OXFORD

Briefings in Bioinformatics, 2023, **24(2)**, 1–15
https://doi.org/10.1093/bib/bbad040
Advance access publication date 2 February 2023
Problem Solving Protocol

scGGAN: single-cell RNA-seq imputation by graph-based generative adversarial network

Zimo Huang, Jun Wang, Xudong Lu, Azlan Mohd Zain and Guoxian Yu

Corresponding author: Jun Wang, Joint SDU-NTU Centre for Artificial Intelligence Research, Shandong University, Jinan 250101, China. Tel: 531-88391516. Fax: 531-88391686. E-mail: kingjun@sdu.edu.cn

Abstract

Single-cell RNA sequencing (scRNA-seq) data are typically with a large number of missing values, which often results in the loss of critical gene signaling information and seriously limit the downstream analysis. Deep learning-based imputation methods often can better handle scRNA-seq data than shallow ones, but most of them do not consider the inherent relations between genes, and the expression of a gene is often regulated by other genes. Therefore, it is essential to impute scRNA-seq data by considering the regional gene-to-gene relations. We propose a novel model (named scGGAN) to impute scRNA-seq data that learns the gene-to-gene relations by Graph Convolutional Networks (GCN) and global scRNA-seq data distribution by Generative Adversarial Networks (GAN). scGGAN first leverages single-cell and bulk genomics data to explore inherent relations between genes and builds a more compact gene relation network to jointly capture the homogeneous and heterogeneous information. Then, it constructs a GCN-based GAN model to integrate the scRNA-seq, gene sequencing data and gene relation network for generating scRNA-seq data, and trains the model through adversarial learning. Finally, it utilizes data generated by the trained GCN-based GAN model to impute scRNA-seq data. Experiments on simulated and real scRNA-seq datasets show that scGGAN can effectively identify dropout events, recover the biologically meaningful expressions, determine subcellular states and types, improve the differential expression analysis and temporal dynamics analysis. Ablation experiments confirm that both the gene relation network and gene sequence data help the imputation of scRNA-seq data.

Keywords: single-cell RNA-seq, data imputation, gene relation network, Graph Convolutional Networks, Generative Adversarial Networks

Introduction

RNA sequencing (RNA-seq) is the canonical transcriptomics sequencing technology to digitalize the expression levels of RNA molecules (i.e. mRNA, miRNA and ncRNA) [1]. RNA-seq is widely evolved in gene expression analysis, novel transcripts discovery and alternatively spliced genes identification, it provides new insights for understanding biological systems [2]. Traditional bulk RNA-seq counts the average gene expression levels of thousands cells; however, it is difficult to reflect cell heterogeneous characteristics and to assess the basic biological units (cells). More advanced single-cell RNA sequencing (scRNA-seq) uses individual cells as the sample unit and digitalizes RNA molecules per cell on a genome-wide high resolution [3], which enables to comprehensively analyze the homogeneity and heterogeneity of gene expression among cells [4, 5]. scRNA-seq has been widely used to analyze the genotype and phenotype heterogeneity of individual cells across tissues, uncover the internal mechanisms of complex diseases and many others [6, 7].

scRNA-seq data are usually stored by a sparse gene-by-cell matrix of transcript counts with a large number of zeros. These zeros are with two situations, one is the 'true' zero of actually unexpressed, the other is the 'false' zero caused by the sequencing techniques, such as the conditions of mRNA captured from a single cell, amplification bias, sequencing depth and so on [8]. The 'false' zero is termed as the '*dropout*' event [9], which causes the observed data failing to reflect the potential expression patterns. Dropout severely impacts downstream analysis and weakens the power of scRNA-seq in a wide range of biomedical applications (i.e. cell clustering, differential expression analysis and cell trajectory inference) [10–12]. Even the most popular single-cell protocol 10X displays more severe dropout problem, especially for genes with low expression levels [13]. Therefore, accurately identifying and imputing the dropout data is an urgent need for scRNA-seq data analysis.

Many methods have been proposed to impute scRNA-seq data. Several solutions apply the linear model (i.e. matrix factorization and factor analysis) [14–16], and some other approaches refer to gene expression of other similar cells for predicting dropout events [17–20], whereas many other methods first assume that the scRNA-seq dataset has a preconceived structure (i.e. negative binomial (NB), zero-inflated negative binomial and low-rank), and then develop computational models to learn the structural pattern and impute dropout [21–24]. However, these solutions still suffer certain limitations. Linear model-based approaches can not

Zimo Huang is an MEng student at School of Software, Shandong University, China. His current research interests include data mining and bioinformatics.
Jun Wang is a professor at the Joint SDU-NTU Centre for Artificial Intelligence Research (C-FAIR), Shandong University, China. Her current research interests include machine learning, data mining and their applications in bioinformatics.
Xudong Lu is an associate professor at the Joint SDU-NTU Centre for Artificial Intelligence Research (C-FAIR), Shandong University, China. His current research interests include intelligent data analysis, social computing, human–computer interaction.
Azlan Mohd Zain is a professor in Big Data Centre, University Teknologi Malaysia, Malaysia. His current research interests include machine learning and big data mining.
Guoxian Yu is a professor at the School of Software, Shandong University, China. His current research interests include machine learning and bioinformatics.
Received: November 8, 2022. **Revised:** December 21, 2022. **Accepted:** January 18, 2023

explore and exploit non-linear relationships in scRNA-seq data, cell–cell similarity-based solutions may lead to over-smoothing or remove natural cell-to-cell stochasticity, which has important significance in biological processes [25, 26]. The preconceived methods depend on the correctness of the prior distribution, but the distribution of real scRNA-seq data is unknown or mixed up with diverse distributions. In addition, some of them are biased toward cell types with a large number of cells, which result in significantly compromised imputations for rare cell types.

Generative Adversarial Networks (GAN) can learn the real data distribution without hypothetical distribution and reduce the impacts of data imbalance [27]. These advantages make GAN a better choice for scRNA-seq generation or imputation. For instance, Wang *et al.* [28] used improved conditional generative adversarial network GGAN that predicts the target genes expression via a fully connected (FC) neural network-based GAN model incorporating both adversarial loss and L1-borm loss terms. Marouf *et al.* [29] built single-cell GAN (scGAN) and conditional scGAN (cscGAN) for generating simulated scRNA-seq data. scGAN and cscGAN utilize a custom library size normalization function and FC network with batch normalization model as their fundamental framework, through adversarial learning they are able to capture gene count distributions and correlations. However, FC network can not model gene expression data well, because it can not capture the internal dependencies of genes in non-Euclidean spaces. This limits the performance of these FC-based GAN models. Xu *et al.* [30] proposed scIGANs that trains GAN to learn real data distribution and imputes scRNA-seq data by cells generated by K nearest neighbors. scIGANs uses generated cells rather than observed ones to avoid the limitations, such as many sources of technical noises and dropouts, and the powerless for rare cells. scIGANs builds on CNN model, and it reshapes the single-cell expression profiles into images and treats each gene as a pixel. The good performance of CNN depends on spatially close points with more complex interactions and translation invariant. Therefore, the relative pixels in the image have great impacts on the result. However, neighborhood genes in the reshaped image may not have any relations, which seriously interfere the learning of real scRNA-seq data distribution and predicting missing values.

The expression of a gene is not an independent process but regulated by its related genes. Therefore, we argue that a more powerful imputation technique should consider two key issues: regional gene-to-gene relations and global real scRNA-seq distribution. The gene relations (i.e. interaction and co-expression) can be modeled as a graph, in which the nodes represent the genes and the edges encode relations between them. Therefore, we try to predict the missing expression values of a gene by aggregating the features of its related genes. In fact, GCN [31] makes it possible to dynamically mine relationships between nodes in a graph and exploit this kind of structural information by working in the domain of graphs. Recently, an adversary-trained graph imputation neural network (GINN) [32] was proposed as a general framework for missing data imputation, it achieves better performance on common missing data imputation datasets. However, GINN cannot make efficient imputation on the datasets with high dropout (missing) rate, whereas scRNA-seq data always has high dropout rate. In addition, most existing methods canonically only use scRNA-seq data for imputation. With the development of sequencing technology, multi-omics data can be easily obtained, which are conducive to comprehensively mine the inherent biological knowledge of single-cell data [33, 34].

To address these issues, we propose a graph-based GAN solution named scGGAN (as illustrated in Figure 1) to accurately impute scRNA-seq data. scGGAN first leverages single-cell and bulk RNA-seq data to construct the gene relation network to capture bulk homogeneous and single-cell heterogeneous gene-to-gene information. Next, a GCN-based GAN is trained in an adversarial learning paradigm to learn regional gene-to-gene relations from different omics by its GCN model and mine global scRNA-seq data distribution by its GAN model. Finally, scGGAN imputes scRNA-seq data by trained generator. Benefited from the advantages of adversarial learning, scGGAN avoids overfitting to some cell types with a large number, while maintains the imputation power for rare cells. Compared with other scRNA-seq imputation methods, scGGAN integrates bulk homogeneous and single-cell heterogeneous gene-to-gene-related information in bulk RNA-seq, scRNA-seq and gene sequence data to make up for information loss caused by high dropout rate and guide efficient imputation. It is more suitable for single-cell data since it does not introduce irrelevant location information (which often appears in CNN-based GAN). scGGAN outperforms competitive methods on both simulated and real scRNA-seq datasets in different downstream tasks (i.e. cell clustering, differential expression analysis and cell trajectory inference). Ablation experiments demonstrate that both gene relation network and genomics sequence data help the imputation of scRNA-seq data.

Methods

As a widely used generative model, GAN can learn real data distribution well without prior hypothesis [27] and has great potential on recovering missing data [32, 35, 36]. Therefore, we propose scGGAN, a graph-based GAN model, to infer scRNA-seq dropout data by leveraging multi-omics data. scGGAN attempts to learn the regional gene-to-gene relations from multi-omics data by GCN model and the global scRNA-seq data distribution by GAN model. In this section, we first briefly introduce the data preprocessing, and then introduce how to construct a gene relation network by scRNA-seq and bulk RNA-seq data. Finally, we describe the graph-based GAN for scRNA-seq imputation and discuss the optimization process of scGGAN.

Data preprocessing

To eliminate the differences in sequencing depth and reduce the negative impacts from raw scRNA-seq data, we first perform data preprocessing (i.e. data filtering and normalization). scRNA-seq data are usually stored in the form of a read count matrix (or sparse matrix) with N rows for genes and M columns for cells. We filter out the genes expressed in fewer than 10 cells and the cells with fewer than 200 expressed genes. Then, we normalize the filtered scRNA-seq data into the range of $[0, 1]$ by dividing with the maximum count of each column (cell). This canonical preprocessing can reduce the impact of differences in the batch effect and the number of transcripts in each cell.

scGGAN also uses bulk RNA-seq data to explore homogeneous information between genes for building a more stable gene relation network. Similar to scRNA-seq data, bulk RNA-seq data are usually provided in form of a count matrix with N' rows (genes) and M' columns (cells). First, we only keep genes that appear in both scRNA-seq and bulk RNA-seq data. After the alignment, the input scRNA-seq data are $\mathbf{X}^s \in \mathbb{R}^{n \times m_1}$, and the bulk RNA-seq is $\mathbf{X}^b \in \mathbb{R}^{n \times m_2}$, where n is the number of genes, m_1 (m_2) is the number of filtered cells (or samples). Next, we also normalize bulk

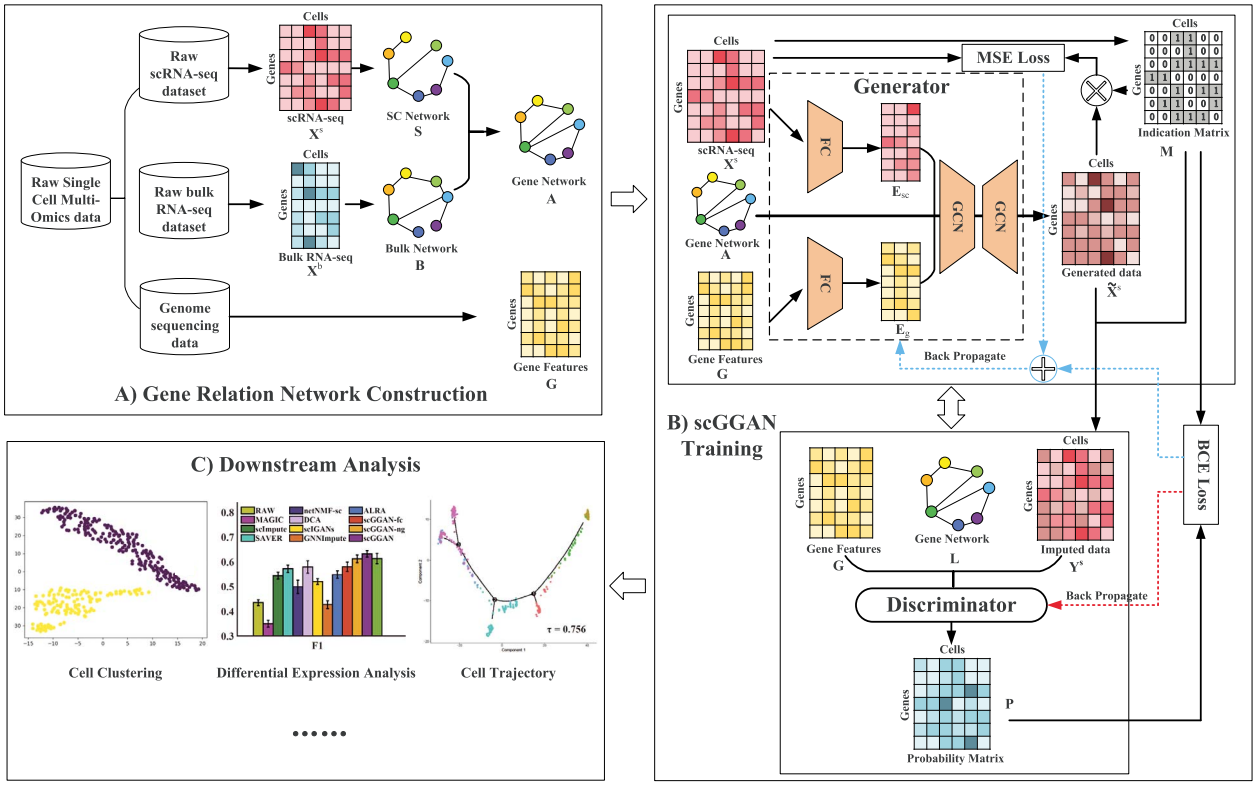

Figure 1. Overview of scGGAN for scRNA-seq data imputation and downstream analysis: (**A**) Gene Relation Network Construction module first constructs the single-cell gene relation network **S** by scRNA-seq $\mathbf{X}^s$ and bulk network **B** by bulk RNA-seq data $\mathbf{X}^b$, and then integrates them as composite gene network **A**. (**B**) scGGAN consists of generator and discriminator with similar structure. The generator uses FC neural networks for embedding scRNA-seq $\mathbf{X}^s$ and gene sequence features **G** to $\mathbf{E}_{sc}$ and $\mathbf{E}_g$, next it exploits a GCN encoder to fuse **A**, $\mathbf{E}_{sc}$ and $\mathbf{E}_g$, and then generates scRNA-seq data through a GCN decoder network. scGGAN imputes raw scRNA-seq data via generated $\widetilde{\mathbf{X}}^s$ and indication matrix **M**, while the discriminator distinguishes the values in the imputed data $\mathbf{Y}^s$ generated or real. The generator and discriminator are trained in an adversary way. (**C**) The quality of imputed data $\mathbf{Y}^s$ by scGGAN can be tested by the downstream analysis experiments including cell clustering, differential expression analysis, cell trajectory.

RNA-seq data by the maximum read count of each column (cell) to scale all gene expression values into [0, 1].

To learn richer genomics information and augment the attribute information of gene nodes in the relation network, scGGAN further integrates inherent genomics information, which is generally contained in gene sequence data of whole transcripts. For this purpose, we search the whole genome sequencing data and genome annotation data for n genes in $\mathbf{X}^s$ and then adopt the widely used K-mers strategy [37] to encode varying length gene sequences and obtain the fixed length representation vectors. The K-mers strategy counts the number of distinct K-mers within the gene sequence by sliding window with length K (K-mers), then represents the overall nucleic acids (or amino acids) sequence by extracting a series of sub-sequences. Finally, we represent gene sequences as a 4^K-dimensional (full permutation of K nucleic acids) data matrix $\mathbf{G} \in \mathbb{R}^{n*(4^K)}$, $g_{ij} \in [0, 1]$, where g_{ij} represents the frequency of j-th K-mer in i-th gene sequence. Here, we just use the typical K-mers instead of more complex DNA or protein language models (i.e. DNABERT [38] and Protein Transformer [39]) to show the usefulness of sequence data for imputation.

Gene relation network construction

The expression of a gene is not an independent process but impacted by other related genes, which can be modeled as a relation graph. An intuitive way to construct this relation network is to calculate the correlation coefficient between different genes from their expression data. The Pearson correlation coefficient (PCC) is the most common correlation calculation, which is used to measure the linear correlation between variables. In the field of biological information analysis, PCC is often used to measure the correlation of gene expression [40]. However, there are a lot of uncertain 'zero' in scRNA-seq data, so the network constructed directly from raw scRNA-seq data will be unreliable. To solve this problem, we construct the gene relation network by fusing bulk RNA-seq and scRNA-seq data, which can integrate homogeneous and heterogeneous information of genes.

The relation network constructed by bulk RNA-seq can catch the gene homogeneity information in a variety of cells, and it can supplement the general genetic information for single-cell networks. Considering that the bulk RNA-seq data is hardly affected by dropout events, its expression values usually are reliable, so we directly use the bulk expression values to calculate the PCC of all gene pairs as:

$$B_{ij} = PCC\left(\mathbf{x}_i^b, \mathbf{x}_j^b\right) \tag{1}$$

where $\mathbf{x}_{i/j}^b$ is the expression vector of i/j-th gene in bulk RNA-seq data $\mathbf{X}^b$.

Unlike bulk RNA-seq data, scRNA-seq data has lots of dropout values, so we calculate the PCC only using the non-zero elements of both vectors (reliable observed expression values) as:

$$S_{ij} = PCC(\mathbf{x}_i^s \odot \mathbf{e}_j,\ \mathbf{x}_j^s \odot \mathbf{e}_i) \tag{2}$$

where $\mathbf{x}^s_{i/j}$ is the expression vector of i/j-th gene in scRNA-seq data $\mathbf{X}^s$, $\odot$ denotes element-wise multiplication (Hadamard product). $\mathbf{e}_{i/j}$ is the indication vector for $\mathbf{x}^s_{i/j}$ (if x^s_{ik} is a non-zero value, e_{ik}=1; otherwise, e_{ik}=0). When the reliable (non-zero) expression values are fewer than 5% cells for a gene pair, its PCC will be considered unreliable and filled with zero.

Finally, we integrate the correlation coefficients between gene pairs at the single-cell and bulk levels to construct a composite gene relation network to preserve both the homogeneous and heterogeneous information as:

$$\mathbf{A} = \frac{\mathbf{S} + \alpha * \mathbf{B}}{1 + \alpha}, \tag{3}$$

where α is the balance factor to balance bulk homogeneous and single-cell heterogeneous information.

Then, we set a threshold θ to remove trivial relations between gene pairs when $|A_{ij}|_i\theta$. Following the meaning of PCC [41], if $|A_{ij}|$ is less than 0.4, there is a weak correlation between two genes, so we set θ=0.4.

Graph-based GAN for data imputation

To identify and impute dropout events in scRNA-seq data and implement data imputation, we train a graph-based GAN model. Compared with other generative models, the main advantage of GAN is that it surpasses the functions of traditional neural network classification and feature extraction and can generate data according to the characteristics of real data. We employ GCN as the core structure of scGGAN generator network that embeds the gene relation network to guide the scRNA-seq data generation for recovering biologically meaningful expression values. Compared with other Euclidean-based generative models [30], graph-based model can dynamically integrate gene-to-gene relations and better guide the imputation by biological knowledge. To quantify the difference between the distribution of real scRNA-seq data and the generated one, we build a discriminator network similar to the generator for adversarial learning. The whole framework of scGGAN is illustrated in Figure 1.

scGGAN consists of two parts: generator and discriminator. The generator aims to learn the regional gene-to-gene relations and global scRNA-seq distribution by integrating multi-omics data (raw scRNA-seq, gene relation network and genomics data) and then imputes scRNA-seq dropout values by generated data. The input of generator are scRNA-seq data $\mathbf{X}^s \in \mathbb{R}^{n\times m_1}$, indicator matrix $\mathbf{M} \in \mathbb{R}^{n\times m_1}$ for $\mathbf{X}^s$, gene sequence feature $\mathbf{G} \in \mathbb{R}^{n\times 4^k}$ and gene relation network $\mathbf{A} \in \mathbb{R}^{n\times n}$. The output of generator is the imputed scRNA-seq data $\mathbf{Y}^s \in \mathbb{R}^{n\times m_1}$. The procedure for the generator to impute scRNA-seq expression data is organized as follows: (1) *Embedding scRNA-seq and genomics data*: to facilitate data integration and reduce the influences of dimension, scGGAN first projects scRNA-seq and gene sequence feature into different feature spaces of the same dimension d through different multi-layer FC neural networks:

$$\mathbf{E}_{sc} = FC_{sc}(\mathbf{X}^s),\ \mathbf{E}_g = FC_g(\mathbf{G}). \tag{4}$$

(2) *Multi-omics data integration by GCN encoder*: to more comprehensively explore gene relations, scGGAN introduces a GCN encoder model to fuse the low-dimensional attribute representation matrices $\mathbf{E}_{sc}$, $\mathbf{E}_g$ and gene relation network $\mathbf{A}$, and the output is graph embedding representation $\mathbf{H} \in \mathbb{R}^{n*h}$.

$$\mathbf{H} = GCN_{en}(Concat(\mathbf{E}_{sc}, \mathbf{E}_g), \mathbf{A}), \tag{5}$$

where *Concat* represents the matrix concatenation. (3) *scRNA-seq data generation by GCN decoder*: Based on graph embedding representation matrix $\mathbf{H}$, scGGAN generates an expression matrix $\widetilde{\mathbf{X}}^s$ through GCN decoder model:

$$\widetilde{\mathbf{X}}^s = Sigmoid(GCN_{de}(\mathbf{H}, \mathbf{A})), \tag{6}$$

where *Sigmoid* function is used to convert the generated data into (0, 1) as expression values. (4) *scRNA-seq imputation*: scGGAN mainly focuses on recovering false zeros in scRNA-seq data. During imputation, non-zero values are assumed to be true expression values and are not modified. scGGAN imputes raw scRNA-seq by the generated expression data $\widetilde{\mathbf{X}}^s$ under the guidance of indicator matrix $\mathbf{M}$. If the corresponding position of the raw observation matrix is zero, imputed data $\mathbf{Y}^s$ uses the generated scRNA-seq data for imputation; otherwise, $\mathbf{Y}^s$ will keep the observed values:

$$\mathbf{Y}^s = \mathbf{X}^s \odot \mathbf{M} + \widetilde{\mathbf{X}}^s \odot (1 - \mathbf{M}) \tag{7}$$

The discriminator of scGGAN aims to distinguish that the expression values in $\mathbf{Y}^s$ are real or generated. Its inputs are imputed scRNA-seq data $\mathbf{Y}^s$, gene sequence feature $\mathbf{G}$ and gene relation network $\mathbf{A}$. Its structure and calculation process are similar to the generator except for the prediction process. Its outputs are the probability matrix $\mathbf{P}$. The specific circumstances are as follows: (1) *Embedding scRNA-seq and genomics data*: the 1st step of discriminator is to calculate the embedding of imputed scRNA-seq $\mathbf{Y}^s$ and gene sequence feature $\mathbf{G}$ by FC model, similar to Equation (4). (2) *Multi-omics data integration by gene relation network*: the discriminator integrates embedded scRNA-seq, genomics data and gene relation network $\mathbf{A}$ by GCN encoder model to obtain the graph embedding representation, similar to Equation (5). (3) *Generated or real scRNA-seq identification*: the discriminator calculates the probability matrix $\mathbf{P}$ ($\mathbf{P} \in \mathbb{R}^{n*m1}$, $p_{ij} \in [0, 1]$) using the graph embedding representation through a FC network, where p_{ij} represents the probability that y^s_{ij} is a true observation.

scGGAN separately and alternately trains its generator and discriminator by an adversarial learning pattern. The generator of scGGAN aims to mimic real scRNA-seq data and impute dropout data, its loss can be divided into two parts: the generative adversarial loss l_1 and the distance loss l_2 between the imputed and observed scRNA-seq data. l_1 is used to evaluate whether the generated data distribution is similar to the real one, in other words, whether the generated data can fool the discriminator. Therefore, scGGAN defines l_1 based on Binary Cross Entropy (BCE) loss between the indicator matrix $\mathbf{M}$ (labels about whether data in $\mathbf{Y}^s$ are real or generated) and probability matrix $\mathbf{P}$ (the prediction by discriminator) as:

$$l_1 = -(1 - \mathbf{M}) * log(\mathbf{P}). \tag{8}$$

In addition, for generated scRNA-seq data $\widetilde{\mathbf{X}}^s$, if the corresponding position of the original observation matrix is zero, it will be used for imputation; otherwise, the generated expression values should be close to the original observed ones as much as possible. Therefore, scGGAN uses the indicator matrix $\mathbf{M}$ to pick out the positions with originally observed values from $\mathbf{X}^s/\widetilde{\mathbf{X}}^s$ and then

calculates the distance between generated and original scRNA-seq data as the loss l_2. Specifically, scGGAN uses mean squared error (MSE) as the loss function of l_2 as:

$$l_2 = MSE(\mathbf{X}^s \odot \mathbf{M}, \widetilde{\mathbf{X}}^s \odot \mathbf{M}). \quad (9)$$

Since the loss functions for two tasks are inconsistent, to reduce the impact of different magnitudes on training, we refer to the multi-task loss balancing strategy [42] to adaptively adjust the loss weight through two trainable parameters (σ_1 and σ_2) as follows:

$$loss_G = \frac{1}{2\sigma_1^2} l_1 + \frac{1}{2\sigma_2^2} l_2 + log\sigma_1\sigma_2. \quad (10)$$

For the discriminator of scGGAN, it aims to distinguish the generated and real data as accurate as possible. We define the BCE loss function for discriminator based on the indicator matrix $\mathbf{M}$ and the probability matrix $\mathbf{P}$ as follows:

$$loss_D = -(\mathbf{M} * log(\mathbf{P}) + (1 - \mathbf{M}) * log(1 - \mathbf{P})). \quad (11)$$

Results and Analysis

Baselines

To comparatively study the performance of scGGAN, we compare it against with some competitive and representative baselines whose implementation strategies are as follow:

(i) **MAGIC** [17] imputes dropout values by sharing information among similar cells via data diffusion on Markov affinity matrix. We adopt the generally shared code (https://github.com/DpeerLab/magic) for experiments.

(ii) **scImpute** [18] can automatically identify possible 'dropout' events ('false' zero values) and perform imputation only on the identified values without introducing new noise to the rest data. We use the shared R package '*scImpute*' for experiments.

(iii) **SAVER** [21] assumes that the count of each gene in each cell follows a negative binomial model and estimates the prior parameters by an empirical Bayes-like method. We use the public R package '*SAVER*' for experiments.

(iv) **netNMF-sc** [14] uses gene–gene interaction network regularized non-negative matrix factorization (NMF) to map scRNA-seq data into two low-dimensional matrices (cell and gene) and imputes scRNA-seq by product of two factor matrices. We directly implement netNMF-sc by MATLAB for experiments.

(v) **DCA** [22] uses the negative binomial noise model with or without zero-inflation and considers the count distribution, over-dispersion and sparsity of the data and nonlinear gene–gene correlation to impute the missing values. We directly use the shared code (https://github.com/theislab/dca) for experiments.

(vi) **scIGANs** [30] transposes scRNA-seq data into images and trains GAN to learn data distribution, then uses the generated series of data for imputation by k Nearest Neighbor. We directly adopt the shared code (https://github.com/xuyungang/scIGANs) for experiments.

(vii) **GNNImpute** [19] is based on graph attention convolution model and focuses on determining the similarity between cells by constructed connection graph. We adopt the original code (https://github.com/Lav-i/GNNImpute) for experiments.

(viii) **ALRA** [23] is a method based on low-rank approximation, which applies non-negativity and correlation structure to selectively impute the missing values. We directly use the shared code (https://github.com/KlugerLab/ALRA) for experiments. In addition, we also design two variants of scGGAN as baselines:

(ix) **scGGAN-fc** replaces the GCN structure of scGGAN with FC network to verify the effectiveness of gene relation network for imputation;

(x) **scGGAN-ng** disregards genomics data to testify whether gene sequence data can improve the imputation performance.

All parameters configuration of compared methods are set with the best parameters as suggested in the original papers or with the default parameters of shared codes. For scGGAN parameter, we set learning rate as 0.001, epochs as 1000, batch size as 256, output dimension of FC as 128, output dimension of GCN encoder as 128. The codes of scGGAN are shared at https://www.sdu-idea.cn/codes.php?name=scGGAN.

Evaluation on simulated scRNA-seq datasets

scRNA-seq data imputation aims to recover biologically meaningful expression from 'dropout' events and then improve the quality of downstream analysis. We explore the effectiveness of scGGAN for imputation by conducting experiments on simulated scRNA-seq datasets. First, we generate simulated data by '*Splatter*' [43] package of the R language, which is a canonical single-cell expression data simulator and also offers some models to simulate 'dropout' in the generated data. The simulated dataset has 10 000 genes and 1000 cells and consists of four clusters with approximate 500, 300, 150 and 50 cells per cluster, respectively. We randomly mask some gene expression values to zero following a logistic function based on each gene's mean expression provided by '*Splatter*' and adjust parameters to make the sparsity levels (zero proportion) of each dataset approximately 60%, 70%, 80%, 90%.

Unlike real scRNA-seq datasets, the corresponding multi-omics data (i.e. bulk RNA-seq data and genomics sequences) are not available for the simulated data, so we only use scRNA-seq data for the imputation. For simulated scRNA-seq data that have known 'true' expression values, in order to quantitatively evaluate the performance of different imputation methods in recovering biologically meaningful gene expression, we use two typical regression metrics: MSE and PCC between the imputed data and real scRNA-seq data (without dropout values). In addition, we separately calculate the MSE loss for different cell types to observe the imputation ability for rare cell types. Table 1 shows the mean MSE for ten experiments between real and imputed scRNA-seq data under different sparsity rates and different cell types for 'Splatter' simulated scRNA-seq datasets. Figure 2 shows the PCC for 'Splatter' simulated scRNA-seq datasets and imputed results by different methods.

It is worth to note that GNNImpute and ALRA fail on this dataset, and they both have MSE >200. From these results, we can find that when the scRNA-seq data sparsity rate increases, the performance of all methods will decrease. This is because the higher the sparsity rate, the more difficult the imputation task is. scGGAN obtains the best MSE and PCC at different sparsity rates compared with all baselines, which proves that scGGAN can

Table 1. The MSE between real and imputed scRNA-seq data under different sparsity rates on 'Splatter' simulated scRNA-seq. For each column in table, the data in the 1st row represents the overall loss of the method, and the data in the 2nd row represent the losses of different cell types (the ratio of different types of cells is about 10:6:3:1), respectively. The results in the table are the average of 10 independent experiments.

Sparsity rates	60%				70%				80%				90%			
MAGIC	42.88±0.36				44.54±0.23				49.63±0.51				71.08± 0.59			
	43.19	41.62	45.13	40.38	44.89	43.01	47.01	42.13	50.03	47.75	52.62	47.64	71.05	67.80	76.32	73.94
scImpute	6.18±0.32				12.78±0.61				30.66±0.85				68.18±1.96			
	5.81	5.83	5.48	13.57	12.48	12.68	12.50	16.96	30.21	29.59	31.60	37.78	68.13	65.42	72.93	79.81
SAVER	24.58±0.88				32.53±0.91				52.92±1.05				104.96±2.20			
	24.75	23.90	26.26	22.09	32.73	31.74	34.29	29.89	52.42	51.17	55.27	48.22	105.36	103.52	107.84	100.81
netNMF-sc	20.44±1.08				21.95±1.51				27.21±1.64				50.19±1.34			
	19.97	20.54	22.88	25.83	22.06	21.49	23.38	26.36	26.86	27.32	28.47	32.59	49.54	54.22	55.69	57.97
DCA	39.82±2.42				48.02±3.08				52.65±2.02				72.40±3.68			
	35.76	42.53	42.64	52.84	43.37	51.54	54.99	59.73	52.53	52.09	51.92	59.12	64.10	76.34	71.55	87.89
scIGANs	6.70±0.52				15.06±0.85				32.74±1.22				87.27±1.68			
	6.86	6.53	6.63	6.48	15.36	14.66	14.97	14.82	32.97	32.06	33.99	30.94	87.86	86.08	88.17	85.96
scGGAN-fc	6.61±0.51				14.53±1.60				32.33±2.38				68.10±3.27			
	6.76	6.46	6.52	6.39	14.83	14.12	14.44	14.32	32.58	31.60	33.55	30.63	67.53	69.18	68.04	67.38
scGGAN	**3.68±0.53**				**6.89±1.06**				**19.35±1.42**				**26.68±1.30**			
	3.65	3.70	3.70	3.75	6.83	6.85	7.08	7.14	19.16	19.44	19.82	19.09	26.23	26.77	27.34	28.41

The best result is highlighted in **bold** font.

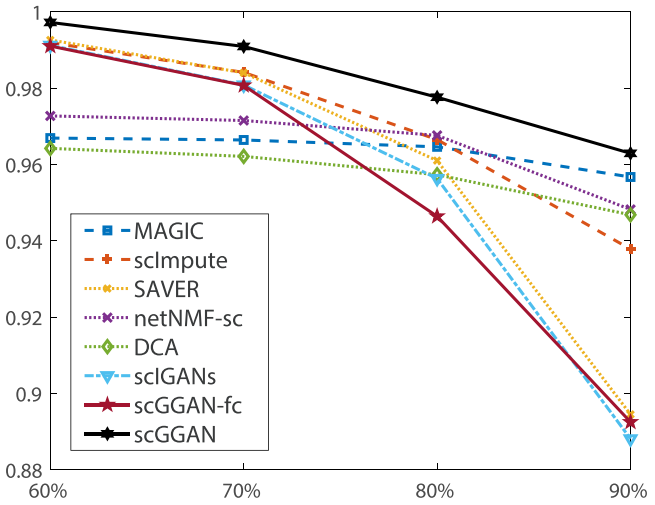

Figure 2. The PCC between real and imputed scRNA-seq data under different sparsity rates.

effectively recover biologically meaningful gene expression, and scGGAN also achieves similar results across different imbalanced cell types. This fact confirms that scGGAN can effectively remedy the bias toward cell types with a larger number of cells. This bias is suffered by most compared methods (i.e., scImpute, DCA and netNMF-sc). In summary, scGGAN can effectively deal with the 'dropout' event to recover the real expression values and achieve better results in unbalanced scRNA-seq data. These results also suggest that scGGAN can be applied to impute scRNA-seq data only by scRNA-seq without other omics data.

In addition, we further design the simulated experiment on real single-cell dataset. First, we select real scRNA-seq and bulk RNA-seq datasets (GEO: GSE75748) [44] for experiment, whose scRNA-seq data is with zero expression as 49%, and then we

Table 2. The MSE and PCC of different methods on masked GSE75748 dataset.

	MSE	PCC
MAGIC	20.78±0.42	0.6638±0.0052
scImpute	15.00±0.51	0.6477±0.0039
SAVER	40.33±0.69	0.4502±0.0037
netNMF-sc	18.17±1.05	0.6833±0.0075
DCA	21.93±1.30	0.5825±0.0131
scIGANs	15.92±0.36	0.6566±0.0128
scGGAN-fc	17.57±0.69	0.6419±0.0171
scGGAN-ng	14.38±0.81	0.7251±0.0081
scGGAN	**11.68±0.68**	**0.7311±0.0061**

The best result is highlighted in **bold** font.

randomly mask the single cell expression matrix to make it with approximately 70% zero. Considering that the real single-cell dataset has some dropout values, to evaluate the quality of the imputed data, we calculate MSE and PCC only between the imputed values and the real data at masked positions. Table 2 reports the results of different methods on this dataset.

From Table 2, we can find that scGGAN achieves best MSE and PCC on masked GSE75748 dataset, this is due to the fact that scGGAN can capture the regional gene-to-gene relations and predict the global scRNA-seq distribution, and it also utilizes genomics information to guide scRNA-seq data generation and predict missing values. The overall results show that scGGAN can more accurately recover gene expression. Compared with its two variants (scGGAN-fc and scGGAN-ng, respectively, disregard the gene relation network and genomics data), scGGAN also achieves better performance on both MSE and PCC, which proves that both gene relation network and genomics sequence data can help the imputation through providing more biological information, and the GCN structure (gene relation network) has a more important role in improving performance. Overall, scGGAN can successfully

recover missing values in scRNA-seq data and obtain an imputed matrix more similar to real scRNA-seq data matrix.

Evaluation on real scRNA-seq datasets

Cell clustering

Among the various downstream analyses of scRNA-seq data, cell clustering is the first step to visualize each cell in a low-dimensional space and to identify known or novel cell types. Dropout events decrease the cell-to-cell similarity within those same kind of cells, which will cause mistakes for cell types identification. Therefore, we utilize the cell type labels reported in the original datasets for cell clustering experiments on four different scRNA-seq datasets with different sequencing protocols: GSE75748 (Fluidigm C1) [44], GSE65525 (CEL-seq protocol) [45], GSE67835 (SMARTer protocol) [46] and 10X peripheral blood mononuclear cells (PBMCs) (10X Genomics) [47]. The details of these datasets are given in the Data and Code availability. For the selection of bulk data, the datasets we selected contain both single-cell and bulk sequencing data, which have the same cell type and can provide gene homogeneity information. In particular, the 10X PBMCs dataset has no associated bulk RNA-seq data so that we just use scRNA-seq data to construct the gene relation network. We use typical clustering metrics (Adjusted Rand Index, ARI and Normalized Mutual Information, NMI) to evaluate the consistency between the predicted results and real labels. For raw and imputed scRNA-seq data, we first use UMAP (Uniform Manifold Approximation and Projection) [48] to map high-dimensional scRNA-seq data into 2D space and visualize the scatter plot in Figure 4 (GSE75748) and Figure 5 (GSE65525), Figure 6 (GSE67835) and Figure 7 (10X PBMCs). Then, we cluster cells on the results of UMAP by *k*-means clustering algorithm and report the cell clustering performance (ARI and NMI) for different imputation methods in Figure 3.

From the visualization results of UMAP in Figures 4, 5, 6 and 7, scGGAN has a clear division boundaries and it also improves the visualization results of single cells compared with raw dataset. From ARI and NMI in Figure 3, scGGAN also has a better clustering performance, which proves that scGGAN can work well on scRNA-seq datasets with different protocols. Compared with other imputation methods, scGGAN can obtain better visualization results and clustering performance on different scRNA-seq datasets, which shows that scGGAN can more credibly impute the dropout values. Other important observations include:

(i) MAGIC, scImpute and GNNImpute all borrow expression information from similar cells to impute the 'dropout' values, but they may lead to over-smoothing or remove natural cell-to-cell stochasticity in gene expression. In contrast, scGGAN imputes scRNA-seq data from gene relations (rather than cell similarity) and real scRNA-seq distribution, and it considers the cell stochasticity. Therefore, scGGAN achieves better clustering results than them.

(ii) Unlike SAVER, DCA and ALRA that build on a prior hypothesis for scRNA-seq distribution, scGGAN directly learns the data distribution by an adversarial learning strategy. Therefore, scGGAN has a better applicability and adaption, it achieves a better imputation performance than them and obtains better visualization and clustering results.

(iii) Although netNMF-sc exploits the gene interaction network, its clustering performance also loses to scGGAN by a clear margin. This is because netNMF-sc builds on network-regularized NMF, whose decomposition results can not be made orthogonal and this will reduce its performance. On the other hand, scGGAN is based on a deep generative model and can learn various distribution relationships in single-cell RNA-seq data, so it achieves a better performance.

(iv) Both scIGANs and scGGAN are based on GAN model to impute the dropout values, they can learn the real data distribution without hypothetical distribution and often have better results than other baselines. This fact demonstrates the effectiveness of using GAN for scRNA-seq data imputation. scGGAN further uses GCN to learn the relations between genes, whereas scIGANs converts the scRNA-seq data into images and then applies CNN to impute dropout value. scIGANs fails to account for objectively present relations and adds noise of relative position information when converting scRNA-seq data to images. For these defects, scGGAN is clearly outperformed by scIGANs.

(v) scGGAN-fc and scGGAN-ng separately disregard the gene relation network and genomics data from scGGAN, they often outperform other baselines, but greatly lose to scGGAN. This fact proves the effectiveness of gene relation and genomics sequence data for scRNA-seq data imputation. scGGAN-ng often gives a clearer boundary and better clustering than scGGAN-fc, this observation suggests that gene relation network and the GCN framework have a greater impact than genomics data on the model performance.

The above analyses also explain why scGGAN can more accurately recover gene expression on simulated scRNA-seq datasets. In summary, compared with the state-of-the-art imputation methods, scGGAN can more credibly impute the dropout values and thus obtain better cell clustering on different scRNA-seq datasets, which proves the superiority of scGGAN.

Differential expression analysis

Differential expression analysis is a basic way to study the similarities and differences of gene expression between two groups of samples. It aims to identify cell-specific genes whose expressions are significantly upregulated or downregulated in one group to another group (i.e. healthy versus disease samples). These differentially expressed genes (DEGs) have important effects on phenotype difference and can be further studied (i.e. enriched pathways or biological processes) for pathology analysis and biomarker discovery. Therefore, the effective imputation methods can reduce dropouts and discover the hidden gene expression patterns from scRNA-seq data, so that we apply '*limma*' [49] to find DEGs on raw and imputed scRNA-seq data of each baseline. The criterion for DEGs is that the log fold changes ≥ 1 (upregulated) or $\leq$-1 (downregulated) with adjusted *P*-value ≤ 0.05. Since the bulk RNA sequencing data are hardly affected by dropout events and considers the higher sensitivity of bulk RNA-seq technology in detecting differential expression at the transcriptome scale, to evaluate the quality of found DEGs, we also download the bulk RNA-seq data from the same dataset (GSE75748) and apply '*limma*' to generate the 'gold standard' by the results of differential expression analysis in bulk RNA-seq data. We introduce another variant (scGGAN-nb) that does not use bulk RNA-seq data (α=0 in Equation (3)) to quantify the contribution of bulk RNA-seq data. To intuitively compare the performance of various methods, we take DEGs obtained from imputed scRNA-seq data of each method as predicted labels. Next, we compute the F1 score and Accuracy between the 'gold standard' and predicted ones and report them in Figure 8.

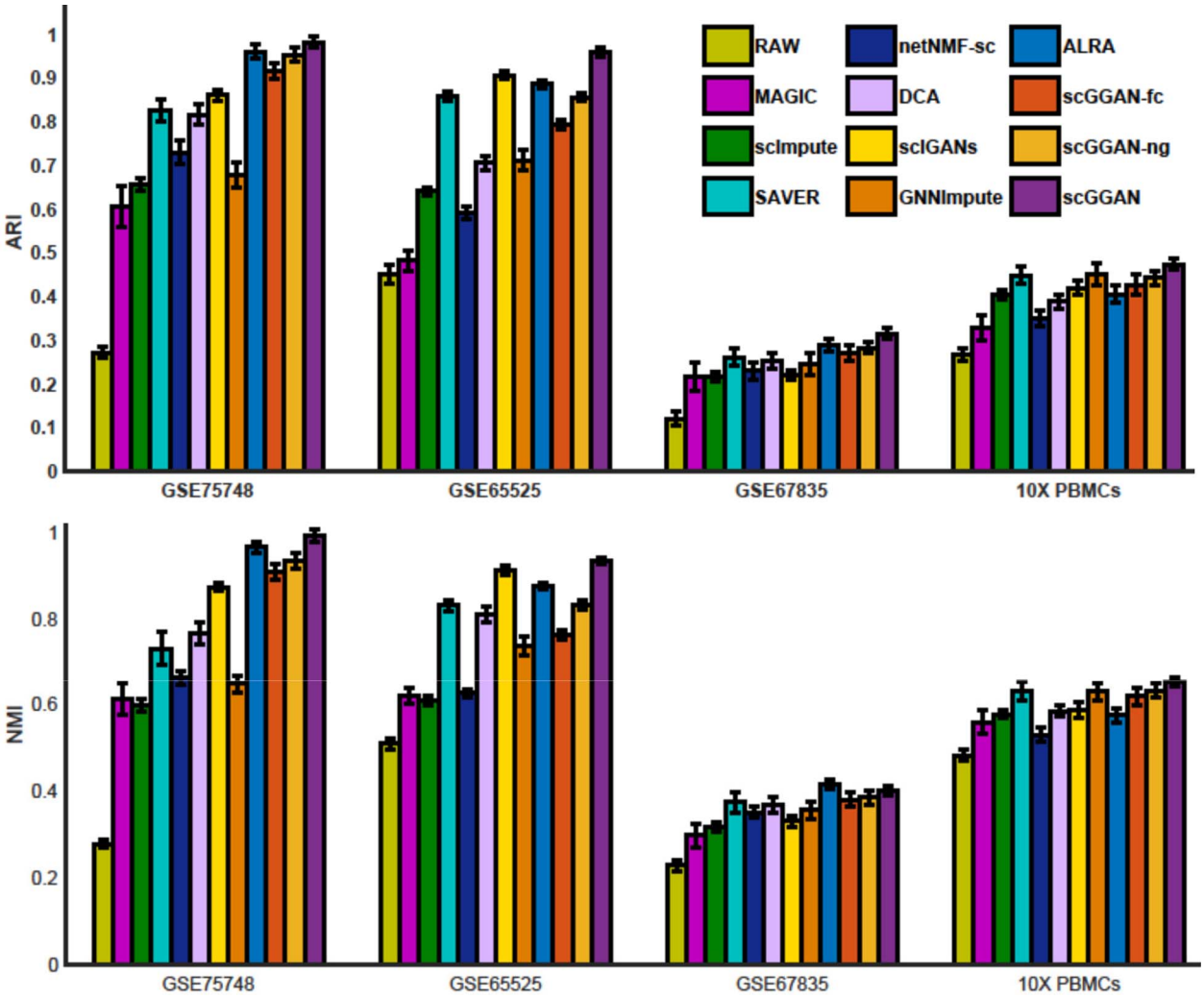

Figure 3. The cell clustering performance (ARI and NMI) for raw data and imputed scRNA-seq data by different methods on four datasets.

From results in Figure 8, we can find that the identification of DEGs using raw scRNA-seq data is low even though both scRNA-seq and bulk RNA-seq are sampled from the same cell types, which proves that the dropout events severely inhibit differential expression analysis. Since the identification of DEGs has a great impact on downstream analysis, it is crucial to reduce errors due to the technical noises. Compared with these baselines, scGGAN achieves the best F1 score and Accuracy, which proves that scGGAN can significantly promote the DEGs identification and indicates the potential of scGGAN in single-cell data analysis. Similar cell-based methods (i.e. MAGIC and GNNImpute) cannot improve the performance of predicting DEGs, because their imputation results are over-smoothed, resulting in a large number of genes with similar expression values and few DEGs identified. Other methods can improve the performance, which indicates that these imputation methods can effectively identify and impute dropouts. Although scGGAN-nb does not use bulk RNA-seq, it often has a better (or comparable) F1 and Accuracy than other methods (except scGGAN) and with a larger variance. This proves that scGGAN can improve the imputation by its network structure, not just by the usage of bulk RNA-seq data.

We further apply Gene Ontology (GO) enrichment analysis on these DEGs and report the results of the bulk RNA-seq and scGGAN in Figure 9 to explore the function of the DEGs identified by scGGAN. From the GO enrichment analysis results, we can find that imputation results of scGGAN are similar with bulk on all biological process (BP), cellular component (CC), molecular function (MF) ontology. In addition to the same GO terms with bulk data, the imputation results of scGGAN are also enriched for some cell-differentiated functions. These results suggest that scGGAN can effectively recover biologically meaningful expression values under the guidance of gene relation network, it can improve the identification of DEGs, which share DEGs with bulk samples and are enriched for similar functions.

Cell trajectory analysis

Cell trajectory analysis is also one of the important tasks in scRNA-seq data analysis. It can reshape the change process of cells over time by constructing the change trajectories between cells, which will help researchers to infer the development and differentiation process between cells from the single-cell level. We employ the time-course scRNA-seq data derived from the differentiation from H1 ESC to DEC (GSE75748) [44], which consists of 758 cells, including 92 cells at 0h, 102 cells at 12h, 66 cells at 24h, 172 cells at 36h, 138 cells at 72h and 188 cells at 96h after the differentiation from H1 ESCs to DECs. To evaluate the performance of imputation methods for reconstructing the cell developmental trajectories, we apply all imputation methods on this dataset, and then use '*Monocle2*' [50] to visualize trajectories and infer pseudo-time. We compute the

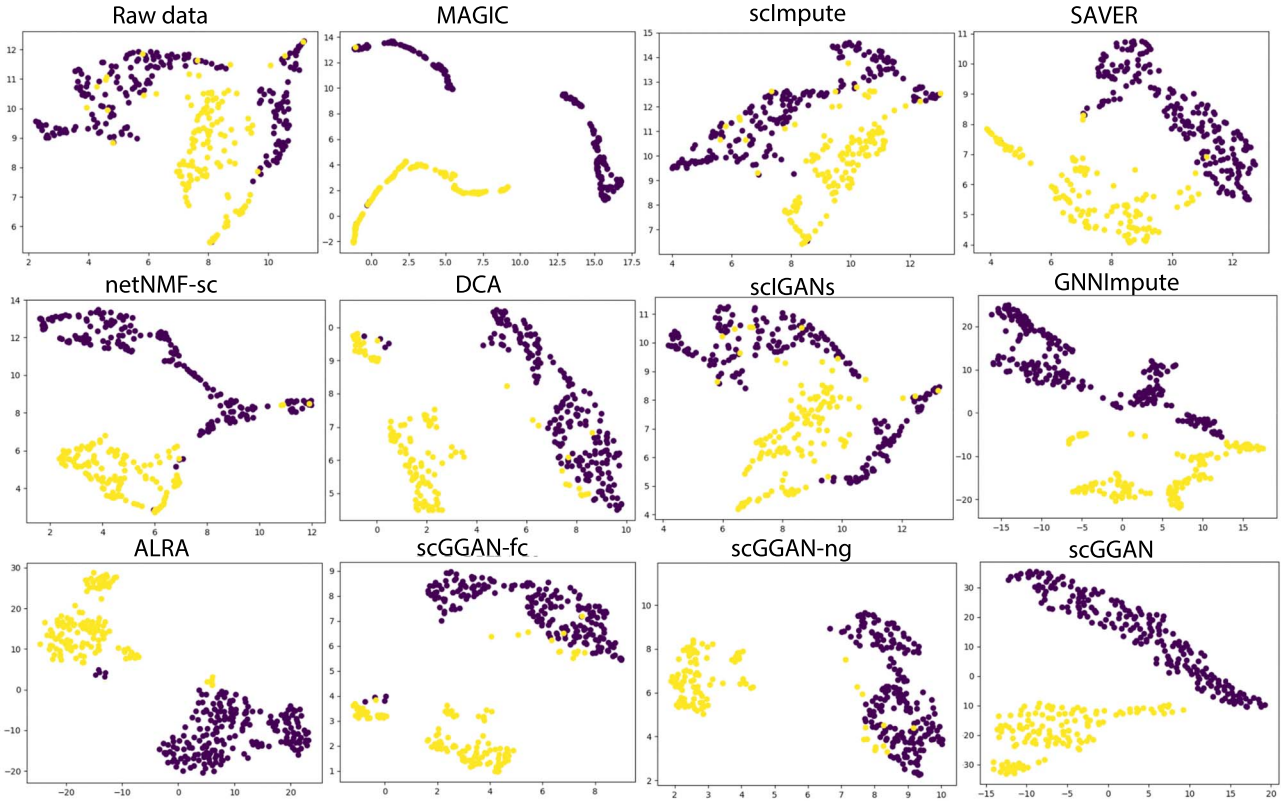

Figure 4. The UMAP dimension reduction and visualization results of raw and imputed scRNA-seq data by different methods on GSE75748.

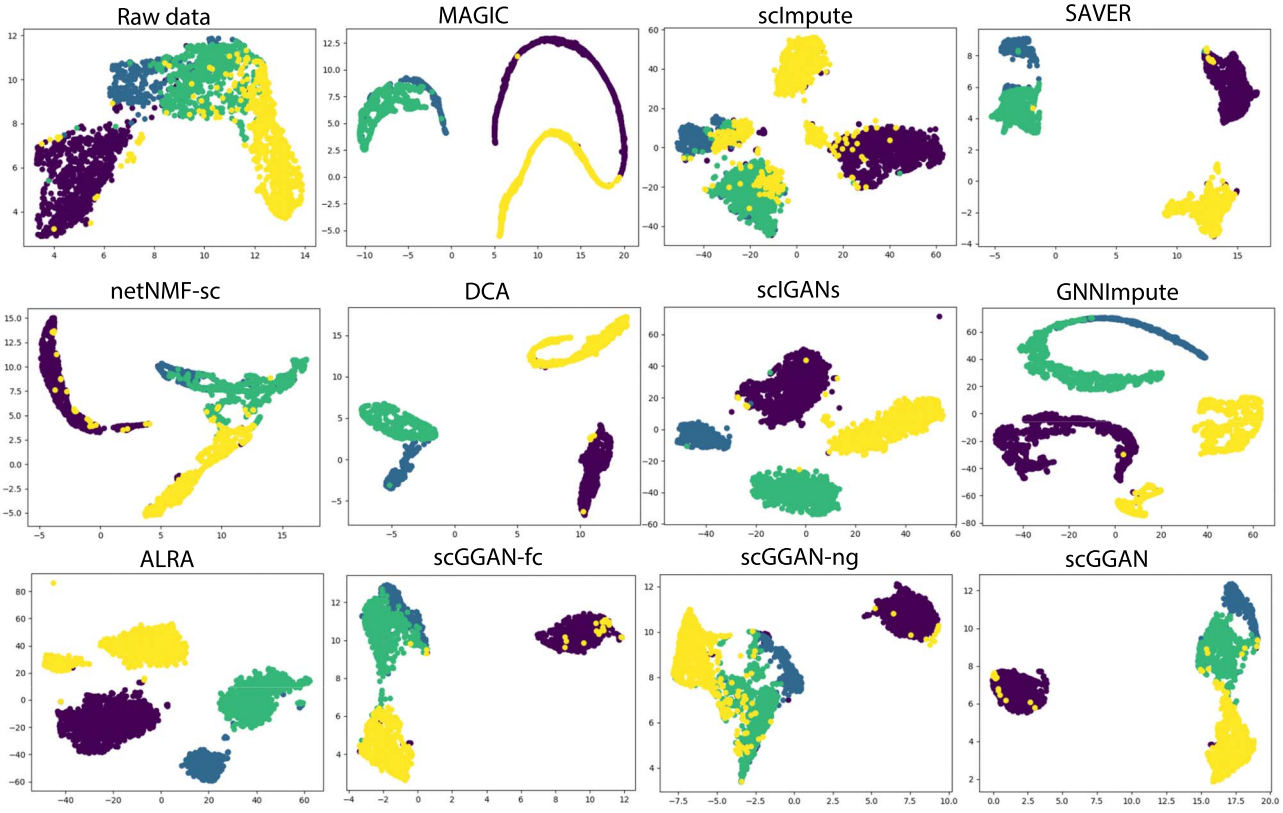

Figure 5. The UMAP dimension reduction and visualization results of raw and imputed scRNA-seq data by different methods on GSE65525.

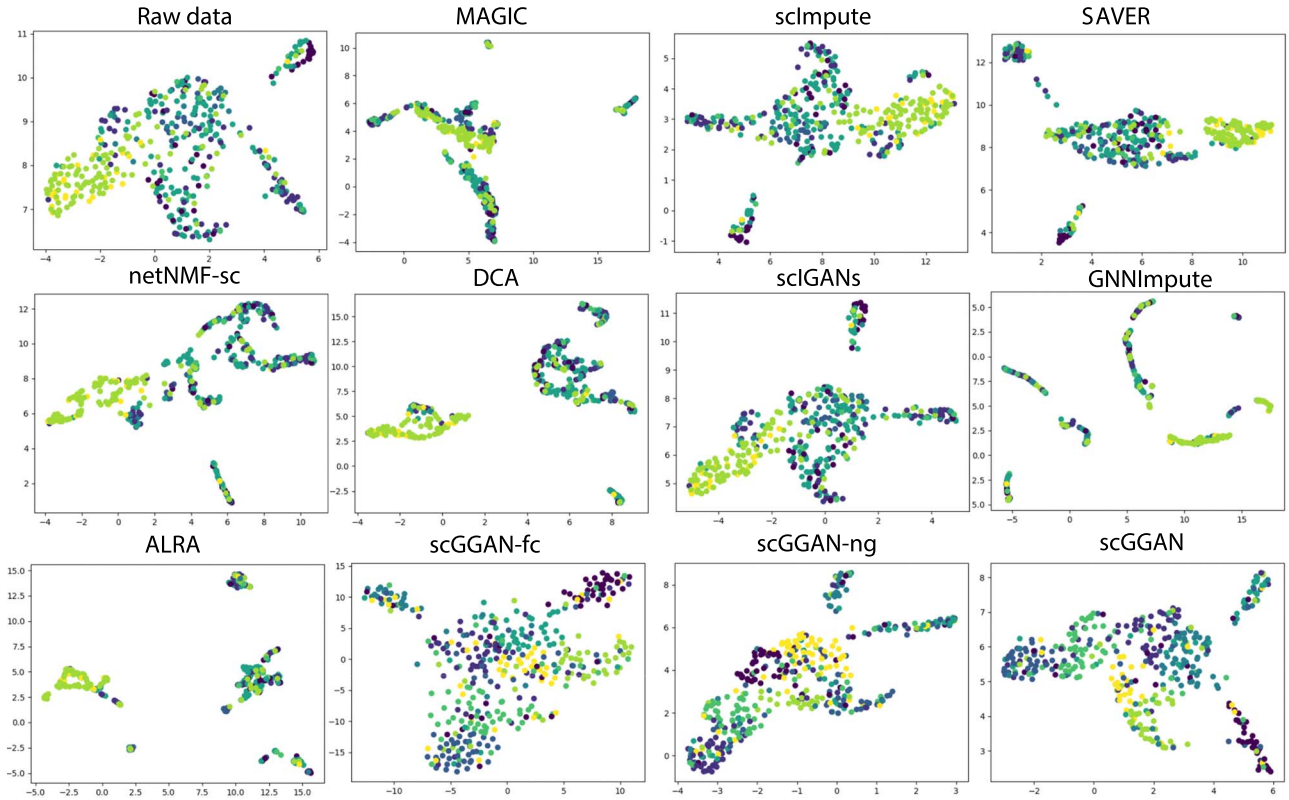

Figure 6. The UMAP dimension reduction and visualization results of raw and imputed scRNA-seq data by different methods on GSE67835.

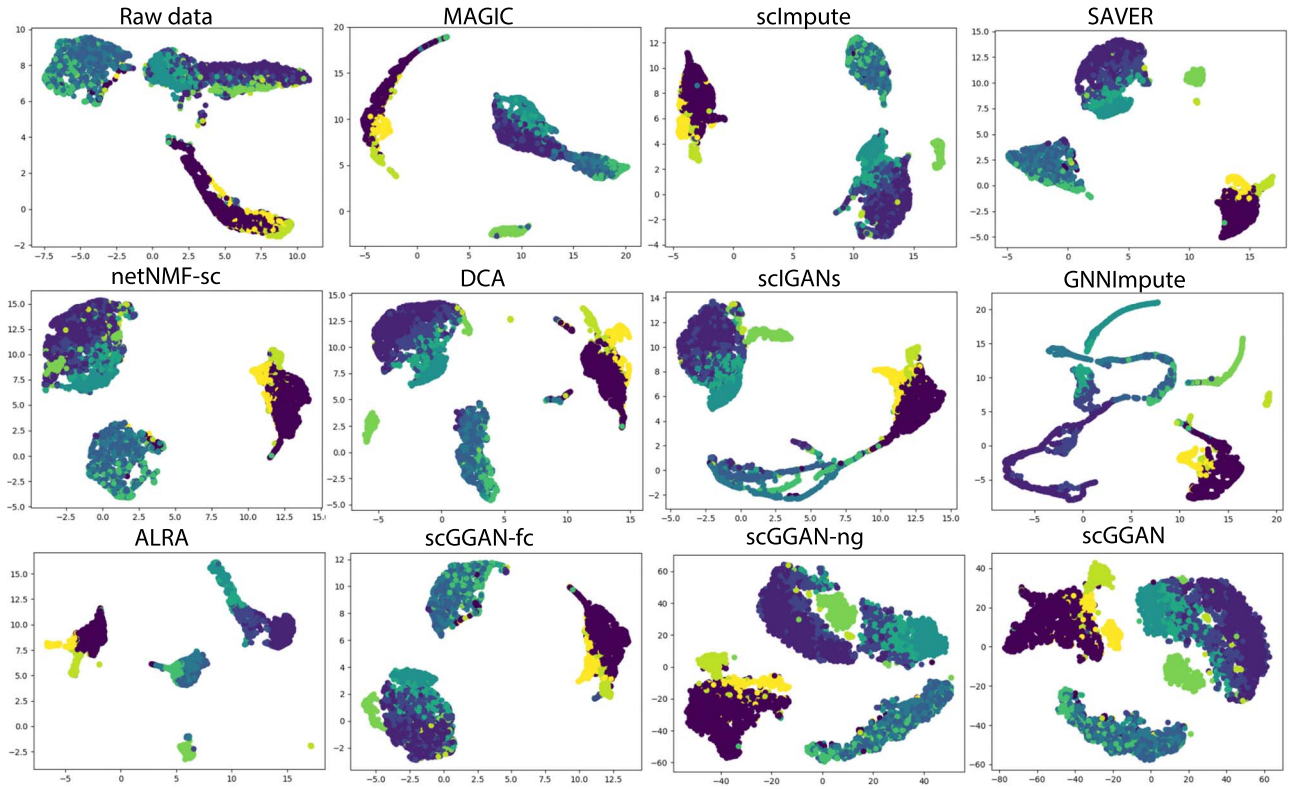

Figure 7. The UMAP dimension reduction and visualization results of raw and imputed scRNA-seq data by different methods on 10X PBMCs.

Kendall's rank correlation score (KRCS) between true-time labels and predicted pseudo-times to quantitatively evaluate the quality of the reconstructed trajectories, which reflect the consistency between two rankings. Figure 10 shows the visualization results of reconstructed trajectories and KRCS (τ) between the inferred pseudo-times and true-time labels for different methods.

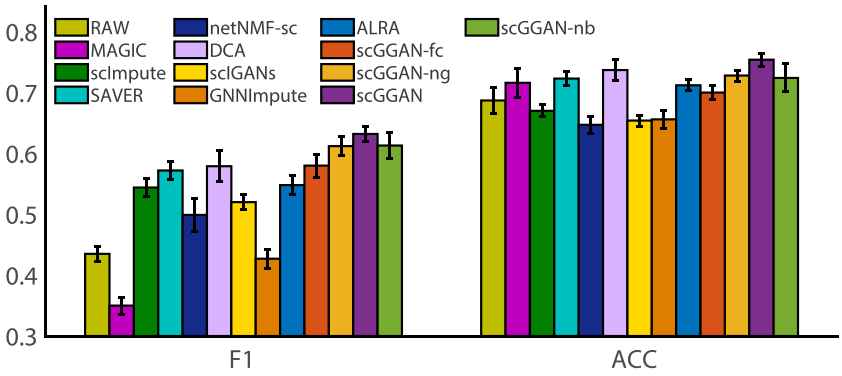

Figure 8. Performance of identifying DEGs with different imputation methods.

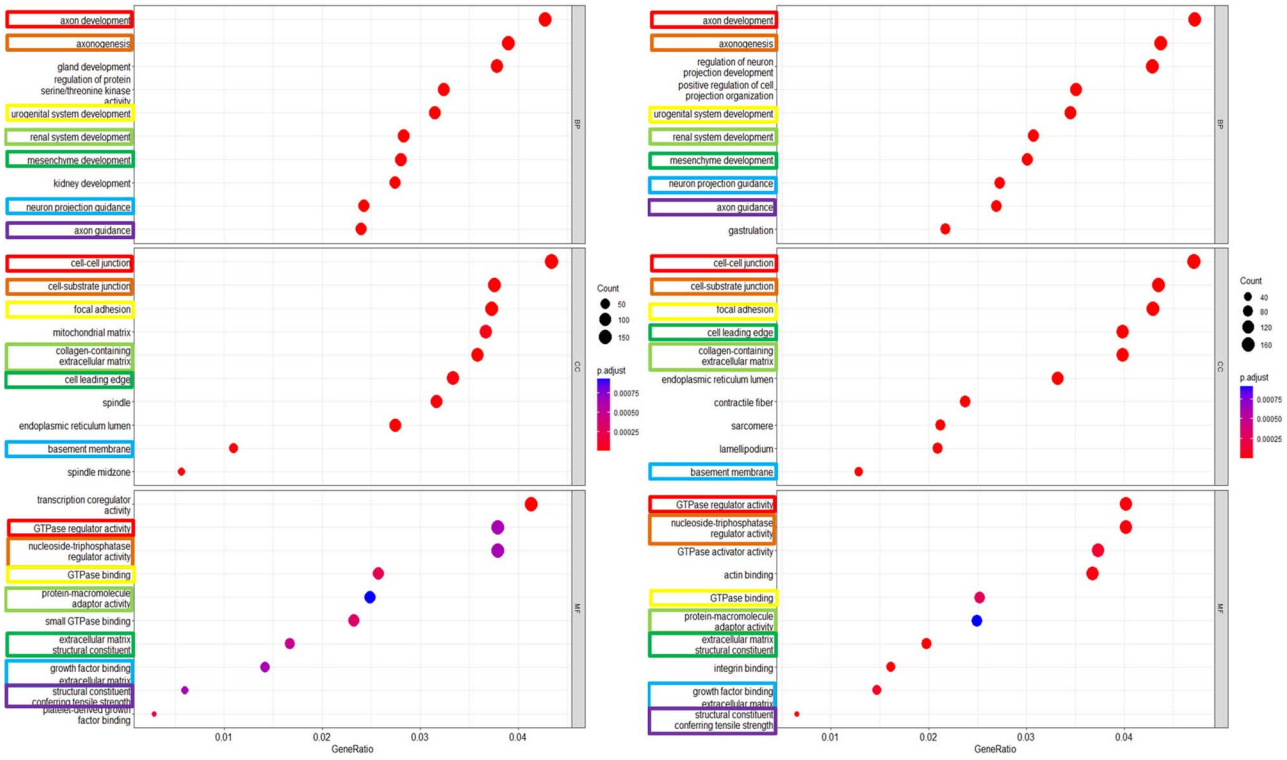

Figure 9. The GO enrichment analysis results for bulk RNA-seq and imputed scRNA-seq by scGGAN.

From the results in Figure 10, we can find that compared with raw scRNA-seq data, scGGAN can increase the KRCS calculated by pseudo-time and true-time from 0.591 to 0.756, which proves the power of scGGAN for increasing cell developmental trajectory inference. Compared with other baselines, scGGAN outperforms other methods in pseudo-time inference when the analysis is conducted by '*Monocle2*' and produces the highest correspondence between the inferred pseudo-times and true-time labels. These results demonstrate that scGGAN can more accurately recover transcriptome dynamics along the time course and improve the performance of pseudo-time inference.

Parameter analysis

We further conduct experiments to study the parameter sensitivity on GSE75748 dataset. For scGGAN, the most important parameter is the balance factor α for single-cell and bulk gene relation network. Since the model is designed to deal with single-cell RNA-seq data dropout events, it should ultimately pay more attention to single-cell heterogeneous information. Therefore, we choose different values ($\alpha \in [0, 1]$) for experiments to study the influence of α. We report the mean values of ARI and NMI under different α in Figure 11.

From Figure 11, we can find that the performance of scGGAN roughly increases first and then decreases with the increase of α. This is because when α is too small, it is difficult to optimize gene networks with homogeneous information from bulk RNA-seq data. With the increase of α, scGGAN can better utilize the bulk homogeneous information of genes, so its performance increases. When α is too large, the gene relation network is more biased towards the bulk homogeneous information and down-weights the single-cell heterogeneous information, so the performance of scGGAN decreases. This parameter analysis proves the necessity of leveraging homogeneous and heterogeneous information of gene relation network and also proves bulk RNA-seq data can improve the imputation performance of scGGAN.

For scGGAN, another important parameter is the length of subsequence (K) in K-mers strategy, which determines the dimensions of the genomics sequence features. We set the value range of $K \in \{2, 3, 4, 5, 6, 7, 8\}$ and exclude genomics sequence features for

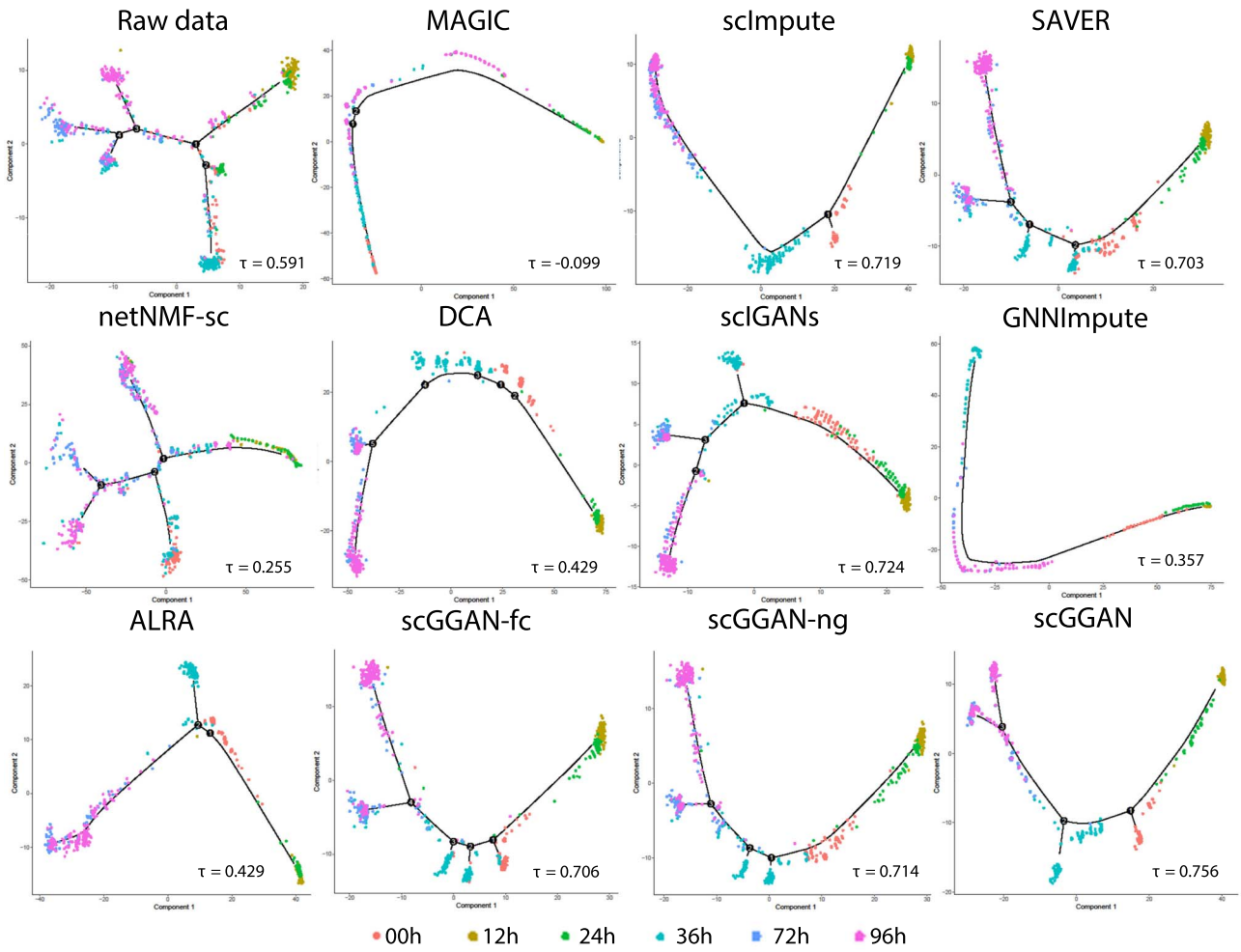

Figure 10. The reconstructed trajectory and Kendall's rank correlation score (τ) from the raw and imputed scRNA-seq data.

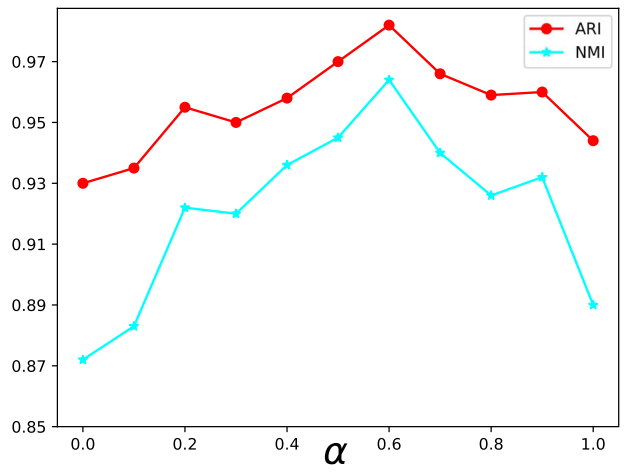

Figure 11. The mean values of ARI and NMI under different balance factor α.

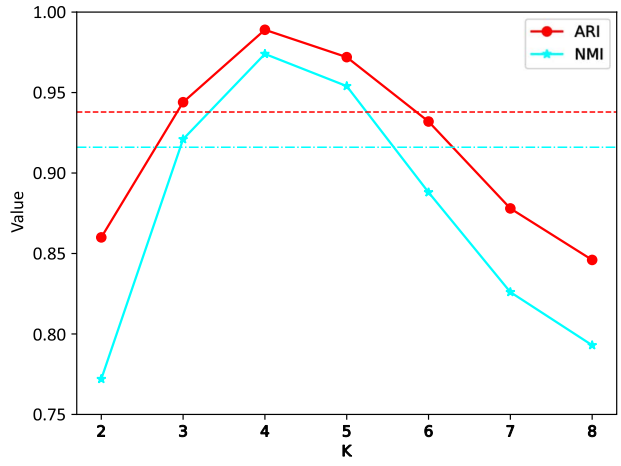

Figure 12. ARI and NMI vs. K-mer length.

experiments. Then, we report the mean values of ARI and NMI in Figure 12.

From the results in Figure 12, we can find that if K is too small or too big, the genomics sequence data can not improve the performance of scGGAN, even give a lower performance than scGGAN without genomics data (dotted line in Figure 12). This is because when K is too small, it is difficult for 4^K-dimensional feature vectors to well represent the sequence information of whole-genome sequencing data, and when K is too big, the 4^K-dimensional feature vector will be too sparse, so the FC network can not encode K-mers features well. Therefore, when the value of K takes a value within the appropriate range, the genomics

Table 3. Ablation experimental results under different bulk datasets.

Dataset	ARI	NMI	H1-recall	DEC-recall
No bulk	0.930±0.017	0.873±0.010	0.972±0.009	0.957±0.014
H1	0.947±0.008	0.913±0.012	**0.993±0.005**	0.949±0.007
DEC	0.942±0.012	0.904±0.010	0.962±0.010	**0.990±0.007**
All bulk	**0.982±0.015**	**0.964±0.011**	0.980±0.013	0.975±0.020

The best result is highlighted in **bold** font.

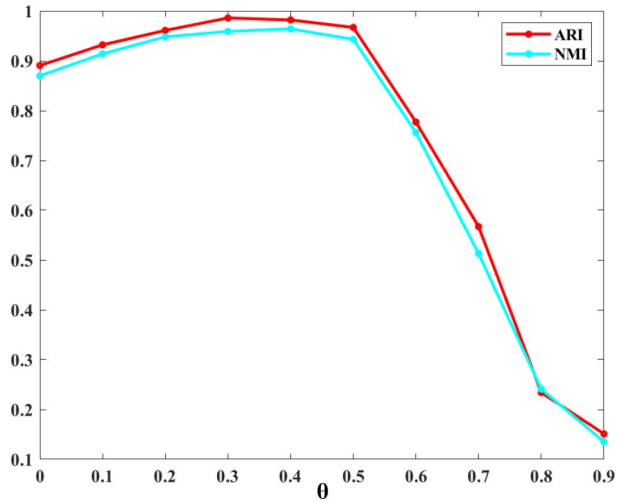

Figure 13. ARI and NMI vs. threshold θ.

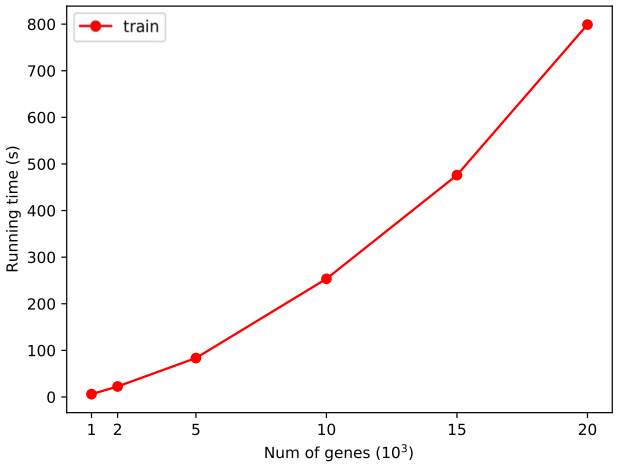

Figure 14. The running time under different numbers of genes.

sequence information represented by the *K*-mers strategy can improve the performance of scGGAN.

For scGGAN, we set a threshold θ to remove trivial relations between gene pairs when $|A_{ij}|_i\theta$, and we choose different values ($\theta \in [0, 1)$) for experiments. Then, we report the mean values of ARI and NMI in Figure 13.

From Figure 13, we can find the performance of scGGAN increases first and then decreases with the increase of θ. ARI and NMI get the maximum value at 0.3 and 0.4, respectively. When the threshold θ is too small, a large number of trivial relations will hinder the imputation performance. And we can find that when θ is greater than 0.6, the model performance drops sharply, this is because when the threshold θ is too large, a lot of gene-to-gene relation are ignored. When the threshold θ is within the appropriate range, reducing some trivial relations will help improve the performance of scGGAN.

Ablation experiment

We further design experiments on different bulk RNA-seq datasets with different cell types on GSE75748 dataset to study the impact of different cell-type bulk RNA-seq data on imputation performance. This dataset includes two cell types (H1 and DEC), we try to use only bulk H1 cell or DEC cell data to construct gene relation network. In addition, we report the mean values of ARI and NMI under different bulk datasets and also report the identification results of different cell types (which will be evaluated by the recall: $\frac{True\ Positive}{True\ Positive+False\ Negative}$) in Table 3, where H1/DEC-recall represents that H1/DEC cells are regarded as positive examples.

From results in Table 3, we can find that bulk RNA-seq data can effectively improve the clustering performance, which proves that the bulk homogeneous information can improve the performance of scGGAN. We also find that if only the part of bulk RNA-seq data (different cell types) are used, the recall of used cell types will increase, but the recall of other cell types will decrease. This is because bulk RNA-seq data of chosen cell types mislead the model to identify more cells as these cell types. Therefore, bulk RNA-seq data can provide homogeneous gene-to-gene-related information and improve the imputation performance of scGGAN. In the best case, scGGAN is designed using corresponding bulk RNA-seq data for different cell types and it can obtain the best results. However, it is difficult to achieve in practical applications. Therefore, scGGAN usually uses all bulk RNA-seq data from single cell types to construct imputation model.

Scalability and efficiency

To verify the scalability and efficiency of the proposed scGGAN, we test it on different simulated datasets and record the runtimes. Specifically, we use the '*Splatter*' package to generate six simulated datasets, which respectively contain 5k cells with 1k, 2k, 5k, 10k, 15k, 20k genes and explore the relationship between runtime and number of genes.

The results on six simulated datasets with different numbers of genes are shown in Figure 14. The runtime for training exponentially increases with respect to the number of genes. Since scGGAN is an imputation method based on gene relation network, its runtime increases with the number of genes. Based on the approximate time complexity, the cell similarity-based methods are exponentially related to the number of cells. In practical applications, the number of genes is limited, the number of cells is unlimited. In addition, scGGAN can be trained by batch training mode, and its runtime increases linearly with the number of cells. Therefore, scGGAN is more suitable than other cell similarity methods for large scRNA-seq datasets [17, 18, 21].

Discussion

Compared with bulk RNA-seq data, the dropout events are much more prevalent in scRNA-seq, resulting in a non-negligible impact

on the scRNA-seq data analysis and application. In this paper, we propose a novel scRNA-seq imputation method called scGGAN. Taking into account that the expression of gene will be regulated by its related genes, scGGAN integrates bulk RNA-seq homogeneous information and single-cell heterogeneous information to construct the gene relation network. Based on the composite gene network, scGGAN leverages a GCN-based GAN to mine gene-to-gene relation and real scRNA-seq distribution by integrating multi-omics data, and then generates 'dropout' values to impute the raw scRNA-seq data. scGGAN can more effectively recover gene expression values and improve the performance of various downstream analysis tasks. To validate the effectiveness of scGGAN, we compared it with some state-of-the-art methods on simulated and real scRNA-seq datasets. The extensive experimental results demonstrate that our scGGAN outperforms most of the advanced imputation methods on these scRNA-seq datasets. For future work, we will fuse other single-cell multi-omics data (i.e. epigenetics and spatial transcriptomics) to more credibly impute scRNA-seq data.

Key Points

- We propose a novel GCN-based GAN model (scGGAN) for scRNA-seq data imputation. scGGAN can learn regional gene-to-gene relations by GCN model and global real scRNA-seq data distribution by GAN model, and leverages gene sequence data to guide the imputation.
- scGGAN integrates bulk RNA homogeneous and single-cell heterogeneous gene-to-gene-related information to construct the gene relation network and guide the imputation of missing gene expression values.
- scGGAN has significantly better performance on recovering dropout gene expression values and achieves better results in unbalanced scRNA-seq data. scGGAN outperforms other imputation methods in different downstream tasks (i.e. cell clustering, differential expression analysis and trajectory inference).

Data and Code availability

For scGGAN, we conduct experiments on four different real scRNA-seq datasets. The 1st dataset is the human embryonic stem cells (H1 ESC) and differentiated definitive endoderm cells (DEC), which can be downloaded from NCBI Gene Expression Omnibus (GEO access number: GSE75748, scRNA-seq Technique is Fluidigm C1) [44]. The 2nd dataset is obtained from mouse embryonic stem cells [45] (GEO access number: GSE65525, scRNA-seq protocol is CEL-seq), which were measured to analyze the heterogeneity of mouse embryonic stem cells in different stages after leukemia inhibitory factor (LIF) withdrawal. We selected four different LIF withdrawal intervals (0, 2, 4, 7 days) with 935, 301, 682 and 799 cells, respectively. The 3rd dataset is the human brain cells, including 420 cells in eight cell types, which can be downloaded from NCBI (GEO access number: GSE67835, scRNA-seq Technique is SMARTer) [46]. scGGAN was further applied to a large dataset generated by the 10X scRNA-seq platform (scRNA-seq protocol is 10X Genomics) [47], which is involved by the scRNA-seq of peripheral blood mononuclear cells (PBMCs) from a healthy donor. The dataset contains 5247 PBMCs and 11 cell types. We pack the source code into a well-documented package at https://www.sdu-idea.cn/codes.php?name=scGGAN. In addition, a detailed **Readme** file is attached to guide the use and parameter setting of scGGAN.

Funding

National Natural Science Foundation of China (No. 62272276, 62072380); Fundamental Research Funds of Shandong University (2020GN061).

References

1. Quinn TP, Erb I, Richardson MF, *et al*.Understanding sequencing data as compositions: an outlook and review. *Bioinformatics* 2018;**34**(16):2870–8.
2. Conesa A, Madrigal P, Tarazona S, *et al*.A survey of best practices for RNA-seq data analysis. *Genome Biol* 2016;**17**(1):1–19.
3. Shapiro E, Biezuner T, Linnarsson S. Single-cell sequencing-based technologies will revolutionize whole-organism science. *Nat Rev Genet* 2013;**14**(9):618–30.
4. Patel AP, Tirosh I, Trombetta JJ, *et al*.Single-cell RNA-seq highlights intratumoral heterogeneity in primary glioblastoma. *Science* 2014;**344**(6190):1396–401.
5. Rashid S, Shah S, Bar-Joseph Z, *et al*.Dhaka: variational autoencoder for unmasking tumor heterogeneity from single cell genomic data. *Bioinformatics* 2021;**37**(11):1535–43.
6. Gawel DR, Serra-Musach J, Lilja S, *et al*.A validated single-cell-based strategy to identify diagnostic and therapeutic targets in complex diseases. *Genome Med* 2019;**11**(1):1–25.
7. Xiang J, Zhang J, Zhao Y, *et al*.Biomedical data, computational methods and tools for evaluating disease–disease associations. *Brief Bioinform* 2022;**23**(2):bbac006.
8. Cheng Jia YH, Kelly D, Kim J, *et al*.Accounting for technical noise in differential expression analysis of single-cell RNA sequencing data. *Nucleic Acids Res* 2017;**45**(19):10978–88.
9. Kharchenko PV, Silberstein L, Scadden DT. Bayesian approach to single-cell differential expression analysis. *Nat Methods* 2014;**11**(7):740–2.
10. Luecken MD, Theis FJ. Current best practices in single-cell RNA-seq analysis: a tutorial. *Mol Syst Biol* 2019;**15**(6):e8746.
11. Zhang Z, Cui F, Lin C, *et al*.Critical downstream analysis steps for single-cell RNA sequencing data. *Brief Bioinformatics* 2021;**22**(5):bbab105.
12. Bao S, Li K, Yan C, *et al*.Deep learning-based advances and applications for single-cell RNA-sequencing data analysis. *Brief Bioinformatics* 2022;**23**(1):bbab473.
13. Wang X, He Y, Zhang Q, *et al*.Direct comparative analyses of 10x genomics chromium and smart-seq2. *Genomics Proteomics Bioinformatics* 2021;**19**(2):253–66.
14. Elyanow R, Dumitrascu B, Engelhardt BE, *et al*.Netnmf-sc: leveraging gene–gene interactions for imputation and dimensionality reduction in single-cell expression analysis. *Genome Res* 2020;**30**(2):195–204.
15. Junlin X, Cai L, Liao B, *et al*.Cmf-impute: an accurate imputation tool for single-cell RNA-seq data. *Bioinformatics* 2020;**36**(10):3139–47.
16. Yinlei H, Li B, Zhang W, *et al*.Wedge: imputation of gene expression values from single-cell RNA-seq datasets using biased matrix decomposition. *Brief Bioinformatics* 2021;**22**(5):bbab085.
17. Van Dijk D, Sharma R, Nainys J, *et al*.Recovering gene interactions from single-cell data using data diffusion. *Cell* 2018;**174**(3):716–29.

18. Li WV, Li JJ. An accurate and robust imputation method scimpute for single-cell RNA-seq data. *Nat Commun* 2018;**9**(1):1–9.
19. Chenyang X, Cai L, Gao J. An efficient scRNA-seq dropout imputation method using graph attention network. *BMC Bioinformatics* 2021;**22**(1):1–18.
20. Xiaobin W, Zhou Y. Ge-impute: graph embedding-based imputation for single-cell rna-seq data. *Brief Bioinformatics* 2022;**23**(5):bbac313.
21. Huang M, Wang J, Torre E, *et al*.Saver: gene expression recovery for single-cell RNA sequencing. *Nat Methods* 2018;**15**(7): 539–42.
22. Eraslan G, Simon LM, Mircea M, *et al*.Single-cell RNA-seq denoising using a deep count autoencoder. *Nat Commun* 2019;**10**(1): 1–14.
23. Linderman GC, Zhao J, Roulis M, *et al*.Zero-preserving imputation of single-cell RNA-seq data. *Nat Commun* 2022;**13**(1):1–11.
24. Gan Y, Huang X, Zou G, *et al*.Deep structural clustering for single-cell RNA-seq data jointly through autoencoder and graph neural network. *Brief Bioinformatics* 2022;**23**(2):bbac018.
25. Kiviet DJ, Nghe P, Walker N, *et al*.Stochasticity of metabolism and growth at the single-cell level. *Nature* 2014;**514**(7522): 376–9.
26. Wimmers F, Subedi N, van Buuringen N, *et al*.Single-cell analysis reveals that stochasticity and paracrine signaling control interferon-alpha production by plasmacytoid dendritic cells. *Nat Commun* 2018;**9**(1):1–12.
27. Goodfellow IJ, Pouget-Abadie J, Mirza M, *et al*.Generative adversarial nets. In *Adv Neural Inform Process Syst* 2014;2672–80.
28. Wang X, Dizaji KG, Huang H. Conditional generative adversarial network for gene expression inference. *Bioinformatics* 2018;**34**(17): i603–11.
29. Marouf M, Machart P, Bansal V, *et al*.Realistic in silico generation and augmentation of single-cell rna-seq data using generative adversarial networks. *Nat Commun* 2020;**11**(1):1–12.
30. Yungang X, Zhang Z, You L, *et al*.Scigans: single-cell RNA-seq imputation using generative adversarial networks. *Nucleic Acids Res* 2020;**48**(15): e85–5.
31. Kipf TN, Welling M. Semi-supervised classification with graph convolutional networks. In *Int Conf Learn Representations*, poster, 2017.
32. Spinelli I, Scardapane S, Uncini A. Missing data imputation with adversarially-trained graph convolutional networks. *Neural Netw* 2020;**129**:249–60.
33. Welch JD, Kozareva V, Ferreira A, *et al*.Single-cell multi-omic integration compares and contrasts features of brain cell identity. *Cell* 2019;**177**(7):1873–87.
34. Yunjin Li L, Ma DW, Chen G. Advances in bulk and single-cell multi-omics approaches for systems biology and precision medicine. *Brief Bioinform* 2021;**22**(5):bbab024.
35. Yoon J, Jordon J, and Schaar M. Gain: missing data imputation using generative adversarial nets. In *Int Conf Mach Learn*, 5689–98, 2018.
36. Lee D, Kim J, Moon W-J, and Ye JC. Collagan: Collaborative gan for missing image data imputation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2487–96, 2019.
37. Shen J, Zhang J, Luo X, *et al*.Predicting protein–protein interactions based only on sequences information. *Proc Natl Acad Sci* 2007;**104**(11):4337–41.
38. Ji Y, Zhou Z, Liu H, *et al*.Dnabert: pre-trained bidirectional encoder representations from transformers model for dna-language in genome. *Bioinformatics* 2021;**37**(15):2112–20.
39. Rives A, Meier J, Sercu T, *et al*.Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *Proc Natl Acad Sci* 2021;**118**(15): e2016239118.
40. Nitzan M, Karaiskos N, Friedman N, *et al*.Gene expression cartography. *Nature* 2019;**576**(7785):132–7.
41. Schober P, Boer C, Schwarte LA. Correlation coefficients: appropriate use and interpretation. *Anesthesia Analgesia* 2018;**126**(5): 1763–8.
42. Kendall A, Gal Y, and Cipolla R. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *IEEE Conf Comput Vis Pattern Recognit*, 7482–91, 2018.
43. Zappia L, Phipson B, Oshlack A. Splatter: simulation of single-cell rna sequencing data. *Genome Biol* 2017;**18**(1):1–15.
44. Chu L-F, Leng N, Zhang J, *et al*.Single-cell RNA-seq reveals novel regulators of human embryonic stem cell differentiation to definitive endoderm. *Genome Biol* 2016;**17**(1):1–20.
45. Klein AM, Mazutis L, Akartuna I, *et al*.Droplet barcoding for single-cell transcriptomics applied to embryonic stem cells. *Cell* 2015;**161**(5):1187–201.
46. Darmanis S, Sloan SA, Zhang Y, *et al*.A survey of human brain transcriptome diversity at the single cell level. *Proc Natl Acad Sci* 2015;**112**(23):7285–90.
47. Zheng GXY, Terry JM, Belgrader P, *et al*.Massively parallel digital transcriptional profiling of single cells. *Nat Commun* 2017;**8**(1): 1–12.
48. McInnes L, Healy J, Melville J. Umap: uniform manifold approximation and projection for dimension reduction. *J Open Source Softw* 2018;**3**(29): 861.
49. Ritchie ME, Belinda Phipson DI, Wu YH, *et al*.Limma powers differential expression analyses for rna-sequencing and microarray studies. *Nucleic Acids Res* 2015;**43**(7): e47–7.
50. Qiu X, Mao Q, Tang Y, *et al*.Reversed graph embedding resolves complex single-cell trajectories. *Nat Methods* 2017;**14**(10): 979–82.