2019 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)

Deep generative autoencoder for low-dimensional embeding extraction from single-cell RNAseq data

Shiquan Sun[1,2,3,4], Yang Liu[1], Xuequn Shang[1,2,4]

Abstract—**Single-cell RNA sequencing (scRNAseq) can reveal biological diversity at the cellular level that are unexplored by bulk RNA sequencing (RNAseq), but they suffer from the excessive zero expression counts and the limitation of the scalability in practice. Here, we propose a non-linear generative autoencoder based method, scSVA, relying on an integration of variational autoencoder and dropout imputations. Specifically, scSVA automatically identifies the dropouts and recovery these values only to avoid introducing new biases. Then, scSVA utilizes stochastic optimization and deep neural network to extract the low-dimensional embedding from gene expression levels. We illustrate the benefits of scSVA through in-depth real analyses of six published scRNAseq data sets. scSVA is up to 1.3 times more powerful cell clustering accuracy than existing approaches. The high power of scSVA allows us to identify new cell types that reveal new biology from scRNAseq data that otherwise cannot be revealed by existing approaches.**

Keywords—Dimensionality reduction; Cell types; Single cell; Variational autoencoder

I. Introduction

Single-cell RNA sequencing (scRNAseq) technologies are now emerging as a powerful tool to characterize the cellular composition of complex tissues [1-4]. Unlike traditional RNA sequencing (i.e., bulk RNA sequencing, bulk RNAseq) [5-7] that profiles the gene specific expression levels within hundreds to tens of thousands of single cells, scRNAseq quantifies the gene specific expression levels within individual cells [8-10]. When thousands of genes are simultaneously profiling in individual cells, their expression levels are usually involved unwanted technical variation effects, such as technical noise or confounding factors, which probably is due to extremely low number of mRNA transcripts in each cell or the stochastic nature of gene expression [11-13]. In addition, current scRNAseq technologies allow to process tens of thousands of cell simultaneously [14]. The increasing number of cells -- generates the extremely large scRNAseq data sets, poses big challenges to the existing methods and computing resources in downstream analyses [15, 16], such as dimensionality reduction (DR) [17] analysis, cell clustering analysis [18].

To overcome these challenges, over 100 analytic tools have been developed in terms of DR analysis in the past few years [19]. DR is an indispensable analytic task for scRNAseq data analysis. A recent survey study, which compared 18 DR methods or matrix factorization methods on 30 publicly available scRNAseq data sets, shows that the simplest or generic DR methods, such as principal component analysis (PCA) [20] and factor analysis (FA) [21], work reasonably well based on the average performance across all data sets; the non-generic DR method zero-inflated negative binomial-based wanted variation extraction (ZINB-WaVE) [22] was proposed to directly model the count nature of scRNAseq data using zero-inflated negative binomial model, but this method meets the computationally challenging when sample size is large; uniform manifold approximation and projection (UMAP) [23] is a Riemannian geometry based non-linear DR method that projecting the high-dimension gene expression into low-dimension feature space, but it is unable to properly deal with the technical variations. In recent years, deep learning-based DR methods have shown superior performances in scRNAseq data analysis, especially in terms of ability of denoising and computation burden reducing [24]. For example, deep count autoencoder network (DCA) [25] is a deep learning based DR method that denoising scRNA-seq data can remove technical variation, but it may lead to over-imputation in case of inadequate hyperparameter choices; and other deep learning -based DR methods for scRNAseq data analysis include deep variational autoencoder for scRNA-seq data (VASC) [26], scvis [27], scNBMF [28], and scScope [29], to name a few.

Here, we developed a deep generative model based method, scSVA, to extract biologically meaningful low-dimensional embedding and visualize the cell distribution of scRNAseq data. In this method, we are using a variational autoencoder network in a completely unsupervised manner and so do not require labeled data. scSVA first imputes the scRNAseq data with an imputation method *scImpute* [30], by fitting a mixed model of each cell type to recover transcriptome dynamics masked by dropouts. Then, with recovered scRNAseq data, scSVA was trained via Bayesian inference with unsupervised fashion of deep learning model. scSVA can capture non-linear variations and automatically learn a hierarchical representation of the data. To illustrate the benefits of scSVA, we also compared the performance of scSVA with three generic DR methods, PCA, UMAP and t-SNE [31], and four bespoke single cell DR methods VASC, ZINB-WaVE, DCA and SIMLR [32]. From the results, we can concluded that scSVA shows obvious advantages, and also confirms the effectiveness of our method for extracting feature information from scRNAseq data.

II. Methods

A. Variational Autoencoder

The variational autoencoder is a multi-layer perceptron neural network that destructs the gene expression matrix X into a latent variable z and then uses the latent variable z to reconstruct the input gene expression matrix X. In this model, when we used the variational self-encoder model for dimensionality reduction, we mainly focus on the generation of the potential representation variable z in its

978-1-7281-1867-3/19/$31.00 ©2019 IEEE

978-1-7281-1867-3/19/$31.00 ©2019 IEEE

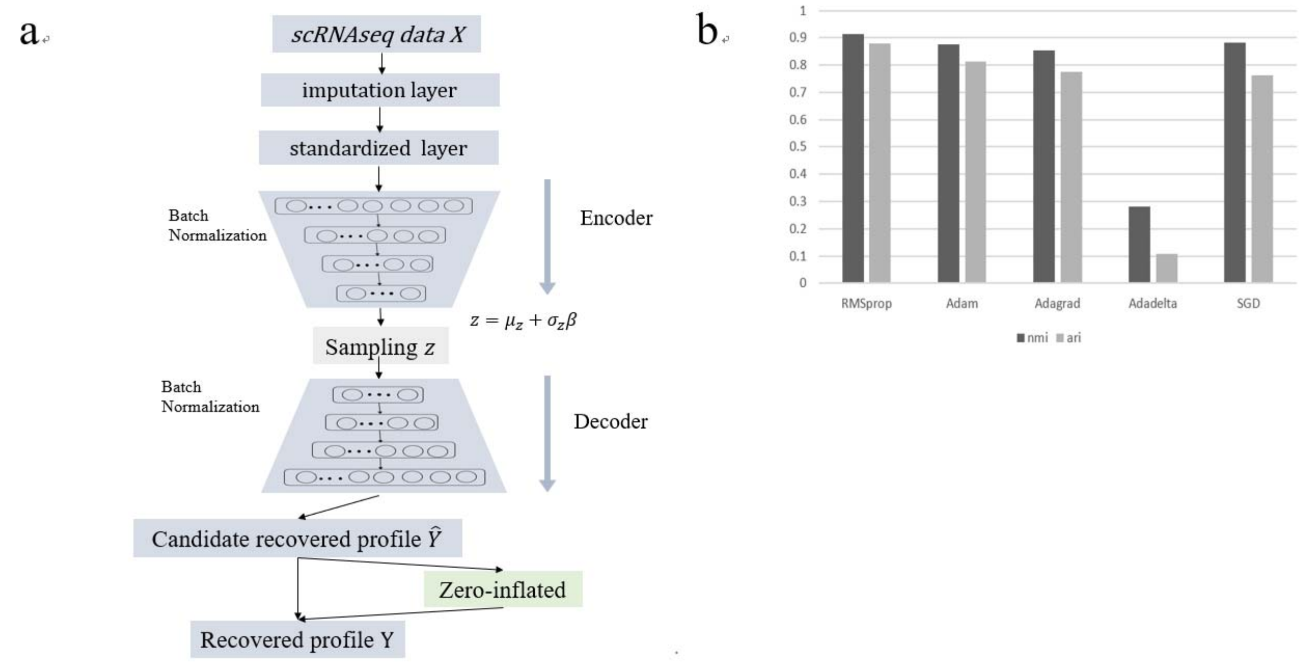

Fig. 1. (a) overview of scSVA method. scSVA consists of four parts: 1) the imputation layer; 2) the encoder network; 3) the decoder network; and 4) the recovered layer. Both the encoder and decoder networks are designed as four-layers fully connected neural networks. (b) The selection of optimization algorithms for network training.

low-dimensional space, so that it can highly restore the original data matrix X. The main advantage of this approach is that the model can learn the inherent or characteristic information of the original data in a completely unsupervised manner. theoretically, the best choice for generating z is the posterior distribution P(z|X), but it is usually too complicated and difficult to handle. The variational autoencoder attempts to approximate it using the variable probability Q(z|X), which is optimized to minimize the KL divergence [33] between Q(z|X) and P(z|X) . Combining unsupervised variational automatic coding Bayesian inference, the neural network did not learn the unconstrained representation of the scRNAseq data, but imposed regularization constraints. By applying Bayes rules and rearranging the order, you can rewrite it as:

$$\mathrm{logP(X)} - D[\mathrm{Q(z|X)||P(z|X)}] = E_{Z\sim Q}[logP(X|z)] - \mathrm{D[Q(z|X)||P(z)]} \qquad (1)$$

where P(X) is a constant, so minimizing the KL divergence is equivalent to maximizing the right side of the above equation. Although the model is fully trained, we are actually interested in the potential vector z representation of the data because it represents the key information needed to accurately reconstruct the input.

$$D(p||q) = \sum_{i=1}^{n} p(x)log\frac{p(x)}{q(x)} \qquad (2)$$

The above formula represents the KL divergence of p to q, where $p(x)$and $q(x)$ are two probability distributions of the value x. The KL divergence, also known as relative entropy, is an asymmetry measure of the difference between two probability distributions.

B. *scSVA*

scSVA was developed based variational autoencoder for scRNAseq data analysis, which was designed for visualization of scRNA-seq data and unsupervised low-dimensional extraction analysis. The workflow of scSVA is given in Figure 1a. scSVA includes four layers: imputation layer, standardized layer, coding network, latent layer, decoding layer, and recovered layer.

1) The imputation layer used the expression matrix from scRNA-seq data as inputs, and the dropout probability is set to a threshold of 0.5. First, by fitting a mixed model to learn the dropout probability of each gene in each cell, the missing information in the cell is estimated by borrowing information from the same gene in other similar cells. This layer interpolates and fills some missing information in the single-cell genetic data to improve the performance of subsequent model learning.

2) The standardized layer added immediately after the imputation layer , and consists of neuron nodes, the number of nodes being equal to the number of genes in each cell we are dealing with. Logarithmic transformation of the data to make the results more robust, then re-adjust the expression of each gene in any single cell by [0, 1] by dividing by the maximum expression of its own cells.

3) In the coding layer, we used four intermediate layers with 1024, 256, 64 and 16 nodes and a two-dimensional potential sampling layer. We use the Batch Normalization (BN) [34] method in all layers, which normalizes the data to the same

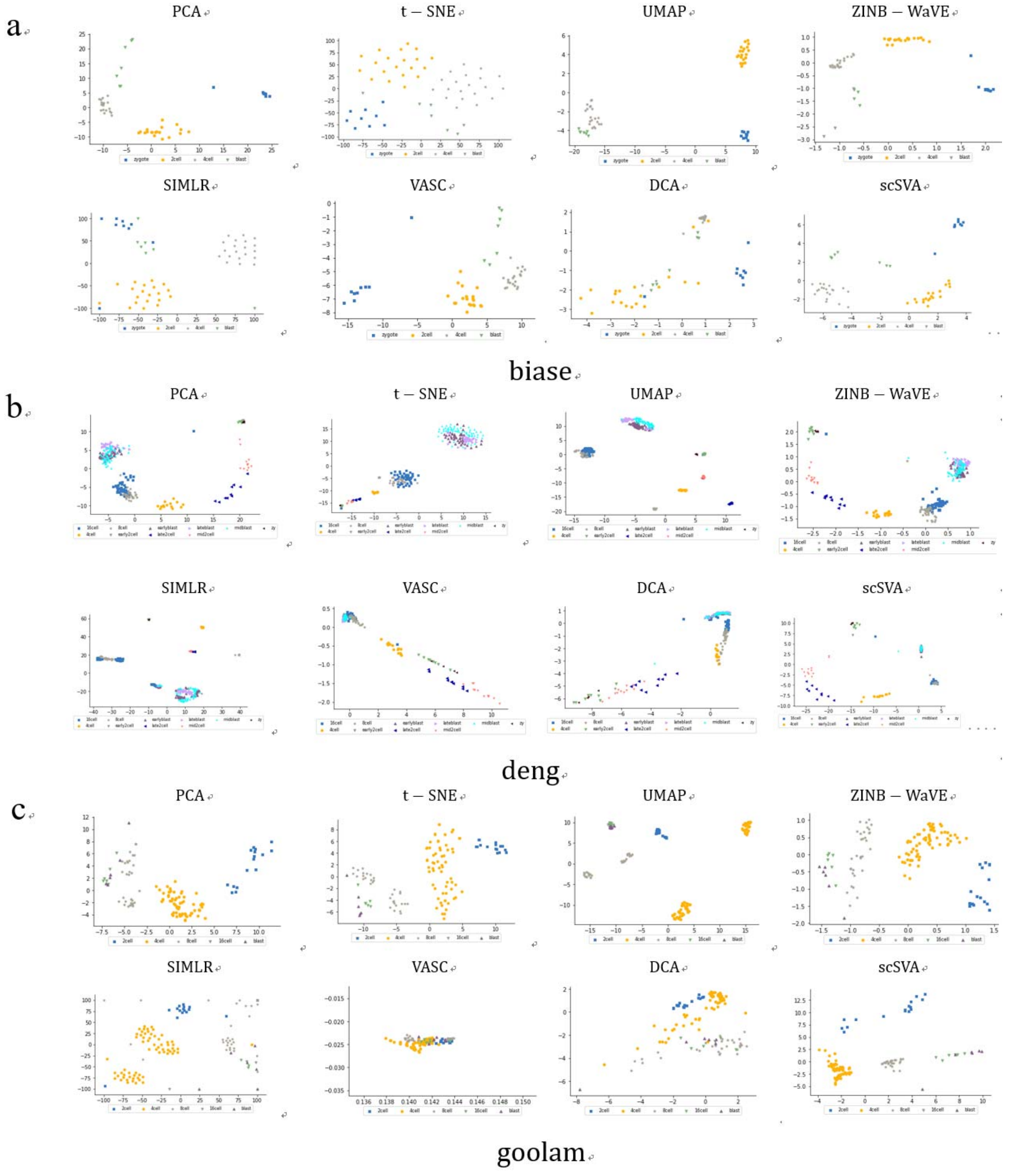

Fig. 2. Visualize the scRNA-seq data set of embryonic cells. Each point represents a cell. Different cell types are marked by different colors and shapes. It can be seen that the scVAE method achieves better performance for cell cluster separation than the other seven DR methods for each scRNAseq data set.

distribution, while reducing the risk of overfitting, which makes the training process in the model learning process more effective. At the same time we used the rectification linear unit (*ReLu*) activation function in all layers, but we used the sigmoid activation function in the last layer. However, in the first layer, non-linear activation is not used, so that it acts as an embedded PCA transform, and the L1 norm regularization is added to the weights in the layer, which is targeted punishment for the sparsity of the model.

4) In the latent sampling layer, there is an average variable μ and a variance variable σ, which can produce a two-dimensional latent variable z. In this model, two potential variables are sufficient to obtain the inherent information or feature information of the original expression data matrix. Increasing this number does not improve the results, so all subsequent analyses are based on this two-dimensional representation. Because neural networks cannot have random layers, this cannot be solved by backpropagation algorithms, using reparameterization techniques to

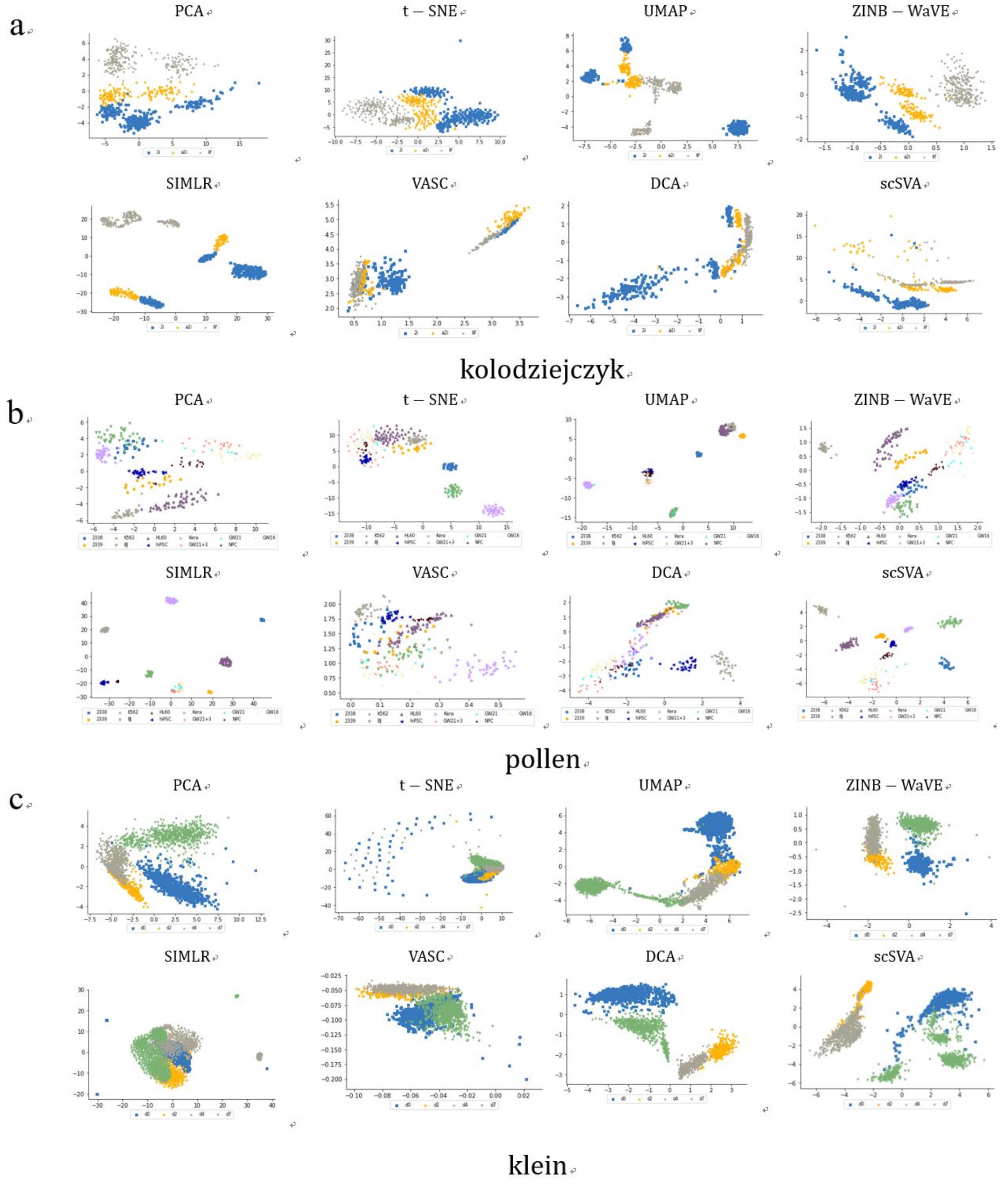

Fig. 3. Visualize scRNAseq data setsof the embryonic stem cells. Each point represents a cell. Different cell types are marked by different colors and shapes. It can be seen that the scVAE method achieves better performance for cell cluster separation than the other seven DR methods

eliminate the randomness of the input. The latent layer and the coding layer form the Encoder network

5) In the decoding layer, the original expression matrix is restored at the decoder network using the z generated by the potential sampling layer, and a four-layer fully connected neural network with nodes of 16, 64, 256 and 1024 and an output layer are designed. The first three layers are activated with *'ReLU'* and the last layer uses the sigmoid activation function. The decoding layer is the Decoder network in Figure 1a.

6) In the recovered layer, we added the ZI function adapted from the ZIFA [35] model. We simulated the dropout event $e^{-\tilde{y}^2}$ by probability, where $\tilde{y}$ is the expression value of the decoder network recovery. Since backpropagation cannot handle random units, it cannot handle discrete units at the same time. The Gumbel-softmax distribution was introduced to address these difficulties. Assuming the drop probability $\mathrm{p}, \mathrm{q} = 1 - \mathrm{p}$, samples from the Gumbel-softmax distribution are obtained by:

$$\mathrm{s} = \frac{\exp\left(\frac{logp + g_0}{\tau}\right)}{\exp\left(\frac{logp + g_0}{\tau}\right) + \exp\left(\frac{logq + g_1}{\tau}\right)} \quad (3)$$

whereg_0, g_1 are sampled from the Gumbel(0,1) distribution. The sample can then be obtained by first drawing u~Uniform(0,1) and then calculating $\mathrm{g} = -\log(-\log \mathrm{u})$. As a hyperparameter $\tau \rightarrow 0$, the generated samples from the Gumbel-softmax distribution should be identical to the samples from the Bernoulli distribution. In fact, too small a τ will make the gradient too small and the optimization algorithm will not work. Our experiments show that it is better to set τ between 0.5 and 1 for a small sample size data set. For data sets with more cells, the annealing strategy can achieve better results.

The loss function shown in equation (2) consists of two components. Due to the [0,1] ratio of our data, the first part is calculated by the binary cross entropy loss function. In the second part, the difference between the control posterior distribution and the previous $\mathrm{N}(0,1)$ can be calculated by analysis.

$$\log(\mathrm{p}(\mathrm{x})) = \mathrm{L}(\emptyset, \theta; \mathrm{x}) + D_{KL}(q_{\emptyset}(z|x)||p_{\theta}(z|x)) \quad (4)$$

The likelihood L can be decomposed as following:

$$\mathrm{L}(\emptyset, \theta; \mathrm{x}) = E_{z\sim q_{\emptyset}(z|x)}\left[\log(p_{\theta}(x|z))\right] - D_{KL}(q_{\emptyset}(z|x)||p_{\theta}(z|x)) \quad (5)$$

where the first term can be considered as a typical reconstruction loss inherent to all autoencoders, and the second term can be considered as a penalty that forces the coded representation to follow the Gaussian prior (regularization part).

$$z = \mu_z + \sigma_z \beta \quad (6)$$

where β is the Gaussian noise, Using β we do not need to sample from the latent layer and so the model is differentiable and RMSprop optimization algorithm [36] can be used to learn model parameters. Using RMSprop, we set the learning rate to 0.0005 to ensure that all test data sets converge. It relies on variants of random small batch gradient drops to minimize the likelihood L. In RMSprop, the learning rate weight is divided by the running average of the recent gradient of the weight, resulting in better convergence. If the training loss does not decrease significantly within 100 epochs, the training process will be stopped.

C. *Comparison and evaluation method*

For each data set, we considered seven DR methods PCA, t-SNE, SIMLR, UMAP, DCA, ZINB-WaVE and VASC for comparison. The same data preprocessing method is used for all methods. For PCA and t-SNE, we use the built-in python *sklearn* package function. For the UMAP, DCA and VASC method we used python packages. For the SIMLR and ZINB-WaVE method we used R packages from Bioconductor for DR. For the benchmarking of different DR methods, we used the clustering method *k*-means to group the cells into different cell types based on reduced dimension obtained by different DR methods. We utilized two criteria NMI and ARI to evaluate the performance of different DR methods:

- Normalized mutual information (NMI) [17]:

$$\mathrm{NMI(P,T)} = \frac{2MI(P,T)}{H(P)H(T)} \quad (7)$$

where $P = (p_1, p_2, \cdots, p_n)^T$ denotes the inferred cell-type cluster labels from clustering analysis while $T = (t_1, t_2, \cdots, t_n)^T$ denotes the known true cell-type labels for n samples in the data

- Adjusted rand index (ARI) [37]:

$$\mathrm{ARI}(L_e, \mathrm{L}) = \frac{\sum_{kt}\binom{n_{kt}}{2} - \left(\sum_k\binom{n_k}{2}\sum_t\binom{n_t}{2}\right)/\binom{n}{2}}{\frac{1}{2}\left(\sum_k\binom{n_k}{2} + \sum_t\binom{n_t}{2}\right) - \left(\sum_k\binom{n_k}{2}\sum_t\binom{n_t}{2}\right)/\binom{n}{2}} \quad (8)$$

where L_e and L are the predicted cell type labels and the true cell type labels, respectively; K_e and K are the predicted cluster number and the true cluster number, respectively; n_k denotes the number of cells assigned to a specific cluster k $(k = 1,2,\cdots,K)$; similarly n_t denotes the number of cells assigned to cluster t $(t = 1,2,\cdots,K_e)$; n_{kt} represents the number of cells shared between cluster k and t; and n is the total number of cells.

D. *Six public scRNAseq data sets*

Six publicly available scRNAseq data sets were got from six studies:

- The *biase* [38] data set, the *deng* [39] data set, and the *goolam* [40] data set were developing mouse embryonic cells that have been studied for embryonic development from fertilized egg cells to embryonic cells.
- The *kolodziejczyk* [41] data set is a mouse embryonic stem cell grown under three different conditions. The *klein* [42] data set is a mouse embryonic stem cell at four different growth stages.
- The *pollen* [43] data set was a developing human cerebral cortical cell that has been sequenced in 11 different cell states.

III. RESULT

A. *Model selection*

The first experiment we conducted was a neural network optimizer that chose a single-cell variational autoencoder model. In order not to lose generality, we chose the human brain scRNAseq data set. As shown in Figure 1b, five optimization methods were compared to optimize the neural networks, namely RMSprop, Adam, Adagrad, Adadelta and SGD. The results show that the RMSprop method is superior to other optimization methods when we choose NMI or ARI. Therefore, in the following experiments, we will choose the RMSprop method to optimize the neural network.

B. *Six public scRNAseq data sets*

We performed eight DR methods on six publicly available scRNAseq real data sets, three mouse scRNAseq data sets that are developing embryonic cells, two mouse scRNAseq data sets that are embryonic stem cells in different growth states, and one human scRNAseq data set that is cerebral cortical cells. Detailed cell type information for six scRNAseq data sets was reported in the original study. For comparison, we compared seven existing DR methods available, PCA, t-SNE, SIMLR, UMAP, DCA, ZINB-WaVE and VASC. In order to effectively evaluate the performance of different DR methods, we performed the same pre-processing procedure on the each data set and finally processed them into a two-dimensional embedding. Finally, using k-means clustering method, repeated 100 times to test each method in the extraction of single-cell data feature information can be strong, the performance of different DR methods was evaluated by NMI and ARI.

Tables I and II show the comparison of scVAE with the other seven DR methods. It can be seen that scVAE method shows more accurate cell type detection results than the other seven DR methods. Specifically, for the NMI criteria, the results of the scVAE evaluation on the data sets biase, deng, goolam, klein, kolodziejczyk, and pollen were: 1.00, 0.76, 0.92, 0.85, 0.70, 0.96. For the ARI standard, the results of the scVAE evaluation were 1.00, 0.46, 0.90, 0.89, 0.68, 0.95 on the data sets biase, deng, goolam, klein, kolodziejczyk and pollen, respectively.

In order to further compare the feature extraction of scSVA with the other seven DR methods, we show the two-dimensional spatial representation of the two-dimensional spatial representation data processed by the eight DR methods on six scRNAseq data sets, as shown in Figures 3 and 4.

For the developing mouse embryonic scRNAseq data set biase, goolam, and deng studied the embryonic development process from fertilized eggs to blast cells, as shown in Figure 3. It can be seen that in the biase data set which has less data, the clustering of t-SNE is sparse and the visualization effect is the worst. In the DCA method, multiple types of cells are confused and cannot be identified. The UMAP method clearly differentiates cells into three categories, failing to distinguish between the two types of cells, 4Cell and Blast. In the Deng data set, other methods have different degrees of cluster confusion, and more than two types of cells are mixed in at least two types of clusters. In the Goolam data set, the rest of the methods show the characteristics of cluster sparseness. t-SNE and SIMLR, both of which use neighbor preserving embedding, showed poor results on these datasets. In these three data sets, scSVA clustering visualization has obvious advantages, indicating that the scSVA method can better simulate cell development status during embryonic development than the other seven DR methods.

TABLE I. COMPARED ALL METHODS (NMI)

dataset	biase	deng	goolam	klein	kolodziejczyk	pollen
PCA	0.92	0.68	0.7	0.71	0.43	0.8
t-SNE	0.85	0.69	0.64	0.54	0.56	0.61
SIMLR	0.86	0.59	0.53	0.24	0.62	0.92
UMAP	1.0	0.74	0.73	0.69	0.65	0.81
ZINB-WaVE	1.0	0.71	0.74	0.83	0.58	0.82
DCA	0.55	0.59	0.45	0.74	0.26	0.68
VASC	1.0	0.66	0.43	0.52	0.21	0.62
scSVA	***1.0***	***0.76***	***0.92***	***0.85***	***0.69***	***0.96***

TABLE II. COMPARED ALL METHODS (ARI)

dataset	biase	deng	goolam	klein	kolodziejczyk	pollen
PCA	0.92	0.44	0.52	0.72	0.38	0.69
t-SNE	0.8	0.43	0.46	0.47	0.56	0.22
SIMLR	0.87	0.3	0.45	0.32	0.59	0.85
UMAP	1.0	0.55	0.54	0.66	0.56	0.7
ZINB-WaVE	1.0	0.44	0.55	0.83	0.58	0.7
DCA	0.47	0.37	0.41	0.73	0.21	0.53
VASC	1.0	0.4	0.28	0.47	0.14	0.41
scSVA	***1.0***	***0.46***	***0.9***	***0.89***	***0.68***	***0.95***

In the data set kolodziejczyk, the differentiation of embryonic stem cells under the growth conditions of serum, 2i and alternative 2i was studied. The data set was used to study the differentiation of embryonic stem cells at different times of d0, d2, d4 and d7. It can be seen that t-SNE, ZINB-WaVE, and scSVA have better effects. In the data set pollen, human developmental cerebral cortical cells were observed. It can be seen that in the pollen data set with the most cluster type of cells, although SIMLR forms the most compact cluster, it can be seen that more than one type of cells are contained in its multiple clusters. The klein data set has more data than others. PCA, DCA, and ZINB-WaVE isolated the cell population of most different growing batches, but erroneously grouped cells under d2 and d4 conditions. scSVA separates most cell populations while maintaining their relative position. In the three data sets, the same type of cell clustering of scSVA is relatively tight, and there is no mixed clustering phenomenon of multiple cell types appearing in the other seven methods, as shown in Figure 4.

IV. Conclusion

In this paper, we present an new variational autoencoder method that integrates the imputation and low-dimensional embedding extraction to analyze the scRNAseq data with unsupervised manner. Using the scSVA method, we can extract the characteristic information in low-dimensional space from the scRNAseq data to effectively detect the cell type, so that there are further biological processes for understanding embryonic development and cell differentiation. We have experimentally tested the effectiveness of the scSVA method in the process of reducing the dimensionality of scRNAseq data, as well as for different data sets with different data structures in the original space. On the six publicly available data sets, the NMI and ARI performance evaluations show their powerful performance compared to the seven existing DR methods.

One advantage of this method is that it can better deal with the dropout events of the scRNAseq data set. We integrate the imputation method in the neural network to correct the dropouts of the input data, and accurately estimate the dropout in the scRNAseq data without introducing new deviations.

scSVA method performs efficient feature extraction on high-dimensional scRNAseq data, and obtains low-dimensional embeddings in its potential space, which provides an effective means for biological research. In this study, we mainly conducted experiments on cell type detection. In future research, we should not only study its effect on single-cell clustering, but also apply to cell differentiation trajectories and detect differential gene expression. This can also be used in future work to identify key mutant genes associated with the evolution of organisms.

ACKNOWLEDGMENT

This study was supported by the National Natural Science Foundation of China (Grant Number: 61902319); the Natural Science Foundation of Shaanxi Province (Grant Number: 2019JQ127); and the National Natural Science Foundation of China (Grant Numbers: 61772426 and 61332014).

AVAILABILITY OF DATA AND MATERIALS

The data sets we used in this experiment are the six data sets published on the website https://hemberg-lab.github.io/scRNA.seq.datasets/;All source code and data sets used in our experiments have been deposited at https://github.com/sqsun/scSVA.

AUTHOR DETAILS

[1]School of Computer Science, Northwestern Polytechnical University, 710129 Xi'an, Shaanxi, P.R. China. [2]Key Laboratory of Big Data Storage and Management, Northwestern Polytechnical University, Ministry of Industry and Information Technology, 710129 Xi'an, Shaanxi, P.R. China. [3]Centre for Multidisciplinary Convergence Computing (CMCC), School of Computer Science, Northwestern Polytechnical University, 710129 Xi'an, Shaanxi, P.R. China. [4]National Engineering Laboratory for Integrated Aero-Space-Ground-Ocean Big Data Application Technology, Northwestern Polytechnical University, Xi'an, Shaanxi, 710129, People's Republic of China.
*Correspondence: sqsun@nwpu.edu.cn and shang@nwpu.edu.cn

REFERENCES

1. Saliba, A.-E., et al., *Single-cell RNA-seq: advances and future challenges.* Nucleic acids research, 2014. **42**(14): p. 8845-8860.
2. Kolodziejczyk, A.A., et al., *The technology and biology of single-cell RNA sequencing.* Molecular cell, 2015. **58**(4): p. 610-620.
3. Liu, S. and C. Trapnell, *Single-cell transcriptome sequencing: recent advances and remaining challenges.* F1000Research, 2016. **5**.
4. Shapiro, E., T. Biezuner, and S. Linnarsson, *Single-cell sequencing-based technologies will revolutionize whole-organism science.* Nature Reviews Genetics, 2013. **14**(9): p. 618-630.
5. Wang, Z., M. Gerstein, and M. Snyder, *RNA-Seq: a revolutionary tool for transcriptomics.* Nature reviews genetics, 2009. **10**(1): p. 57.
6. Sun, S., et al., *Differential expression analysis for RNAseq using Poisson mixed models.* Nucleic acids research, 2017. **45**(11): p. e106-e106.
7. Sun, S., et al., *Heritability estimation and differential analysis of count data with generalized linear mixed models in genomic sequencing studies.* Bioinformatics, 2018. **35**(3): p. 487-496.
8. Islam, S., et al., *Quantitative single-cell RNA-seq with unique molecular identifiers.* Nature methods, 2014. **11**(2): p. 163.
9. Tang, X., et al., *The single-cell sequencing: new developments and medical applications.* Cell & Bioscience, 2019. **9**(1): p. 53.
10. Hwang, B., J.H. Lee, and D. Bang, *Single-cell RNA sequencing technologies and bioinformatics pipelines.* Experimental & molecular medicine, 2018. **50**(8): p. 1-14.
11. Gong, W., et al., *DrImpute: imputing dropout events in single cell RNA sequencing data.* BMC bioinformatics, 2018. **19**(1): p. 220.
12. Tracy, S., G.-C. Yuan, and R. Dries, *RESCUE: imputing dropout events in single-cell RNA-sequencing data.* BMC bioinformatics, 2019. **20**(1): p. 388.
13. Chen, M. and X. Zhou, *VIPER: variability-preserving imputation for accurate gene expression recovery in single-cell RNA sequencing studies.* Genome biology, 2018. **19**(1): p. 196.
14. Zheng, G.X., et al., *Massively parallel digital transcriptional profiling of single cells.* Nature communications, 2017. **8**: p. 14049.
15. Sinha, D., et al., *dropClust: efficient clustering of ultra-large scRNA-seq data.* Nucleic acids research, 2018. **46**(6): p. e36-e36.
16. Iacono, G., et al., *bigSCale: an analytical framework for big-scale single-cell data.* Genome research, 2018. **28**(6): p. 878-890.
17. Sun, S., et al., *Accuracy, Robustness and Scalability of Dimensionality Reduction Methods for Single Cell RNAseq Analysis.* bioRxiv, 2019: p. 641142.
18. Ren, X., L. Zheng, and Z. Zhang, *SSCC: A Novel Computational Framework for Rapid and Accurate Clustering Large-scale Single Cell RNA-seq Data.* Genomics, proteomics & bioinformatics, 2019.
19. Zappia, L., B. Phipson, and A. Oshlack, *Exploring the single-cell RNA-seq analysis landscape with the scRNA-tools database.* PLoS computational biology, 2018. **14**(6): p. e1006245.
20. Jolliffe, I., *Principal component analysis for special types of data.* Principal component analysis, 2002: p. 338-372.
21. Cattell, R.B., *The three basic factor-analytic research designs—their interrelations and derivatives.* Psychological bulletin, 1952. **49**(5): p. 499.
22. Risso, D., et al., *A general and flexible method for signal extraction from single-cell RNA-seq data.* Nature communications, 2018. **9**(1): p. 284.
23. McInnes, L., J. Healy, and J. Melville, *Umap: Uniform manifold approximation and projection for dimension reduction.* arXiv preprint arXiv:1802.03426, 2018.
24. Hinton, G.E. and R.R. Salakhutdinov, *Reducing the dimensionality of data with neural networks.* science, 2006. **313**(5786): p. 504-507.

25. Eraslan, G., et al., *Single-cell RNA-seq denoising using a deep count autoencoder.* Nature communications, 2019. **10**(1): p. 390.

26. Wang, D. and J. Gu, *VASC: dimension reduction and visualization of single-cell RNA-seq data by deep variational autoencoder.* Genomics, proteomics & bioinformatics, 2018. **16**(5): p. 320-331.

27. Ding, J., A. Condon, and S.P. Shah, *Interpretable dimensionality reduction of single cell transcriptome data with deep generative models.* Nature communications, 2018. **9**(1): p. 2002.

28. Sun, S., et al., *A fast and efficient count-based matrix factorization method for detecting cell types from single-cell RNAseq data.* BMC systems biology, 2019. **13**(2): p. 28.

29. Deng, Y., et al., *Scalable analysis of cell-type composition from single-cell transcriptomics using deep recurrent learning.* Nature methods, 2019. **16**(4): p. 311.

30. Li, W.V. and J.J. Li, *An accurate and robust imputation method scImpute for single-cell RNA-seq data.* Nature communications, 2018. **9**(1): p. 997.

31. Maaten, L.v.d. and G. Hinton, *Visualizing data using t-SNE.* Journal of machine learning research, 2008. **9**(Nov): p. 2579-2605.

32. Wang, B., et al., *SIMLR: a tool for large-scale single-cell analysis by multi-kernel learning.* bioRxiv, 2017: p. 118901.

33. Vidyasagar, M. *Kullback-Leibler divergence rate between probability distributions on sets of different cardinalities*. in *49th IEEE Conference on Decision and Control (CDC)*. 2010. IEEE.

34. Ioffe, S. and C. Szegedy, *Batch normalization: Accelerating deep network training by reducing internal covariate shift.* arXiv preprint arXiv:1502.03167, 2015.

35. Pierson, E. and C. Yau, *ZIFA: Dimensionality reduction for zero-inflated single-cell pneumophila analysis.* Genome Biology, 2015. **16**(243).

36. Hinton, G., N. Srivastava, and K. Swersky, *Neural networks for machine learning lecture 6a overview of mini-batch gradient descent.* Cited on, 2012. **14**: p. 8.

37. Hubert, L. and P. Arabie, *Comparing partitions.* Journal of classification, 1985. **2**(1): p. 193-218.

38. Biase, F.H., X. Cao, and S. Zhong, *Cell fate inclination within 2-cell and 4-cell mouse embryos revealed by single-cell RNA sequencing.* Genome research, 2014. **24**(11): p. 1787-1796.

39. Deng, Q., et al., *Single-cell RNA-seq reveals dynamic, random monoallelic gene expression in mammalian cells.* Science, 2014. **343**(6167): p. 193-196.

40. Goolam, M., et al., *Heterogeneity in Oct4 and Sox2 targets biases cell fate in 4-cell mouse embryos.* Cell, 2016. **165**(1): p. 61-74.

41. Kolodziejczyk, A.A., et al., *Single cell RNA-sequencing of pluripotent states unlocks modular transcriptional variation.* Cell stem cell, 2015. **17**(4): p. 471-485.

42. Klein, A.M., et al., *Droplet barcoding for single-cell transcriptomics applied to embryonic stem cells.* Cell, 2015. **161**(5): p. 1187-1201.

43. Pollen, A.A., et al., *Low-coverage single-cell mRNA sequencing reveals cellular heterogeneity and activated signaling pathways in developing cerebral cortex.* Nature biotechnology, 2014. **32**(10): p. 1053.