scCAN: Clustering With Adaptive Neighbor-Based Imputation Method for Single-Cell RNA-Seq Data

Shujie Dong, Yuansheng Liu, Yongshun Gong, Xiangjun Dong, and Xiangxiang Zeng

***Abstract*—Single-cell RNA sequencing (scRNA-seq) is widely used to study cellular heterogeneity in different samples. However, due to technical deficiencies, dropout events often result in zero gene expression values in the gene expression matrix. In this paper, we propose a new imputation method called scCAN, based on adaptive neighborhood clustering, to estimate the zero value of dropouts. Our method continuously updates cell-cell similarity information by simultaneously learning similarity relationships, clustering structures, and imposing new rank constraints on the Laplacian matrix of the similarity matrix, improving the imputation of dropout zero values. To evaluate the performance of this method, we used four simulated and eight real scRNA-seq data for downstream analyses, including cell clustering, recovered gene expression, and reconstructed cell trajectories. Our method improves the performance of the downstream analysis and is better than other imputation methods.**

***Index Terms*—Adaptive neighborhood clustering, imputation, laplace matrix, single -cell RNA sequencing.**

I. Introduction

SEQUENCING is a high-throughput technique that reads and analyzes nucleic acids, allowing us to understand information such as gene structure and variations, making it an essential tool for life science and technology research. With scientific progress, we understood the code of life through sequencing technology [1], [2]. Currently, sequencing technology has improved. Sanger sequencing is a first-generation sequencing technology that determines DNA sequences by nucleotides with fluorescent markers. High-throughput sequencing is a second-generation sequencing technology that is efficient and fast. The third-generation sequencing technology based on nanopores sequences individual molecules with great accuracy and speed [3]. The continuous development of sequencing technology provides efficient tools for research in the life sciences and is critical for research in genomics, transcriptomics, and other fields [4], [5]. Bulk RNA sequencing (bulk RNA-seq) [6] can simultaneously determine the expression of all genes in a sample. Thus being widely used in transcriptome analysis of specific cell populations to study grafting variants, gene fusions, exons, and transcription start and stop sites. The transcriptome patterns of cell populations can help solve relevant biological problems. Bulk RNA sequencing technology mixes cells of different types and states and does not capture the gene expression heterogeneity and isoform differences between cells; only an average of the overall expression levels can be obtained. This may mask variations in the expression of critical genes and the presence of specific isoforms [7]. In contrast, through single-cell RNA sequencing (scRNA-seq) technology [8], it is feasible to acquire data on the transcriptome of individual cells, better reflecting the heterogeneity of cells within a population, revealing dynamic changes in the type, subtype, and transcriptional expression. The existing single-cell sequencing technology allows studying the cellular functions of organisms and related diseases accurately and clearly [9]. However, single-cell RNA-sequencing techniques have several drawbacks. One of the main challenges is the low RNA content of individual cells, which makes it difficult to obtain sufficient RNA sequencing data from a single cell. To address this issue, scRNA-seq experiments usually require RNA molecule amplification methods. However, RNA molecules may be lost during amplification, leading to missing data; this is called dropout [10]. scRNA-seq generates noisier and more variable data compared to bulk RNA-seq. scRNA-seq data are noisy, highly sparse, and massive, making it difficult for existing machine learning algorithms to analyze and process scRNA-seq data effectively. Therefore, developing machine-learning algorithms suitable for processing and analyzing scRNA-seq data is essential for understanding the pathogenesis of human diseases and their treatment [11], [12], [13]. To address the shortcomings of scRNA-seq technology, scRNA-seq data can be processed using single-cell sequencing data imputation [14]. This technique can recover noisy sequencing data, enhancing the performance of the imputed data in subsequent downstream

Manuscript received 19 April 2023; revised 22 October 2023; accepted 23 November 2023. Date of current version 5 February 2024. This work was supported by the National Natural Science Foundation of China under Grants 62076143, 62372159, 62122025, 62102140, 62202270, and 62172002; in part by the Science and Technology Innovation Program of Hunan Province under Grant 2022RC1100; in part by the Hunan Provincial Natural Science Foundation of China under Grant 2021JJ10020; in part by the Fundamental Research Promotion Plan of Qilu University of Technology (Shandong Academy of Sciences) under Grant 2021JC02009; and in part by the Taishan Scholar Project of Shandong Province under Grant tsqn202306066. *(Shujie Dong and Yuansheng Liu contributed equally to this work.) (Corresponding author: Xiangjun Dong.)*

Shujie Dong and Xiangjun Dong are with the Key Laboratory of Computing Power Network and Information Security, Ministry of Education, Shandong Computer Science Center (National Supercomputer Center in Jinan), Qilu University of Technology (Shandong Academy of Sciences), Jinan 250353, China, and also with the Shandong Provincial Key Laboratory of Computer Networks, Shandong Fundamental Research Center for Computer Science, Jinan 250014, China (e-mail: 10431210695@stu.qlu.edu.cn; d-xj@163.com).

Yuansheng Liu and Xiangxiang Zeng are with the College of Computer Science and Electronic Engineering, Hunan University, Changsha 410086, China (e-mail: yuanshengliu@hnu.edu.cn; xzeng@hnu.edu.cn).

Yongshun Gong is with the School of Software, Shandong University, Jinan 250100, China (e-mail: ysgong@sdu.edu.cn).

This article has supplementary downloadable material available at https://doi.org/10.1109/TCBB.2023.3337231, provided by the authors.

Digital Object Identifier 10.1109/TCBB.2023.3337231

analyses. Consequently, imputation has become an essential component of downstream cell analyses.

In recent years, imputation algorithms developed for recovering dropout events in single-cell transcriptome data have proliferated. For example, MAGIC [15] used the distance between cells to construct a Markov affinity matrix, used a data diffusion approach to obtain a smoother similarity matrix, and finally imputed the dropout values using similar expressions in similar cells. SAVER [16] used a negative binomial distribution, a Poisson distribution, and the expression of similar genes to estimate the dropout values, which were then weighted and averaged with the true expression values to obtain the final expressions. scImpute [17] used a gamma-normal mixture model on genes to distinguish between zero values owing to sequencing technology and biologically significant structural zeros. SCDD [18] is a two-stage imputation method that first identifies potential dropout values using a gamma-normal mixture model and then introduces a diffusion method for the initial estimation based on the expression values of similar cells. Finally the noise was then filtered using a joint model of a graph convolutional neural network and a shrinkage autoencoder to mitigate over–smoothing. ALRA [19] defines gene thresholds to distinguish dropout values. A low-rank estimation of the gene expression matrix was calculated using the singular value decomposition method to complete the imputation. DrImpute [20] uses the cell-to-cell distance to cluster cells into subgroups and then calculates the average expression values of the cell subgroups to impute the dropout values. DeepImpute [21] divides genes into subsets and builds subnetworks to construct models with improved efficacy and efficiency. scVI [22] proposed a hierarchical Bayesian model (HBM) based on deep neural networks. WEDGE [23] sets different weight parameters for zero and nonzero values of the matrix, uses low weight values for zero values to reduce the impact on the results, and uses a biased low-rank matrix decomposition to calculate the gene expression matrix.

Many imputation methods estimated dropout values based on similarity matrices of the data [24], [25], resulting in the recovery of gene expression by relying on the learning of similarity matrices, which comprise cell-to-cell and gene-to-gene similarities [26]. The point-to-point similarity matrices ignore the structure among potential cell subpopulations. Thus, the learned data similarity may not be the best for the class structure, leading to suboptimal results. In this study, we propose a new imputation method based on the adaptive neighbor clustering algorithm [27], that learns similarity by automatically assigning the best neighbor to each cell via local distance while applying rank constraints to the Laplacian matrix of the similarity matrix of the data to learn the structure of potential cell subgroups [28], thereby estimating the dropout values accurately. We applied the scCAN method to four published scRNA-seq datasets and six simulated datasets and compared their results with imputation methods. scCAN effectively used the cell subpopulation structure to improve the similarity matrix and demonstrate its downstream advantages for scRNA-seq. scCAN could accurately distinguish cell subpopulations and, compared to other imputation methods, reconstruct the cell differentiation trajectory from the fertilized egg through the 2-, 4-, 8-, and 16-cell stages to blastocyst cells. In addition, scCAN demonstrated the effect of recovering gene expression for different dropout rates, with the results being consistent with true expression. scCAN is currently an open-source method, available at https://https://github.com/dsj1998/scCAN.

II. Preliminary Discussion

A. Problem Description

Dropout events in single-cell sequencing are caused by technical limitations of the sequencing process, specifically DNA amplification. Single-cell sequencing involves the amplification of the DNA of a single cell to generate sufficient material for sequencing. However, this imperfect amplification process can lead to random biases and errors in the resulting DNA sequences. Thus, dropout events can generate technical zeros values, and this study aims to accurately fill in zeros entries by updating the intercellular similarity network.

B. Identify Potential Dropout Sites

Zeros contain biological and technical zeros values. They should not be confused when making imputations, and a distinction must be made. We used spectral clustering [29], [30] to reduce the dimensionality of the original data, and then roughly divided it into $k(k = 1, \ldots, K)$ cell subgroups using the K-means [31] clustering algorithm [32]. In the realm of cellular subpopulations, we developed a confidence level $\beta_i^{(k)}$ for each gene in order to differentiate between biological zeros and technological zeros. A higher confidence level indicates a greater probability that the zero value in the gene is a dropout value. The logic of this approach is derived from scImpute, whereby a zero value for a gene with high expression and low variability in a particular cellular subpopulation is assumed to be a technical zero value. Conversely, genes that are consistently expressed at low to moderate levels in cellular subpopulations with high variability are assumed to reflect true biological variability if they have a zero expression value. The confidence level $\beta_i^{(k)}$ is defined as follows:

$$\beta_i^{(k)} = \frac{(1-\alpha_i^{(k)})\mu_i^{(k)}}{(1-\alpha_i^{(k)})\mu_i^{(k)} + \alpha_i^{(k)}\sigma_i^{2(k)}}, \tag{1}$$

where $\alpha_i^{(k)}$, $\mu_i^{(k)}$, and $\sigma_i^{2(k)}$ denote the zero expression rate, the mean expression value, and the variance of the gene i in the cellular subpopulation k, respectively.

To distinguish between expression and dropout values, we construct a coefficient matrix P of the same size as the matrix V. The elements of the matrix P have only the values 1 and 0, indicating the presence or absence of expression in V_{ij}. To avoid over-imputation, genes with confidence below a threshold t are all set to 1 in the matrix P. This means that we consider zero expression values in these genes to be biologically significant and do not need to impute them. The matrix P is defined as follows:

$$P_{ij} = \begin{cases} 1 & \text{if } V_{ij} > 0, \\ 1 & \text{if } \beta_j^{(k)} < t \in (0,1), \\ 0 & \text{otherwise,} \end{cases} \tag{2}$$

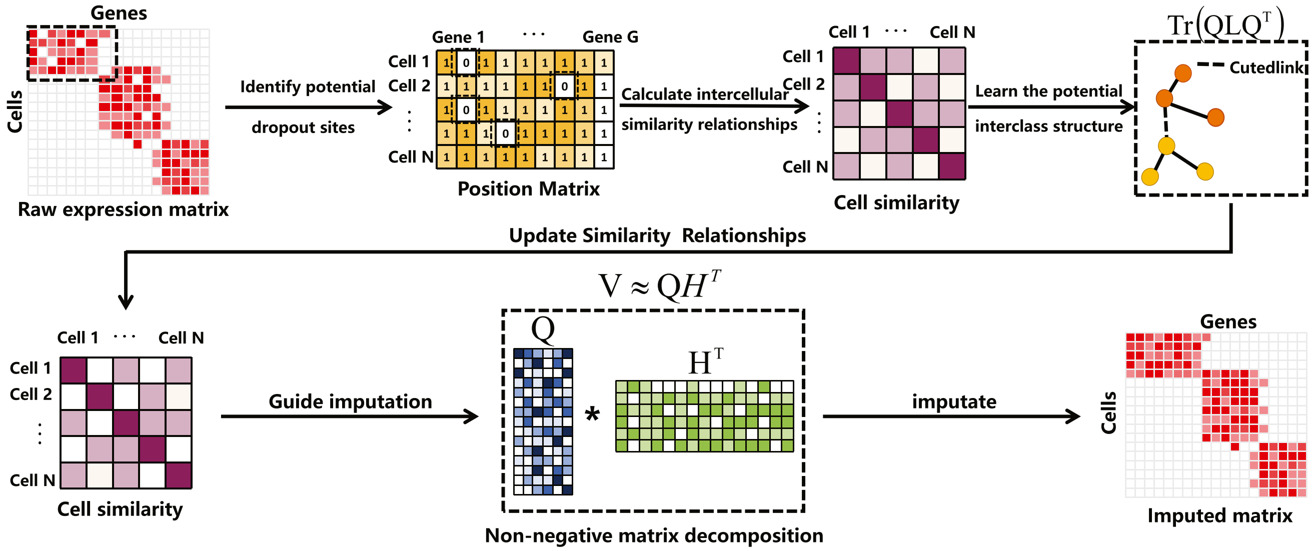

Fig. 1. Workflow of the scCAN algorithm.

where the element P_{ij} indicates the position of the true biological variability and dropout values, and the element V_{ij} indicates the expression value of the gene.

C. Non-Negative Matrix Factorization (NMF)

Single-cell sequencing data are non-negative. Using this non-negative matrix decomposition [33] to estimate dropout values is based on the data characteristics. In the problem of imputing dropout data, the non-negative matrix decomposition method reduces the original single-cell gene expression matrix V into two matrices, Q represents the latent features of cells, and H represents the latent features of the genes. The following optimization objectives can be used to describe the NMF-based basic data imputation model:

$$\min_{Q\geq 0, H\geq 0} \left\| P \odot \left(V - QH^{\top}\right)\right\|_F^2, \tag{3}$$

where $H^{\top}$ is the transpose matrix of H and $\odot$ is the Hadamard product operation.

III. The Proposed Method

This section describes how the scCAN algorithm can be implemented to update the similarity network using adaptive best neighbors while considering the cell subpopulation structure to improve accuracy. The proposed method is outlined in Fig. 1.

A. Self-Applicable Neighborhoods

In estimating deletion values, the cells or genes associated with them are often selected to construct a similarity matrix. The deletion values are estimated based on the cells or genes with high similarity to them. In this study, we assumed that the smaller the distance, the higher the similarity of the data points. In each iteration round, we used the KNN algorithm to traverse all data points and retain only the nearest r cells of each data point as the best nearest neighbors for similarity learning, calculating the similarity relationships between cells using the cosine distances. The total number of cells and the total number of genes are defined by N and G, respectively. The expression of cell i and cell j in the full set of genes is represented by x_{ig} and x_{jg}. The cosine distance between two cells is defined as

$$c_{ij} = 1 - \frac{\sum_{g=1}^{G} x_{ig} x_{jg}}{\sqrt{\sum_{g=1}^{G} x_{ig}^2}\sqrt{\sum_{g=1}^{G} x_{jg}^2}}. \tag{4}$$

The similarity matrix S is initialized as follows:

$$S_{ij} = \begin{cases} c_{ij} & x_i \in KNN\left(x_j\right) \ \ or \ \ x_j \in KNN\left(x_i\right), \\ 0 & x_i \notin KNN\left(x_j\right) \ \ and \ \ x_j \notin KNN\left(x_i\right), \end{cases} \tag{5}$$

where $KNN(x_i)$ and $KNN(x_j)$ denote the sets of cells closest to the cells x_i and x_j.

B. Potential Cluster Structure

Similarity networks are learned through the local distances between data points to consider point-to-point relationships. The underlying cellular subpopulation structure was ignored. Our method makes the connected components in the similarity matrix precisely equal to the number of clusters by imposing a new rank constraint on the Laplacian matrix of the data similarity matrix. Learning potential cell subpopulations while updating the similarity network improves accuracy [34]. The standardized Laplacian matrix is defined as:

$$L := D^{-1/2}\hat{L}D^{-1/2} = I - D^{-1/2}WD^{-1/2}, \tag{6}$$

where $\hat{L}$ is the graph Laplace matrix defined by $\hat{L} = D - W, W$ is the adjacency matrix and $W_{ij} = S_{ij}$, D is the degree matrix and $d_i = \sum_{i=1}^{n} w_{ij}$.

In order to learn the structure of potential cell subpopulations, it is necessary that W preserves the intrinsic geometry of V. In particular, if two cells v_i and v_j are close in the original data, then their potential spatial representations q_i and q_j are necessarily close, and vice versa. Cai et al. [35] have proved that local topological structure preservation can be formulated as trace optimization, i.e.

$$\frac{1}{2}\sum_{i,j} w_{ij}\left\|q_i - q_j\right\|^2 = \mathrm{Tr}\left(Q^\top L Q\right). \tag{7}$$

In summary, our final optimization objective is

$$\begin{aligned}\mathcal{L} = \min_{q\geq 0, H\geq 0} &\left\|P \odot (V - QH^\top)\right\|_F^2 \quad + \lambda_1\left(\|Q\|_F^2 + \|H\|_F^2\right) \\ &+ \lambda_2\,\mathrm{Tr}\left(Q^\top L Q\right),\end{aligned} \tag{8}$$

where λ_1 and λ_2 are the regularization parameters.

After solving (8), the learned matrices Q and H can be used to perform dropout data imputation. The completed data were estimated using

$$\hat{V} = (1-P)\odot(QH^\top) + V, \tag{9}$$

where $\hat{V}$ is defined as the imputed matrix.

C. Learning Process

To resolve the optimisation question in (8), we denote the objective function by $\mathcal{L}$. By setting $\frac{\partial\mathcal{L}}{\partial Q} = 0$ and $\frac{\partial\mathcal{L}}{\partial H} = 0$, we derived the updating rules as follows:

$$Q = Q \odot \frac{\left((P\odot V)H + \lambda_2 D^{-1/2} S D^{-1/2}\right)}{((P\odot(QH^\top)H) + \lambda_1 Q + \lambda_2 D^{-1/2} D D^{-1/2}}, \tag{10}$$

$$H = H \odot \frac{\left(Q^\top(P\odot V)\right)}{(Q^\top(P\odot(QH^\top)) + \lambda_1 H)}, \tag{11}$$

where the (10) and (11) are used to update the matrices Q and H until convergence.

To dynamically update the similarity between cells, we calculate the similarity matrix S of the matrix Q with potential cell characteristics. The similarity matrix S is updated as follows:

$$S_{ij} = \begin{cases} Q_{c_{ij}} & q_i \in KNN\left(q_j\right) \;\; or \;\; q_j \in KNN\left(q_i\right), \\ 0 & q_i \notin KNN\left(q_j\right) \;\; and \;\; q_j \notin KNN\left(q_i\right), \end{cases} \tag{12}$$

where $Q_{c_{ij}}$ represents the cosine distance between potential features of the cells, the $KNN(q_i)$ and $KNN(q_j)$ denote the feature sets closest to the cell potential features q_i and q_j.

D. Evaluation Measures

Two clustering metrics were used to assess the effectiveness of scCAN in cell clustering. Normalized mutual information (NMI) and adjusted random index (ARI) were used to measure the agreement between the estimated cell clusters and the intended cell clusters in scRNA-seq data. Pseudo-temporal ordering (POS) and Kendall's rank correlation score (KOR) were used to assess the cell movement trajectory. We used two metrics to measure the behavior of various imputation methods in recovering gene expression: pearson correlation coefficient (PCC) and root mean square error (RMSE).

NMI: It is a measurement of the degree of similarity between the two clustering results. The label collection of the raw clusters is denoted as A, and $\hat{A}$ denotes the set of labels obtained by clustering. It is defined by:

$$\mathrm{NMI} = \frac{2I(A,\hat{A})}{H(A) + H(\hat{A})}, \tag{13}$$

where $I(\hat{A},\hat{A})$ is the

$$I(\hat{A},\hat{A}) = \sum_{a\in A, b\in\hat{A}} p(a,b)\times\log\frac{p(a,b)}{p(a)p(b)}, \tag{14}$$

$H(\hat{A})$ is given by:

$$H(\hat{A}) = \sum_{a\in\hat{A}} p(a)\times\log p(a), \tag{15}$$

where the probability of cluster a is denoted as $p(a)$, the probability of cluster b is denoted as $p(b)$, and the probability of a sample belonging to both a and b is denoted as $p(a,b)$.

ARI: It is a common cluster evaluation metric that measures the degree of similarity between two clustering results by counting the number of sample points in the same class and different classes of clusters. We assumed that there were m cells which are clustered into k clusters. $\{u_i\}_{i=1}^m$ means the predicted cluster label and $\{v_j\}_{j=1}^m$ represents the true cluster label. It is defined by:

$$\mathrm{ARI} = \frac{\sum_{ij}\binom{n_{ij}}{2} - \left[\sum_i\binom{a_i}{2}\sum_j\binom{b_j}{2}\right]/\binom{n}{2}}{\left[\sum_i\binom{a_i}{2} + \sum_j\binom{b_j}{2}\right]/2 - \left[\sum_i\binom{a_i}{2}\sum_j\binom{b_j}{2}\right]/\binom{n}{2}}, \tag{16}$$

where i and j enumerate the k clusters, and $n_{ij} = \sum_{k,g} I(u_k = i)I(v_g = j)$, $a_i = \sum_k I(u_k = i)$, and $b_j = \sum_g I(v_g = j)$. The indicator function $I(x = y)$ is defined as

$$I(x=y) = \begin{cases} 1, & x = y, \\ 0, & \text{otherwise}, \end{cases} \tag{17}$$

POS: POS can be used to evaluate the performance of reconstructing pseudotimes. The POS is defined by:

$$\mathrm{POS} = \sum_{i=1}^{n-1}\sum_{j>i} f(i,j), \tag{18}$$

where n is the number of samples and the degree to which the order of the ith and jth cells in the ordered path matches the order based on external information is defined as $f(i,j)$.

KOR: It is used to assess the level of similarity between two rankings or ratings. It is defined by:

$$\tau = \frac{4\,C}{c(c-1)} - 1, \tag{19}$$

where c is the number of samples and C is the sum of the number of samples ranked after the given sample by two out of all samples.

RMSE: It measures the discrepancy between the imputed and true matrices by calculating the difference between the predicted and true values. It is defined by:

$$\mathrm{RMSE}(V, \hat{V}) = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\widehat{V}_i - V_i\right)^2}, \qquad (20)$$

where the gene expression matrix is denoted by V and $\hat{V}$ is the imputed matrix.

PCC: Its purpose is to measure the level of correlation among the imputation matrix and the raw matrix. The PCC is defined by:

$$\rho_{\mathrm{V},\hat{V}} = \frac{E(\mathrm{V} \odot \hat{V}) - E(\mathrm{V})E(\hat{V})}{\sqrt{E(V^2)(E(V))^2}\sqrt{E(\hat{V}^2)(E(\hat{V}))^2}}, \qquad (21)$$

where $E(V)$ represents the mean of V.

IV. Results

A. scCAN Improves Cell Clustering Analysis

Cell clustering is a critical part of downstream analysis and is essential in studying cell heterogeneity and identifying cell types. Therefore, cell clustering caused by dropout events reduces the accuracy of the downstream analysis. To validate scCAN, we processed the data using scCAN and other imputation methods before clustering them using the Seurat clustering algorithm [36] and comparing them with other imputation methods. We set two metrics to assess the agreement between the true and predicted labels, including the adjusted Rand index (ARI) and normalized mutual information (NMI). To validate the effectiveness of our method, we use seven real-world datasets, three of which are from the benchmark dataset generated by Tian et al. [37]. The other four real-world datasets are from Li et al. [38], Camp et al. [39], Baron et al. [40], and Usoskin et al. [41]. The dataset of Tian et al. contains three cell types and five cell types. Li dataset containing nine cell types, a liver dataset with seven cell types, a mouse dataset comprising thirteen cell types, and a usoskin dataset covering four cell types. Figs. 2 and 3, and Supplementary Figs. 1–2 show the results of the different imputation methods visualized on four real data sets. Our method can clearly distinguish different cell subpopulations. In the li dataset, our method clearly categorizes cells into nine types. The scVI and DrImpute imputation methods clearly show the outline of the cell subpopulations and accurately classify the cells into nine subpopulations. The DeepImpute method and the scImpute method confuse some of the cells. The cell types in the liver dataset contain fewer cells, which makes the distinction more difficult, but our method can still distinguish them by the visualization results out different cell types to achieve the best results. Other imputation methods distinguish too many cell types and have too much distance between classes. The three cell types in the unimputed sc_Droseq dataset have obvious confusing cells, and our method can clearly distinguish the confusing cells and get the cell types with clear outlines. sc_Celseq2_5cl_p2 dataset has the best results for WEDGE, and the structural relationship within the cell subpopulations is strengthened. Figs. 2(c) and 3(c) illustrate the clustering accuracies of the different imputation methods on the seven real datasets, and we evaluated the clustering effect of each method using the ARI and NMI clustering metrics. Our accuracy on the five real datasets ranked first on the sc_Dro-seq dataset after WEDGE, and second on the sc_Celseq2_5cl_p2 dataset after scVI. Combining the ARI and NMI scores with the UMAP visualization results, we conclude that our scCAN method significantly improves cell clustering performance.

B. scCAN Improves Cell Trajectory Inference

Trajectory inference, which is the ordering of cells based on the similarity of their expression patterns, is an essential tool in single-cell sequencing to infer cell developmental trajectories, thus, many algorithms have been developed in this field. For example, Monocle 3 [42] first selected genes with differential expression, used a reverse graph-embedding algorithm for dimensionality reduction, and finally sorted the cells along the trajectory. However, Monocle 3 does not consider the impact of dropout events on trajectory inference performance. Therefore, we compared the performance of different imputation methods for trajectory inference from scRNA-seq data to verify the effectiveness of imputation in trajectory inference. We used the temporal scRNA-seq dataset [43], which included single cells from fertilized eggs and 2-, 4-, 8-/16- cell stages to embryonic stages. Different imputation methods were run on the scRNA-seq dataset, and then we used Monocle 3 to perform trajectory inferences on the imputation data. We used Kendall rank correlation scores and pseudo-time ordering (POS) to measure the agreement between the true and estimated timestamps.

Fig. 4 shows the reconstructed trajectories on the Deng dataset using Monocle 3 for both the original and estimated data. Our method produces a pseudo-time order that is most consistent with the true timestamp. We noted that the MAGIC and WEDEG trajectories were more completely reconstructed. In contrast, the trajectories of ALRA, SAVER, and scVI could not be fully reconstructed. The other methods are far from the real timestamps and have performance gaps despite having complete trajectories. Fig. 4 also shows the Pseudo Timing Scores (POS) and Kendall Rank Correlation Scores (KOR) of the various imputation methods for the post-imputation cell trajectory analysis of the original dataset. scCAN, MAGIC, and WEDEG achieved the best rankings, with scCAN ranking first. In summary, the proposed method achieves the best results for the Deng dataset. scCAN inferred the complete trajectory from the Deng dataset, producing a more accurate pseudo-time order closest to the true timestamp. In contrast, the imputation of ALRA, SAVER, and scVI did not. Altogether, we conclude that scCAN improves the ability of Monocle 3 pseudo-time inference.

C. scCAN Improves the Recovery of Gene Expression

We used Splatter [44] to generate six simulated datasets with different deletion rates to test the imputation performance of scCAN, consisting of 500 cells and 1000 genes. The zero-expression rates for the six simulated datasets were 0.78, 0.71, 0.63, 0.55, 0.48, and 0.42. We imputed deletion values using scCAN and all other imputation methods and compared their filling matrices [45]. To facilitate visualizing the improvement in

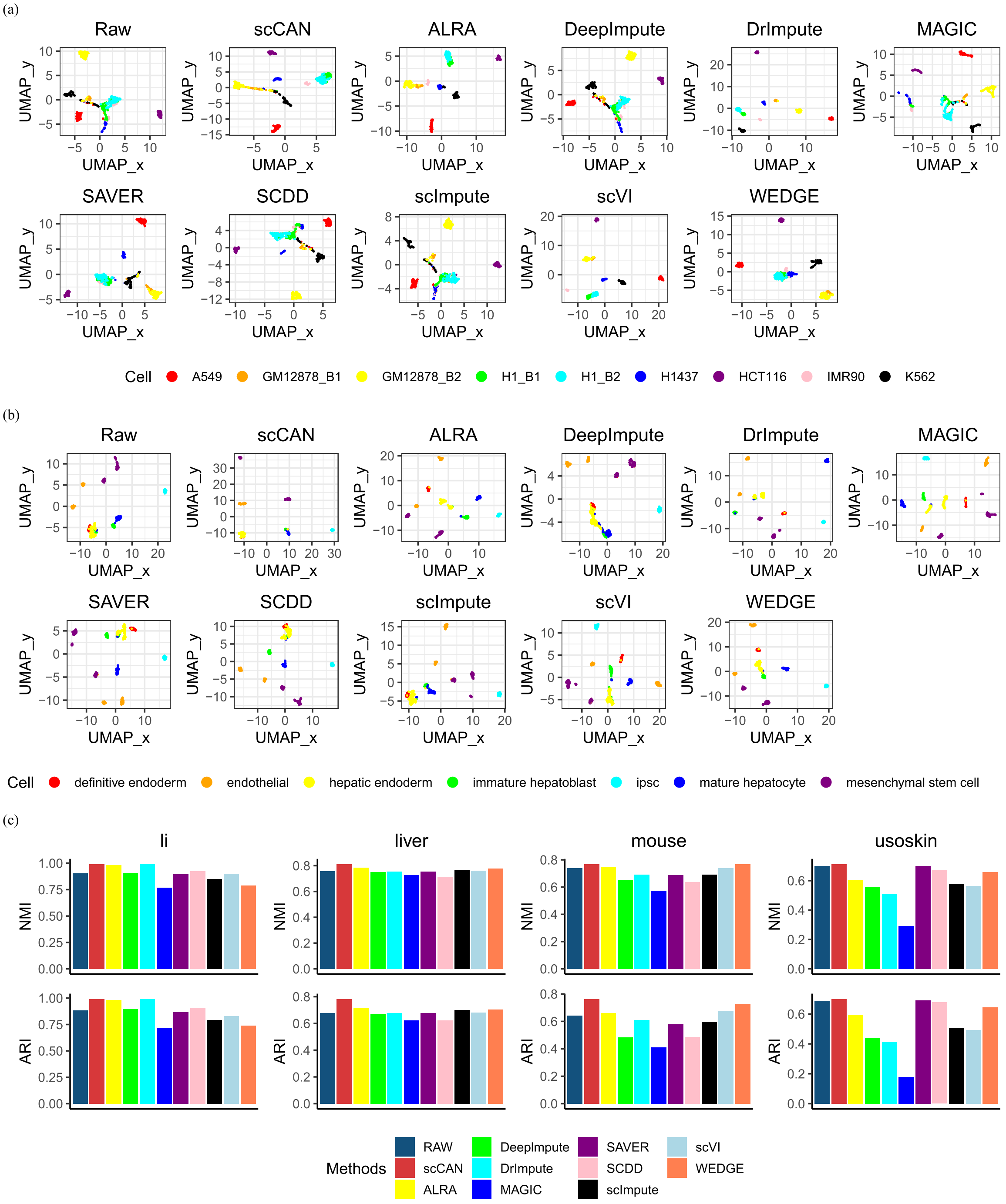

Fig. 2. Visualization of estimation data from different estimation methods through UMAP. (a) UMAP visualization results for li data. (b) UMAP visualization results for liver data. (c) NMI and ARI scores on four real-world datasets for all imputation methods. For all four real-world datasets, the scCAN ARI and NMI scores were ranked in first place when compared to the other nine imputation methods.

the clustering effect of each method on the simulated dataset, we reduced the dimensionality of the estimated data matrix using the UMAP method and plotted a scatter plot of the reduced dimensionality, as shown in Fig. 5 and Supplementary Figs. 3–6. We calculated the agreement between the estimated and true values at different deletion rates using the root mean square error (RMSE) and Pearson's correlation coefficient (PCC) to compare the recovered gene expression results. For example, in Fig. 5(a), we downscaled and visualized the simulated dataset with a zero expression rate of 0.78 using UMAP. In the absence of dropout events, four subpopulations of cells with distinct boundaries were observed in the real-count matrix. Our method, scCAN, distinguished between these four cell subpopulations. We obtained the closest results to the real count matrix with clear cell subpopulation boundaries and reasonable distances. Other methods did not have clear cell subpopulation boundaries or

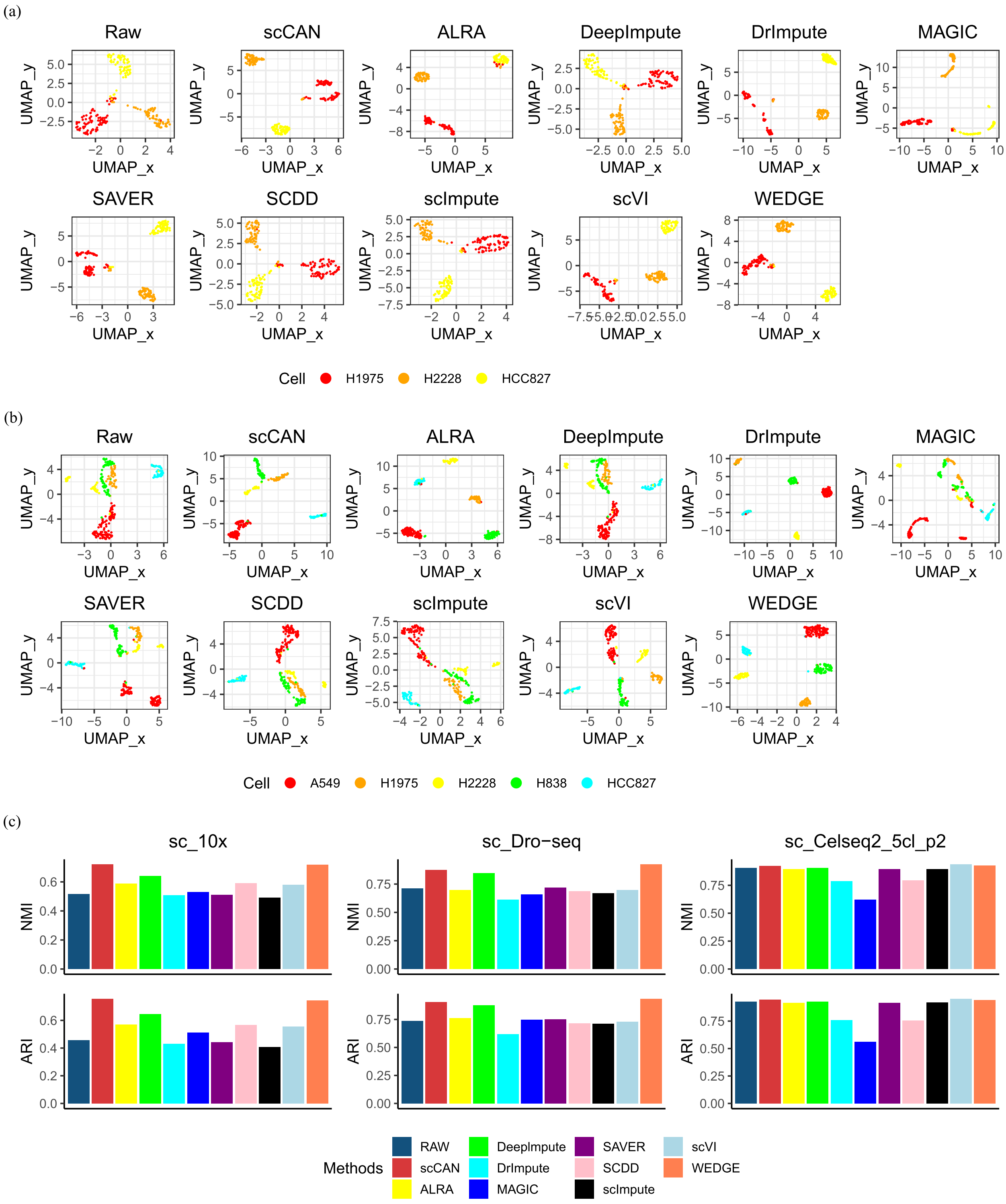

Fig. 3. Visualization of estimation data from different estimation methods through UMAP. (a) UMAP visualization results for sc_Droseq data. (b) UMAP visualization results for sc_Celseq2_5cl_p2 data. (c) NMI and ARI scores on four real-world datasets for all imputation methods. The scCAN algorithm score achieved the first place in the sc_10x dataset and the top two in the sc_Droseq and sc_Celseq2_5cl_p2 datasets.

accurately distinguish between cell subpopulation types. As the zero expression rate decreased, other imputation methods, such as DeepImpute, DrImpute, scVI, and WEDGE, distinguished four cell subpopulations, as seen in Fig. 5(b). WEDGE has clear cell subpopulation boundaries; however, the intra-class distance is far from the true count matrix. As shown in Fig. 6, our method has the smallest RMSE and the highest PCC ranking among the six simulated datasets, indicating that it accurately estimates dropout values and causes the least damage to the source dataset. The experiments demonstrate that our method

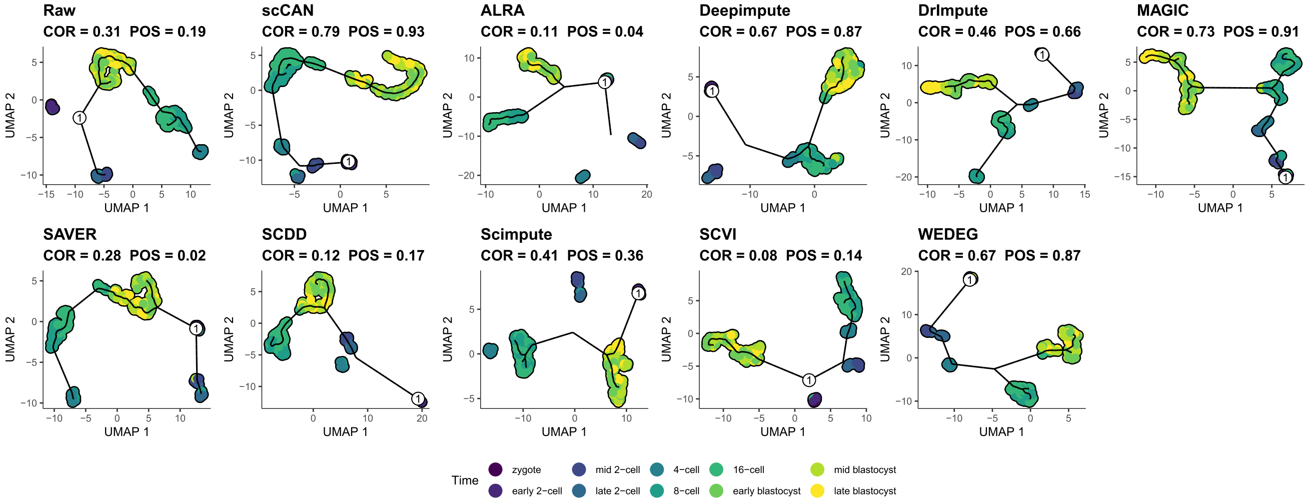

Fig. 4. Trajectories reconstructed on the Deng dataset by different imputation methods.

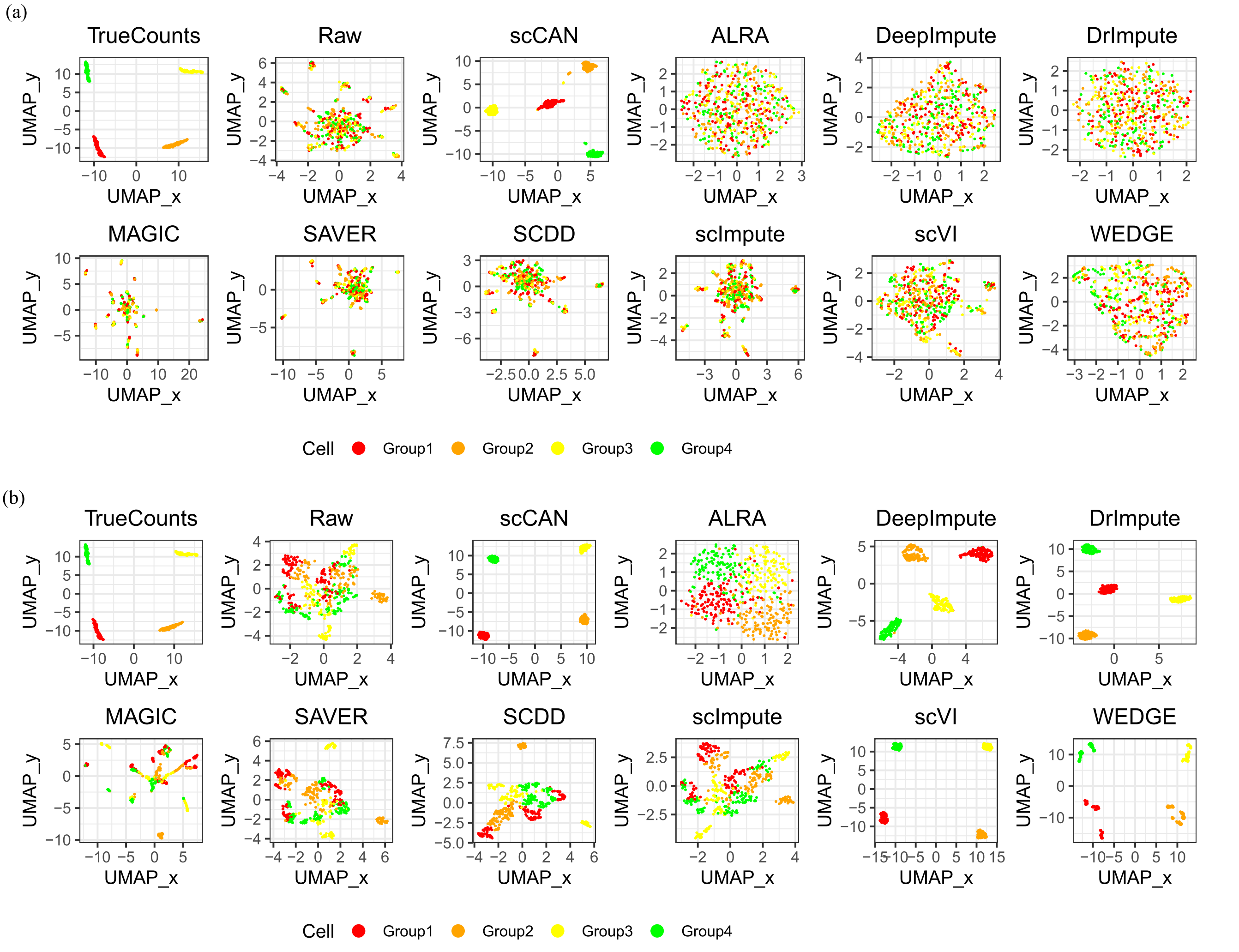

Fig. 5. scCAN can help recover dropout values and minimize damage to the original dataset. (a) UMAP visualization results on a simulated dataset with a zero expression rate of 0.78. (b) UMAP visualization results on a simulated dataset with a zero expression rate of 0.42.

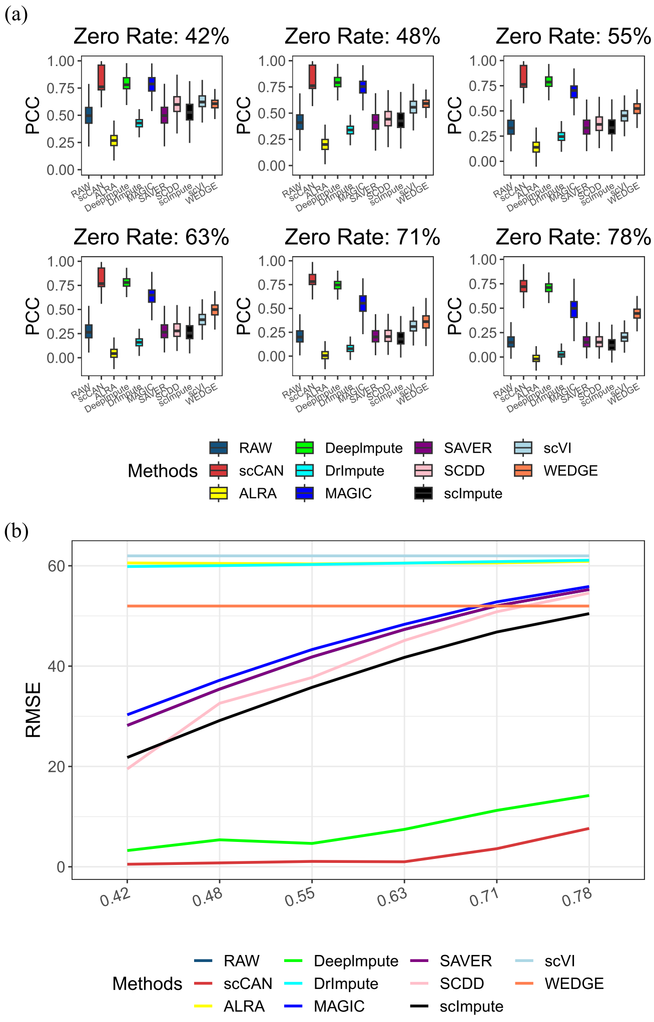

Fig. 6. PCC and RMSE scores for different imputation methods on different simulated datasets. (a) PCC of imputed gene expression data recovered on six simulated datasets with different zero expression rates. (b) RMSE of the imputed data for simulated data with different zero expression rates.

accurately recovers the dropout values when estimating the simulated datasets and effectively reduces the damage to the original datasets.

D. Parameter Setting

The scCAN algorithm involves three sets of hyperparameters and is studied on the liver dataset, we find the optimal hyperparameter configuration by tuning through a grid search. The first parameter is K, which determines the initial clustering number, fixed at $\{3, 7, 11\}$. The second parameter is the threshold t, which imputes only genes with confidence greater than t, set to 0.1 to 0.9 with an interval of 0.1, which controls the number of imputed genes. The third parameter consists of λ_1 and λ_2, which control the complexity of the model and the ability to preserve the structure of the data during optimization, with combinations of $\{0.2, 0.4\}$, $\{2, 4\}$, respectively. Table I shows the clustering accuracy of scCAN measured by ARI at different parameter settings with a fixed threshold t. Where setting K to a value close to the true number of cell subpopulations ARI scored the highest. Fig. 7 demonstrates the sensitivity analysis for the threshold t. From Fig. 7(a), it can be seen that the clustering accuracy of the scCAN algorithm is stable when different thresholds t are set and does not depend heavily on the choice of threshold t. Fig. 7(b) demonstrates that the confidence level of most genes in different cell subgroups is very close to 0 and 1, and the choice of threshold t only affects a small number of genes. To summarize, based on parametric analysis, scCAN is robust to different parameter values.

TABLE I
ARI CLUSTERING ACCURACY OF SCCAN AT FIXED THRESHOLD 0.1 FOR OTHER PARAMETER SETTINGS

	3	7	11	15	19
$(0.2, 0.4)$	0.730	0.671	0.650	0.613	0.613
$(2, 4)$	0.756	0.784	0.672	0.672	0.681

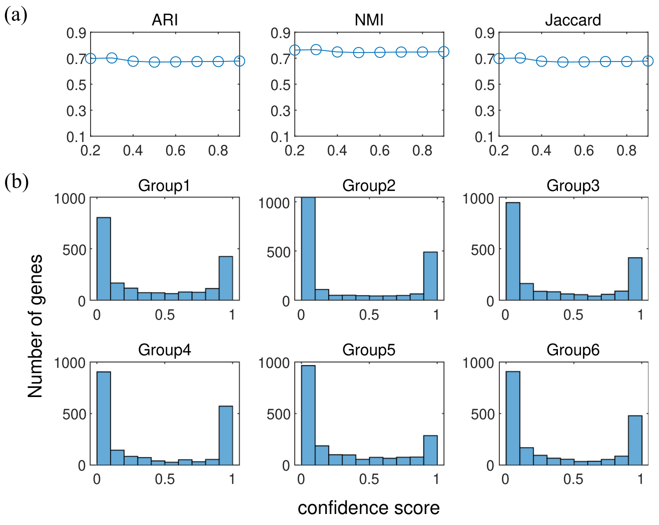

Fig. 7. Sensitivity analysis of the threshold t based on the liver dataset with fixed parameters $\{7, (2, 4)\}$ for K, λ_1 and λ_2. (a) Clustering scores of scCAN when using different thresholds t. (b) Gene confidence distributions for the six initial clusters in the liver dataset.

E. Complexity Analysis

The scCAN algorithm consists of three main steps: constructing the coefficient matrix P, initialization, and iterative updating of matrices Q, H, and S. The algorithm proceeds in three parts. In order to enhance clarity, we define G as the number of rows and N as the number of columns in the original gene expression matrix. Also, K represents both the total number of clusters and the dimensionality of the reduced data set, and Z specifies the maximum number of iterations. First, the spectral clustering algorithm is used to divide the dataset into K clusters and construct the coefficient matrix P with a complexity of $O(N^3 + KNG)$. Next, the matrices $Q \in R^{N\times K}$, $H \in R^{K\times G}$, and the similarity matrix S are initialized with a complexity of $O(ZNKG + N^2 K)$. Similarly, the iterative update process has a complexity of $O(ZNKG + N^2 K)$. As

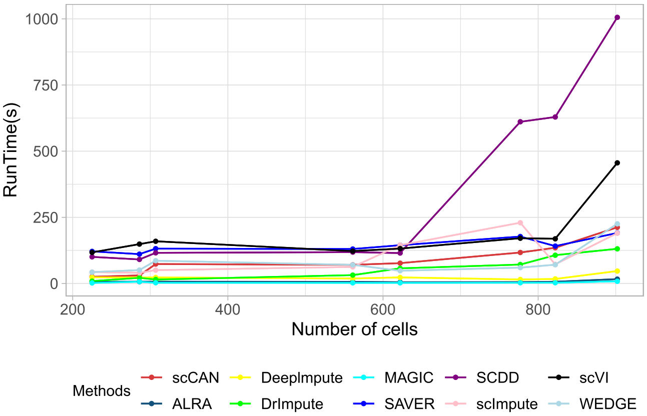

Fig. 8. Computational time of different imputation methods on all real datasets.

a result, the complexity of the scCAN algorithm is $O(N^3 + KNG + 2(ZNKG + N^2 K))$. Since the reduced dimension K is typically much smaller than the original dimension N, the complexity effectively becomes $O(N^3)$. Fig. 8 illustrates the running time of the scCAN method and nine other estimation methods across seven real datasets. Fast computation is crucial for estimation methods, and reduced computation times imply that the algorithms are more scalable and can handle tasks with varying data sizes. It is evident that MAGIC has the fastest runtime, scCAN is comparable to DrImpute and WEDGE, and SCDD has the slowest computation time.

V. Conclusion

With the rapid development of gene sequencing technology, accurately analyzing gene expression matrices has become a growing concern. However, until there is a novel breakthrough in sequencing technology, the processing of single-cell gene expression matrices will directly impact the effectiveness of downstream analysis. In this study, we propose a new imputation method, scCAN, to impute gene expression matrices with high noise. To show the efficiency of scCAN, we conducted an experiment using eight real and six simulated datasets. The results show that scCAN significantly improves downstream analysis compared to other imputation methods. In future work, we will extract more nonlinear features and variations and explore more nonlinear variation-based imputation methods on gene expression datasets.

References

[1] E. A. Winkler et al., “A single-cell atlas of the normal and malformed human brain vasculature,” *Science*, vol. 375, no. 6584, 2022, Art. no. eabi7377.

[2] Y. Cheng, Y. Gong, Y. Liu, B. Song, and Q. Zou, “Molecular design in drug discovery: A comprehensive review of deep generative models,” *Brief. Bioinf.*, vol. 22, no. 6, 2021, Art. no. bbab344.

[3] S. H. Gohil, J. B. Iorgulescu, D. A. Braun, D. B. Keskin, and K. J. Livak, “Applying high-dimensional single-cell technologies to the analysis of cancer immunotherapy,” *Nature Rev. Clin. Oncol.*, vol. 18, no. 4, pp. 244–256, 2021.

[4] Y. Liu, X. Shen, Y. Gong, Y. Liu, B. Song, and X. Zeng, “Sequence alignment/map format: A comprehensive review of approaches and applications,” *Brief. Bioinf.*, vol. 22, no. 5, 2023, Art. no. bbad320.

[5] Y. Wang, Y. Zhai, Y. Ding, and Q. Zou, “SBSM-Pro: Support bio-sequence machine for proteins,” 2023, *arXiv:2308.10275*.

[6] T. Peng, Q. Zhu, P. Yin, and K. Tan, “SCRABBLE: Single-cell RNA-seq imputation constrained by bulk RNA-seq data,” *Genome Biol.*, vol. 20, no. 1, pp. 1–12, 2019.

[7] E. Papalexi and R. Satija, “Single-cell RNA sequencing to explore immune cell heterogeneity,” *Nature Rev. Immunol.*, vol. 18, no. 1, pp. 35–45, 2018.

[8] S. S. Potter, “Single-cell RNA sequencing for the study of development, physiology and disease,” *Nature Rev. Nephrol.*, vol. 14, no. 8, pp. 479–492, 2018.

[9] J. Kolasa and C. D. Rollo, “Introduction: The Heterogeneity of Heterogeneity: A Glossary,” in *Ecological Heterogeneity*. Berlin, Germany: Springer, 1991, pp. 1–23.

[10] W. Hou, Z. Ji, H. Ji, and S. C. Hicks, “A systematic evaluation of single-cell RNA-sequencing imputation methods,” *Genome Biol.*, vol. 21, no. 1, pp. 1–30, 2020.

[11] T. Tang et al., “Machine learning on protein–protein interaction prediction: Models, challenges and trends,” *Brief. Bioinf.*, vol. 24, no. 2, 2023, Art. no. bbad076.

[12] B. Song, X. Luo, X. Luo, Y. Liu, Z. Niu, and X. Zeng, ”Learning spatial structures of proteins improves protein–protein interaction prediction,” *Brief. Bioinf.*, vol. 23, no. 2, 2022, Art. no. bbab558.

[13] X. Yang et al., “Modality-DTA: Multimodality fusion strategy for drug–target affinity prediction,” *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 20, no. 2, pp. 1200–1210, Mar./Apr. 2023.

[14] L.-H. Ly and M. Vingron, “Effect of imputation on gene network reconstruction from single-cell RNA-seq data,” *Patterns*, vol. 3, no. 2, 2022, Art. no. 100414.

[15] D. V. Dijk et al., “Recovering gene interactions from single-cell data using data diffusion,” *Cell*, vol. 174, no. 3, pp. 716–729, 2018.

[16] M. Huang et al., “SAVER: Gene expression recovery for single-cell RNA sequencing,” *Nature Methods*, vol. 15, no. 7, pp. 539–542, 2018.

[17] W. V. Li and J. J. Li, “An accurate and robust imputation method scimpute for single-cell RNA-seq data,” *Nature Commun.*, vol. 9, no. 1, p. 997, 2018.

[18] J. Liu, Y. Pan, Z. Ruan, and J. Guo, “SCDD: A novel single-cell RNA-seq imputation method with diffusion and denoising,” *Brief. Bioinf.*, vol. 23, no. 5, 2022, Art. no. bbac398.

[19] G. C. Linderman et al., “Zero-preserving imputation of single-cell RNA-seq data,” *Nature Commun.*, vol. 13, no. 1, pp. 1–11, 2022.

[20] W. Gong, I.-Y. Kwak, P. Pota, N. Koyano-Nakagawa, and D. J. Garry, “DrImpute: Imputing dropout events in single cell RNA sequencing data,” *BMC Bioinf.*, vol. 19, no. 1, pp. 1–10, 2018.

[21] C. Arisdakessian, O. Poirion, B. Yunits, X. Zhu, and L. X. Garmire, “DeepImpute: An accurate, fast, and scalable deep neural network method to impute single-cell RNA-seq data,” *Genome Biol.*, vol. 20, no. 1, pp. 1–14, 2019.

[22] J. Ding, A. Condon, and S. P. Shah, “Interpretable dimensionality reduction of single cell transcriptome data with deep generative models,” *Nature Commun.*, vol. 9, no. 1, pp. 1–13, 2018.

[23] Y. Hu et al., “WEDGE: Imputation of gene expression values from single-cell RNA-seq datasets using biased matrix decomposition,” *Brief. Bioinf.*, vol. 22, no. 5, 2021, Art. no. bbab085.

[24] Y. Gong, Z. Li, J. Zhang, W. Liu, B. Chen, and X. Dong, “A spatial missing value imputation method for multi-view urban statistical data,” in *Proc. 29th Int. Conf. Int. Joint Conferences Artif. Intell.*, 2021, pp. 1310–1316.

[25] Y. Gong, Z. Li, J. Zhang, W. Liu, Y. Yin, and Y. Zheng, “Missing value imputation for multi-view urban statistical data via spatial correlation learning,” *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 1, pp. 686–698, Jan. 2023.

[26] J. Xu, L. Cai, B. Liao, W. Zhu, and J. Yang, “CMF-Impute: An accurate imputation tool for single-cell RNA-seq data,” *Bioinformatics*, vol. 36, no. 10, pp. 3139–3147, 2020.

[27] F. Nie, X. Wang, and H. Huang, “Clustering and projected clustering with adaptive neighbors,” in *Proc. 20th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2014, pp. 977–986.

[28] W. Wu and X. Ma, “Network-based structural learning nonnegative matrix factorization algorithm for clustering of scRNA-seq data,” *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 20, no. 1, pp. 566–575, Jan./Feb. 2023.

[29] A. Ng, M. Jordan, and Y. Weiss, “On spectral clustering: Analysis and an algorithm,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2001, pp. 849–856.

[30] Z. Li, J. Zhang, Q. Wu, Y. Gong, J. Yi, and C. Kirsch, “Sample adaptive multiple kernel learning for failure prediction of railway points,” in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2019, pp. 2848–2856.
[31] J. Huang, F. Nie, and H. Huang, “Spectral rotation versus K-means in spectral clustering,” in *Proc. AAAI Conf. Artif. Intell.*, 2013, pp. 431–437.
[32] W. Wu, Z. Liu, and X. Ma, “jSRC: A flexible and accurate joint learning algorithm for clustering of single-cell RNA-sequencing data,” *Brief. Bioinf.*, vol. 22, no. 5, 2021, Art. no. bbaa433.
[33] Y.-X. Wang and Y.-J. Zhang, “Nonnegative matrix factorization: A comprehensive review,” *IEEE Trans. Knowl. Data Dngineering*, vol. 25, no. 6, pp. 1336–1353, Jun. 2013.
[34] H. Wang and X. Ma, “Learning deep features and topological structure of cells for clustering of scRNA-sequencing data,” *Brief. Bioinf.*, vol. 23, no. 3, 2022, Art. no. bbac068.
[35] D. Cai, X. He, J. Han, and T. S. Huang, “Graph regularized nonnegative matrix factorization for data representation,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 8, pp. 1548–1560, Aug. 2011.
[36] R. Satija, J. A. Farrell, D. Gennert, A. F. Schier, and A. Regev, “Spatial reconstruction of single-cell gene expression data,” *Nature Biotechnol.*, vol. 33, no. 5, pp. 495–502, 2015.
[37] L. Tian et al., “Benchmarking single cell RNA-sequencing analysis pipelines using mixture control experiments,” *Nature Methods*, vol. 16, no. 6, pp. 479–487, 2019.
[38] H. Li et al., “Reference component analysis of single-cell transcriptomes elucidates cellular heterogeneity in human colorectal tumors,” *Nature Genet.*, vol. 49, no. 5, pp. 708–718, 2017.
[39] J. G. Camp et al., “Multilineage communication regulates human liver bud development from pluripotency,” *Nature*, vol. 546, no. 7659, pp. 533–538, 2017.
[40] M. Baron et al., “A single-cell transcriptomic map of the human and mouse pancreas reveals inter-and intra-cell population structure,” *Cell Syst.*, vol. 3, no. 4, pp. 346–360, 2016.
[41] D. Usoskin et al., “Unbiased classification of sensory neuron types by large-scale single-cell rna sequencing,” *Nature Neurosci.*, vol. 18, no. 1, pp. 145–153, 2015.
[42] X. Qiu et al., “Reversed graph embedding resolves complex single-cell trajectories,” *Nature Methods*, vol. 14, no. 10, pp. 979–982, 2017.
[43] Q. Deng, D. Ramsköld, B. Reinius, and R. Sandberg, “Single-cell RNA-seq reveals dynamic, random monoallelic gene expression in mammalian cells,” *Science*, vol. 343, no. 6167, pp. 193–196, 2014.
[44] L. Zappia, B. Phipson, and A. Oshlack, “Splatter: Simulation of single-cell RNA sequencing data,” *Genome Biol.*, vol. 18, no. 1, pp. 1–15, 2017.
[45] C. Dai et al., “scIMC: A platform for benchmarking comparison and visualization analysis of scRNA-seq data imputation methods,” *Nucleic Acids Res.*, vol. 50, no. 9, pp. 4877–4899, 2022.

Shujie Dong is currently working toward the ME degree with the Faculty of Computer Science and Technology, Qilu University of Technology (Shandong Academy of Sciences), Jinan, China. His research interests include sequential data mining and bioinformatics.

Yuansheng Liu received the bachelor’s and master’s degrees in computer science from Xiangtan University, China, in 2012 and 2015, respectively, and the PhD degree from the University of Technology Sydney, Australia, in 2019. He is currently an associate professor with the College of Information Science and Engineering, Hunan University, China. In 2020, he was postdoctoral research fellow with the University of Technology Sydney. His current research interests include bioinformatics and deep learning.

Yongshun Gong received the PhD degree from the University of Technology Sydney. He is an associate professor with the School of Software, Shandong University, China. His Principal research interests include covers the data science and machine learning, in particular, the following areas: adaptive model; spatiotemporal data mining; traffic prediction; recommender system, and sequential pattern mining. He has published above 40 papers in top journals and refereed conference proceedings, including *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *IEEE Transactions on Knowledge and Data Engineering*, *IEEE Transactions on Neural Networks and Learning Systems*, *IEEE Transactions on Cybernetics*, *IEEE Transactions on Multimedia*, NeurIPS, CVPR, KDD, CIKM, AAAI, IJCAI, etc.

Xiangjun Dong received the PhD degree in computer applications from the Beijing Institute of Technology, Beijing, China, in 2005. From 2007 to 2009, he worked as a postdoctoral fellow with the School of Management and Economics, Beijing Institute of Technology. From 2009 to 2010, he was a visiting scholar with the University of Technology Sydney, Sydney, Australia. He is currently a professor with the School of Computer Science and Technology, Qilu University of Technology (Shandong Academy of Sciences), Jinan, China. His research interests include data mining, artificial intelligence and Big Data. He has published more than 100 journal/conference publications including Artificial Intelligence, *IEEE Transactions on Neural Networks and Learning Systems*, the *IEEE Transactions on Cybernetics, Pattern Recognition*, The Conference on Information and Knowledge Management (CIKM), and so on.

Xiangxiang Zeng received the BS degree in automation from Hunan University, Changsha, China, in 2005, and the PhD degree in system engineering from the Huazhong University of Science and Technology, Wuhan, China, in 2011. Before joining Hunan University in 2019, he was with Department of Computer Science, Xiamen University. In 2019, he is a Yuelu distinguished professor with the College of Information Science and Engineering, Hunan University. His main research interests include membrane computing, neural computing, and bioinformatics.