (2018).

76. Fornes, O. et al. JASPAR 2020: update of the open-access database of transcription factor binding profiles. *Nucleic Acids Res.* **48**, D87–D92 (2020).

77. Grant, C. E., Bailey, T. L. & Noble, W. S. FIMO: scanning for occurrences of a given motif. *Bioinformatics* **27**, 1017–1018 (2011).

78. Muraro, M. J. et al. A Single-cell transcriptome atlas of the human pancreas. *Cell Syst.* **3**, 385–394.e3 (2016).

79. Li, H. et al. Reference component analysis of single-cell transcriptomes elucidates cellular heterogeneity in human colorectal tumors. *Nat. Genet.* **49**, 708–718 (2017).

80. Deng, Q., Ramsköld, D., Reinius, B. & Sandberg, R. Single-cell RNA-seq reveals dynamic, random monoallelic gene expression in mammalian cells. *Science* **343**, 193–196 (2014).

81. Segerstolpe, Å. et al. Single-cell transcriptome profiling of human pancreatic islets in health and type 2 diabetes. *Cell Metab.* **24**, 593–607 (2016).

82. Tian, L. et al. Benchmarking single cell RNA-sequencing analysis pipelines using mixture control experiments. *Nat. Methods* **16**, 479–487 (2019).

83. Shu, H. et al. Code for paper 'Modeling gene regulatory networks using neural network architectures'. *Zenodo* https://doi.org/10.5281/zenodo.4915754 (2021).

Acknowledgements

Author contributions

J. Zeng and J.M. designed the study and developed the conceptual ideas. H.S. and Q.L. implemented the main algorithms. H.S. performed the model training and experimental validation task. H.S. and J. Zhou collected all the input data sources and interpreted the results. H.S., J. Zhou, H.L., D.Z., J. Zeng and J.M. wrote the manuscript with support from all authors.

Competing interests

J. Zeng is founder and CTO of Silexon AI Technology Co. Ltd. and has an equity interest. The remaining authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s43588-021-00099-8.

Correspondence and requests for materials should be addressed to J.Z. or J.M.

Peer review information *Nature Computational Science* thanks Jun Ding, Yafei Lyu and the other, anonymous, reviewer(s) for their contribution to the peer review of this work. Handling editor: Fernando Chirigati, in collaboration with the Nature Computational Science team.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.