Feng *et al. BMC Genomics* 2020, **21**(Suppl 10):618
https://doi.org/10.1186/s12864-020-07007-w

BMC Genomics

METHODOLOGY Open Access

I-Impute: a self-consistent method to impute single cell RNA sequencing data

Xikang Feng[1,2†], Lingxi Chen[2†], Zishuai Wang[2] and Shuai Cheng Li[2,3*]

From The 18th Asia Pacific Bioinformatics Conference
Seoul, Korea. 18-20 August 2020

Abstract

Background: Single-cell RNA-sequencing (scRNA-seq) is becoming indispensable in the study of cell-specific transcriptomes. However, in scRNA-seq techniques, only a small fraction of the genes are captured due to "dropout" events. These dropout events require intensive treatment when analyzing scRNA-seq data. For example, imputation tools have been proposed to estimate dropout events and de-noise data. The performance of these imputation tools are often evaluated, or fine-tuned, using various clustering criteria based on ground-truth cell subgroup labels. This limits their effectiveness in the cases where we lack cell subgroup knowledge. We consider an alternative strategy which requires the imputation to follow a "self-consistency" principle; that is, the imputation process is to refine its results until there is no internal inconsistency or dropouts from the data.

Results: We propose the use of "self-consistency" as a main criteria in performing imputation. To demonstrate this principle we devised I-Impute, a "self-consistent" method, to impute scRNA-seq data. I-Impute optimizes continuous similarities and dropout probabilities, in iterative refinements until a self-consistent imputation is reached. On the in silico data sets, I-Impute exhibited the highest Pearson correlations for different dropout rates consistently compared with the state-of-art methods SAVER and scImpute. Furthermore, we collected three wetlab datasets, mouse bladder cells dataset, embryonic stem cells dataset, and aortic leukocyte cells dataset, to evaluate the tools. I-Impute exhibited feasible cell subpopulation discovery efficacy on all the three datasets. It achieves the highest clustering accuracy compared with SAVER and scImpute.

Conclusions: A strategy based on "self-consistency", captured through our method, I-Impute, gave imputation results better than the state-of-the-art tools. Source code of I-Impute can be accessed at https://github.com/xikanfeng2/I-Impute.

Keywords: scRNA-seq, Imputation, Self-consistency, Cell subpopulation identification

*Correspondence: shuaicli@cityu.edu.hk
†Xikang Feng and Lingxi Chen contributed equally to this work.
[2]Department of Computer Science, City University of Hong Kong, Tat Chee Avenue, Kowloon, Hong Kong, China
[3]Department of Biomedical Engineering, City University of Hong Kong, Tat Chee Avenue, Kowloon, Hong Kong, China
Full list of author information is available at the end of the article

Background

Single-cell RNA-sequencing (scRNA-seq) is becoming indispensable in studying the landscapes of cell-specific transcriptomes [1]. It demonstrates robust efficacy in capturing transcriptome-wide cell-to-cell heterogeneity with high resolution [2–5]. With meta information such as time series or patient histology, scRNA-seq has the potential to decipher the underlying patterns in cell cycles [6–8], complex diseases [9–11], and cancers [8, 12–16].

As with other sequencing techniques, scRNA-seq produces a count matrix which captures expression profiles, with genes as rows, cells as columns, and the gene counts as the matrix elements. scRNA-seq only captures a small fraction of the genes due to "dropout" events. That is, it produces a zero-inflated count matrix where only about 10% entries are non-zero values [17]. This is mainly due to the missing of truly expressed transcripts from some cells during sequencing. The dropout rate is protocol-dependent [18]. When analyzing scRNA-seq data, the excess zero counts from dropout events needs to be remedied. Otherwise, the zero count distribution from different protocols may lead to diverging potency, which will affect downstream analyses [18], such as clustering, cell type recognition, dimension reduction, differential gene expression analysis, identification of cell specific genes and reconstruction of differentiation trajectory on zero-inflated single-cell gene expression data [18]. The correctness of all these analyses are contingent on the correctness of the expression profile.

As a remedy, downstream scRNA-seq-based analyses such as clustering, cell type recognition, and dimension reduction, can be adapted to implicitly incorporate considerations for dropout events [19–22]. On the other hand, dropout events can be treated prior to downstream analysis with scRNA-seq imputation tools. Two such leading tools are SAVER and scImpute. SAVER [23] imputes by borrowing information across genes using a Bayesian approach which estimates the expression levels. It aims to reduce meaningless biological variation and retain valuable biological variation. One caveat is that SAVER would unfairly adjust all gene expression levels including the actual non-expression of genes, hence possibly interject new biases and abolish real biological meanings. scImpute [18] is designed to first identify dropout values with Gamma-Normal mixture model, and then impute the dropout events by borrowing information from similar cells, with the expression level of un-dropout events unchanged. It automatically excludes the outlier cells and their gene information, which are likely to influence the original imputation values. While scImpute is able to avoid the problem which SAVER faces, it is not good with extremely sparse datasets.

On in silico data where the ground truth counts are known, the root mean square error (RMSE) between imputed and ground truth entries is the most common metrics for imputation evaluation [24]. For wetlab data sets, the ground truth counts for missing events are unknown. One common practice is to randomly remove non-zeros entries and employ an imputation tool to impute these removed entries. Then, the RMSE for the removed entries is calculated as a criterion to evaluate the performance of the imputation [24, 25]. Another common practice is to implicitly validate imputation efficacy by checking whether the imputation improves the downstream analysis result. This check, on the other hand, typically requires additional knowledge. For instance, clustering measurements such as adjusted Rand index (ARI), normalized mutual information (NMI), silhouette width (SW), and within-cluster sum of squares are commonly adopted for scRNA-seq imputation evaluation [18, 26], but these evaluations all require the true cluster labels, which are often hard to obtain.

As an explicit measurement, *imputation consistency* has been discussed in several studies. Buuren et al. [27] stated that the imputed entries should remain internal homogeneous to the non-missing data. Liang et al. adopts consistent estimate after imputation step for high-dimensional data [28]. Here, we propose a new interpretation for imputation consistency. As a reliable imputation tool should assume its output contains no dropout or errors. We want the imputation tool to be consistent in its output: If we are to feed the output to the imputation tool again after eliminating a number of entries, the tool should be able to reproduce these entries. We refer this property as *self-consistency*.

Therefore, in this study, to study the effects of the new criterion, we developed a self-consistent method called I-Impute for scRNA-seq data imputation. We compared I-Impute with the state-of-the-art imputation tools, by evaluating their imputation performance as well as their self-consistency. On the in silico data sets, I-Impute exhibited consistently the highest Pearson correlations for different dropout rates compared to SAVER and scImpute. Furthermore, several discrete cell subpopulations have been reported in scRNA-Seq data collected from the wet lab; the identification of subpopulations of cells is crucial [29]. Here, we collected three wetlab datasets, mouse bladder cells dataset, embryonic stem (ES) cells dataset, and aortic leukocyte cells dataset to evaluate the tools. I-Impute exhibited feasible cell subpopulation discovery efficacy on all the three datasets. It achieves the highest clustering accuracy compared to SAVER and scImpute.

Results

Evaluating the self-consistency of existing imputation tools in synthetic data

To evaluate the imputation tools, we applied the R package Splatter [30] to generate scRNA-seq reads count data.

Table 1 Self-consistency on synthetic data. NA denotes not applicable. $\theta = 0.1$

	SAVER	scImpute	I-Impute
88.45% dropout	0.5613	7.3460	0.0936
Self-consistent ($< \theta$)	x	x	✓
63.29% dropout	1.0245	0.2392	0.0806
Self-consistent ($< \theta$)	x	x	✓
45.16% dropout	1.3561	0.2677	0.0381
Self-consistent ($< \theta$)	x	x	✓

We simulated 150 cells of three groups, each with 2,000 genes. Then we generated three sparse matrices by setting the dropout rates as 88.45%, 63.29%, and 45.16%; and their corresponding zero rates are 90.87%, 70.98%, and 56.65%, respectively.

We first validated whether the existing imputation tools are self-consistent. We consider the imputation process as a complex function $f : x \rightarrow x$ that maps the zero-inflated matrix into an output matrix of the same shape. We say that f is *self-consistent* if and only if the root mean square error (RMSE) between x and $f(x)$ is less than a pre-determined threshold θ, that is, $||x - f(x)||_2 \leq \theta$. The results are shown in Table 1. We found that SAVER and scImpute are not self-consistent. scImpute has RMSE values of 7.346 at 88.45% dropout data, 0.2392 at 63.29% dropout data, and 0.2677 at 45.16% dropout data. For these data sets, SAVER has RMSE value of 0.5613, 1.0245, and 1.3561 respectively. Nevertheless, when ground truth group labels are incorporated, traditional evaluation metrics show SAVER to outperformed scImpute with respect to adjusted Rand index (ARI), normalized mutual information (NMI), and silhouette width (SW) (see Additional file 1, Table S1).

Our tool, I-Impute, is constructed on both the principle of self-consistency as well as to optimizing the existing imputation metrics (ARI, NMI, and SW). As illustrated in Fig. 1a, I-Impute first calls an internal subroutine (called C-Impute), which uses continuous similarities and dropout probabilities to infer missing entries. Then, I-Impute invokes SAVER as a subroutine to preprocess the data. Finally, it deploys C-Impute iteratively on the processed data (see Fig. 1b). As illustrated in Additional file 1, Fig. S1, after some number of iterations, the RMSE of I-Impute approaches to below 0.1, which is much smaller than SAVER and scImpute. That is, assume $\theta = 0.1$, the imputed result converges to a self-consistent matrix, with RMSE values of 0.0936, 0.0806, and 0.0381 in three synthetic datasets, respectively (see Table 1).

I-Impute recovers gene expression affected by dropouts in synthetic data

To validate the performance of I-Impute, we plotted the heatmap of the raw matrix, the 88.45% dropout matrix, and the imputed matrices, respectively (see Fig. 2a-f). I-Impute's output are closest to the raw matrices, compared to SAVER, scImpute, and C-Impute. As illustrated in

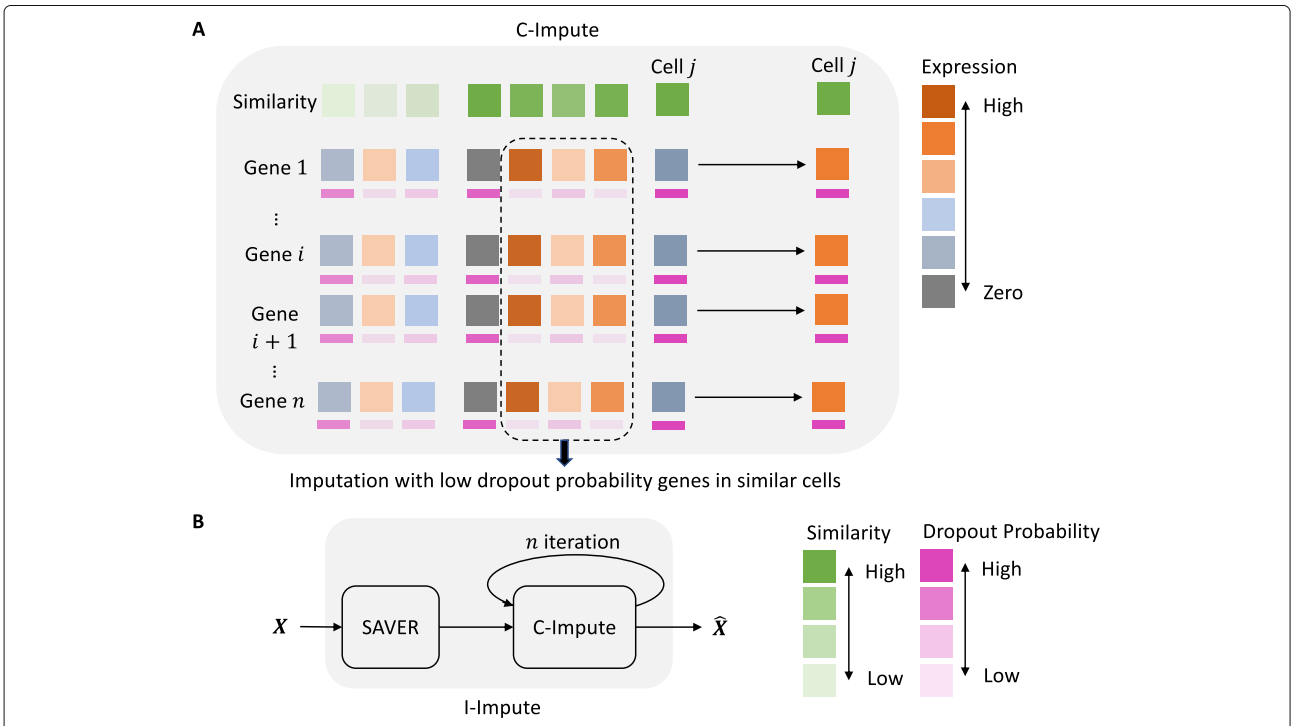

Fig. 1 Illustration of I-Impute architecture

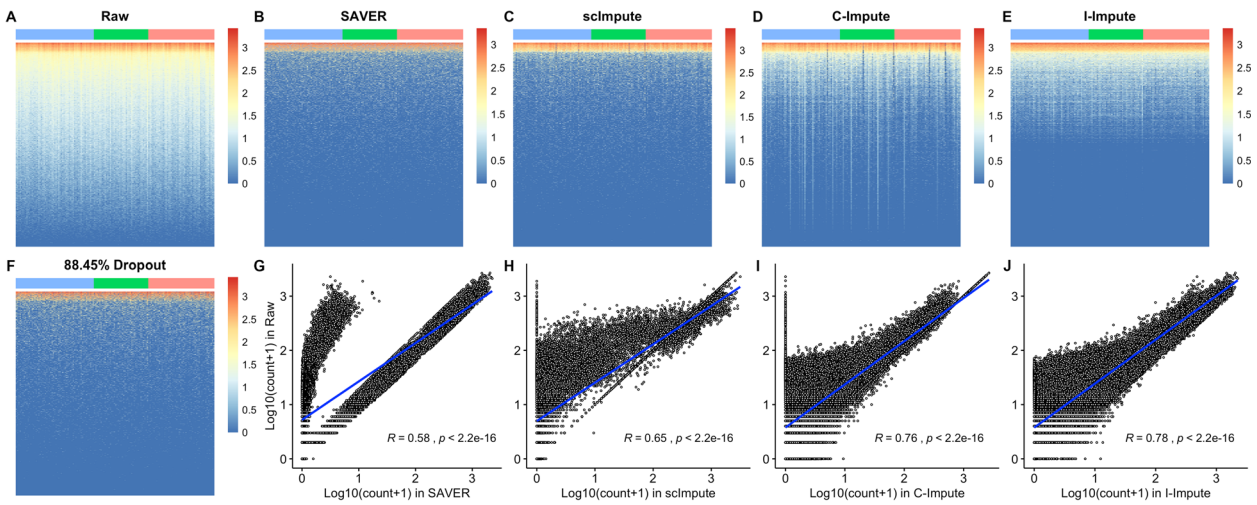

Fig. 2 Imputation performance on synthetic data. **a**-**f** Heatmap plots. Blue, green, and red tiles represent different cell groups **g**-**j** Scatter plots, Pearson correlation between ground-truth entries and imputed values are calculated

Fig. 2g, SAVER failed in reproducing many entries in the raw matrices, leading to the lowest Pearson correlation 0.58 between its output and the ground truth. scImpute and C-Impute changed some highly expressed elements into zero, hence introducing new bias after imputation (see Fig. 2h-i). With no extreme pull-down or pull-up prediction, I-Impute exhibited the most robust recovery power, with the highest Pearson correlation 0.78 (see Fig. 2j). On data with 63.29% and 45.16% dropout rate, I-Impute also gave the highest Pearson correlation of 0.90 and 0.94, respectively (see Additional file 1, Table S4).

The t-SNE embedding plots of the raw matrix, 88.45% dropout matrix, and recovered matrices show that SAVER, C-Impute, and I-Impute recover the missing entries, while preserving cell subgroups structures well (see Fig. 3a-f). Silhouette width (SW) further validated that the in-group similarity and out-group separation were enhanced after the imputation by SAVER, C-Impute, and I-Impute. That is, the average silhouette value increased from 0.0862 (dropout data) to 0.1075 (SAVER), 0.1705 (C-Impute), and 0.2429 (I-Impute), respectively (see Additional file 1, Table S1). Figure 3g demonstrates

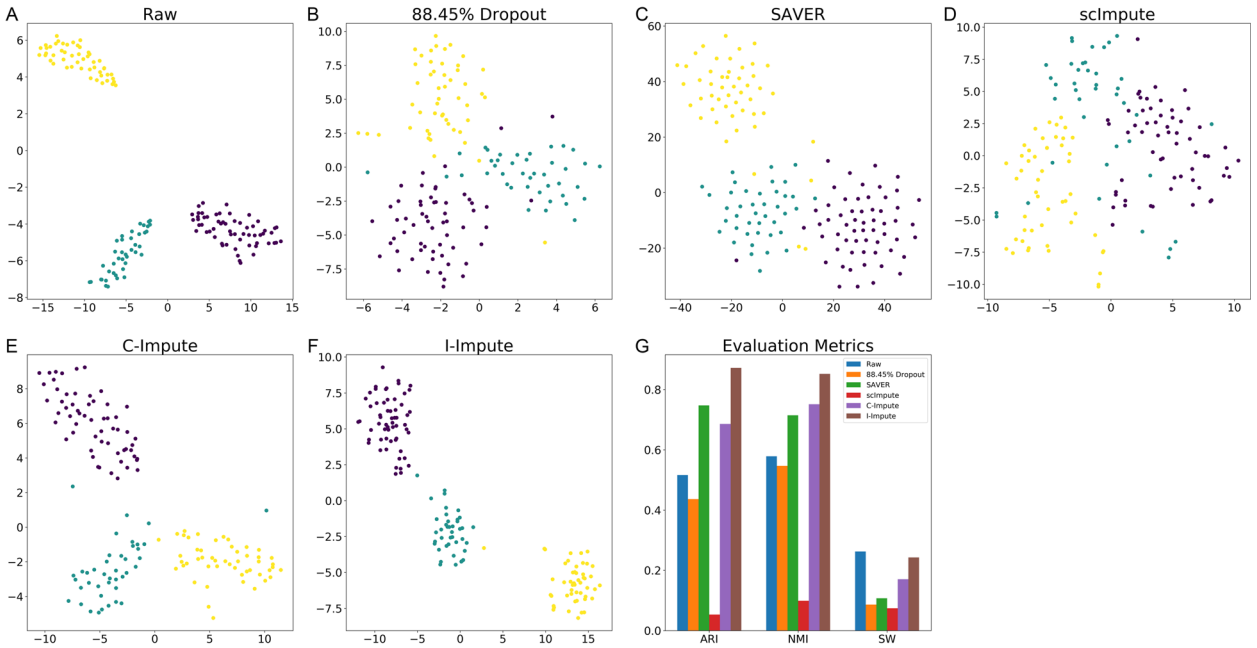

Fig. 3 Imputation performance on synthetic data. **a**-**f** t-SNE plots **g** evaluation metrics

that I-Impute achieves the most noticeable improvement, while scImpute illustrates lower SW values than dropout data. Next, we applied hierarchical clustering into all matrices, and computed the adjusted Rand index (ARI) and normalized mutual information (NMI) to evaluate the clustering accuracy. ARI and NMI measure the overlap between the inferred groups and ground-truth clusters; a score of 0 implies random labeling while 1 indicates perfect inference. In Fig. 3g, I-Impute outperforms all other tools and exhibits the best sub-population identification strength, with the highest clustering accuracy (ARI: 0.8721, NMI: 0.8521, see Additional file 1, Table S1). Experiments on data sets of 63.2% and 45.16% dropout rate also proved that I-Impute produced the best recovered matrices; with ARI 1.0, NMI 1.0, SW 0.3908 for 63.2% dropout rate, and ARI 0.9801, NMI 0.9710, and SW 0.4123 for 45.16% dropout rate (see Additional file 1, Table S2-S3).

Overall, the synthetic experiment demonstrates that by incorporating C-Impute to refine the SAVER processed data iteratively, I-Impute is able to mitigate the inconsistency in SAVER's result and this resulted in improved imputation.

I-Impute promotes cell subpopulation identification in real data sets

To examine the effects of I-Impute on the identification of cell sub-populations, we performed tests on three real scRNA-Seq datasets. The first test involves a dataset of mouse Bladder cells which contains 162 cells of three cell types. Due to dropout events, 73.5% of the read counts in the raw count matrix are zeros. We evaluated the imputation power by reviewing the tSNE embedding result and silhouette width (SW). ScImpute mixes part of Unknown-type cells (purple dots) with the Fibroblasts-1 cells (blue dots) and Fibroblasts-2 cells (yellow dots); SAVER, C-Impute, and I-Impute distinguish the Unknown-type cells from Fibroblasts-1 cells and Fibroblasts-2 cells well. Compared with raw and other imputed data, I-Impute produced the most compact clusters with highest silhouette width of 0.1758 (Fig. 4a). We then compared the hierarchical clustering accuracy, ARI and NMI. Both measurements show that with 0.6054 ARI and 0.7892 NMI, I-Impute resulted in the best clustering (ARI:0.1937, NMI:0.45), compared to those based on the imputations by SAVER (ARI:0.5253, NMI:0.7085), scImpute (ARI:0.1937, NMI:0.45), or C-Impute (ARI:0.1664, NMI:0.4317) (Fig. 4a, Additional file 1, Table S5).

We next tested the tools on a mouse embryonic stem (ES) cells dataset. This dataset contains 2717 cells of four cell types (mouse ES cells sample 1, mouse ES cells LIF 2 days, mouse ES cells LIF 4 days and mouse ES cells LIF 7 days). Due to the high running time of scImpute on large cells dataset, we randomly selected 200 cells and no sub-populations and genes were excluded during this process. Due to dropout events, 67.0% of read counts in the raw count matrix are zeros. Figure 4b shows that SAVER and I-Impute achieved overwhelmingly better imputation power than other tools. In the 2D t-SNE embedding space, the results from SAVER and I-Impute both separate the 2 days cells (the yellow dot) from the 4 days cells (the green dots) and the 7 days cells (the blue dots) well. From the Silhouette width, adjusted Rand index, and normalized mutual information, we found that I-Impute (ARI:0.7047, NMI:0.7444, SW:0.2275) produced a tighter and more accurate in-cluster structure than SAVER (ARI:0.692, NMI:0.7329, SW:0.2235)(Additional file 1, Table S6). Hence I-Impute was able to allow identification of the cell sub-populations in spite of the 67.0% missing rate.

Finally, we performed test with a mouse Aortic Leukocyte cells dataset. This dataset contains 378 cells of six cell types (B cells, T cells, T memory cells, Macrophages, Nuocytes, and Neutrophils). Due to dropout events, 91.2% of read counts in the raw count matrix are zeros. Both SAVER and I-Impute grouped the T memory cells (the yellow dots) into big cluster, while in raw data and other imputed matrices, T memory cells are separated into different clusters (see Fig. 4c). In this test, I-Impute gave a silhouette width of 0.0711, which is poorer than the result from SAVER. Nevertheless, I-Impute outperformed all other tools in hierarchical clustering tasks with the highest ARI (0.522) and NMI (0.7728) (Additional file 1, Table S7).

Discussion

In this paper, we introduced I-Impute, which is designed to impute scRNA missing entries iteratively. Experiments using synthetic and real data demonstrated I-Impute to be particularly suited for cell subpopulation discovery.

There are some advantages of I-Impute compared with scImpute and SAVER. First, I-Impute produces results which will be treated consistently when they are given back as input, and the imputed matrix are of tighter hierarchical structure. Second, scImpute requires the user to decide the cell groups number K and assign cells in the same group equal weights during imputation, whereas I-Impute does not require such a hyper-parameter K but instead builds a continuous affinity matrix by leveraging on the Gaussian kernel. Last but not least, Lasso regression makes unimportant weights zero, which can help to filter the distant cells for the regression.

Concerning the hyper-parameter pruning, the parameter t denotes the threshold of dropout probabilities. We have conducted experiments to guide the pruning. The result in Additional file 1, Fig. S2 suggests that the value of parameter t should not be too small, and $t = 0.5$ is adequate as the default setting.

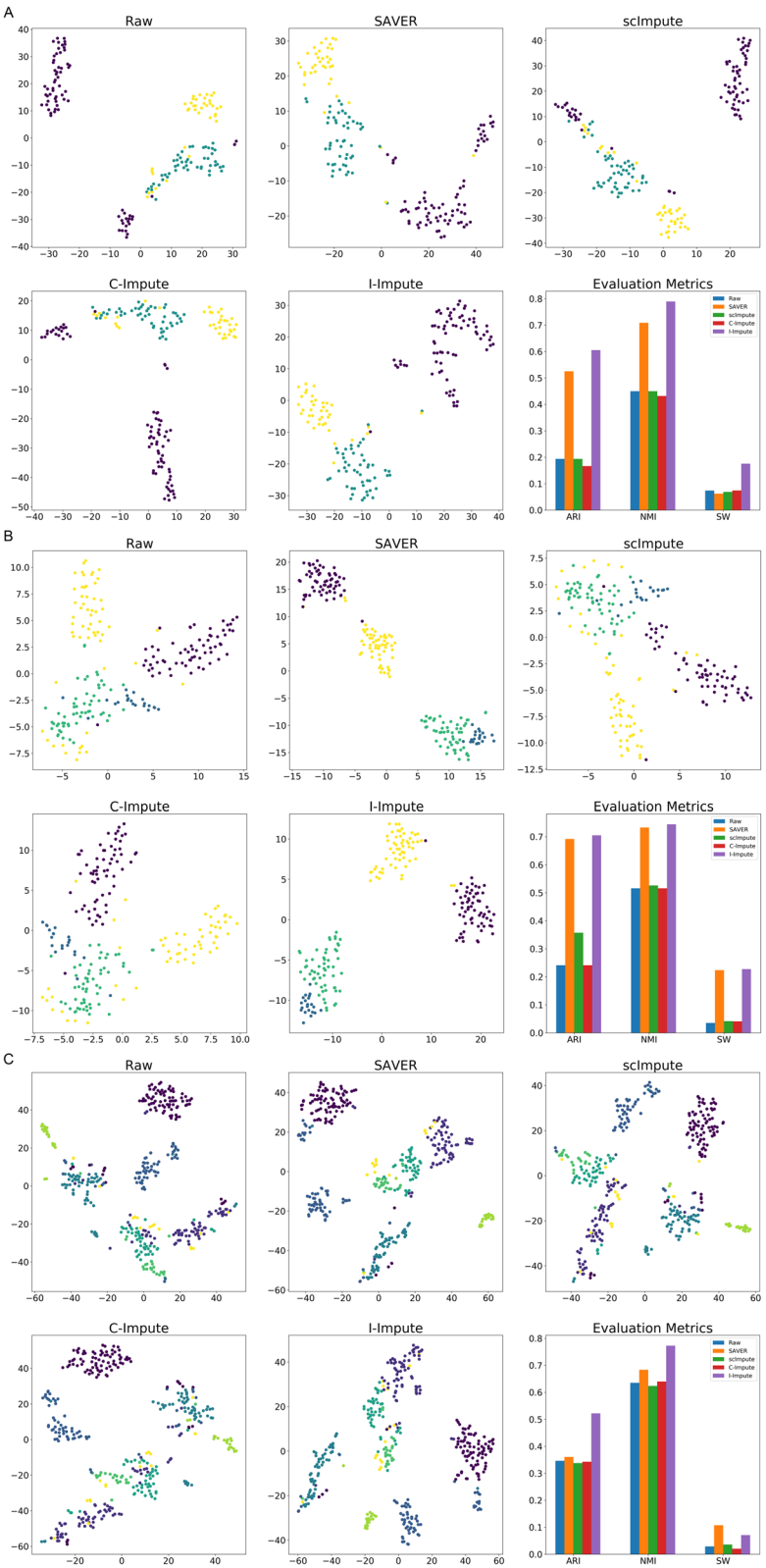

Fig. 4 Imputation performance on real datasets. **a**-**c** t-SNE plots and evaluation metrics for mouse bladder cells, embryonicstem cells, and aortic leukocyte cells, respectively

Conclusions

Imputation is an essential step in the use of scRNA-seq. In this work we introduced an imputation criterion called self-consistency and demonstrated the effectiveness of this criterion with an iterative imputation tool called I-Impute. Experiments on simulation data and real data sets showed I-Impute to be highly feasible in imputation and in the discovery of cell sub-population.

Methods

C-Impute

I-Impute utilizes a subroutine called C-Impute, which performs imputation with an objective function based on continuous similarity and Lasso penalty (see **Fig.** 1a). The following describes this subroutine.

Data prepossessing

The input of C-Impute is a count matrix $\dot{\boldsymbol{X}}^C \in M \times N_{total}$ which contains rows as genes and columns as cells, where M and N_{total} represent the total number of genes and cells correspondingly. The dropout values are replaced by zero counts.

First, C-Impute performs normalization, dimension reduction, and outlier removal as in scImpute [18]. This results in a matrix $\boldsymbol{X} \in M \times N$ and $\boldsymbol{Z} \in K \times N$, where K is the reduced dimensionality of metagenes, N is the number of remained cells.

Affinity matrix constructing

From $\boldsymbol{Z}$, a cell affinity matrix $\boldsymbol{A} \in N \times N$ is computed with Euclidean distance and Gaussian Kernel:

$$\mathrm{Dist}(i,j) = \left|\left|\boldsymbol{Z}_i^\top - \boldsymbol{Z}_j^\top\right|\right|_F^2 \tag{1}$$

$$\sigma_i = \mathrm{Dist}(i,k) \tag{2}$$

$$\boldsymbol{A}_{ij} = \begin{cases} \exp^{-\frac{\mathrm{Dist}(i,j)}{2\sigma_i^2}}, \ \mathrm{Dist}(i,j) \le \sigma_i, \\ 0, \ \mathrm{Dist}(i,j) > \sigma_i. \end{cases} \tag{3}$$

where i, j represent two different cell indices, $\boldsymbol{Z}_i^\top$ and $\boldsymbol{Z}_j^\top$ indicate the principle components of i-th and j-th cell respectively, $||\cdot||_F$ is the Frobenius norm. For the i-th cell, the kernel width will be set to the distance between it and its n-nearest neighbor, cell k, which stands for the cell whose distance to cell i is n-th smallest in all other cells, where n is a hyper-parameter.

Identification of dropout values and calculating dropout rate

With preprocessed gene expression matrix $\boldsymbol{X}$, we utilize a statistical model to infer which entries are influenced by the dropout effects. Instead of treating all zero values as missing entries, we use the Gamma-Normal mixture model to learn whether a zero observation originates from dropout or not. We use the Normal distribution to present the actual gene expression level and Gamma distribution to take the dropout events into account. Since the preprocessed matrix $\boldsymbol{X}$ is no longer of integral values, we cannot adopt zero-inflated negative binomial (ZINB) distribution.

For the i-th gene and its observed value x in prepossessed gene profiling $\boldsymbol{X}_i$, the Gamma-Normal mixture model will be:

$$\begin{aligned} &f_{\text{Gamma-Normal}}\left(x; \boldsymbol{\pi}_i, \boldsymbol{\alpha}_i, \boldsymbol{\beta}_i, \boldsymbol{\mu}_i, \boldsymbol{\sigma}_i\right) \\ =&\boldsymbol{\pi}_i \text{Gamma}\left(x; \boldsymbol{\alpha}_i, \boldsymbol{\beta}_i\right) + (1-\boldsymbol{\pi}_i)\,\text{Normal}\left(x; \boldsymbol{\mu}_i, \boldsymbol{\sigma}_i\right) \end{aligned} \tag{4}$$

where $\boldsymbol{\pi}_i$ is the dropout rate of gene i, $\boldsymbol{\alpha}_i$ and $\boldsymbol{\beta}_i$ is the shape and rate parameter of Gamma distribution respectively, $\boldsymbol{\mu}_i$ and $\boldsymbol{\sigma}_i$ are the mean and standard deviation of Normal distribution. The estimated model parameters $\hat{\boldsymbol{\pi}}$, $\hat{\boldsymbol{\alpha}}$, $\hat{\boldsymbol{\beta}}$, $\hat{\boldsymbol{\mu}}$, and $\hat{\boldsymbol{\sigma}}$ are obtained by Expectation-Maximization (EM) algorithm. Then, we can calculate the dropout probability matrix $\boldsymbol{D} \in M \times N$.

$$\boldsymbol{D}_{ij} = \frac{\boldsymbol{\pi}_i \text{Gamma}\left(\boldsymbol{X}_{ij}; \boldsymbol{\alpha}_i, \boldsymbol{\beta}_i\right)}{f_{\text{Gamma-Normal}}\left(\boldsymbol{X}_{ij}; \boldsymbol{\pi}_i, \boldsymbol{\alpha}_i, \boldsymbol{\beta}_i, \boldsymbol{\mu}_i, \boldsymbol{\sigma}_i\right)} \tag{5}$$

This mixture model enables the identification of whether an observed value is a dropout value or not, since a zero value can be either caused by a technical error or may reflect the actual expression value. If a gene has high expression and low variation in most of its similar cells, a zero count will have high dropout probability and more likely to be a dropout value; otherwise, the zero value may exhibit real biological variability [18].

Imputation of dropout values

To impute the gene expression levels, we first define a hyper-parameter t which is used as the threshold to determine if $\boldsymbol{X}_{ij}$ is a dropout event. An entry of dropout probability less than t is considered a real observation, in which case its value is retained. Otherwise, while values with dropout probability higher than t will be replaced by imputation result. We perform imputation by linear regression weighted by dropout probability and cell affinity.

$$\hat{\boldsymbol{X}}_{ij} = \begin{cases} \boldsymbol{X}_{ij}, \boldsymbol{D}_{ij} < t, \\ \left(\left(1 - \boldsymbol{D}_{\bar{j}}^\top\right) \circ \left(\boldsymbol{A}_{j\bar{j}} \odot \boldsymbol{X}_{\bar{j}}^\top\right)\right) \boldsymbol{B}_j^\top, \boldsymbol{D}_{ij} \ge t. \end{cases} \tag{6}$$

where $\boldsymbol{D}_j^\top$ and $\boldsymbol{X}_j^\top$ are the j-th column of $\boldsymbol{D}$ and $\boldsymbol{X}$ respectively. The $\circ$ operator is the Hadamard product which follows $(\boldsymbol{P} \circ \boldsymbol{Q})_{ij} = \boldsymbol{P}_{ij}\boldsymbol{Q}_{ij}$. $\bar{j}$ denotes all indices except index j, thus $\boldsymbol{D}_{\bar{j}}^\top$ and $\boldsymbol{X}_{\bar{j}}^\top$ denotes the sub-matrix of $\boldsymbol{D}$ and $\boldsymbol{X}$ which contains all cells except the j-th cell, respectively. $\boldsymbol{A}_{j\bar{j}}$ stores the pairwise affinity between j-th cell and all other cells; $\boldsymbol{X}_{\bar{j}}^\top$ is a sub-matrix of $\boldsymbol{X}$ which contains all cells except the j-th cell. $\odot$ operator represents the vector and matrix multiplication, e.g. $(\boldsymbol{p} \odot \boldsymbol{Q})_{ij} = \boldsymbol{p}_i\boldsymbol{Q}_{ij}$. Leveraging

$\left(1-\boldsymbol{D}_j^\top\right)\circ\boldsymbol{X}_j^\top$ as target indicates that genes with high dropout probability in j-th cell will not contribute to optimization. Furthermore, the multiplication of $\left(1-\boldsymbol{D}_{\bar{j}}^\top\right)$ and $\boldsymbol{A}_{j\bar{j}}$ ensures that the information is only borrowed from the trusted genes with low dropout probabilities in the similar cells. Non-negative weights $\boldsymbol{B}_j^\top$ are extra contributions of all other cells learned from regression.

For j-th cell, the objective is:

$$\begin{aligned}\min_{\boldsymbol{B}_j^\top}\sum_{j=1}^{N}\Big|\Big|&\left(1-\boldsymbol{D}_j^\top\right)\circ\boldsymbol{X}_j^\top\\&-\left[\left(1-\boldsymbol{D}_{\bar{j}}^\top\right)\circ\left(\boldsymbol{A}_{j\bar{j}}\odot\boldsymbol{X}_{\bar{j}}^\top\right)\right]\boldsymbol{B}_j^\top\Big|\Big|_F^2\\&+\lambda\left|\left|\boldsymbol{B}_j^\top\right|\right|_1,\\\text{subject to }&\boldsymbol{B}_j^\top\geq 0\end{aligned}\tag{7}$$

$\mathcal{L}1$ is applied to avoid over-fitting and further ensure that the imputation borrow information from the cell's most similar neighbors.

Assume $y\in\mathbb{R}^M=\left(1-\boldsymbol{D}_j^\top\right)\circ\boldsymbol{X}_j^\top,\beta\in\mathbb{R}^N=\boldsymbol{B}_j^\top,X\in\mathbb{R}^{M\times N}=\left(1-\boldsymbol{D}_{\bar{j}}^\top\right)\circ\left(\boldsymbol{A}_{j\bar{j}}\odot\boldsymbol{X}_{\bar{j}}^\top\right)$, for each j-th cell we can simplify the objective to non-negative least squares lasso regression $\min_\beta\left|\left|y-X\beta\right|\right|_2^2+\lambda\left|\left|\beta\right|\right|_1,\beta\geq 0$, and solve it by coordinate descent [31].

I-Impute

As mentioned, I-Impute performs a self-consistent imputation on scRNA-seq data. The method is as illustrated in **Fig.** 1b. I-Impute utilizes C-Impute to iteratively refine SAVER processed data. After a few iterations, the result converges to a self-consistent matrix ($<\theta$) and is given as I-Impute's output.

We define self-consistency of a functional mapping $f:x\rightarrow x$ given by input data $\boldsymbol{X}\in M\times N$:

$$\begin{aligned}\boldsymbol{X}_{output}&=f(\boldsymbol{X})\\\text{self-consistency}\,(f;\boldsymbol{X})&=\frac{\left|\left|\boldsymbol{X}_{output}-f\left(\boldsymbol{X}_{output}\right)\right|\right|_F^2}{M\times N}\\\text{self-consistency}\,(f;\boldsymbol{X})&<\theta\rightarrow f\text{ is self-consistent}\end{aligned}\tag{8}$$

Evaluation metrics

Adjusted rand index and normalized mutual information

The adjusted Rand index (ARI) [32] and normalized mutual information (NMI) [33] are adopted as clustering accuracy. They measure the similarity between a clustering result and the actual clusters. A value close to 0 indicates random labeling or no mutual information, and a value of 1 demonstrates 100% consistency between the clustering and the actual clusters.

Silhouette width

The silhouette width (SW) measures the similarity of a sample to its class compared to other categories [34]. It ranges from -1 to 1. A higher silhouette value suggests a more appropriate clustering. A silhouette value near 0 indicates overlapping clusters and a negative value indicates that the clustering has been performed incorrectly. We adopted the silhouette width to evaluate the model's imputation power. We used the ground-truth subtype classes as the input cluster labels.

Simulation and benchmark settings

Splatter are used to generate simulated scRNA-seq data. The parameters used for our simulation dataset are nGroups=3, nGenes=2000, batchCells=150, seeds=42, dropout.type="experiment", dropout.shape=-1 and dropout.mid=2, 3, 5 for three different dropout rate data.

SAVER and scImpute are the state-of-the-art tools which I-Impute is compared against. For the SAVER R package, we used the "saver" function with the parameters ncores=12 and estimates.only=TRUE to perform the imputation tasks. The parameters for scImpute are drop_thre=0.5, ncores=10, Kclusters=(number of true clusters in input data).

On synthetic data, I-Impute configuration is n=40, normalize=False, and iteration=True. On real data sets, I-Impute configuration is n=40, and iteration=True when tested with the mouse Bladder cell dataset and ES cell dataset, and is n=20, and iteration=True when tested with the mouse Aortic Leukocyte cell dataset.

Supplementary information

Supplementary information accompanies this paper at https://doi.org/10.1186/s12864-020-07007-w.

Additional file 1: The PDF file includes all the supporting materials for the manuscript.

Abbreviations

scRNA-seq: Single-cell RNA-sequencing; ARI: Adjusted Rand Index; NMI: Normalized Mutual Information; SW: Silhouette Width; RMSE: Root Mean Square Error

Acknowledgements

We would like to express sincere gratitude to Yen Kaow Ng from Kotai Biotechnologies, Japan for manuscript revision.

About this supplement

This article has been published as part of *BMC Genomics Volume 21 Supplement 10, 2020: Selected articles from the 18th Asia Pacific Bioinformatics Conference (APBC 2020): genomics*. The full contents of the supplement are available online at https://bmcgenomics.biomedcentral.com/articles/supplements/volume-21-supplement-10.

Authors' contributions

SCL conceived the idea and supervised the project. XF, LC, ZW, SCL discussed the algorithm and designed the experiments. XF implemented the code and conducted the analysis. LC, XF drafted the manuscript. ZW, SCL revised the manuscript. All author(s) read and approved the final manuscript.

Funding
This work and publication costs are funded by the GRF Research Projects 9042348 (CityU 11257316). The funding body did not play any role in the design of the study and collection, analysis, and interpretation of data and in writing the manuscript.

Availability of data and materials
The real scRNA-seq datasets analysed during the current study are all publicly available. The mouse ES cell dataset [35] was downloaded from the Gene Expression Omnibus (GEO) with the accession code GSE65525. The mouse Bladder cell dataset and Aortic Leukocyte cell dataset were downloaded from the PanglaoDB [36] with the accession code SRS3044239 and SRS2747908 respectively. The Python package I-Impute is freely available at https://github.com/xikanfeng2/I-Impute.

Ethics approval and consent to participate
Not applicable.

Consent for publication
Not applicable.

Competing interests
The authors declare that they have no competing interests.

Author details
[1]School of Software, Northwestern Polytechnical University, 710072 Xi'an, Shaanxi, China. [2]Department of Computer Science, City University of Hong Kong, Tat Chee Avenue, Kowloon, Hong Kong, China. [3]Department of Biomedical Engineering, City University of Hong Kong, Tat Chee Avenue, Kowloon, Hong Kong, China.

Published: 18 November 2020

References

1. McDavid A, Finak G, Chattopadyay PK, Dominguez M, Lamoreaux L, Ma SS, Roederer M, Gottardo R. Data exploration, quality control and testing in single-cell qpcr-based gene expression experiments. Bioinformatics. 2012;29(4):461–7.
2. Saliba A-E, Westermann AJ, Gorski SA, Vogel J. Single-cell rna-seq: advances and future challenges. Nucleic Acids Res. 2014;42(14):8845–60.
3. Vallejos CA, Marioni JC, Richardson S. Basics: Bayesian analysis of single-cell sequencing data. PLoS Comput Biol. 2015;11(6):1004333.
4. Kolodziejczyk AA, Kim JK, Svensson V, Marioni JC, Teichmann SA. The technology and biology of single-cell rna sequencing. Mol Cell. 2015;58(4):610–20.
5. Liu S, Trapnell C. Single-cell transcriptome sequencing: recent advances and remaining challenges. F1000Research. 2016;5:182.
6. Trapnell C, Cacchiarelli D, Grimsby J, Pokharel P, Li S, Morse M, Lennon NJ, Livak KJ, Mikkelsen TS, Rinn JL. The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering of single cells. Nat Biotechnol. 2014;32(4):381.
7. Liu Z, Lou H, Xie K, Wang H, Chen N, Aparicio OM, Zhang MQ, Jiang R, Chen T. Reconstructing cell cycle pseudo time-series via single-cell transcriptome data. Nat Commun. 2017;8(1):22.
8. Horning AM, Wang Y, Lin C-K, Louie AD, Jadhav RR, Hung C-N, Wang C-M, Lin C-L, Kirma NB, Liss MA, et al. Single-cell rna-seq reveals a subpopulation of prostate cancer cells with enhanced cell-cycle–related transcription and attenuated androgen response. Cancer Res. 2018;78(4):853–64.
9. Baruch K, Deczkowska A, Rosenzweig N, Tsitsou-Kampeli A, Sharif AM, Matcovitch-Natan O, Kertser A, David E, Amit I, Schwartz M. Pd-1 immune checkpoint blockade reduces pathology and improves memory in mouse models of alzheimer's disease. Nat Med. 2016;22(2):135.
10. Segerstolpe Å, Palasantza A, Eliasson P, Andersson E-M, Andréasson A-C, Sun X, Picelli S, Sabirsh A, Clausen M, Bjursell MK, et al. Single-cell transcriptome profiling of human pancreatic islets in health and type 2 diabetes. Cell Metab. 2016;24(4):593–607.
11. Lawlor N, George J, Bolisetty M, Kursawe R, Sun L, Sivakamasundari V, Kycia I, Robson P, Stitzel ML. Single-cell transcriptomes identify human islet cell signatures and reveal cell-type–specific expression changes in type 2 diabetes. Genome Res. 2017;27(2):208–22.
12. Chung W, Eum HH, Lee H-O, Lee K-M, Lee H-B, Kim K-T, Ryu HS, Kim S, Lee JE, Park YH, et al. Single-cell rna-seq enables comprehensive tumour and immune cell profiling in primary breast cancer. Nat Commun. 2017;8:15081.
13. Karaayvaz M, Cristea S, Gillespie SM, Patel AP, Mylvaganam R, Luo CC, Specht MC, Bernstein BE, Michor F, Ellisen LW. Unravelling subclonal heterogeneity and aggressive disease states in tnbc through single-cell rna-seq. Nat Commun. 2018;9(1):3588.
14. Guo X, Zhang Y, Zheng L, Zheng C, Song J, Zhang Q, Kang B, Liu Z, Jin L, Xing R, et al. Global characterization of t cells in non-small-cell lung cancer by single-cell sequencing. Nat Med. 2018;24(7):978.
15. Kim C, Gao R, Sei E, Brandt R, Hartman J, Hatschek T, Crosetto N, Foukakis T, Navin NE. Chemoresistance evolution in triple-negative breast cancer delineated by single-cell sequencing. Cell. 2018;173(4):879–93.
16. Bartoschek M, Oskolkov N, Bocci M, Lövrot J, Larsson C, Sommarin M, Madsen CD, Lindgren D, Pekar G, Karlsson G, et al. Spatially and functionally distinct subclasses of breast cancer-associated fibroblasts revealed by single cell rna sequencing. Nat Commun. 2018;9(1):5150.
17. Kharchenko PV, Silberstein L, Scadden DT. Bayesian approach to single-cell differential expression analysis. Nat Methods. 2014;11(7):740.
18. Li WV, Li JJ. An accurate and robust imputation method scimpute for single-cell rna-seq data. Nat Commun. 2018;9(1):997.
19. Xu C, Su Z. Identification of cell types from single-cell transcriptomes using a novel clustering method. Bioinformatics. 2015;31(12):1974–80.
20. Lin P, Troup M, Ho JW. Cidr: Ultrafast and accurate clustering through imputation for single-cell rna-seq data. Genome Biol. 2017;18(1):59.
21. Satija R, Farrell JA, Gennert D, Schier AF, Regev A. Spatial reconstruction of single-cell gene expression data. Nat Biotechnol. 2015;33(5):495.
22. Pierson E, Yau C. Zifa: Dimensionality reduction for zero-inflated single-cell gene expression analysis. Genome Biol. 2015;16(1):241.
23. Huang M, Wang J, Torre E, Dueck H, Shaffer S, Bonasio R, Murray JI, Raj A, Li M, Zhang NR. Saver: gene expression recovery for single-cell rna sequencing. Nat Methods. 2018;15(7):539.
24. Deng Y, Bao F, Dai Q, Wu LF, Altschuler SJ. Scalable analysis of cell-type composition from single-cell transcriptomics using deep recurrent learning. Nat Methods. 2019;16(4):311.
25. Lopez R, Regier J, Cole MB, Jordan MI, Yosef N. Deep generative modeling for single-cell transcriptomics. Nat Methods. 2018;15(12):1053.
26. Eraslan G, Simon LM, Mircea M, Mueller NS, Theis FJ. Single-cell rna-seq denoising using a deep count autoencoder. Nat Commun. 2019;10(1):390.
27. Van Buuren S, Van Rijckevorsel JL. Imputation of missing categorical data by maximizing internal consistency. Psychometrika. 1992;57(4):567–80.
28. Liang F, Jia B, Xue J, Li Q, Luo Y. An imputation–regularized optimization algorithm for high dimensional missing data problems and beyond. J R Stat Soc Ser B Stat Methodol. 2018;80(5):899–926.
29. Wang Y, Hoinka J, Przytycka TM. Subpopulation detection and their comparative analysis across single-cell experiments with scpopcorn. Cell Syst. 2019;8:506–13.
30. Zappia L, Phipson B, Oshlack A. Splatter: simulation of single-cell rna sequencing data. Genome Biol. 2017;18(1):174.
31. Friedman J, Hastie T, Tibshirani R. Regularization paths for generalized linear models via coordinate descent. J Stat Softw. 2010;33(1):1.
32. Rand WM. Objective criteria for the evaluation of clustering methods. J Am Stat Assoc. 1971;66(336):846–50.
33. Cover TM, Thomas JA. Elements of Information Theory, vol. 68. New York: Wiley; 1991, pp. 69–73.
34. Rousseeuw PJ. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. J Comput Appl Math. 1987;20:53–65.
35. Klein AM, Mazutis L, Akartuna I, Tallapragada N, Veres A, Li V, Peshkin L, Weitz DA, Kirschner MW. Droplet barcoding for single-cell transcriptomics applied to embryonic stem cells. Cell. 2015;161(5):1187–201.
36. Franzén O, Gan L-M, Björkegren JL. Panglaodb: a web server for exploration of mouse and human single-cell rna sequencing data. Database. 2019;2019:baz046.

Publisher's Note