RIA: a novel Regression-based Imputation Approach for single-cell RNA sequencing

Bang Tran
Computer Science & Engineering
University of Nevada, Reno
Reno, USA
bang.t.s@nevada.unr.edu

Duc Tran
Computer Science & Engineering
University of Nevada, Reno
Reno, USA
duct@nevada.unr.edu

Hung Nguyen
Computer Science & Engineering
University of Nevada, Reno
Reno, USA
hungnp@nevada.unr.edu

Nam Sy Vo
Computational Biomedicine
Vingroup Big Data Institute
Hanoi, Vietnam
v.namvs@vintech.net.vn

Tin Nguyen*
Computer Science & Engineering
University of Nevada, Reno
Reno, USA
tinn@unr.edu

***Abstract*—Advances in single-cell technologies have shifted genomics research from the analysis of bulk tissues toward a comprehensive characterization of individual cells. This holds enormous opportunities for both basic biology and clinical research. As such, identification and characterization of short-lived progenitors, stem cells, cancer stem cells, or circulating tumor cells are essential to better understand both normal and diseased tissue biology. However, quantifying gene expression in each cell remains a significant challenge due to the low amount of mRNA available within individual cells. This leads to the excess amount of zero counts caused by dropout events. Here we introduce RIA, a regression-based approach, that is able to reliably recover the missing values in single-cell data and thus can effectively improve the performance of downstream analyses. We compare RIA with state-of-the-art methods using five scRNA-seq datasets with a total of 3,535 cells. In each dataset analyzed, RIA outperforms existing approaches in improving the identification of cell populations while preserving the biological landscape. We also demonstrate that RIA is able to infer temporal trajectories of embryonic development stages.**

***Index Terms*—single cell, scRNA-seq, imputation, sequencing**

I. Introduction

Single-cell RNA sequencing (scRNA-seq) was first known in 2009 when Tang et al. [1] monitored how individual cells respond to signals and other environmental cues at critical stages of cell-fate. However, scRNA-seq had not gain major attention until 2014 when sequencing cost became more affordable. Since then, a number of scRNA-seq protocols have been developed to isolate single cells and to prepare cDNA libraries using next generation sequencing (NGS) platforms [2, 3]. These advancements in single-cell sequencing hold enormous opportunities for both basic biology and clinical applications. For example, scRNA-seq disclosed diverse characteristic of cells within a seemingly analogous cell population or tissue, and revealed insights into cell identity, cell fate, and cellular functions [4]. Single-cell data was also used to detect highly variable genes (HVGs) that contribute for heterogeneity across cells in a cell population, to discover the relationship between genes and cellular phenotypes, or to identify new rare cell types via dimensionality reduction and clustering [5].

However, scRNA-seq data come with additional challenges [6, 7]. One challenge is that sequencing mRNA within individual cells requires artificial amplification of DNA materials millions of times, leading to disproportionate distortions of relative transcript abundance and gene expression. Another outstanding challenge is the "dropout" phenomenon where a gene was highly expressed in one cell but did not express at all in another cell [8]. These dropout events usually occur due to the limitation of sequencing technologies when only a low amount of starting mRNA in individual cells can be captured, leading to low sequencing depth and failed amplification [9–11]. Since downstream analyses of scRNA-seq is heavily relied on expression measurement's accuracy, it is very crucial to impute the false zero expression introduced by dropout phenomenon and sequencing error.

There have been a number of imputation methods developed to address this challenge for single-cell data. MAGIC [12, 13] was one of the first imputation method that is able to impute single-cell data on a genomic scale. MAGIC imputes zero expression value by using heat diffusion [14] concept. It first constructs the affinity matrix between cells using Gaussian kernel and then constructs a Markov transition matrix by normalizing the sc-RNA similarity matrix. Next, the weights of the other cells are estimated by the transition matrix. Another method is DrImpute [15], which is based on the cluster ensemble strategy [16] using consensus clustering [17, 18] as the basic clustering algorithm. It performs clustering for a predefined number of times and imputes the data by averaging value of similar cells. If the number of clusters is not provided by users, DrImpute will use some default values that might not be optimal for the data. The major drawback of MAGIC and

978-1-7281-3003-3/19/$31.00 ©2019 IEEE

DrImpute is that they rely on many parameters to fine-tune their model, which often leads to overfitting. This makes their results unreliable, i.e., the imputation is sensitive to a slight change in the input data or in parameter settings.

SAVER [19] and scImpute [20] are statistical methods that model dropouts in scRNA dataset as a mixture of different distributions. scImpute models the gene expression as a mixture of two different distributions: the Gaussian distribution represents the actual gene expression while the Gamma distribution accounts for the dropout events. scImpute estimates the parameters of the mixture model using the Expectation-Maximization (EM) algorithm [21]. Genes with a high dropout rate are consider imputable while genes with low dropout rate do not need imputation. The method then uses a non-negative least square to impute genes with high dropout rates. Similarly, SAVER [19] models read counts as a mixture of Poisson-Gamma and then uses a Bayesian approach to estimate true expression values of genes by borrowing information across genes. Similar to SAVER and scImpute, BISCUIT [22] uses the Dirichlet process mixture model [23] to repeatedly perform the processing steps such as normalization, sc-RNA data imputation, and cells clustering by simultaneously inferring clustering parameters, estimating technical variations (e.g. library size), and learning co-expression structures of each cluster.

Despite initial success, these statistical methods have some important limitations that need to be addressed. First, the EM-based strategy involves estimation of many parameters for all genes across the whole genome. This makes the methods very slow and vulnerable to overfitting. Second, these methods attempt to alter the expression of all genes, including those that are not affected by dropout events.

Here we propose a new approach, RIA, that can reliably impute missing values from single-cell data. Our method consists of two modules. The first module performs a hypothesis testing to identify the values that are likely to be impacted by the dropout events. The second module estimates the missing value using a robust regression approach. All of the parameters are learned from the data themselves. The approach is tested using five benchmarking datasets with a total of 3,535 cells. We demonstrate that RIA outperforms existing imputation methods in improving the identification of cell population and temporal trajectories.

II. Methods

Figure 1 shows the overall analysis pipeline of RIA. The input of RIA is a matrix in which rows represent genes/components and columns represent cells/samples. RIA first performs a hypothesis testing to determine genes that have accurate values with high confidence. These genes are will be used as the training set. The rest of the genes (genes that need to be imputed) will be the imputable set. The method then uses a generalized linear model to learn from the training set and to impute the missing data in the imputable set. Finally, RIA concatenates the two sets of genes and outputs a matrix that has the same number of rows and columns as of the input matrix.

A. Hypothesis Testing and Identification of Dropout

In order to impute the missing data without introducing false signals to the original data, it is important to determine which genes are impacted by dropouts and which genes do not need imputation. Therefore, we have developed a hypothesis testing approach to determine the set of genes that are likely to be impacted by dropouts.

Our approach is based on the observation that for genes that are not impacted by dropouts, the log-transformed expression values are normally distributed [20, 24]. Therefore, we use z-test to determine whether a zero value is observed by chance or by the impact of dropout events. For each gene g, we use the non-zero expression values to determine the parameters μ and σ of the Gaussian distribution. Next, we use z-test to estimate how likely a zero value occurs, given that the expression values follow the estimated Gaussian distribution. If the chance of observing a zero value is less than the significance threshold (0.05), we conclude that gene g is likely to be affected by dropout. By repeating this process for all genes, we can divide our data into two sets of genes: a set G that include genes affected by dropout, and a set M that have high confidence of not being affected by dropout.

B. Regression-based Imputation

Based on the hypothesis testing described above, we divide the data into two groups of genes: i) a group G in which all of the genes are likely to be affected by dropouts (imputable set), and ii) a group of genes M that have accurate gene expression that do not need imputation (training set). The linear regression process consists of two steps. The first step is to select genes from the training set that are highly correlated with the gene we need to impute. In the second step, we train the linear model using these highly-correlated genes and then estimate the missing values.

For a gene $g \in G$ (imputable set), let us denote y as the non-zero part of g. In the first step we calculate the Pearson correlation coefficient of y with the corresponding values of every gene in M (training set). We then select 10 genes from M with the highest correlation coefficients. Denoting $\{m_{i_1}, \ldots, m_{i_{10}}\}$ as the selected genes in M, we have $\{x_{i_1}, \ldots, x_{i_{10}}\}$ as the vectors obtained from $\{m_{i_1}, \ldots, m_{i_{10}}\}$ that are highly correlated with y. Note that each vector x_{i_j} is a part of m_{i_j}. We train the generalized linear model in which $\{x_{i_1}, \ldots, x_{i_{10}}\}$ are the predictor variables and y is the outcome variable. In our implementation, we adopt the lm function that is available in the *stats* package. Next, we use the trained linear model to estimate the missing values in g, using $\{m_{i_1} \backslash x_{i_1}, \ldots, m_{i_{10}} \backslash x_{i_{10}}\}$ as the predictors, where $m_{i_j} \backslash x_{i_j}$ is that part of m_{i_j} that do not belong to x_{i_j}.

III. Results

Here we assess the performance of RIA using five single-cell datasets that are available in NIH Gene Expression

Fig. 1. The overall pipeline of RIA. The algorithm consists of two modules. In the first module, we apply a hypothesis testing approach to determine which genes need to be imputed and which genes can be used as training. In the second module, we adopt the generalized linear model to impute the missing values from the imputable set. The algorithm outputs the imputed matrix that has the same number of rows and columns as of the input data.

Omnibus (GEO) [25] and Array Express [26]: Biase's [27], Yan's [28], Goolam's [29], Deng's [30], and Zeisel's [31]. The processed data were downloaded from Hemberg lab's website (https://hemberg-lab.github.io/scRNA.seq.datasets).

In each dataset, the cell populations and developmental stages are known. This informatinom are only used *a posteriori* to assess the performance of each method in improving the identification of cell populations and the recovery of temporal trajectories. We compare our method with two state-of-the-art methods for single-cell imputation: MAGIC [13] and scImpute [20]. Both methods are widely used and each represents a different imputation strategy. MAGIC uses Markov affinity matrix to smooth the data while scImpute is a statistical approach that models the data as a mixture of Gamma and Gaussian distributions.

The details for each dataset (accession ID, number of cells, number of cell types, organism, and single-cell protocol) are described in Table I. The first four studies, Biase [27], Yan's [28], Goolam [29] and Deng [30], measure the gene expression of embryonic cells at different stages, from zygote to the cells of the late blastocyst. Cell types of these datasets were labeled according to their developmental stages (timestamp). The fifth dataset, Zeisel [31], was obtained from a mouse brain tissue. The cell labels of this dataset were assigned based on expert knowledge of the underlying biology [31].

TABLE I
SINGLE-CELL DATA OBTAINED FROM NIH GEO

Dataset	Accession ID	Size	K	Organism	Protocol
Biase[27]	GSE57249	49	4	Mouse Embryo	SMARTer
Yan[28]	GSE36552	90	6	Human Embryo	Tang
Goolam[29]	E-MTAB-3321	124	5	Mouse Embryo	Smart-Seq2
Deng[30]	GSE45719	268	6	Mouse Embryo	Smart-Seq2
Zeisel[31]	GSE60361	3,005	9	Mouse Brain	STRT-Seq

For each dataset, we downloaded the already processed expression data, in which genes are represented in rows and cells are in different columns. We only perform log_2 transformation to re-scale sc-RNAseq data, i.e., $log_2(\mathbf{A}+1)$ where $\mathbf{A}$ is the expression matrix. Genes that do not express across any cells will be removed.

A. RIA improves the identification of sub-populations while preserving the biological landscape

For each of the five datasets described in Table I, the cell types are known. We use this information *a posteriori* to assess how separable the cell populations are after imputation. For each dataset, we have a raw matrix that serves as the input of each imputation method. After imputation, we have four matrices: the raw data and three imputed matrices (from RIA, MAGIC, and scImpute). In order to assess how separable the cell types in each matrix, we use k-means [32] to cluster each matrix and then compare the obtained partitionings with the known cell types. We use three different metrics for comparing the obtained partitionings with the known types: adjusted Rand index (ARI) [33], Jaccard index [34] and Purity [35] (see Appendix A for more details of the three metrics).

TABLE II
COMPARISONS USING ADJUSTED RAND INDEX (ARI).

Dataset	Adjusted Rand Index			
	Raw	RIA	scImpute	MAGIC
Biase	0.558	**0.711**	-0.009	0.154
Yan	0.558	**0.573**	0.507	0.029
Goolam	0.501	**0.914**	0.321	0.197
Deng	0.549	**0.815**	0.229	0.483
Zeisel	0.738	**0.768**	0.689	0.289

Table II shows the ARI values obtained for each method and for the raw data. For each row, cells highlighted in bold

have the highest ARI values. For each of the five datasets analyzed, the ARI values obtained for RIA are substantially higher than those of scImpute and MAGIC, demonstrating the superiority of the developed method over existing approaches. More importantly, the ARI values for RIA are higher than those obtained for raw data, demonstrating the ability of RIA in recovering the true expression of missing values due to dropout events. At the same time, it also demonstrates that RIA do not introduce false signals. In contrast, the ARI values obtained for scImpute and MAGIC are consistently lower than those obtained for raw data. There might be two reasons. First, these methods rely on sophisticated models that are prone to overfitting. Second, they lack of an efficient mechanism to verify whether a low expression value is due to sequencing limitation (i.e., dropout) or indeed due to biological phenomena. Therefore, they are likely to add false signals to the imputed data.

Tables III and IV show the Jaccard index and Purity values obtained for raw data and imputed data using RIA, scImpute, and MAGIC. Again, these metrics confirm that RIA is the best among the competing methods. All of the three benchmarking metrics show that RIA consistently outperforms scImpute and MAGIC in every single analysis.

TABLE III
COMPARISON USING JACCARD INDEX

Dataset	Jaccard Index			
	Raw	RIA	scImpute	MAGIC
Biase	0.589	**0.708**	0.339	0.289
Yan	0.498	**0.498**	0.473	0.146
Goolam	0.496	**0.892**	0.375	0.312
Deng	0.524	**0.781**	0.395	0.518
Zeisel	0.651	**0.683**	0.605	0.285

TABLE IV
COMPARISON USING PURITY INDEX

Dataset	Purity Index			
	Raw	RIA	scImpute	MAGIC
Biase	0.795	**0.836**	0.449	0.612
Yan	0.711	**0.778**	0.733	0.467
Goolam	0.822	**0.952**	0.693	0.621
Deng	0.805	**0.839**	0.627	0.750
Zeisel	0.876	**0.893**	0.840	0.668

Here we will also demonstrate that RIA improves the quality of the data without altering the transcriptomics landscapes. Since single-cell data are high-dimensional and are hard to interpret, it is desirable to visualize them in low dimensional space with two or three dimensions. Traditionally, researchers use t-distributed Stochastic Neighbourhood Embedding (t-SNE) [36, 37] for this purpose, which preserve local structure among cells. We first use Principal Component Analysis (PCA) [38] to reduce the number of dimensions to 20, and then use t-distributed Stochastic Neighbourhood Embedding (t-SNE) [39] to visualize the data. The purpose of using PCA is to reduce the running time of the visualization process.

Figures 2 and 3 show the visualization of the raw data and the imputed data. For all of the five datasets, the transcriptomics landscape of RIA is similar to that of the original data, demonstrating that RIA did not alter the transcriptomics landscape. On the contrary, the transcriptomics landscapes obtained from scImpute and MAGIC are very different from the those of the original data.

Regarding time complexity, both MAGIC and RIA are extremely fast. These two methods are able to analyze any of the five datasets in minutes. On the other hand, scImpute is slow because it needs to iteratively estimate the mixture parameters for every single gene across the genome. It takes scImpute an hour to analyze the Zeisel datasets using 20 cores.

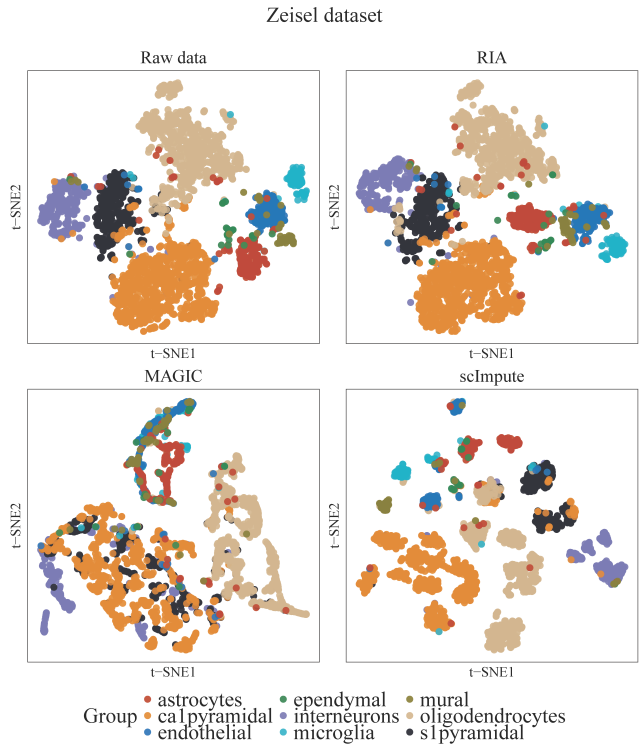

Fig. 2. Transcriptomics landscape of the Zeisel dataset. The scatter plot shows first two principle components calculated by t-SNE for raw and imputation data using RIA, scImpute, and MAGIC. RIA preserve the transcriptomics landscape of the data whereas scImpute and MAGIC introduces artificial signals and complete change the landscape.

B. RIA recovers temporal trajectories in embryonic developmental stages

We use the four embryonic datasets to demonstrate RIA's ability in recovering the temporal dynamics. The Biase dataset consists of 49 inter-blastomere cells from mouse embryonic stem cells (mESCs), including *zygote, 2-cell and 4-cell*. The Goolam dataset includes transcriptome data of 124 individual cells in mouse pre-implantation development stages: *2-cell, 4-cell, 8-cell, 16-cell and blast*. The Yan dataset consists of

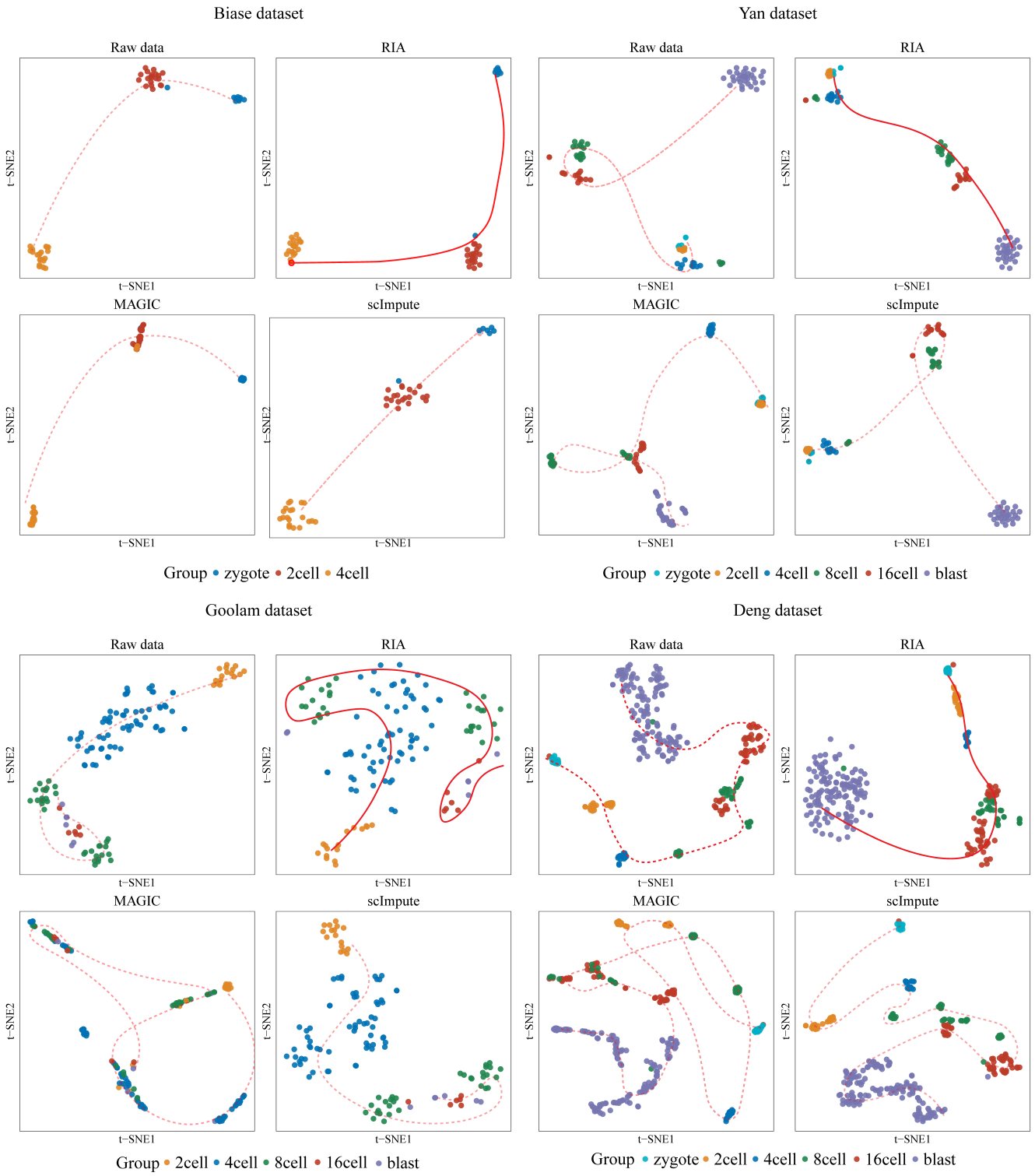

Fig. 3. Transcriptomics landscape and temporal development stages. The scatter plots show the first two dimensions of the t-SNE results calculated from Biase, Yan, Goolam, and Deng datasets. Due to dropouts, it is difficult to recognize different temporal dynamics of cells. The raw data and imputed data using scImpute and DrImpute do not show clear patterns. On the contrary, RIA significantly elucidates the cell lineage identification such that it is clearly recognized in the 2-D scatter plots.

90 cells from human pre-implantation embryos and human embryonic stem cells (hESCs). The Deng dataset includes the expression profiles of 268 individual cells of mouse pre-implantation embryos of mixed background.

Figure 3 shows the transcriptomics landscape and temporal development stages using the raw data and imputation data produced by RIA, MAGIC, and scImpute. The lines in each scatter plot connect cell groups in consecutive developmental stages. For example, for the Biase dataset, the zygote group is directly connected with the 2-cell class while the 2-cell class is connected with the 4-cell class. For this dataset, raw data and data imputed by any of the three imputation methods clearly distinguish cells at different time points. The pseudotime ordering is consistent with the time labels. For the Goolam dataset, the landscapes of the raw data and data imputed by RIA and scImpute have similar pattern. On the contrary, the transcriptomics landscape of MAGIC is very different from the rest.

For the Yan and Deng datasets, the data imputed by RIA better distinguish cell groups of different time points. The pseudotime ordering for RIA accurately reflects the transcriptome dynamics along the time course. On the contrary, the raw data and data imputed by MAGIC and scImpute fail to depict a clear time trajectory. Overall, RIA better recovers temporal trajectories than existing state-of-the-art imputation methods.

IV. Conclusion

In this article, we present a new method to recover missing values caused by dropout events in scRNA-seq data. The contribution of this approach is two folds. First, we introduce a statistical hypothesis testing to identify the set of genes that are likely to be affected by dropouts. Second, we impute missing values by using highly correlated genes that share similar biological characteristics. This strategy avoids introducing false signals. Our extensive analysis shows that RIA dramatically outperforms existing state-of-the-art approaches in improving the identification of cell populations. Our analysis also demonstrates that RIA is able to recover temporal trajectories in embryonic development stages. Regarding time complexity, RIA is fast and is able to impute thousands of cells with tens of thousands of genes in minutes.

For future work, we plan to utilize the perturbation clustering (PINSPlus) [40–42] to group genes and samples with similar patterns together before performing linear regression. This will improve the performance of the regression model and imputation. Another direction is to perform meta-analysis [43–46] to learn common bias and dropout patterns of certain platforms and protocols. This will provide us with more prior knowledge to further customize our model in order to improve the imputation procedure. Finally, we plan extend this work to improve the data for omics integration and network analysis [47–52].

V. Acknowledgments

This work was partially supported by the National Aeronautics and Space Administration (NASA) under Grant Number 80NSSC19M0170. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of any of the funding agencies.

Appendix

Rand index (RI) evaluates the similarity between predicted clusters and true cell types. Given P as a set of clusters and Q is a set of true cell types then RI is calculated as:

$$RI = \frac{t+u}{t+u+v+s} = \frac{t+u}{\binom{N}{2}} \tag{1}$$

where t is the number of pairs belonging to the same cell type in Q and are grouped together in the same cluster in P, u is the number of pairs of different cell types in Q and are grouped to different clusters in P, v is the number of pairs of same cell types in Q and are grouped to different clusters in P, s is the number of pairs in different cell types in Q and are grouped together in the same cluster in P, N is the total number of cells, and $\binom{N}{2}$ is the number of possible pairs. In brief, RI measures the ratio of pairs that are clustered in the same way (either together or different) from two partitions (e.g. 0.80 means 80% of pairs are grouped in the same way). The Adjusted Rand Index (ARI) [33] is the corrected-for-chance version of the Rand Index. The ARI values ranged from -1 to 1 in which 0 indicates for a random grouping. The ARI score is calculated as :

$$ARI = \frac{RI - exptected\_RI}{max(RI) - expected\_RI} \tag{2}$$

The Jaccard index is also known as Intersection over Union. In our context, The Jaccard index basically measures the number of pairs in same true cell type and are grouped together, divided by the number of pairs that are either in the same true cell type or are clustered together. The Jaccard index is measured by following formula:

$$J = \frac{t}{t+u+v} \tag{3}$$

Finally, the Purity metric measures the extent to which clusters contain a single true cell type. Denoting X as the clusters and Y as the classes, Purity is calculated as follows:

$$Purity = \frac{1}{N} \sum_{x \in X} max_{y \in Y} |x \cap y| \tag{4}$$

References

[1] F. Tang, C. Barbacioru, Y. Wang, E. Nordman, C. Lee, N. Xu, X. Wang, J. Bodeau, B. B. Tuch, A. Siddiqui, K. Lao, and A. Surani, "mRNA-Seq whole-transcriptome analysis of a single cell," *Nature Methods*, vol. 6, no. 5, pp. 377–382, 2009.

[2] S. Liu and C. Trapnell, "Single-cell transcriptome sequencing: recent advances and remaining challenges," *F1000Research*, vol. 5, 2016.

[3] C. Ziegenhain, B. Vieth, S. Parekh, B. Reinius, A. Guillaumet-Adkins, M. Smets, H. Leonhardt,

H. Heyn, I. Hellmann, and W. Enard, "Comparative analysis of single-cell RNA sequencing methods," *Molecular Cell*, vol. 65, no. 4, pp. 631–643, 2017.

[4] C. A. Herring, B. Chen, E. T. McKinley, and K. S. Lau, "Single-cell computational strategies for lineage reconstruction in tissue systems," *Cellular and Molecular Gastroenterology and Hepatology*, vol. 5, no. 4, pp. 539–548, 2018.

[5] Lun, Aaron TL and McCarthy, Davis J and Marioni, John C, "A step-by-step workflow for low-level analysis of single-cell RNA-seq data with Bioconductor," *F1000Research*, vol. 5, 2016.

[6] Brennecke, Philip and Anders, Simon and Kim, Jong Kyoung and Kolodziejczyk, Aleksandra A and Zhang, Xiuwei and Proserpio, Valentina and Baying, Bianka and Benes,Vladimir and Teichmann, Sarah A and Marioni, John C and others, "Accounting for technical noise in single-cell RNA-seq experiments," *Nature Methods*, vol. 10, no. 11, p. 1093, 2013.

[7] J. . N. Buettner, Florian and Natarajan, Kedar N and Casale, F Paolo and Proserpio, Valentina and Scialdone, Antonio and Theis, Fabian J and Teichmann, Sarah A and Marioni, John C and Stegle, Oliver, "Computational analysis of cell-to-cell heterogeneity in single-cell rna-sequencing data reveals hidden subpopulations of cells," *Nature Biotechnology*, vol. 33, no. 2, pp. 155–160, 2015.

[8] P. V. Kharchenko, L. Silberstein, and D. T. Scadden, "Bayesian approach to single-cell differential expression analysis," *Nature Methods*, vol. 11, no. 7, pp. 740–742, 2014.

[9] Rizzetto, Simone and Eltahla, Auda A and Lin, Peijie and Bull, Rowena and Lloyd, Andrew R and Ho, Joshua WK and Venturi, Vanessa and Luciani, Fabio, "Impact of sequencing depth and read length on single cell RNA sequencing data of T cells," *Scientific Reports*, vol. 7, no. 1, p. 12781, 2017.

[10] Parekh, Swati and Ziegenhain, Christoph and Vieth, Beate and Enard, Wolfgang and Hellmann, Ines, "The impact of amplification on differential expression analyses by RNA-seq," *Scientific Reports*, vol. 6, p. 25533, 2016.

[11] A. Haque, J. Engel, S. A. Teichmann, and T. Lönnberg, "A practical guide to single-cell RNA-sequencing for biomedical research and clinical applications," *Genome Medicine*, vol. 9, no. 1, p. 75, 2017.

[12] D. van Dijk, J. Nainys, R. Sharma, P. Kathail, A. J. Carr, K. R. Moon, L. Mazutis, G. Wolf, S. Krishnaswamy, and D. Pe'er, "MAGIC: A diffusion-based imputation method reveals gene-gene interactions in single-cell RNA-sequencing data," *BioRxiv*, p. 111591, 2017.

[13] D. Van Dijk, R. Sharma, J. Nainys, K. Yim, P. Kathail, A. J. Carr, C. Burdziak, K. R. Moon, C. L. Chaffer, D. Pattabiraman, B. Bierie, L. Mazutis, G. Wolf, S. Krishnaswamy, and D. Pe'er, "Recovering gene interactions from single-cell data using data diffusion," *Cell*, vol. 174, no. 3, pp. 716–729, 2018.

[14] Z. I. Botev, J. F. Grotowski, D. P. Kroese *et al.*, "Kernel density estimation via diffusion," *The Annals of Statistics*, vol. 38, no. 5, pp. 2916–2957, 2010.

[15] Gong, Wuming and Kwak, Il-Youp and Pota, Pruthvi and Koyano-Nakagawa, Naoko and Garry, Daniel J, "DrImpute: imputing dropout events in single cell RNA sequencing data," *BMC Bioinformatics*, vol. 19, no. 1, p. 220, 2018.

[16] A. Strehl and J. Ghosh, "Cluster ensembles—a knowledge reuse framework for combining multiple partitions," *The Journal of Machine Learning Research*, vol. 3, pp. 583–617, 2003.

[17] S. Monti, P. Tamayo, J. Mesirov, and T. Golub, "Consensus clustering: a resampling-based method for class discovery and visualization of gene expression microarray data," *Machine Learning*, vol. 52, no. 1-2, pp. 91–118, 2003.

[18] V. Y. Kiselev, K. Kirschner, M. T. Schaub, T. Andrews, A. Yiu, T. Chandra, K. N. Natarajan, W. Reik, M. Barahona, A. R. Green *et al.*, "SC3: consensus clustering of single-cell RNA-seq data," *Nature Methods*, vol. 14, no. 5, p. 483, 2017.

[19] Huang, Mo and Wang, Jingshu and Torre, Eduardo and Dueck, Hannah and Shaffer, Sydney and Bonasio, Roberto and Murray, John I and Raj, Arjun and Li, Mingyao and Zhang, Nancy R, "Saver: gene expression recovery for single-cell rna sequencing," *Nature Methods*, vol. 15, no. 7, p. 539, 2018.

[20] W. V. Li and J. J. Li, "An accurate and robust imputation method scImpute for single-cell RNA-seq data," *Nature Communications*, vol. 9, no. 1, p. 997, 2018.

[21] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum-likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society. Series B.*, vol. 39, pp. 1–39, 1977.

[22] Azizi, Elham and Prabhakaran, Sandhya and Carr, Ambrose and Pe'er, Dana, "Bayesian inference for single-cell clustering and imputing," *Genomics and Computational Biology*, vol. 3, no. 1, pp. e46–e46, 2017.

[23] Görür, Dilan and Rasmussen, Carl Edward, "Dirichlet process gaussian mixture models: Choice of the base distribution," *Journal of Computer Science and Technology*, vol. 25, no. 4, pp. 653–664, 2010.

[24] Bengtsson, Martin and Ståhlberg, Anders and Rorsman, Patrik and Kubista, Mikael, "Gene expression profiling in single cells from the pancreatic islets of langerhans reveals lognormal distribution of mrna levels," *Genome Research*, vol. 15, no. 10, pp. 1388–1392, 2005.

[25] T. Barrett, S. E. Wilhite, P. Ledoux, C. Evangelista, I. F. Kim, M. Tomashevsky, K. A. Marshall, K. H. Phillippy, P. M. Sherman, M. Holko, A. Yefanov, H. Lee, N. Zhang, C. L. Robertson, N. Serova, S. Davis, and A. Soboleva, "NCBI GEO: archive for functional genomics data sets–update," *Nucleic Acids Research*, vol. 41, no. D1, pp. D991–D995, 2013.

[26] G. Rustici, N. Kolesnikov, M. Brandizi, T. Burdett,

M. Dylag, I. Emam, A. Farne, E. Hastings, J. Ison, M. Keays, N. Kurbatova, J. Malone, R. Mani, A. Mupo, R. P. Pereira, E. Pilicheva, J. Rung, A. Sharma, Y. A. Tang, T. Ternent, A. Tikhonov, D. Welter, E. Williams, A. Brazma, H. Parkinson, and U. Sarkans, "ArrayExpress update–trends in database growth and links to data analysis tools," *Nucleic Acids Research*, vol. 41, no. D1, pp. D987–D990, 2013.

[27] F. H. Biase, X. Cao, and S. Zhong, "Cell fate inclination within 2-cell and 4-cell mouse embryos revealed by single-cell RNA sequencing," *Genome Research*, vol. 24, no. 11, pp. 1787–1796, 2014.

[28] L. Yan, M. Yang, H. Guo, L. Yang, J. Wu, R. Li, P. Liu, Y. Lian, X. Zheng, J. Yan *et al.*, "Single-cell RNA-seq profiling of human preimplantation embryos and embryonic stem cells," *Nature Structural and Molecular Biology*, vol. 20, no. 9, p. 1131, 2013.

[29] M. Goolam, A. Scialdone, S. J. Graham, I. C. Macaulay, A. Jedrusik, A. Hupalowska, T. Voet, J. C. Marioni, and M. Zernicka-Goetz, "Heterogeneity in Oct4 and Sox2 targets biases cell fate in 4-cell mouse embryos," *Cell*, vol. 165, no. 1, pp. 61–74, 2016.

[30] 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.

[31] Zeisel, Amit and Muñoz-Manchado, Ana B and Codeluppi, Simone and Lönnerberg, Peter and La Manno, Gioele and Juréus, Anna and Marques, Sueli and Munguba, Hermany and He, Liqun and Betsholtz, Christer and others, "Cell types in the mouse cortex and hippocampus revealed by single-cell RNA-seq," *Science*, vol. 347, no. 6226, pp. 1138–1142, 2015.

[32] J. A. Hartigan and M. A. Wong, "Algorithm AS 136: A k-means clustering algorithm," *Applied Statistics*, pp. 100–108, 1979.

[33] L. Hubert and P. Arabie, "Comparing partitions," *Journal of Classification*, vol. 2, no. 1, pp. 193–218, 1985.

[34] P. Jaccard, "Étude comparative de la distribution florale dans une portion des Alpes et des jura," *Bull Soc Vaudoise Sci Nat*, vol. 37, pp. 547–579, 1901.

[35] C. Manning, P. Raghavan, and H. Schütze, "Introduction to information retrieval," *Natural Language Engineering*, vol. 16, no. 1, pp. 100–103, 2010.

[36] L. Van Der Maaten, "Accelerating t-SNE using tree-based algorithms," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 3221–3245, 2014.

[37] L. v. d. Maaten and G. Hinton, "Visualizing data using t-SNE," *Journal of Machine Learning Research*, vol. 9, no. Nov, pp. 2579–2605, 2008.

[38] Baglama, James and Reichel, Lothar, "Augmented implicitly restarted Lanczos bidiagonalization methods," *SIAM Journal on Scientific Computing*, vol. 27, no. 1, pp. 19–42, 2005.

[39] J. Krijthe, "Rtsne: T-distributed stochastic neighbor embedding using barnes-hut implementation," *R package version 0.13, URL https://github. com/jkrijthe/Rtsne*, 2015.

[40] H. Nguyen, S. Shrestha, S. Draghici, and T. Nguyen, "PINSPlus: A tool for tumor subtype discovery in integrated genomic data," *Bioinformatics*, p. bty1049, 2018.

[41] T. Nguyen, R. Tagett, D. Diaz, and S. Draghici, "A novel approach for data integration and disease subtyping," *Genome Research*, vol. 27, no. 12, pp. 2025–2039, 2017.

[42] H. Nguyen, S. Shrestha, and T. Nguyen, *PINSPlus: Clustering Algorithm for Data Integration and Disease Subtyping*, 2018, r package version 1.0.2. [Online]. Available: https://CRAN.R-project.org/package=PINSPlus

[43] T. Nguyen and S. Draghici, *BLMA: A package for bi-level meta-analysis*, Bioconductor, 2017, r package.

[44] T. Nguyen, R. Tagett, M. Donato, C. Mitrea, and S. Draghici, "A novel bi-level meta-analysis approach-applied to biological pathway analysis," *Bioinformatics*, vol. 32, no. 3, pp. 409–416, 2016.

[45] T. Nguyen, C. Mitrea, R. Tagett, and S. Draghici, "DANUBE: Data-driven meta-ANalysis using UnBiased Empirical distributions - applied to biological pathway analysis," *Proceedings of the IEEE*, vol. 105, no. 3, pp. 496–515, 2017.

[46] T. Nguyen, D. Diaz, R. Tagett, and S. Draghici, "Overcoming the matched-sample bottleneck: an orthogonal approach to integrate omic data," *Nature Scientific Reports*, vol. 6, p. 29251, 2016.

[47] H. Nguyen, S. Shrestha, D. Tran, A. Shafi, S. Draghici, and T. Nguyen, "A comprehensive survey of tools and software for active subnetwork identification," *Frontiers in Genetics*, vol. 10, p. 155, 2019.

[48] A. Shafi, T. Nguyen, A. Peyvandipour, H. Nguyen, and S. Draghici, "A multi-cohort and multi-omics meta-analysis framework to identify network-based gene signatures," *Frontiers in Genetics*, vol. 10, p. 159, 2019.

[49] A. Shafi, T. Nguyen, A. Peyvandipour, and S. Draghici, "GSMA: an approach to identify robust global and test Gene Signatures using Meta-Analysis." *Bioinformatics*, p. btz561, 2019.

[50] J. Stansfield, D. Tran, T. Nguyen, and M. Dozmorov, "R tutorial: Detection of differentially interacting chromatin regions from multiple Hi-C datasets." *Current Protocols in Bioinformatics*, vol. 66, no. 1, pp. e76–e76, 2019.

[51] M. Menden, D. Wang, Y. Guan, M. Mason, B. Szalai, K. Bulusu, T. Yu, J. Kang, M. Jeon, R. Wolfinger, T. Nguyen, M. Zaslavskiy, S. Jang, Z. Ghazoui, M. Ahsen, R. Vogel, E. Neto, T. Norman, E. Tang, M. Garnett, G. Veroli, C. Zwaan, S. Fawell, G. Stolovitzky, J. Guinney, J. Dry, and J. Saez-Rodriguez, "Community assessment to advance computational prediction of cancer drug combinations in a pharmacogenomic screen," *Nature Communications*, vol. 10, no. 1, p. 2674, 2019.

[52] E. Cruz, H. Nguyen, T. Nguyen, and I. Wallace, "Functional analysis tools for post-translational modification: a post-translational modification database for analysis of proteins and metabolic pathways," *The Plant Journal*,

vol. DOI: 10.1111/tpj.14372, 2019.