InterLibrary Loan
Article Delivery Service
http://www.library.uthscsa.edu/illiad
ariel-ill@uthscsa.edu

Briscoe Library
7703 Floyd Curl Drive
San Antonio, TX 78229
(210) 567-2460
(210) 567-2463 (fax)

Desktop Delivery Cover Sheet

If you receive a copy that is missing pages, smudged, or unreadable, please contact the UT HSC Libraries at (210) 567-2460 so we may obtain a clean copy for you as quickly as possible.

Notice: Warning concerning copyright restrictions

WE MAKE LIVES BETTER
UT HEALTH SCIENCE CENTER™
SAN ANTONIO

Libraries
UT Health Science Center

OXFORD

Briefings in Bioinformatics, 2022, 1–14

https://doi.org/10.1093/bib/bbac313

Problem Solving Protocol

GE-Impute: graph embedding-based imputation for single-cell RNA-seq data

Xiaobin Wu and Yuan Zhou

Corresponding author: Yuan Zhou, Department of Biomedical Informatics, Center for Noncoding RNA Medicine, School of Basic Medical Sciences, Peking University, 38 Xueyuan Rd, Haidian District, Beijing 100191, China. Tel.: 86-10-82801585; E-mail: zhouyuanbioinfo@hsc.pku.edu.cn

Abstract

Single-cell RNA-sequencing (scRNA-seq) has been widely used to depict gene expression profiles at the single-cell resolution. However, its relatively high dropout rate often results in artificial zero expressions of genes and therefore compromised reliability of results. To overcome such unwanted sparsity of scRNA-seq data, several imputation algorithms have been developed to recover the single-cell expression profiles. Here, we propose a novel approach, GE-Impute, to impute the dropout zeros in scRNA-seq data with graph embedding-based neural network model. GE-Impute learns the neural graph representation for each cell and reconstructs the cell–cell similarity network accordingly, which enables better imputation of dropout zeros based on the more accurately allocated neighbors in the similarity network. Gene expression correlation analysis between true expression data and simulated dropout data suggests significantly better performance of GE-Impute on recovering dropout zeros for both droplet- and plated-based scRNA-seq data. GE-Impute also outperforms other imputation methods in identifying differentially expressed genes and improving the unsupervised clustering on datasets from various scRNA-seq techniques. Moreover, GE-Impute enhances the identification of marker genes, facilitating the cell type assignment of clusters. In trajectory analysis, GE-Impute improves time-course scRNA-seq data analysis and reconstructing differentiation trajectory. The above results together demonstrate that GE-Impute could be a useful method to recover the single-cell expression profiles, thus enabling better biological interpretation of scRNA-seq data. GE-Impute is implemented in Python and is freely available at https://github.com/wxbCaterpillar/GE-Impute.

Keywords: single-cell RNA-sequencing, imputation, graph embedding, neural graph representation, similarity network

Introduction

Single-cell RNA-sequencing (scRNA-seq) has emerged as a powerful technique to characterize cellular heterogeneity, advancing our understanding of human disease by measuring gene expression and transcriptome states at the single-cell resolution [1–3]. Based on the protocols for single-cell library generation, the methods for scRNA-seq can be summarized into two categories: 1) the plate-based methods, which sort one single cell into one well of multiple-well plate, such as Fluidigm C1 [4] and Smart-Seq2 [5]; 2) the droplet-based methods, which distribute each cell into a tiny droplet containing reagents and a specific barcode to uniquely quantify the transcriptome, such as 10x Genomics [6]. The plate-based methods are often of lower throughput but higher sensitivity that enables the detection of more genes for each cell, while the droplet-based are of higher throughput but lower sensitivity in comparison with the plate-based methods. Despite rapid growth in the scale and robustness of the scRNA-seq protocols, drop-out events (i.e. missed detection of gene expression which results in artificial zero expressions of many genes) in scRNA-seq data have remained as the major obstacle in downstream functional analysis for either plate- or droplet-based scRNA-seq methods [7, 8]. Therefore, it is necessary to develop efficient algorithms to overcome this unwanted sparsity in single-cell expression matrix and recover the incomplete expression profiles.

Recently, several computational methods have been established to impute the dropout zeros in scRNA-seq data. Generally, these computational methods can be categorized into three classes [9]. The first class consists of methods that focus on smoothing all expression values among the cells with similar expression profiles, such as MAGIC [10], kNN-smoothing [11] and DrImpute [12]. MAGIC imputes the dropout values of the scRNA-seq count matrix through data diffusion across similar cells. kNN-smoothing reconstructs the count matrix for each cell by smoothing the expression values of its k-nearest neighbors. DrImpute first performs cell clustering to identify similar cells and further imputes data by averaging the expression values from similar cells. The second class of methods reconstructs the expression matrix from the latent spaces estimated by low-rank matrix-based methods or deep-learning methods, like WEDGE [13], scScope [14], DeepImpute [15], scVI [16] and scGNN [17]. WEDGE is a recently proposed algorithm to impute gene expression matrix by using biased low-

Xiaobin Wu is a PhD candidate at Department of Biomedical Informatics, Peking University. His research interest is single-cell omics data modeling and analysis. **Yuan Zhou** is an associate investigator at Department of Biomedical Informatics, Peking University. His research interest includes transcriptomic and epitranscriptomic bioinformatics.

Received: April 13, 2022. **Revised:** June 27, 2022. **Accepted:** July 11, 2022

rank matrix decomposition method. scScope is a deep-learning-based method that employs a recurrent network layer to iteratively impute scRNA-seq data matrix. DeepImpute and scVI both apply a deep neural network to learn expression patterns of the scRNA-seq data, thus allowing for fast imputation of missing values. scGNN is a graph neural network-based imputation method which learns cell–cell relationships in the graph autoencoder. The third class of methods models the sparsity using probabilistic models, such as SAVER [18]. SAVER imputes the gene expression of cells by estimating prior parameters for an empirical Bayes-like method with Poisson least absolute shrinkage and selection operator (LASSO) regression model.

It is noteworthy that there are multiple aspects when assessing the performance of an imputation method. First, the primary objective of the scRNA-seq imputation methods is to recover the real expression profiles. Since there is no golden-standard single-cell expression matrix, one alternative approach is to randomly mask non-zero expression values to simulate the dropout zeros in scRNA-seq datasets and assess the similarity between imputation data and true background data. Moreover, there are several important tasks when analyzing scRNA-seq data, including identification of differentially expressed genes, unsupervised clustering of cells, marker genes-based cell-type annotation and trajectory analysis. Therefore, how one imputation method could facilitate these downstream analysis tasks is also of prominent biological significance. One recent benchmarking study [19] has systematically evaluated the imputation methods for recovering biological signals in downstream analysis. In this evaluation, MAGIC, SAVER and kNN-smoothing have outperformed other imputation methods in denoising scRNA-seq data. Nonetheless, the ability of these methods to improve the quality of analysis in all aspects is still lacking. Therefore, more robust methods should be developed to improve the feasibility of data analysis while preserving the original information of single-cell data as much as possible.

Here, we propose a new method, GE-Impute, to impute singe-cell data matrix based on cell–cell similarity links predicted by graph embedding neural network. We first constructed a raw cell–cell similarity network (graph) and embedded all cells into low-dimension vectors using biased random walks and skip-gram model [20, 21]. After training feature embeddings, the new links between cells were predicted based on the embedded low-dimension features to obtain a reconstructed cell–cell similarity network. Finally, dropout zero values for each cell were estimated by smoothing the expression values of all neighbors in the reconstructed cell–cell similarity network. We applied GE-Impute on computer-simulated, droplet- and plate-based scRNA-seq data and compared it with other nine state-of-the-art methods, including MAGIC [10], kNN-smoothing [11], DrImpute [12], WEDGE [13], scScope [14], DeepImpute [15], scVI [16], scGNN [17] and SAVER [18], and found GE-Impute outperformed other imputation methods across multiple evaluations on data quality and analysis feasibility. Sections below will firstly describe the method framework of GE-Impute, and then provide the detailed performance evaluation results.

Methods

The design of GE-Impute pipeline

Graph embedding (or graph representation learning) emerges as a promising technique in various machine learning tasks, such as node classification, link prediction and community detection [22]. In recent years, the graph embedding method has been further applied to several important biological issues. For example, Zhao et al. [23] performed graph embedding on a heterogeneous network to predict novel drug-disease associations. Zhang et al. [24] utilized a deep learning model of graph convolution network and graph factorization to predict the potential association of circRNA and disease. In this study, we applied node2vec [21] graph embedding algorithm which simultaneously considering breadth-first sampling (BFS) and depth-first sampling (DFS) search strategies for random walks sampling. The skip-gram model is a neural network to create a word vector and is widely used in natural language processing. We further applied skip-gram model to learn continuous feature representations for cells in the raw cell–cell similarity network based on the sampling walks. Since there are multiple effective graph representation learning methods, to test if other models work better than node2vec, we have also tried four other commonly used graph embedding methods including DeepWalk [22], LINE [25], Struct2vec [26] and Snore [27]. We compared the Pearson correlation coefficients calculated by different models using 10x Genomics cell lines dataset (see sections below) and found node2vec performs best in missing values recovering analysis (Supplementary Figure 1**A**). To further evaluate the accuracy of new links predicted by different methods in the cell–cell similarity network, we calculated the ratio of true links (i.e. links within the same cell type) to total predicted links (Supplementary Figure 1**B**). The result shows that node2vec and DeepWalk predict 100% links of the same cell type while other methods may predict the false links (i.e. links between the different cell types). Since DeepWalk has taken more time and memory cost than node2vec when learning feature representation. We have decided to select node2vec as the graph representation learning neural network model to develop our imputation task. Taking advantage of the node2vec algorithm, GE-Impute mapped cells into a low-dimension space and maximized the likelihood of co-occurrence of their neighbors in network. The similarities among cells could be re-calculated from low-dimension feature representations to predict new link-neighbors for the cells and reconstruct cell–cell similarity network. Finally, imputation for scRNA-seq

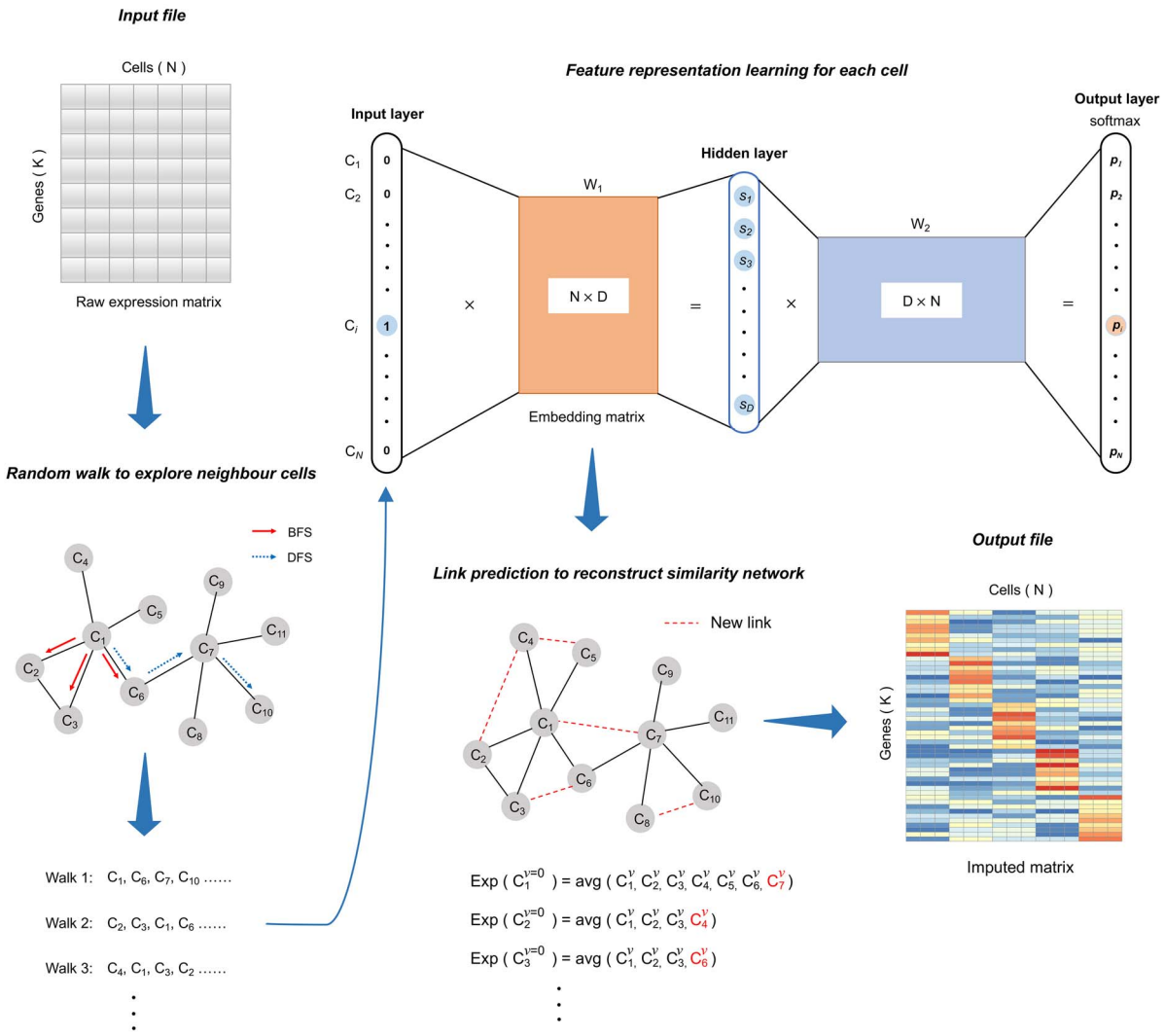

Figure 1. Overall workflow of GE-Impute algorithm pipeline. GE-Impute constructs a raw cell–cell similarity network based on Euclidean distance. For each cell, it simulates a random walk of fixed length using BFS and DFS strategy. Next, graph embedding-based neural network model was employed to train the embedding matrix for each cell based on sampling walks. The similarity among cells could be re-calculated from embedding matrix to predict new link-neighbors for the cells and reconstruct cell–cell similarity network. Finally, GE-Impute imputes the dropout zeros for each cell by averaging the expression value of its neighbors in reconstructed similarity network.

expression data matrix was implemented based on the reconstructed cell–cell similarity network. The workflow of GE-Impute is summarized in Figure 1.

More specifically, we firstly normalized the raw scRNA-seq count matrix using Freeman-Tukey transform to reduce the technical variance.

$$Y = \sqrt{X} + \sqrt{X+1}$$

where X denotes raw expression value and Y represents normalized value. The Freeman-Tukey transform [28] was proposed to stabilize the variance of Poisson-distributed variables and was verified to outperform the regular logarithm transcript per million (log-TPM) transform when calculating cell–cell distance [11]. The Euclidean distances between cells were calculated and adjacency matrix of raw cell–cell similarity network was established based on the k nearest neighbors (KNN) of each cell:

$$W_{ij} = W_{ji} = \begin{cases} 0 & C_i \notin \boldsymbol{KNN}\left(\boldsymbol{C_j}\right) \text{ and } C_j \notin \boldsymbol{KNN}\left(\boldsymbol{C_i}\right) \\ 1 & C_i \in \boldsymbol{KNN}\left(\boldsymbol{C_j}\right) \text{ or } C_j \in \boldsymbol{KNN}\left(\boldsymbol{C_i}\right) \end{cases}$$

As for the sampling strategy, the biased random walk was used to explore the neighbors considering both breadth-first and depth-first sampling strategy (Figure 1). Let $G = (V, E)$ be the raw cell–cell similarity network. Given a source node u, the IM_i was defined as the ith intermediate cell in sampling walks of given length L, the transition probability is defined as follows:

$$P\left(IM_i = x \mid IM_{i-1} = v\right) = \begin{cases} \frac{w_{vx}}{Z}, \text{if } (v,x) \in \boldsymbol{E} \\ 0, \text{otherwise} \end{cases}$$

where w_{vx} denotes transition probability between cell v and cell x, and Z is the normalizing constant. To achieve moderate sampling strategy, two parameters p and q were used to calculate the bias of walk. Let t be the upper cell of v and suppose walks just traversed edge (t, v). From t to x, the α is defined as follows:

$$\alpha_{pq}(t,x) = \begin{cases} 1/p \text{ if } d_{tx} = 0 \\ 1 \text{ if } d_{tx} = 1 \\ 1/q \text{ if } d_{tx} = 2 \end{cases}$$

where the $w_{vx} = \alpha_{pq}$ and d_{tx} denotes the distance between cell t and cell x. The values p and q control the bias of walks. If $p > 1$, the walk strategy is biased toward searching cells away from t. If $p < 1$, the walk strategy is biased toward revisiting t. If $q > 1$, the walks strategy is biased toward breadth-first sampling. If $q < 1$, the walks strategy is biased toward depth-first sampling. By default, p and q are set to 0.25 and 4, respectively, according to the preliminary optimization. After sampling walks from similarity network, GE-Impute trained features representation for each cell using skip-gram model (Figure 1). The model aims to optimize the following objective function:

$$\max_{S} \sum_{u \in \mathbf{V}} \log \Pr\left(N_g(u) \mid S(u)\right)$$

where $S(u)$ is the mapping function from cells to feature representations, $N_g(u)$ is defined as the neighborhood set of node u deriving from sampling function g. To make this optimization solvable, the assumptions of conditional independence and symmetry in feature space were proposed, which could simplify the objective function [29] as follows:

$$\max_{S} \sum_{u \in \mathbf{V}} \left[-\log Z_u + \sum_{v_i \in \boldsymbol{N_g(u)}} S(v_i) \bullet S(u) \right],$$

$$\text{where } Z_u = \sum_{w \in \mathbf{V}} e^{S(u) \cdot S(w)}$$

The learning features were used to predict new links and therefore reconstruct cell–cell similarity network. In a detailed manner, for one cell i, the distances to all linked cells were calculated. Let E_i represent the number of its neighbor links with other cells in the initial adjacency matrix W. We ranked the distance scores in ascending order and the top E_i neighbors of cell i were described as 'features-related neighbors' ($N_{feature}$). By further combining with W, a union of neighbor links of cell i was used to impute its dropout values. For a specific gene y that got 0 value in the raw expression data matrix, GE-Impute filled it by averaging the expression data of gene y in its neighbors:

$$\text{Expression}\left(C_i^{y=0}\right) = \text{average}\left\{\text{Expression}\left(C_n^{y}\right)\right\}$$

$$\text{where } C_n \in \mathbf{N_{feature}} \cup \mathbf{V_j}\left(\mathbf{W_{i,j}} = \mathbf{1}\right)$$

To demonstrate the improvement of GE-Impute by the graph neural network model, we compared the performance of GE-Impute with the original similarity network derived from KNN algorithm as well as the raw features from node2vec. The result shows that the raw KNN similarity network or the raw learning features obtained by node2vec individually cannot perform as well as GE-Impute in missing values recovering analysis on either 10x Genomics dataset or Fludigm C1 dataset (Supplementary Figure 2). Moreover, to test if exclusion outlier cells when averaging expression values would help improve the imputation performance. The interquartile range metric (IQR) is used to define outlier cells. Instead of averaging all the similar cells, the cells whose expression levels are more than Median + 1.5*IQR or less than Median - 1.5*IQR are removed and the average expression value is calculated by the remaining cells. However, we cannot observe a significant improvement in the missing value recovering analysis (Supplementary Figure 3). Therefore, to simplify the algorithm, we did not consider adding this procedure to our imputation model.

For parameter setting in GE-Impute, we have run an optimization for p (i.e. the bias of walks), q (i.e. the bias of walks), L (i.e. the length of each random walk), N_W (i.e. the number of random walks) and W_S (i.e. the window size). For p and q, different combinations of values (range of 0.25, 0.5, 1, 2, 4) are used to explore the most suitable combination of p and q (Supplementary Table 1). For parameters L, N_W and W_S, we calculate performance in missing value recovering analysis when considering different range of values (Supplementary Table 1). According to the optimization results, we have determined the values of those parameters in GE-Impute model, with $p = 0.25$, $q = 4$, $L = 5$, $N_W = 20$, $W_S = 3$.

Dataset collection and imputation

The scRNA-seq and bulk RNA-seq datasets used to test the performance of GE-Impute are summarized in Supplementary Table 2. Notably, these datasets have been commonly used in previous studies and have proved to be effective for imputation methods benchmarking [19, 30]. To comprehensively evaluate the performance of GE-Impute for different scRNA-seq protocols, we considered datasets from droplet-based methods (e.g. 10x Genomics) and plate-based methods (e.g. Fluidigm C1 and Smart-Seq2). Several datasets were used to perform missing value recovering analysis and differentially expressed gene identification, including a 10x Genomics scRNA-seq data of five cell lines (i.e. A549, H1975, H2228, H838 and HCC828) [31] and a Fluidigm C1 scRNA-seq data of five cell lines (i.e. A549, GM12878, H1, K562 and IMR90) [32]. For the 10x Genomics scRNA-seq data, the corresponding bulk RNA-seq data including A549, H1975, H2228, H838 and HCC828 was downloaded

from GEO database [33], and for the Fluidigm C1 scRNA-seq data, the corresponding bulk RNA-seq data including A549, GM12878, H1, K562 and IMR90 was downloaded from ENCODE database [34] (Supplementary Table 2). One dataset containing six cell types of peripheral blood mononuclear cells (PBMC) [6] from 10x Genomics and one dataset including four conventional dendritic cell subtypes from Smart-Seq2 [35] were utilized to perform clustering analysis and marker genes visualization. One dataset which contains 1529 cells from five stages of human preimplantation embryonic development from E3 to E7 was used to perform trajectory analysis [36]. In addition to experimentally derived datasets, we also considered several scRNA-seq datasets simulated by splatSimulate function in Splatter R package [37]. One dataset including five cell groups without dropout rate was simulated to perform missing value recovering analysis and two other datasets with batch effects were simulated to evaluate batch effect benchmark. For scRNA-seq data, only cells with at least 500 detected genes were retained and genes that expressed at least 10% of cells were retained to ensure the quality of data. We compared GE-Impute with several extensively used imputation methods, including DeepImpute, DrImpute, MAGIC, kNN-smoothing, SAVER, scGNN, scScope, scVI and WEDGE. All methods were implemented in R version 4.0.3 or Python version 3.8.8 with respective default parameters.

Comparison of imputation methods for dropout zero recovering and differentially expressed gene identification

To evaluate our imputation method for recovering the dropout zeros in scRNA-seq data, the similarity between imputation data and true background data was calculated based on Pearson correlation analysis. Firstly, we randomly mask 10%, 20% and 30% of non-zero values for each cell in 10x Genomics dataset, Fluidigm C1 dataset and Splatter-generated dataset to simulate the dropout events in scRNA-seq data. After imputation for the simulated dropout data, the raw data and imputed data are both adjusted for library size with NormalizeData function in Seurat 4.0 R package. The Pearson correlation coefficients for each cell between imputation data and true background data were calculated. To test the ability of GE-Impute on capturing and identifying differentially expressed genes (DEGs) among different cell states, we regarded DEGs identified by bulk RNA-seq data as the 'gold standard' gene set following the idea of the previous benchmarking [19]. We first identified DEGs between all pairs of cell types for bulk RNA-seq data using package DESeq2 [38] in R version 4.0.3. Genes with absolute value of log2-fold change >1 and adjusted *P*-value <0.05 were retained and considered as 'gold standard' DEGs sets from bulk data. For each pair of cell types in the (imputed) scRNA-seq data, the Seurat normalized log2-transformed expression profiles were used to identify DEGs. We applied the Wilcoxon Rank-Sum test [39] to calculate *P*-value for all genes and further corrected them using Benjamini-Hochberg method. Genes with absolute value of log2-fold change >0.5 or 1 and FDR < 0.05 were identified as the single-cell DEG sets. To evaluate the similarity between bulk-derived gold-standard DEGs and single-cell DEGs, the Jaccard index [40] was used to measure the amount of overlap between these two gene sets and was defined as follows:

$$\text{Jaccard}\,(\boldsymbol{B},\boldsymbol{S}) = \frac{|\boldsymbol{B}\cap\boldsymbol{S}|}{|\boldsymbol{B}\cup\boldsymbol{S}|}$$

where the *B* and *S* denote bulk-derived gold standard DEGs and single-cell DEGs, respectively.

Evaluation of GE-Impute for unsupervised clustering of cells

To investigate whether GE-Impute can outperform other methods in improving clustering of cells that belong to the same cell type or subtypes, we considered two datasets from 10x Genomics and Smart-Seq2 platform, respectively. The 10x Genomics dataset contains 61,213 sorted PBMCs [6] including CD14+ monocytes, CD19+ B cells, CD34+ cells, CD4+ T cells and CD8+ cytotoxic T cells; while the Smart-Seq2 dataset contains 957 conventional dendritic cells [35] with four predefined subtypes (i.e. blood pre-cDCs, cord pre-cDCs, CD141+ cDC and CD1c + cDC). We employed Seurat 4.0 pipeline [41], the most commonly used scRNA-seq data analysis pipeline, to perform cell clustering for the imputed expression matrix of each method. Briefly, the expression profiles were first normalized using NormalizeData function with default parameters. Then highly variant genes were identified using FindVariableGenes function and scaled by ScaleData function. The top 30 significant principal components were selected to perform Louvain clustering using FindNeighbors function and FindClusters function. For a comparable configuration, we adjusted the resolution parameter of FindClusters function until the number of clusters reaches the same number of the predefined cell types or subtypes. For different imputation methods, the expression characteristics of imputed data are different, so their final resolutions to get the same number of clusters are also different. The exact resolution parameters of clustering for each method are summarized in Supplementary Table 4.

Purity, Adjust Rand Index (ARI) and Normalized Mutual Information (NMI) are commonly used indices to compare clustering results against known labels. Therefore, the cluster labels and known cell (sub)type labels were employed to evaluate the performance of imputation method in improving unsupervised clustering. The Purity was defined as the percent of the total number of cells that were classified correctly and was implemented by purity function in NMF package [42]. Let *K* be the number of clusters inferring by *N* cells, P_i be the

cluster i and T_j be the true cell (sub)type j. Formally:

$$\text{Purity}(P,T) = \frac{1}{N}\sum_{i=1}^{K}\max_j |P_i \cap T_j|$$

The Adjust Rand Index aims to calculate similarity measure between two clustering results by counting pairs of cells that are assigned to the same or different clusters in the predicted and true clustering results:

$$\text{ARI}(P,T) = \frac{\sum_{i,j}\binom{N_{ij}}{2} - \left[\sum_i \binom{N_i}{2}\sum_j \binom{N_j}{2}\right]/\binom{N}{2}}{\frac{1}{2}\left[\sum_i \binom{N_i}{2} + \sum_j \binom{N_j}{2}\right] - \left[\sum_i \binom{N_i}{2}\sum_j \binom{N_j}{2}\right]/\binom{N}{2}}$$

where N_{ij} denotes the number of cells of the cell type label T_j assigned to cluster P_i. N_i is the number of cells in cluster P_i, while N_j is the number of cells in cell type T_j. The ARI was calculated using the adjustedRandIndex function in mclust package [43]. The Normalized Mutual Information (NMI) is also a good measure for estimating clustering quality and is implemented by NMI function in aricode package (https://github.com/jchiquet/aricode). Let L be the number of true cell (sub)type labels and the NMI is defined as:

$$\text{NMI}(P,T) = \frac{\sum_{i=1}^{L}\sum_{j=1}^{K} N_{ij}\ \log\frac{N \bullet N_{ij}}{N_i \bullet N_j}}{\max\left(-\sum_{i=1}^{L} N_i \bullet \log\frac{N_i}{N}, -\sum_{j=1}^{K} N_j \bullet \log\frac{N_j}{N}\right)}$$

where the numerator denotes the mutual information between P and T and the denominator denotes the entropy of P and T.

Results

GE-Impute shows effective improvement in recovering missing value in scRNA-seq data

To evaluate the ability of GE-Impute in imputing the missing value of scRNA-seq expression data, nine other state-of-the-art imputation methods including DeepImpute, MAGIC, kNN-smoothing, SAVER, scGNN, scVI and WEDGE, DrImpute and scScope were used to perform comparison analysis. Three datasets generated from droplet-based (10x Genomics), plate-based (Fluidigm C1) and Splatter-generated protocols are used to systematically evaluate the performance of GE-Impute on various scRNA-seq data. We simulated the dropout events in scRNA-seq data by randomly masking 10%, 20% and 30% of non-zero values for each cell. The three simulated dropout datasets were first imputed to follow each method's guideline (see Methods). Pearson correlation coefficients (PCCs) between true background data and imputation data were calculated to measure the difference, where larger PCCs indicate the better performance of the imputation method. As shown in Figure 2, GE-Impute has shown excellent performance in recovering the missing value and provides higher PCCs than any other methods in all cell lines (groups) of three datasets. We observed that DrImpute performed better on 10x Genomics dataset imputation than on Fluidigm C1 and simulated dataset, while DeepImpute performed better on Fluidigm C1 and simulated dataset than on 10x Genomics dataset, indicating these two methods are applicable to scRNA-seq data from different protocols. We also found that several methods show more compromised performance in missing value imputation such as scGNN and scScope. scGNN utilized the imputation autoencoder and pre-processed matrix to recover gene expression matrix which may lead to an exaggerated deviation between raw data matrix and imputation data matrix. scScope allows the recurrent network layer to perform imputation on dropout entries iteratively, which may overcorrect the raw expression data. Overall, GE-Impute can successfully recover missing value in scRNA-seq data and obtain an imputed matrix similar to real data matrix.

GE-Impute promotes correct identification of differentially expressed genes in downstream analysis

One of the important tasks in scRNA-seq downstream analysis is to identify cell type-specific DEGs under various conditions [44] (i.e. healthy versus disease samples). Through DEG analysis, one can further explore which biological pathways related to the variation between cells under different conditions. Therefore, accurate acquisition of DEGs in the context of dropout noises is one of the hallmarks to demonstrate the biological significance of imputation results. Here, considering the higher sensitivity of bulk RNA-seq technology in detecting differential expression at the transcriptome scale, DEGs that were calculated based on bulk RNA-seq data were treated as the "gold standard". The DEGs of bulk RNA-seq and scRNA-seq data were determined following the method described above (see Methods). Also, we compared GE-Impute with other imputation methods for capturing DEGs of bulk RNA-seq. The Jaccard index was used to measure the overlap between DEGs from scRNA-seq data and bulk RNA-seq data. We also measured the performance of raw data (no imputation) in identifying DEGs of bulk RNA-seq as the baseline. As a result, GE-Impute can significantly improve the performance of DEGs identification compared with other imputation methods as well as the no imputation baseline when considering different fold change thresholds (Figure 3 and Supplementary Figure 4). In 10x Genomics dataset, GE-Impute can improve the identification of DEGs in all pairs of cell types compared with no imputation baseline. While kNN-smoothing and DrImpute can only improve several pairs of cell types. In Fluidigm C1 dataset, in addition to GE-Impute, scVI can also facilitate DEGs identification compared with the no imputation baseline. Since the identification of DEGs has a great impact on downstream analysis, it is crucial to reduce false positives and false negatives due to the technical noises. In our results, GE-Impute can significantly promote the

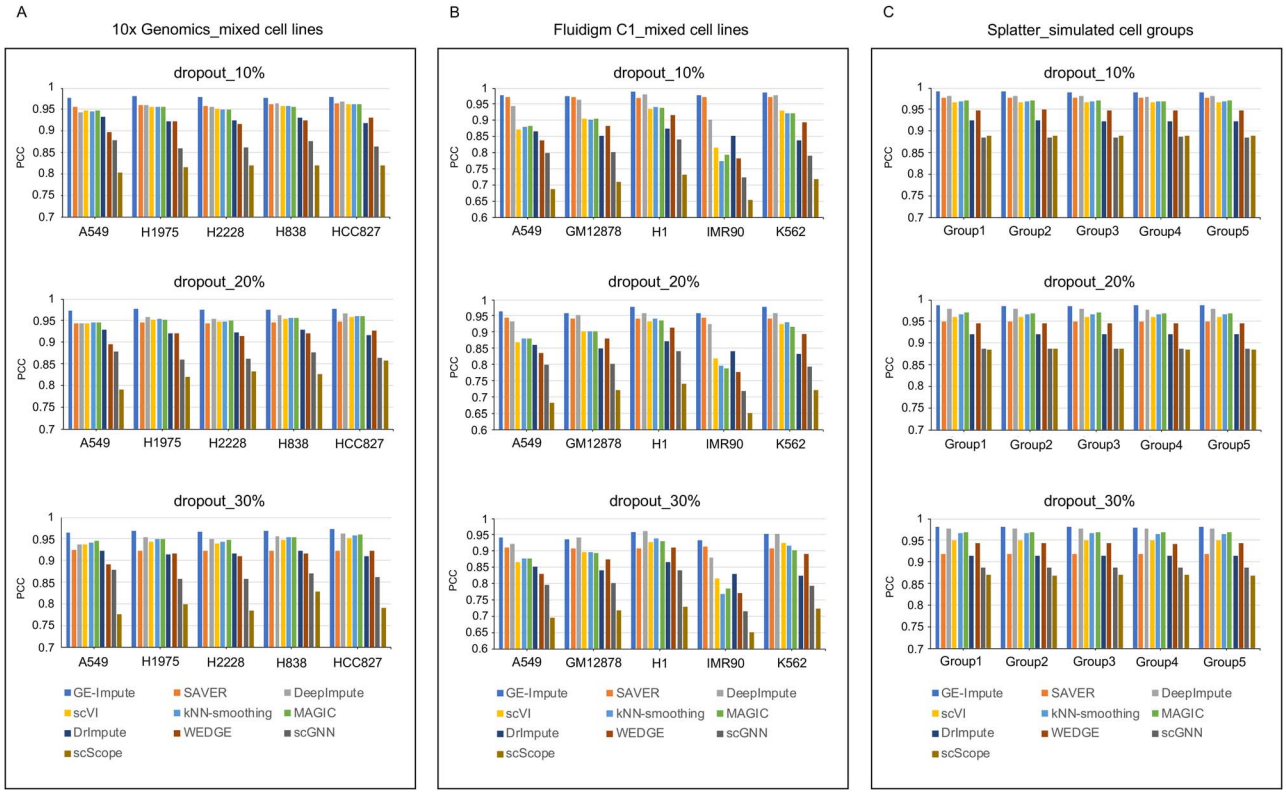

Figure 2. Performance comparison of GE-Impute with other imputation methods in recovering missing values in scRNA-seq data. The barplot shows a comparison of Pearson correlation coefficients between real and imputed expression profiles between different imputation methods on 10x Genomics dataset (**A**), Fluidigm C1 dataset (**B**) and simulated dataset (**C**), respectively. Different colors represent different imputation methods and each cell line is grouped accordingly.

identification of DEGs from bulk RNA-seq, indicating its potential in single cell RNA-seq analysis.

GE-Impute significantly improves the performance of unsupervised clustering of cells

Unsupervised clustering is essential for defining cell type heterogeneity and cell type annotation in scRNA-seq data analysis [45]. Mapping unbiased clusters to known cell types is one of the commonly used methods for cell type annotation, thus the clustering result would directly affect the accuracy of downstream interpretation [46]. Accordingly, we used the standard Seurat pipeline to cluster cells, and compared GE-Impute with other imputation methods on improving the performance of unsupervised clustering (see Methods). In this part of analysis, the known cell types or cell subtypes were regarded as the true labels and the clustering results were treated as predicted labels. Three indices were introduced to evaluate the consistency between the true and predicted labels, including Purity, ARI and NMI (see Methods). We first explored the effect of GE-Impute on droplet-based 10x Genomic PBMC dataset including CD14+ monocytes, CD19+ B cells, CD34+ cells, CD56+ cells, CD4+ T cells and CD8+ /cytotoxic T cells. The clustering results were visualized by uniform manifold approximation and projection (UMAP) method. In the result of no imputation data, several CD8 T cells (Cluster 1) are dispersedly distributed on UMAP plot and show prominent discrepancy between unsupervised clustering labels and true cell types labels (Figure 4**A**). Whereas in GE-Impute imputation's UMAP plot, the same cell types are more cohesively distributed and show better consistency between unsupervised clustering results and true cell types labels (Figure 4**B**). The Cluster 1 and Cluster 5 are dominated by CD8 T cells and CD4 T cells, respectively, though they distribute very closely to each other on UMAP plot. Moreover, the clustering results of all imputation methods on 10x Genomics data were quantitatively evaluated using the above-mentioned three indices (Figure 4**C** and Table 1). In general, most imputation methods can improve the unsupervised clustering compared with no imputation (Purity = 0.757, ARI = 0.563, NMI = 0.674), except kNN-smoothing (Purity = 0.809, ARI = 0.425, NMI = 0.552) and scScope (Purity = 0.755, ARI = 0.578, NMI = 0.649) which show reduced performance for one or more indices. The result also shows that GE-Impute could achieve the best clustering accuracies among all imputation methods, with Purity = 0.972, ARI = 0.936 and NMI = 0.894. Meanwhile, WEDGE (Purity = 0.968, ARI = 0.927, NMI = 0.886) and DrImpute (Purity = 0.965, ARI = 0.899, NMI = 0.854) also performed well in improving the clustering accuracy, although their performances in missing value recovering and differential gene identification are not such satisfactory, suggesting these methods are particularly suitable for cell clustering analysis. The clustering accuracy can

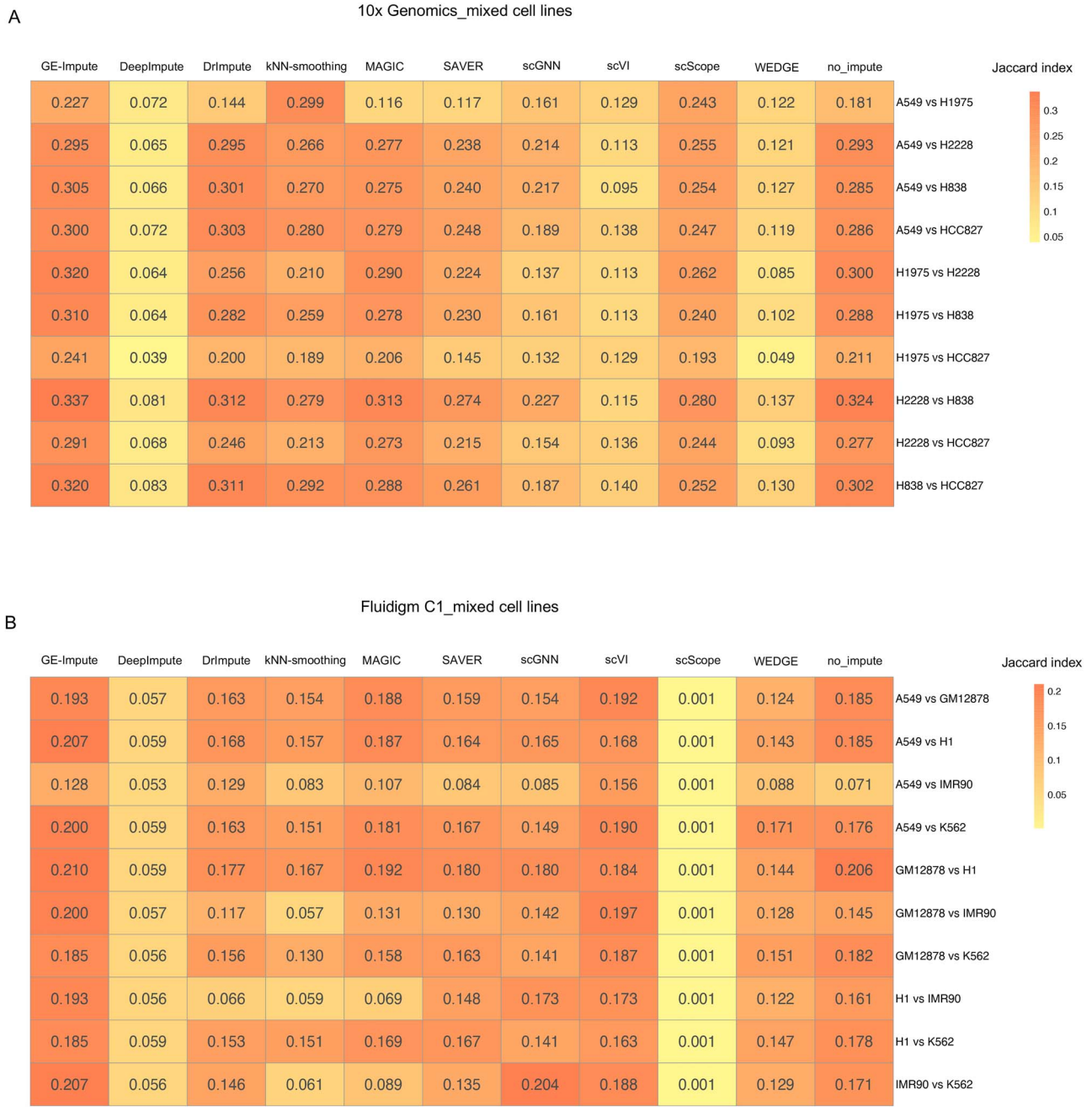

Figure 3. Comparison of different imputation methods on identifying differential expressed genes in 10x Genomics dataset and Fluidigm C1 dataset. Heatmap showing the value of Jaccard index between bulk (as the golden standard) and single-cell DEGs (log2-fold change >0.5) on the 10x Genomics dataset (**A**) and Fluidigm C1 dataset (**B**), respectively. Each row represents pair of cell types and each column represents an imputation method.

also be intuitively reflected by the UMAP plots. For example, in the clustering results of DeepImpute, MAGIC, SAVER, scGNN, scScope and scVI, a substantial fraction of CD4 T cells were wrongly assigned to the clusters of CD8 T cells, while kNN-smoothing showed a more dispersed distribution of clusters in UMAP plot, suggesting that this algorithm significantly changed the cell clustering topology.

In comparison with droplet-based scRNA-seq method like 10x Genomics, plated-based scRNA-seq platform like Smart-Seq2 often results in an scRNA-seq dataset with much fewer cells and therefore more challenging for clustering analysis. Here, a Smart-Seq2 dataset which contains four dendritic cell subtypes was introduced for the clustering accuracy assessment. Similar to the aforementioned method, we treated the clustering results as predicted labels and the known cell subtypes as true labels. Notably, GE-Impute (Purity = 0.727, ARI = 0.272, NMI = 0.391) showed better clustering accuracy than the raw data (Purity = 0.601, ARI = 0.147, NMI = 0.343) and other nine imputation methods for nearly all cases except the NMI index of DrImpute (Supplementary Figure 5 and Table 1). DrImpute (Purity = 0.704, ARI = 0.205, NMI = 0.440) performed well in

Figure 4. Performance comparison of unsupervised clustering on 10x Genomics PBMC dataset. The UMAP plot showing unsupervised clustering of PBMC raw data (**A**) or GE-Impute data (**B**). The left subpanel represents clusters information of unsupervised clustering. The right sub-panel presents the known cell type labels information. Each color represents a cell type. (**C**) Unsupervised clustering for other imputation methods. The upper sub-panel represents the unsupervised clustering information and the lower sub-panel represents the known cell labels. Intuitively, higher consistency between clustering labels and known cell type labels indicates a better cell clustering result.

both 10x Genomics dataset and Smart-Seq2 dataset, which would be attributed to its cell clustering-based expression imputation nature that enhances the intra-cluster expression homogeneity [12]. On the other hand, DeepImpute, MAGIC, scGNN, scVI and WEDGE could not perform as well on Smart-Seq2 dataset as they did on 10x Genomics dataset, suggesting these methods were not the recommended choice for performing clustering analysis of plated-based scRNA-seq data. In all, GE-Impute can significantly improve the clustering analysis accuracy of scRNA-seq data and make the expression characteristics of different cell types more straightforward, no matter on droplet- or plate-based scRNA-seq datasets.

Batch effect is common when analyzing scRNA-seq dataset and emerges as an obstacle in downstream analysis. Therefore, effective batch correction is vital in scRNA-seq data analysis. To investigate if GE-Impute

Table 1. Performance comparison of different imputation with multiple clustering evaluation indices

Index	Purity	ARI	NMI
10x Genomics_PBMC dataset			
GE-Impute	0.972	0.936	0.894
DeepImpute	0.874	0.666	0.801
DrImpute	0.965	0.899	0.854
kNN-smoothing	0.809	0.425	0.552
MAGIC	0.865	0.657	0.794
SAVER	0.866	0.651	0.772
scGNN	0.865	0.522	0.701
scScope	0.755	0.578	0.649
scVI	0.874	0.664	0.802
WEDGE	0.968	0.927	0.886
Raw	0.757	0.563	0.674
Smart-Seq2_cDC dataset			
GE-Impute	0.727	0.272	0.391
DeepImpute	0.596	0.095	0.250
DrImpute	0.704	0.205	0.440
kNN-smoothing	0.587	0.147	0.211
MAGIC	0.581	0.029	0.240
SAVER	0.615	0.157	0.332
scGNN	0.575	0.061	0.223
scScope	0.531	0.028	0.032
scVI	0.527	0.029	0.207
WEDGE	0.553	0.120	0.222
Raw	0.601	0.147	0.343

affects the batch effect benchmarks in scRNA-seq data analysis, we applied FindIntegrationAnchors and IntegrateData function of Seurat 4.0 R package to perform batch correction and evaluate the performance of raw and imputed data. We have considered both experimentally derived [6] and Splatter-simulated [37] datasets with different batches (Supplementary Figure 6 and Supplementary Table 2). The performance was evaluated using local inverse Simpson's index (LISI) [47] and adjusted rand index (ARI) [48]. LISI and ARI are two commonly used metrics to measure batch mixing. A higher LISI indicates superior batch correction while a low ARI denotes superior batch mixing. The result shows that GE-Impute does not significantly affect the batch correction of scRNA-seq dataset. LISI and ARI calculated by GE-Impute data are almost equivalent to the raw non-imputed data (Supplementary Table 5).

GE-Impute enhances the identification and visualization of cell type marker genes

To investigate whether GE-impute could help facilitate cell type annotation through enhancing cell type marker genes expression, we used Seurat package [41] to identify the expression of several key marker genes. We first explored the expression of CD14 (marker gene for CD14+ cells) and CD8A (marker gene for CD8+ T cells) in 10x_Genomics PBMC raw data (Figure 5**A**). As the result shown in violin plot, the CD14 was identified as marker gene in Cluster 5 while CD8A was not significantly enriched in any clusters, and their expression signatures were not obvious in the feature plot. After GE-Impute processing, both CD14 and CD8A were found to be stably expressed in CD14+ cell cluster and CD8+ T cell cluster, respectively (Figure 5**B**). The feature plot shows that the expression characteristic of these two marker genes is also more distinguishable between different clusters. Furthermore, the expression of several other marker genes is found to be enhanced after GE-Impute processing, such as CD1C marker for CD19+ B cell, GZMH and PTGD5 markers for CD56+ NK cell, EGFL7 maker for CD34+ cell, and CORO1B maker for CD4+ T cell (Figure 5**C**). In addition to 10x Genomics dataset, we also observed significantly elevated expression of marker genes in dendritic cell subtypes in scRNA-seq data from Smart-Seq2, such as GBP1 for blood pre-cDC, CCL23 for cord pre-cDCs, PPY and ERICH5 for CD1c + dendritic cells (Supplementary Figure 7). Although there are many outstanding methods and software available to automatically annotate cell types [49–51], clear expression of marker genes is still an essential feature for cell annotation in scRNA-seq data analysis [52]. These results suggest that GE-Impute can help identify cell types for scRNA-seq data by enhancing the expression of marker genes, thereby improving the efficiency of cell type annotation analysis.

GE-Impute improves the performance of cell trajectory inference

Trajectory analysis is also one of the important tasks in scRNA-seq data analysis. To evaluate if GE-Impute can improve the accuracy of trajectory inference and pseudotime ordering, we utilized a dataset containing 1529 cells from five stages of human preimplantation embryonic development from E3 to E7 [36]. We applied GE-Impute and other nine imputation methods to the raw data and then reconstructed the trajectory using SlingShot R package [53]. The results demonstrate that GE-Impute can improve cell trajectory reconstruction compared to the raw data in both t-SNE and UMAP reduction plots (Figure 6**A** and Supplementary Figure 8). Besides GE-Impute, some other imputation methods (but not all the methods) can also help to reconstruct the cell trajectory such as DrImpute, SAVER and scGNN in t-SNE plot. Whereas in UMAP reduction plot, MAGIC, scVI and WEDGE can also improve the trajectory inference but DrImpute fails to reconstruct the trajectory. To quantitatively compare their performance in improving the accuracy of pseudotime inference, the consistency between the true-time labels (i.e. E3 to E7) and pseudotime ordering was measured by the Pearson correlation coefficients. Two widely used methods, SlingShot [53] and PAGA [54], were used to predict the pseudotime labels. We found GE-Impute outperforms other methods in pseudotime inference when the analysis was conducted by SlingShot, and it ranks only second to scGNN when using PAGA. While the pseudotime ordering of imputed data from MAGIC and kNN-smoothing cannot be inferred by PAGA, suggesting these two methods overcorrect the transcriptome dynamics along the time course. In summary, these

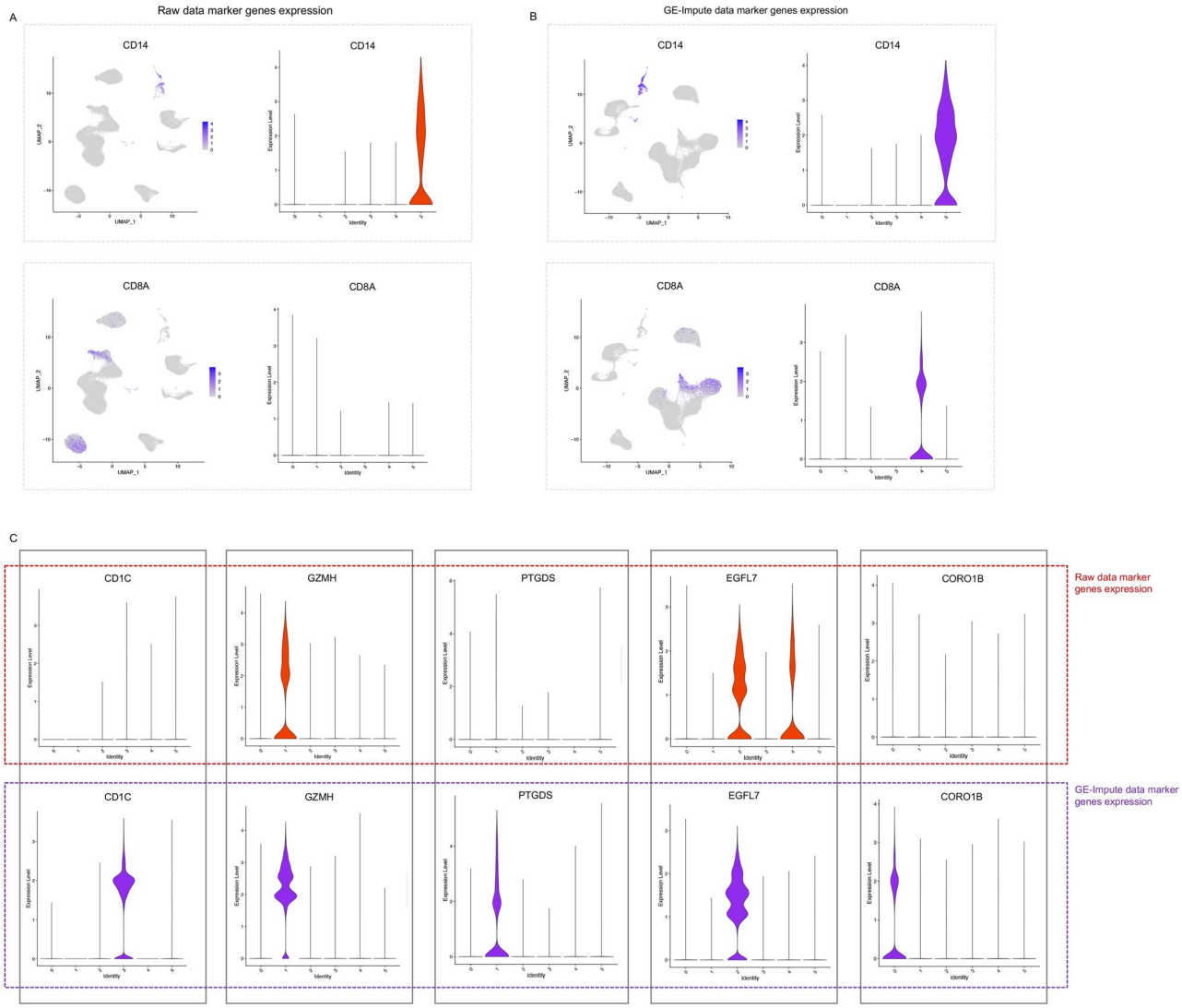

Figure 5. GE-Impute facilitates identification of marker genes in specific cell types. Feature plot and violin plot showing CD14 (top panel) and CD8A (lower panel) expression in (**A**) raw 10x_PBMC data or (**B**) GEImpute 10x_PBMC data. CD14 is the marker for CD14+ cell type and CD8A is the marker for CD8+ T cell type. (**C**) Violin plot showing several marker genes for specific cell types in raw (top red panel) and GEImpute 10x_PBMC data (lower purple panel), including CD1C (marker for CD19+ B cell), GZMH and PTGDS (marker for CD56+ NK cell), EGFL7 (marker for CD34+ stem cell), CORO1B (marker for CD4+ T cell).

results demonstrate that GE-Impute can improve the performance of pseudotime inference.

Time and memory cost evaluation

To evaluate the efficiency of GE-Impute and other imputation algorithms, we have counted the time and peak memory usage when imputing the aforementioned 10x Genomics scRNA-seq data of five cell lines (containing 3817 cells and 11 786 genes). We found GE-Impute only cost 39 s and 2242 MiB memory to finish the imputation work (Supplementary Figure 9**A**), which was comparable to kNN-smoothing and MAGIC methods. Moreover, we applied GE-Impute to impute datasets of various sizes, ranging in size from 5 k to 50 k cells, which were sampled from 10x Genomics PBMC dataset. The computational cost of GE-Impute is at a moderate level among all the methods (Supplementary Figure 9**B** and **C**), which indicates its effectiveness in imputation task.

Discussion

Compared with bulk RNA-seq, the dropout events are much more prevalent in scRNA-seq, resulting in a non-negligible impact on the accuracy of scRNA-seq analysis results. Generally, there are two approaches to solve this issue. The first is to capture more transcripts in scRNA-seq by improving the sensitivity of sequencing platform, such as switching to the plate-based platforms like Fluidigm C1 and Smart-Seq2. However, the cost per sample of these plate-based methods is much higher than droplet-based method, and the library preparation of plate-based protocols is very complex. Besides, as also shown by the above analysis, plate-based scRNA-seq data are also more challenging for downstream cell clustering analysis. Therefore, another promising solution is to develop new bioinformatics methods to handle the sparsity and technical noises in scRNA-seq data, where the simplest and most effective way is to impute the

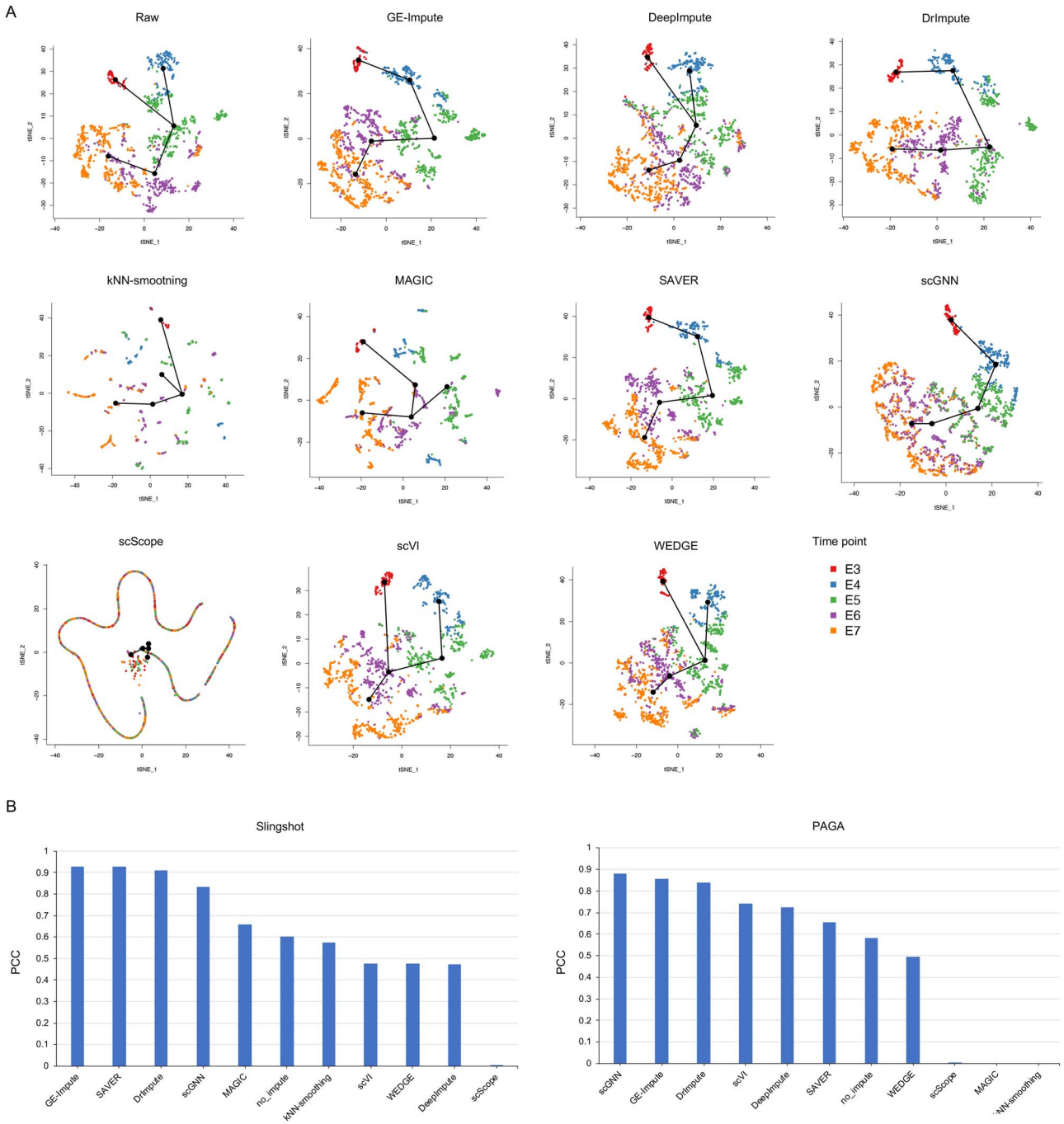

Figure 6. GE-Impute enhances the inference of cell trajectory. The trajectories reconstructed by SlingShot from raw and imputed scRNA-seq data (**A**). Each color represents a specific time point. The barplot shows a comparison of Pearson correlation coefficients between true-time points and predicted pseudotime labels which were calculated by SlingShot (**B**) and PAGA (**C**).

dropout zeros in scRNA-seq data matrix, thus improving the flexibility and accuracy of downstream analysis.

In this study, we propose a novel imputation method GE-Impute based on graph embeddings. GE-Impute first constructs a similarity matrix to learn feature representation using graph embedding neural network model and then predicts new links for the similarity network based on the learning features. Indeed, previous studies have applied similarity matrix to perform scRNA-seq clustering such as spectral clustering [55, 56]. Those methods rely on the similarity metrics for categorizing individual cells and show good performance in the clustering results, revealing the significance of similarity matrix in scRNA-seq data analysis. Compared with the original KNN cell similarity network, GE-Impute has significantly improved the performance of imputation, which indicates the advantage of the predicted new links between connected cells with similar characteristics. Unlike other graph-based imputation methods, GE-Impute only imputes the dropout values and retains the original expression characteristics as much as possible, while other imputation cells may overcorrect

the data such as kNN-smoothing and MAGIC. In missing value recovering analysis, GE-Impute provides higher Pearson correlation coefficients than the other nine methods on both experimentally derived and computer-simulated datasets. Through differential expression analysis, we compared the degree of overlap between DEGs from scRNA-seq and bulk RNA-seq. The results demonstrate that GE-Impute performs best in identifying DEGs in downstream analysis whether in 10x Genomics dataset or Fluidigm C1 dataset. Moreover, GE-Impute can significantly improve the unsupervised clustering of cells and promote cell type annotation through enhancing visualization of the cell type marker genes. Finally, GE-Impute can also improve the performance of cell trajectory analysis. During the above performance assessment, we also note that expression smoothing-based methods like kNN-smoothing and MAGIC often show better expression recovering performance, but are not very effective in delating with cell clustering tasks since such methods do not emphasize the latent topology of cell clusters. On the contrary, machine learning and deep learning methods can better depict the cell cluster topology and improve the cell clustering results, but often show compromised performance in expression recovering test, perhaps due to its overestimation of cell heterogenicity and topology complexity. While the pipeline of GE-Impute is somewhat in-between, the overall cell–cell similarity network is reconstructed by a sophisticated neural graph representation model, which is conceptually similar to the methods that depend on latent topology of cell clusters. But after reconstruction of cell–cell similarity network, a simple KNN-like approach was used for expression smoothing. Therefore, it is plausible that GE-Impute takes advantage of the traits of these two distant categories of imputation methods to achieve more robust performances. We believe future improvement of either network embedding models or expression smoothing algorithms is likely to further improve GE-Impute and methods alike.

Key Points

- GE-Impute is an imputation method for scRNA-seq data based on graph embedding.
- GE-Impute has significantly better performance on recovering dropout zeros in both droplet- and plated-based scRNA-seq data than other imputation methods.
- GE-Impute outperforms other imputation methods in identifying biological differentially expressed genes and improving the accuracy of unsupervised clustering analysis.
- GE-Impute enhances the identification and visualization of cell type-specific marker genes.
- GE-Impute improves the performance of cell trajectory inference.

Supplementary data

Supplementary data are available online at http://bib.oxfordjournals.org/.

Funding

National Key Research and Development Program of China (2021YFF1201201).

Data availability

The source code of GE-Impute is freely available at https://github.com/wxbCaterpillar/GE-Impute.

References

1. Tang F, Barbacioru C, Wang Y, *et al.* mRNA-Seq whole-transcriptome analysis of a single cell. *Nat Methods* 2009;**6**(5):377–82.
2. Maynard A, McCoach CE, Rotow JK, *et al.* Therapy-induced evolution of human lung cancer revealed by single-cell RNA sequencing. *Cell* 2020;**182**(5):1232–1251.e22.
3. Paik DT, Cho S, Tian L, *et al.* Single-cell RNA sequencing in cardiovascular development, disease and medicine. *Nat Rev Cardiol* 2020;**17**(8):457–73.
4. Xin Y, Kim J, Ni M, *et al.* Use of the Fluidigm C1 platform for RNA sequencing of single mouse pancreatic islet cells. *Proc Natl Acad Sci U S A* 2016;**113**(12):3293–8.
5. Picelli S, Faridani OR, Björklund AK, *et al.* Full-length RNA-seq from single cells using Smart-seq2. *Nat Protoc* 2014;**9**(1):171–81.
6. Zheng GX, Terry JM, Belgrader P, *et al.* Massively parallel digital transcriptional profiling of single cells. *Nat Commun* 2017;**8**(1):14049.
7. Hicks SC, Townes FW, Teng M, *et al.* Missing data and technical variability in single-cell RNA-sequencing experiments. *Biostatistics* 2018;**19**(4):562–78.
8. Ding J, Adiconis X, Simmons SK, *et al.* Systematic comparison of single-cell and single-nucleus RNA-sequencing methods. *Nat Biotechnol* 2020;**38**(6):737–46.
9. Lähnemann D, Köster J, Szczurek E, *et al.* Eleven grand challenges in single-cell data science. *Genome Biol* 2020;**21**(1):31.
10. van Dijk D, Sharma R, Nainys J, *et al.* Recovering gene interactions from single-cell data using data diffusion. *Cell* 2018;**174**(3):716–729.e27.
11. Wagner F, Yan Y, Yanai I. K-nearest neighbor smoothing for high-throughput single-cell RNA-Seq data. *bioRxiv* 2017. https://doi.org/10.1101/217737.
12. Gong W, Kwak IY, Pota P, *et al.* DrImpute: imputing dropout events in single cell RNA sequencing data. *BMC Bioinform* 2018;**19**(1):220.
13. Hu Y, Li B, Zhang W, *et al.* WEDGE: imputation of gene expression values from single-cell RNA-seq datasets using biased matrix decomposition. *Brief Bioinform* 2021;**22**(5):bbab085.
14. Deng Y, Bao F, Dai Q, *et al.* Scalable analysis of cell-type composition from single-cell transcriptomics using deep recurrent learning. *Nat Methods* 2019;**16**(4):311–4.
15. Arisdakessian C, Poirion O, Yunits B, *et al.* DeepImpute: an accurate, fast, and scalable deep neural network method to impute single-cell RNA-seq data. *Genome Biol* 2019;**20**(1):211.
16. Lopez R, Regier J, Cole MB, *et al.* Deep generative modeling for single-cell transcriptomics. *Nat Methods* 2018;**15**(12):1053–8.

17. Wang J, Ma A, Chang Y, *et al.* scGNN is a novel graph neural network framework for single-cell RNA-Seq analyses. *Nat Commun* 2021;**12**(1):1882.
18. Huang M, Wang J, Torre E, *et al.* SAVER: gene expression recovery for single-cell RNA sequencing. *Nat Methods* 2018;**15**(7):539–42.
19. Hou W, Ji Z, Ji H, *et al.* A systematic evaluation of single-cell RNA-sequencing imputation methods. *Genome Biol* 2020;**21**(1):218.
20. Perozzi B, Al-Rfou R, Skiena S. DeepWalk: online learning of social representations. In: *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. New York, NY, USA: Association for Computing Machinery, 2014, 701–10.
21. Grover A, Leskovec J. Node2vec: scalable feature learning for networks. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, San Francisco, CA, USA, 2016, 855–64. New York, NY, USA: Association for Computing Machinery.
22. Palash G, Emilio F. Graph embedding techniques, applications, and performance: A survey. *Knowl Based Syst* 2018;**151**:78–94.
23. Zhao BW, Hu L, You ZH, *et al.* HINGRL: predicting drug-disease associations with graph representation learning on heterogeneous information networks. *Brief Bioinform* 2022;**23**(1):bbab515.
24. Zhang HY, Wang L, You ZH, *et al.* iGRLCDA: identifying circRNA-disease association based on graph representation learning. *Brief Bioinform* 2022;**23**(3):bbac083.
25. Tang J, Qu M, Wang M *et al.* LINE: large-scale information network embedding. In: *Proceedings of the 24th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, Florence, Italy, 2015, 1067–77. Republic and Canton of Geneva, Switzerland: International World Wide Web Conferences Steering Committee.
26. Ribeiro LFR, Savarese PHP, Figueiredo DR. struc2vec: learning node representations from structural identity. In: *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*, Halifax, NS, Canada, 2017, 385–94. New York, NY, USA: Association for Computing Machinery.
27. Mežnar S, Lavrač N, Škrlj B. SNoRe: Scalable Unsupervised Learning of Symbolic Node Representations. *IEEE Access* 2020;**8**: 212568–88.
28. Freeman MF, Tukey JW. Transformations Related to the Angular and the Square Root. *Annals of Mathematical Statistics* 1950;**21**: 607–11.
29. Mikolov T, Sutskever I, Chen K *et al.* Distributed representations of words and phrases and their compositionality. In: *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*, Lake Tahoe, NV, USA, 2013, 3111–9. Red Hook, NY, USA: Curran Associates Inc.
30. Li X, Li S, Huang L, *et al.* High-throughput single-cell RNA-seq data imputation and characterization with surrogate-assisted automated deep learning. *Brief Bioinform* 2022;**23**(1):bbab368.
31. Tian L, Su S, Dong X, *et al.* scPipe: A flexible R/Bioconductor preprocessing pipeline for single-cell RNA-sequencing data. *PLoS Comput Biol* 2018;**14**(8):e1006361.
32. Li H, Courtois ET, Sengupta D, *et al.* Reference component analysis of single-cell transcriptomes elucidates cellular heterogeneity in human colorectal tumors. *Nat Genet* 2017;**49**(5): 708–18.
33. Barrett T, Wilhite SE, Ledoux P, *et al.* NCBI GEO: archive for functional genomics data sets–update. *Nucleic Acids Res* 2013;**41**(Database issue):D991–5.
34. Davis CA, Hitz BC, Sloan CA, *et al.* The Encyclopedia of DNA elements (ENCODE): data portal update. *Nucleic Acids Res* 2018;**46**(D1):D794–d801.
35. Breton G, Zheng S, Valieris R, *et al.* Human dendritic cells (DCs) are derived from distinct circulating precursors that are precommitted to become CD1c+ or CD141+ DCs. *J Exp Med* 2016;**213**(13): 2861–70.
36. Petropoulos S, Edsgärd D, Reinius B, *et al.* Single-Cell RNA-Seq Reveals Lineage and X Chromosome Dynamics in Human Preimplantation Embryos. *Cell* 2016;**165**(4):1012–26.
37. Zappia L, Phipson B, Oshlack A. Splatter: simulation of single-cell RNA sequencing data. *Genome Biol* 2017;**18**(1):174.
38. Love MI, Huber W, Anders S. Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. *Genome Biol* 2014;**15**(12):550.
39. Bauer DF. Constructing Confidence Sets Using Rank Statistics. *J Am Stat Assoc* 1972;**67**(339):687–90.
40. Levandowsky M, Winter D. Distance between Sets. *Nature* 1971;**234**(5323):34–5.
41. Hao Y, Hao S, Andersen-Nissen E, *et al.* Integrated analysis of multimodal single-cell data. *Cell* 2021;**184**(13):3573–3587.e29.
42. Gaujoux R, Seoighe C. A flexible R package for nonnegative matrix factorization. *BMC Bioinform* 2010;**11**(1):367.
43. Scrucca L, Fop M, Murphy TB, *et al.* mclust 5: Clustering, Classification and Density Estimation Using Gaussian Finite Mixture Models. *R j* 2016;**8**(1):289–317.
44. Wu X, Zhao X, Xiong Y, *et al.* Deciphering Cell-Type-Specific Gene Expression Signatures of Cardiac Diseases Through Reconstruction of Bulk Transcriptomes. *Front Cell Dev Biol* 2022;**10**: 792774.
45. Lukowski SW, Lo CY, Sharov AA, *et al.* A single-cell transcriptome atlas of the adult human retina. *EMBO J* 2019;**38**(18):e100811.
46. Kiselev VY, Andrews TS, Hemberg M. Challenges in unsupervised clustering of single-cell RNA-seq data. *Nat Rev Genet* 2019;**20**(5):273–82.
47. Büttner M, Miao Z, Wolf FA, *et al.* A test metric for assessing single-cell RNA-seq batch correction. *Nat Methods* 2019;**16**(1): 43–9.
48. Hubert L, Arabie P. Comparing partitions. *Journal of Classification* 1985;**2**(1):193–218.
49. Xu Y, Baumgart SJ, Stegmann CM, *et al.* MACA: marker-based automatic cell-type annotation for single-cell expression data. *Bioinformatics* 2021;**38**:1756–60.
50. Wei Z, Zhang S. CALLR: a semi-supervised cell-type annotation method for single-cell RNA sequencing data. *Bioinformatics* 2021;**37**:i51–8.
51. Shao X, Yang H, Zhuang X, *et al.* scDeepSort: a pre-trained cell-type annotation method for single-cell transcriptomics using deep learning with a weighted graph neural network. *Nucleic Acids Res* 2021;**49**(21):e122.
52. Zhang X, Lan Y, Xu J, *et al.* CellMarker: a manually curated resource of cell markers in human and mouse. *Nucleic Acids Res* 2019;**47**(D1):D721–d728.
53. Street K, Risso D, Fletcher RB, *et al.* Slingshot: cell lineage and pseudotime inference for single-cell transcriptomics. *BMC Genomics* 2018;**19**(1):477.
54. Wolf FA, Hamey FK, Plass M, *et al.* PAGA: graph abstraction reconciles clustering with trajectory inference through a topology preserving map of single cells. *Genome Biol* 2019;**20**(1):59.
55. Qi R, Wu J, Guo F, *et al.* A spectral clustering with self-weighted multiple kernel learning method for single-cell RNA-seq data. *Brief Bioinform* 2021;**22**(4):bbaa216.
56. Li Y, Luo P, Lu Y, *et al.* Identifying cell types from single-cell data based on similarities and dissimilarities between cells. *BMC Bioinform* 2021;**22**(S3):255.