The Annals of Applied Statistics
2019, Vol. 13, No. 3, 1733–1752
https://doi.org/10.1214/19-AOAS1250

A HIERARCHICAL BAYESIAN MODEL FOR SINGLE-CELL CLUSTERING USING RNA-SEQUENCING DATA

BY YIYI LIU[1], JOSHUA L. WARREN[2] AND HONGYU ZHAO[1]

Yale University

Understanding the heterogeneity of cells is an important biological question. The development of single-cell RNA-sequencing (scRNA-seq) technology provides high resolution data for such inquiry. A key challenge in scRNA-seq analysis is the high variability of measured RNA expression levels and frequent dropouts (missing values) due to limited input RNA compared to bulk RNA-seq measurement. Existing clustering methods do not perform well for these noisy and zero-inflated scRNA-seq data. In this manuscript we propose a Bayesian hierarchical model, called BasClu, to appropriately characterize important features of scRNA-seq data in order to more accurately cluster cells. We demonstrate the effectiveness of our method with extensive simulation studies and applications to three real scRNA-seq datasets.

1. Introduction. Identifying cell subtypes with distinct transcriptomic signatures is important for understanding the functionalities of complex tissues, such as brain [Tasic et al. (2016)], and provides useful information for disease diagnostics and treatment [Patel et al. (2014)]. Single-cell RNA-sequencing (scRNA-seq) technology, which shows unprecedented spatial resolution, provides great promise to delineate cell heterogeneity. However, unlike bulk RNA-seq measurement, which usually involves millions of cells, scRNA-seq has much less input RNA, making it more difficult to measure transcript levels [Stegle, Teichmann and Marioni (2015)]. The technical noise level observed in scRNA-seq data is usually much higher than that observed in bulk RNA-seq data [Brennecke et al. (2013)]. In addition a direct result of low starting material is that a transcript is more likely to be missed in reverse transcription or not detected in sequencing. This leads to frequent dropout events (i.e., false quantification of a gene as absent, can be treated as a "missing value" in a statistical sense) in scRNA-seq data [Vallejos et al. (2017)]. These unique features of scRNA-seq data impact the effectiveness of clustering approaches developed for bulk RNA-seq data and necessitate the development of new statistical and computational methods [Stegle, Teichmann and Marioni (2015)].

Received March 2018; revised September 2018.

[1]Supported by P01 CA154295 and P50 CA196530.

[2]Supported by CTSA Grant Number UL1 TR001863 and KL2 TR001862 from the National Center for Advancing Translational Science.

Key words and phrases. Bayesian hierarchical model, clustering, Dirichlet process, Gaussian mixture model, missing data, single-cell RNA-sequencing.

Several methods have been developed recently for scRNA-seq data clustering, such as Single-cell Interpretation via Multikernel Learning (SIMLR) [Wang et al. (2017)] and SNN-Cliq [Xu and Su (2015)]. These methods typically focus on learning "secondary" similarity metrics for cells that might be more robust to experimental noises than conventional Euclidean distance- or correlation-based similarities. However, much less attention has been paid to properly modeling dropout events, a common phenomenon in scRNA-seq data, with the majority of existing methods treating genes with zero counts as truly "unexpressed." We note that a method proposed for dimension reduction, Zero Inflated Factor Analysis (ZIFA) [Pierson and Yau (2015)], and a recently developed clustering method, Clustering through Imputation and Dimension Reduction (CIDR) [Lin, Troup and Ho (2017)], do model dropout events by defining the probability of missingness as a function of the true expression level. However, empirical data suggest that dropout rates are not only associated with the true gene expression values, but also are affected by other important factors in sequencing such as library size (Figure 1(a)).

To overcome these limitations, we propose a novel hierarchical clustering method under a Bayesian setting, BasClu (**Ba**yesian **s**cRNA-seq **Clu**stering). Unlike existing methods that cluster cells based on gene expression values directly, BasClu infers and utilizes a dichotomized gene expression status to cluster cells. Due to the high level of experimental noise, binary states (expressed or not) for genes could provide more reliable signals for a cell's identity compared to expression values. In addition BasClu explicitly models the missing probability for each gene to account for dropout events. Along with the true gene expression level, BasClu also incorporates sequencing features, such as gene length and library size, into the dropout modeling framework. Finally, we employ a Gaussian mixture model to connect the binary (hidden) state of each gene and observed expression levels.

The remainder of our manuscript is organized as follows. In Section 2 we describe the details of our statistical model and method of conducting inference. In Section 3 we evaluate the performance of our method through simulation. Three real scRNA-seq datasets are analyzed in Section 4. We conclude the paper with a brief discussion in Section 5.

2. Methods.

2.1. *Statistical model.* Let x_{ij} be the natural log of the observed read counts (without normalization) of gene j in cell i, where $i = 1, 2, \ldots, N$ and $j = 1, 2, \ldots, G$. If the count for a gene is 0, we define x_{ij} as missing (N.A.). Since single-cell RNA-seq data are very noisy, we dichotomize the expression values into two states, expressed or not expressed. Denote z_{ij} as the binary indicator representing whether gene j is expressed in cell i, where $z_{ij} = 1$ if it is expressed and

0 otherwise. We assume that

$$z_{ij}|\pi_{ij} \sim \text{Bernoulli}(\pi_{ij}),$$

where π_{ij} is the probability that gene j is expressed in cell i.

Cell clustering is induced through use of a Dirichlet Process (DP) prior distribution [Ferguson (1973)] assigned to the vector of cell-specific expression probabilities $(\pi_{i1}, \pi_{i2}, \ldots, \pi_{iG})^T$, that is,

$$(\pi_{i1}, \pi_{i2}, \ldots, \pi_{iG})^T \sim \text{DP}(\alpha_\lambda, G_0).$$

We model G_0, the base distribution of the DP, as a product of G independent $\text{Beta}(\alpha_\pi, \beta_\pi)$ distributions, allowing for semiconjugacy during model fitting.

We then connect y'_{ij}, the true (unobserved) expression level of gene j in cell i (natural log scale), with z_{ij} via a Gaussian mixture model. Specifically, we assume that y'_{ij} follows a Gaussian distribution conditional on z_{ij} such that

$$y'_{ij}|z_{ij} = 0 \sim \text{N}(g'_j, \sigma^2_{0j})$$

and

$$y'_{ij}|z_{ij} = 1 \sim \text{N}(g'_j + \delta_j, \sigma^2_{1j}), \quad \text{where } \delta_j > 0.$$

In addition to true expression levels, it is known that measured gene expression values in RNA-seq are influenced by other factors such as gene length and library size. Here, we consider Reads Per Kilobase of transcript per Million mapped reads (RPKM) [Mortazavi et al. (2008)], a widely used RNA-seq normalization method, as the unit for measured transcript abundance and assume that the measured expression level (without dropout, natural log scale) is $y_{ij} = y'_{ij} + s_i + l_j + \mu$, where s_i is the log library size of cell i, l_j is the length of gene j on the log scale, and μ is a scaling constant in logarithm. Note, we adopt the idea of RPKM to link raw count measurements y_{ij} with the underlying gene expression level y'_{ij} here, however, this does not mean RPKM-normalized gene expression data are necessary in the analysis. The resulting model for the measured expression levels can be written as

$$y_{ij}|z_{ij}, g'_j, \delta_j, \sigma^2_{0j}, \sigma^2_{1j} \sim \text{N}\big(g'_j + \delta_j z_{ij} + s_i + l_j + \mu, \sigma^2_{0j}(1 - z_{ij}) + \sigma^2_{1j} z_{ij}\big).$$

Moreover, if we define $g_j = g'_j + l_j + \mu$, we can simplify the above formula as

$$y_{ij}|z_{ij}, g_j, \delta_j, \sigma^2_{0j}, \sigma^2_{1j} \sim \text{N}\big(g_j + \delta_j z_{ij} + s_i, \sigma^2_{0j}(1 - z_{ij}) + \sigma^2_{1j} z_{ij}\big).$$

Finally, we address the issue of the dropout events by assuming

$$x_{ij}|y_{ij} = \begin{cases} y_{ij} & \text{w.p. } 1 - \Phi(\gamma_1 y_{ij} + \gamma_2), \\ \text{N.A.} & \text{w.p. } \Phi(\gamma_1 y_{ij} + \gamma_2), \end{cases}$$

where $\Phi(\cdot)$ is the cumulative distribution function of the standard normal distribution. Previous methods that model dropout events, such as ZIFA [Pierson and Yau

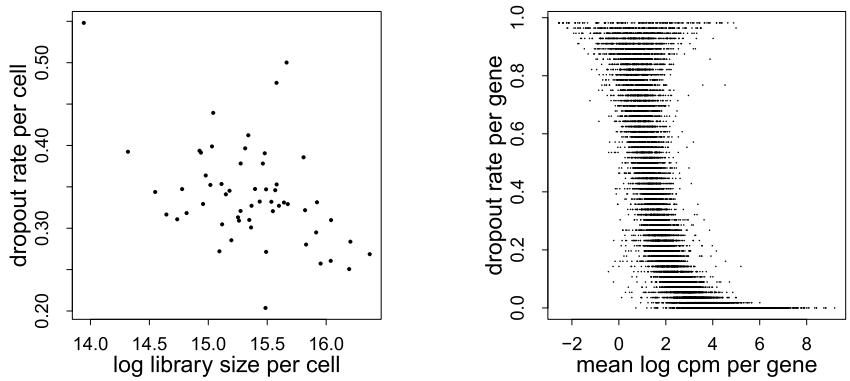

FIG. 1. *Dropout rate per cell versus library size* (a) *and dropout rate per gene versus mean expression level* (b), *estimated in the Okaty data*.

(2015)], assume that the dropout probability depends on the underlying true expression level y'_{ij}. A key difference in our method is that we model the dropout probability as a function of the "amplified" gene expression measurement y_{ij}, based on our observation that dropout rate is highly correlated with library size (Figure 1(a)).

2.2. *Clustering inference*. We adopt two approaches to infer the clustering structures of single cells. The first one is based on a posterior similarity matrix of cells, $\boldsymbol{S}$, which is defined by the elements

$$S_{ii'} = \frac{1}{M}\sum_{m=1}^{M} \delta\big(c_i^{(m)}, c_{i'}^{(m)}\big), \quad i, i' = 1, 2, \ldots, N,$$

where M is the total number of posterior samples (after removing burn-in and thinning of the chains), $c_i^{(m)}$ is cell i's component label in the mth iteration and $\delta(\cdot,\cdot)$ is the kronecker delta function ($\delta(x, y) = 1$ if $x = y$ and 0 otherwise). This posterior similarity matrix is robust to the commonly observed "label switching" problem associated with Bayesian mixture models of this form, and is commonly used to estimate the clustering structure. We utilize the method of [Fritsch and Ickstadt (2009)] to obtain the clusters. Specifically, this method selects a partition $\boldsymbol{c}^*$ from the posterior samples $\boldsymbol{c}^{(m)}, m = 1, 2, \ldots, M$ such that

$$\begin{aligned}\boldsymbol{c}^* = \arg\max_{\boldsymbol{c}^{(m)}} &\left(\sum_{i<i'} \delta\big(c_i^{(m)}, c_{i'}^{(m)}\big) S_{ii'} - \sum_{i<i'} \delta\big(c_i^{(m)}, c_{i'}^{(m)}\big) \sum_{i<i'} S_{ii'} \Big/ \binom{n}{2}\right) \\ &\Big/ \left(\left[\sum_{i<i'} \delta\big(c_i^{(m)}, c_{i'}^{(m)}\big) + \sum_{i<i'} S_{ii'}\right]\Big/2 - \sum_{i<i'} \delta\big(c_i^{(m)}, c_{i'}^{(m)}\big) \sum_{i<i'} S_{ii'} \Big/ \binom{n}{2}\right).\end{aligned} \tag{1}$$

The objective function above is the adjusted Rand index [Hubert and Arabie (1985)] (details in Section 2.5) between $\boldsymbol{c}^{(m)}$ and $E(\boldsymbol{c}|\boldsymbol{x})$. Note that in this approach both the number of clusters and the clustering structures are automatically selected. For convenience we denote this approach as BasCluS in subsequent text.

In the second approach we apply hierarchical clustering to posterior estimates of the $N \times G$ binary expression status matrix $\boldsymbol{z}$ (with elements $z_{ij}, i = 1, 2, \ldots, N, j = 1, 2, \ldots, G$). We consider this approach for comparison with other methods that require a prespecified number of clusters. Since the clustering accuracy evaluation metrics (details in Section 2.5) could be affected by the selected number of clusters, evaluating the performance of these methods with the number of clusters fixed at the same value could provide a fairer comparison. We denote our proposed approach as BasCluZ.

2.3. *Prior specification.* Because we are working in the Bayesian setting, we specify prior distributions for the unknown model parameters. We choose weakly informative conjugate prior distributions when possible in order to allow the data to drive the inference rather than our prior beliefs while maintaining important computational benefits during model fitting.

We specify the concentration parameter in the DP model, α_λ, using Gamma(a_l, b_l) prior distribution (we adopt the $(shape, rate)$ parameterization for the Gamma distribution), and $\alpha_\pi = 1$ and $\beta_\pi \sim$ Gamma(a_p, b_p) in the base distribution definition, Beta(α_π, β_π). In the Gaussian mixture model we specify $g_j \overset{\text{i.i.d.}}{\sim} \mathrm{N}(\nu_g, \tau_g^2)$, $\delta_j \overset{\text{i.i.d.}}{\sim}$ Gamma$(\alpha_\delta, \beta_\delta)$, $\sigma_{0j}^2 \overset{\text{i.i.d.}}{\sim}$ Inverse Gamma(α_0, β_0) and $\sigma_{1j}^2 \overset{\text{i.i.d.}}{\sim}$ Inverse Gamma(α_1, β_1). Furthermore, we set hyperparameters $\nu_g \sim \mathrm{N}(m_g, d_g^2)$, $\tau_g^2 \sim$ Inverse Gamma(a_g, b_g), $\alpha_\delta = 2$, $\alpha_0 = \alpha_1 = 1$, and β_δ, β_0, and β_1 are given independent Gamma(a_d, b_d) prior distributions. In the dropout model we assume $\gamma_1 \sim \mathrm{N}(\nu_1, \tau_1^2)$ and $\gamma_2 \sim \mathrm{N}(\nu_2, \tau_2^2)$. In our data analysis we set $a_l = a_p = a_d = 2$, $b_l = b_p = b_d = 1$, $m_g = 0$, $d_g^2 = 100$, $a_g = 1$, $b_g = 0.01$, $\nu_1 = \nu_2 = 0$ and $\tau_1^2 = \tau_2^2 = 100$.

2.4. *Posterior sampling.* We analytically integrate out several nuisance parameters in order to reduce the dimension of our parameter space and therefore, increase the computational efficiency of our Markov Chain Monte Carlo (MCMC) sampling algorithm.

Let c_i be the mixture component label of cell i in a particular MCMC iteration (assume $c_i \in \{1, 2, \ldots, K\}$, where K is the total number of components in this particular MCMC iteration), $\boldsymbol{c}$ be the vector $(c_1, c_2, \ldots, c_N)^T$ and $\tilde{\boldsymbol{\pi}}$ be the $K \times G$ matrix with elements $\tilde{\pi}_{kj}$ representing the expression probability of gene j for cells with component label k, $k = 1, 2, \ldots, K$, $j = 1, 2, \ldots, G$. The prior distribution and likelihood with respect to the expression probabilities matrix $\tilde{\boldsymbol{\pi}}$ are

$$p(\tilde{\boldsymbol{\pi}}|\boldsymbol{c}, \alpha_\pi, \beta_\pi) = \prod_{k=1}^{K} \prod_{j=1}^{G} \frac{\Gamma(\alpha_\pi + \beta_\pi)}{\Gamma(\alpha_\pi)\Gamma(\beta_\pi)} \tilde{\pi}_{kj}^{\alpha_\pi - 1} (1 - \tilde{\pi}_{kj})^{\beta_\pi - 1} \tag{2}$$

and

$$p(\boldsymbol{z}|\boldsymbol{c},\tilde{\boldsymbol{\pi}}) = \prod_{k=1}^{K}\prod_{i:c_i=k}\prod_{j=1}^{G}\tilde{\pi}_{kj}^{z_{ij}}(1-\tilde{\pi}_{kj})^{1-z_{ij}} \tag{3}$$

respectively. The products of (2) and (3) define the full conditional posterior density of $(\tilde{\boldsymbol{\pi}}, \boldsymbol{z})$ given $\boldsymbol{c}$ (up to a proportionality constant). Due to the conjugacy of the Bernoulli and beta distributions, we can easily integrate out $\tilde{\boldsymbol{\pi}}$ and have

$$\begin{aligned} &p(\boldsymbol{z}|\boldsymbol{c},\alpha_\pi,\beta_\pi) \\ &\quad= \prod_{k=1}^{K}\prod_{j=1}^{G}\frac{\Gamma(\alpha_\pi+\beta_\pi)}{\Gamma(\alpha_\pi)\Gamma(\beta_\pi)} \\ &\qquad\cdot\frac{\Gamma(\alpha_\pi+\sum_{i:c_i=k}z_{ij})\Gamma(\beta_\pi+n_k-\sum_{i:c_i=k}z_{ij})}{\Gamma(\alpha_\pi+\beta_\pi+n_k)}, \end{aligned}$$

where n_k is the total number of cells with component label k.

In addition for the dropout elements (i.e., $(i,j): x_{ij} = \text{N.A.}$), we have

$$\begin{aligned} &p(y_{ij}|g_j,\delta_j,z_{ij},\sigma_{0j}^2,\sigma_{1j}^2) \\ &\quad= \exp\left\{-\frac{(y_{ij}-g_j-\delta_j z_{ij}-s_i)^2}{2(\sigma_{0j}^2(1-z_{ij})+\sigma_{1j}^2 z_{ij})}\right\}\Big/\sqrt{2\pi\left(\sigma_{0j}^2(1-z_{ij})+\sigma_{1j}^2 z_{ij}\right)} \end{aligned}$$

and

$$p(x_{ij}=\text{N.A.}|y_{ij},\gamma_1,\gamma_2) = \Phi(\gamma_1 y_{ij}+\gamma_2).$$

Integrating out the unobserved y_{ij}, we have

$$\begin{aligned} &p(x_{ij}=\text{N.A.}|g_j,\delta_j,z_{ij},\sigma_{0j}^2,\sigma_{1j}^2,\gamma_1,\gamma_2) \\ &\quad= \Phi\left(\frac{\gamma_1(g_j+\delta_j z_{ij}+s_i)+\gamma_2}{\sqrt{1+\gamma_1^2(\sigma_{0j}^2(1-z_{ij})+\sigma_{1j}^2 z_{ij})}}\right) \end{aligned}$$

for the dropout elements.

We then adopt Gibbs sampling, when conjugacy is satisfied, and use a Metropolis algorithm otherwise for the remaining parameters. The detailed sampling algorithm is described in Section S1 [Liu, Warren and Zhao (2019)].

2.5. *Competing methods*. We compare our newly developed method with five methods designed specifically for scRNA-seq data analysis, BackSPIN [Zeisel et al. (2015)], CIDR [Lin, Troup and Ho (2017)], SIMLR [Wang et al. (2017)], SNN-Cliq [Xu and Su (2015)] and ZIFA [Pierson and Yau (2015)].

BackSPIN is a bi-clustering algorithm. It starts by sorting both cells and genes using the SPIN algorithm [Tsafrir et al. (2005)]. Then, on the cell dimension, it

finds a splitting point that maximizes a within-group cell-cell correlation measure. Genes are assigned to the cell group with the highest expression levels. This process is repeated on the two halves iteratively until a given number of split cycles is reached. Unlike BasCluS, BackSPIN requires a user-specified number of split cycles to determine the parsimony of clustering structures (however, the number of clusters cannot be directly specified by the user). In our analysis we adjust the number of split cycles parameter so that the final number of clusters detected is close to the true value.

CIDR fits a logistic regression model to estimate the relationship between the dropout probability and the normalized gene expression level. For each gene and cell the estimated dropout probability is used as a weight for dropout imputation in the calculation of a cell-cell dissimilarity matrix. CIDR then performs a principal coordinate analysis on the obtained dissimilarity matrix and groups cells using hierarchical clustering on the first few principal coordinates. CIDR implements an approach based on the Calinski–Harabasz index [Caliński and Harabasz (1974)] to determine the number of clusters. In our analysis we apply CIDR with both the true number of clusters and its automatically selected number of clusters (we denote the latter approach as "CIDR0").

SIMLR begins by training a cell-to-cell similarity matrix using multikernel learning. The similarity matrix is then input into a stochastic neighbor embedding method [Maaten and Hinton (2008)] to project the cells into a lower dimensional space. Afterward, SIMLR applies k-means [Forgy (1965)] in the latent space to cluster cells. SIMLR was shown to have superior performance over alternative clustering methods based on other similarity measurements. SIMLR requires user-specified number of clusters and number of reduced dimensions as input parameters. In subsequent analyses we provide the true number of clusters to SIMLR. As recommended by the authors, we set the number of reduced dimensions equal to the number of clusters (default value in the associated R function).

SNN-Cliq first computes the Euclidean distance between cells and then constructs a cell-cell similarity matrix as a function of the shared nearest neighbors (SNN). The clustering structure of cells is determined using a graph partition algorithm on the cell-cell similarity matrix. Like BackSPIN, the number of clusters cannot be directly specified in SNN-Cliq. Instead, it has two parameters that control the parsimony of the clustering results, one parameter for quasi-clique finding, r, and one parameter for cluster merging, m. In our analysis we fix m with its default value of 0.5 and test SNN-Cliq with $r = 0.1, 0.2, \ldots, 0.9$ (only results with $r = 0.3, 0.4, 0.5$ are shown for simplicity).

Although ZIFA was originally designed for dimension reduction, we include it because it models dropout events explicitly and the latent space representation of cells was demonstrated to be informative for classification purpose. ZIFA utilizes the framework of factor analysis to model true gene expressions. In addition it assumes that the dropout probability has the form $p_{ij} = \exp(-\lambda y_{ij}'^2)$, where y_{ij}' is the

true underlying expression level of gene j in cell i (without amplification). Similar to SIMLR, ZIFA also requires user-specified number of clusters and number of reduced dimensions. We provide the true number of clusters to ZIFA. Due to the lack of guidance on how to choose an appropriate number of reduced dimensions, we test ZIFA with this value set to be 5, 10 and 20.

We also test the performance of setting the binary expression status z to be 1, if the raw RNA-seq measurement is nonzero and zero otherwise, as well as estimating it with method like MAST [Finak et al. (2015)]. To have a fair comparison with BasCluZ, we apply hierarchical clustering to these z matrices and provide the true number of clusters to call cell types.

To evaluate the performance of each method, we compute the normalized mutual information (NMI) [Vinh, Epps and Bailey (2010)] and adjusted Rand index (ARI) [Hubert and Arabie (1985)] between cluster labels assigned by a method and the ground truth. Given N samples and two clustering results U (P classes) and V (Q classes), NMI between U and V is defined as

$$\mathrm{NMI}(U,V)=\frac{\mathrm{MI}(U,V)}{\max\{H(U),H(V)\}},$$

where

$$\mathrm{MI}(U,V)=\sum_{p=1}^{P}\sum_{q=1}^{Q}\frac{|U_p\cap V_q|}{N}\log\frac{N|U_p\cap V_q|}{|U_p|\times|V_q|},$$

$$H(U)=-\sum_{p=1}^{P}\frac{|U_p|}{N}\log\frac{|U_p|}{N},$$

$$H(V)=-\sum_{q=1}^{Q}\frac{|V_q|}{N}\log\frac{|V_q|}{N}$$

and $|\cdot|$ denotes the cardinality of a set. NMI takes on values between 0 and 1. With P and Q fixed, the larger NMI is, the more concordant the two clustering results are.

ARI between U and V is defined as

$$\mathrm{ARI}(U,V)=\frac{\sum_{p=1}^{P}\sum_{q=1}^{Q}\binom{|U_p\cap V_q|}{2}-\sum_{p=1}^{P}\binom{|U_p|}{2}\sum_{q=1}^{Q}\binom{|V_q|}{2}/\binom{n}{2}}{[\sum_{p=1}^{P}\binom{|U_p|}{2}+\sum_{q=1}^{Q}\binom{|V_q|}{2}]/2-\sum_{p=1}^{P}\binom{|U_p|}{2}\sum_{q=1}^{Q}\binom{|V_q|}{2}/\binom{n}{2}}$$

which is equal to

$$\frac{\sum_{i<l}\delta(c_i^{(U)},c_l^{(U)})\delta(c_i^{(V)},c_l^{(V)})-\sum_{i<l}\delta(c_i^{(U)},c_l^{(U)})\sum_{i<l}\delta(c_i^{(V)},c_l^{(V)})/\binom{n}{2}}{[\sum_{i<l}\delta(c_i^{(U)},c_l^{(U)})+\sum_{i<l}\delta(c_i^{(V)},c_l^{(V)})]/2-\sum_{i<l}\delta(c_i^{(U)},c_l^{(U)})\sum_{i<l}\delta(c_i^{(V)},c_l^{(V)})/\binom{n}{2}}.$$

Note, substituting $\delta(c_i^{(U)}, c_l^{(U)})$ with $\delta(c_i^{(m)}, c_l^{(m)})$ and $\delta(c_i^{(V)}, c_l^{(V)})$ with S_{il}, which equals to $E(\delta(c_i, c_l))$ asymptotically, will give us the objective function in equation (1) to determine the clustering structure using the posterior similarity matrix $\boldsymbol{S}$. ARI has a maximum value of 1 and takes values around 0 for two randomly assigned clustering labels (it can occasionally become negative). Similarly to NMI, with P and Q fixed, a larger ARI indicates better agreement between two clustering results.

3. Simulation studies.

3.1. *Simulation settings*. We conduct simulation studies to evaluate the effectiveness of our method in identifying cell subtypes as well as inferring underlying gene expression patterns.

We simulate datasets with 300 cells and 200 genes. The cells are randomly assigned to 10 classes with equal probability. For each class we sample an expression probability vector $\tilde{\boldsymbol{\pi}}_k$ with 200 independent elements from Beta(0.9, 0.9). Given the class assignment and the expression probability, we simulate an expression status indicator vector for each cell with corresponding Bernoulli distribution. We then simulate $g_j \overset{\text{i.i.d.}}{\sim} \mathrm{N}(\tilde{\nu_g}, \tilde{\tau_g}^2)$, $s_i \overset{\text{i.i.d.}}{\sim} \mathrm{N}(\tilde{\nu_s}, \tilde{\tau_s}^2)$, $\delta_j \overset{\text{i.i.d.}}{\sim}$ truncated $\mathrm{N}(\tilde{\nu_\delta}, \tilde{\tau_\delta}^2; >0)$, $\sigma_{0j}^2 \overset{\text{i.i.d.}}{\sim}$ truncated $\mathrm{N}(\tilde{\nu_0}, \tilde{\tau_0}^2; >0)$, $\sigma_{1j}^2 \overset{\text{i.i.d.}}{\sim}$ truncated $\mathrm{N}(\tilde{\nu_1}, \tilde{\tau_1}^2; >0)$, $i = 1, 2, \ldots, 300$, $j = 1, 2, \ldots, 200$. Parameters $\tilde{\nu_g}$, $\tilde{\tau_g}$, $\tilde{\nu_s}$, $\tilde{\tau_s}$ are set as -23, 3, 20 and 2, respectively. We choose these values based on analysis results of a scRNA-seq dataset [Okaty et al. (2015)] (means are matched approximately while standard deviations are set to be slightly larger than real data estimates) in order to ensure that our simulation results are useful in practice. To simulate δ_j, σ_{0j}^2 and σ_{1j}^2, we fix $\tilde{\tau_\delta}$, $\tilde{\tau_0}$, and $\tilde{\tau_1}$ to be 2, $\tilde{\nu_0}$ and $\tilde{\nu_1}$ to be 6, and choose two sets of $\tilde{\nu_\delta}$, 5 and 8, to have varying "noise levels" in the Gaussian mixture model component of the model. Smaller δ_j will result in more overlap between the two Gaussian distributions, hence making expressed and unexpressed states more difficult to separate. Given these parameters, we simulate expression value (without dropout) as $y_{ij} \sim \mathrm{N}(g_j + \delta_j z_{ij} + s_i, \sigma_{0j}^2 z_{ij} + \sigma_{1j}^2(1 - z_{ij}))$ independently for $i = 1, 2, \ldots, 300$, $j = 1, 2, \ldots, 200$. Finally, we sample binary variables $m_{ij} \sim \text{Bernoulli}(\Phi(\gamma_1 y_{ij} + \gamma_2))$ and simulate dropout events by letting $x_{ij} = y_{ij}$, if $m_{ij} = 0$ and $x_{ij} = \text{N.A.}$ otherwise. We choose eight sets of (γ_1, γ_2): $(-0.05, -0.10)$, $(-0.05, -0.50)$, $(-0.10, -0.50)$, $(-0.10, -1.00)$, $(-1.00, -2.00)$, $(-1.00, -4.00)$, $(-5.00, -5.00)$, $(-5.00, -10.00)$, so that the final dropout rates range from 0.15 to 0.50. A summary of the 16 simulation scenarios (two sets of $\tilde{\nu_\delta}$ times eight sets of (γ_1, γ_2)) are included in Table 1. We simulate a total of 80 datasets for analysis, five for each of the 16 settings.

3.2. *Simulation results*. We apply BasClu, BackSPIN, CIDR, SIMLR, SNN-Cliq, ZIFA and hierarchical clustering with binary expression status $\boldsymbol{z}$ (estimated

TABLE 1
Summary of simulated datasets

Scenario	1	2	3	4	5	6	7	8
$\tilde{\nu}_\delta$	5.00	5.00	5.00	5.00	5.00	5.00	5.00	5.00
γ_1	−0.05	−0.05	−0.10	−0.10	−1.00	−1.00	−5.00	−5.00
γ_2	−0.10	−0.50	−0.50	−1.00	−2.00	−4.00	−5.00	−10.00
Dropout rate	0.471	0.321	0.344	0.199	0.377	0.254	0.437	0.389
	(0.005)	(0.003)	(0.007)	(0.007)	(0.036)	(0.013)	(0.068)	(0.016)
Scenario	9	10	11	12	13	14	15	16
$\tilde{\nu}_\delta$	8.00	8.00	8.00	8.00	8.00	8.00	8.00	8.00
γ_1	−0.05	−0.05	−0.10	−0.10	−1.00	−1.00	−5.00	−5.00
γ_2	−0.10	−0.50	−0.50	−1.00	−2.00	−4.00	−5.00	−10.00
Dropout rate	0.442	0.299	0.306	0.173	0.331	0.221	0.390	0.331
	(0.004)	(0.005)	(0.005)	(0.004)	(0.010)	(0.010)	(0.010)	(0.009)

Mean dropout rate across the five replicates in each scenario is reported with standard deviation in the parenthesis.

with raw counts and MAST) to the simulated datasets and calculate NMI and ARI between the inferred clustering results and the true class labels in each scenario (Table 2, Table S1 and Table S2 [Liu, Warren and Zhao (2019)]).

For BasClu we run four independent chains using our MCMC algorithm, each for 500,000 iterations. Initial values for the chains are randomly sampled from overdispersed distributions (with respect to the true posteriors) based on the posterior distribution estimates from preliminary runs of the model (e.g., we sample initial values for z_{ij} from Bernoulli(0.5), δ_j from Uniform(1, 50)). After discarding the first 300,000 samples in each chain as burn-in, the remaining samples are thinned such that we collect every tenth posterior sample. Convergence is assessed using the Gelman–Rubin diagnostic [Gelman and Rubin (1992)]. Mean Gelman–Rubin statistics across all parameters (for binary expression status z we check its mean value across genes and cells) are reported in Table S2. Samples from all chains are combined when making inference on the model parameters. We also report the number of clusters estimated by the posterior similarity matrix-based clustering approach of our method (BasCluS, Table S2). Comparing it with the true number of clusters (10), we see that in most settings our method provides accurate estimation.

Since the number of clusters could affect the value of NMI and ARI, we provide the true number of clusters in the z-based hierarchical clustering version of our method (BasCluZ), SIMLR, CIDR, ZIFA and binary expression status defined by whether the raw count is zero (denoted as "rawZ" in subsequent analysis) as well as MAST (denoted as "MASTZ") for a fair comparison (Table 2 and Table S1); for

TABLE 2
NMI of BasClu, CIDR, SIMLR and ZIFA in the simulation study

Scenario	1	2	3	4
BasCluZ	0.965 (0.025)	1.000 (0.000)	0.996 (0.004)	0.999 (0.003)
CIDR	0.182 (0.046)	0.288 (0.070)	0.361 (0.059)	0.543 (0.081)
SIMLR	0.271 (0.084)	0.552 (0.115)	0.758 (0.089)	0.888 (0.044)
ZIFA (5)	0.438 (0.085)	0.621 (0.050)	0.680 (0.022)	0.767 (0.037)
Scenario	5	6	7	8
BasCluZ	0.999 (0.003)	1.000 (0.000)	0.999 (0.003)	0.999 (0.003)
CIDR	0.670 (0.071)	0.652 (0.069)	0.647 (0.073)	0.654 (0.086)
SIMLR	0.970 (0.014)	0.952 (0.019)	0.959 (0.006)	0.966 (0.016)
ZIFA (5)	0.812 (0.045)	0.826 (0.046)	0.785 (0.052)	0.795 (0.066)
Scenario	9	10	11	12
BasCluZ	1.000 (0.000)	1.000 (0.000)	1.000 (0.000)	1.000 (0.000)
CIDR	0.289 (0.027)	0.508 (0.065)	0.797 (0.047)	0.909 (0.030)
SIMLR	0.772 (0.064)	0.956 (0.019)	0.987 (0.013)	0.985 (0.013)
ZIFA (5)	0.711 (0.064)	0.817 (0.037)	0.881 (0.041)	0.920 (0.037)
Scenario	13	14	15	16
BasCluZ	1.000 (0.000)	1.000 (0.000)	1.000 (0.000)	1.000 (0.000)
CIDR	0.952 (0.014)	0.958 (0.015)	0.951 (0.028)	0.949 (0.032)
SIMLR	0.995 (0.011)	1.000 (0.000)	0.984 (0.016)	0.995 (0.007)
ZIFA (5)	0.930 (0.021)	0.926 (0.023)	0.921 (0.046)	0.937 (0.024)

The number in each cell is the mean value across the five replicates, with standard deviation in the parentheses. In the parentheses following "ZIFA," the value k denotes the prespecified number of latent factors (we list the result with $k = 5$ here and the other cases in Table S2).

BackSPIN and SNN-Cliq the number of clusters could not be specified directly, so we test these two methods by setting the tuning parameters that control the parsimony of the clusterings to different values and report the NMI and ARI as well as the number of clusters identified in each setting (Table S2). As previously discussed, for SIMLR we set the number of reduced dimensions as the true number of subtypes, and for ZIFA we explore a range of values (5, 10 and 20).

We can see from the NMI and ARI values that our method performs well in all scenarios while the other methods perform much worse, especially in Scenarios 1 to 9. In fact, we observe all methods tend to perform better in Scenarios 9 to 16 compared to Scenarios 1 to 8, where $\tilde{\nu}_\delta$ is smaller. This is because in our simulations larger $\tilde{\nu}_\delta$ leads to greater differences between expressed and unexpressed states for the genes (z), hence resulting in stronger signals to separate

cells from different classes. In addition we also notice that in Scenarios 1 and 9, the dropout rate is large (around 0.4 to 0.5); more importantly, γ_1 in these two cases is small (-0.05) which means that the dropout probability is only weakly related to the expression measurements. This suggests that not only genes barely expressed can drop out, but also genes with relatively large ("amplified") expression values also have a fair chance of being missing, hence treating all dropouts as truly unexpressed or neglecting certain factors in the dropout probability would more severely impact the clustering accuracy. Compared to the other settings, such cases present greater challenges to correctly extract the information from the zero-counts. Indeed, in less challenging cases (Scenarios 10 to 16) we see that SIMLR, CIDR and ZIFA can estimate the clustering patterns much better.

In addition to accurately estimating the clustering structures, our method can also reveal information regarding the underlying gene expression patterns. In Figure S1 we compare the posterior means of $\boldsymbol{g}((g_1, g_2, \ldots, g_G)^T)$, $\boldsymbol{\delta}((\delta_1, \delta_2, \ldots, \delta_G)^T)$, $\boldsymbol{\sigma_0^2}((\sigma_{01}^2, \sigma_{02}^2, \ldots, \sigma_{0G}^2)^T)$, $\boldsymbol{\sigma_1^2}((\sigma_{11}^2, \sigma_{12}^2, \ldots, \sigma_{1G}^2)^T)$ and $\boldsymbol{z}$ estimated by our method and their true values (results from one replicate of Scenario 1 are illustrated as an example, but results from other settings are similar or better). We can see that the estimates of $\boldsymbol{g}$ (Figure S1(a)), $\boldsymbol{\delta}$ (Figure S1(b)) and $\boldsymbol{z}$ (Figures S1(e) and S1(f)) are close to the corresponding true values. In particular we evaluate the average of the absolute difference between the posterior mean of $\boldsymbol{z}$ and the true value $\boldsymbol{z}^0$, $\text{ave}_{ij}(|\,\text{ave}_m(z_{ij}^{(m)}) - z_{ij}^0|)$ which is equal to $\text{ave}_{ij}(\text{ave}_m\,|z_{ij}^{(m)} - z_{ij}^0|) \approx \text{ave}_{ij}(P(z_{ij}^{(m)} \neq z_{ij}^0))$ (Table S2). In comparison we also calculate $\text{ave}_{ij}(|\,P(z_{ij} = 1) - z_{ij}^0|)$ in the Gaussian mixture models fitted for each gene individually using its true expression levels (without missing values, using function "normalmixEM" in the R package "mixtools" [Benaglia et al. (2009)]). For Senarios 1 to 8 with $\tilde{\nu}_\delta = 5$, this value is around 0.255 with standard deviation 0.008 across the five replicates, and for Scenarios 9 to 16, with $\tilde{\nu}_\delta = 8$, it is around 0.102 with standard deviation 0.009. If there is no information shared across genes and cells, we could expect error in our model to be greater than the latter one due to the high dropout rates. However, we observe that in most cases they are roughly the same level (sometimes error in our model is even smaller), suggesting that leveraging clustering structure information assists in estimating the expression status of individual genes and cells. We notice that overall posterior estimates of $\boldsymbol{\sigma_0^2}$ (Figure S1(c)) and $\boldsymbol{\sigma_1^2}$ (Figure S1(d)) are also close to their true values except in a few cases where the posterior means are much larger. A further investigation into "genes" suggests that they either have large dropout rates or two highly overlapped Gaussian distributions (due to small δ_j, large σ_{0j}^2 or large σ_{1j}^2).

In order to test the performance of our method with more diverse cluster sizes, we also repeat similar analysis with the the probability for the 10 classes set as (0.067, 0.067, 0.067, 0.067, 0.067, 0.133, 0.133, 0.133, 0.133, 0.133) (Table S3). We can see that in these scenarios, our method also estimates the clustering structure accurately.

TABLE 3
Summary of scRNA-seq datasets in real data analysis

Dataset	Cell population	No. subtypes
Okaty [Okaty et al. (2015)]	56 cells from mouse serotonin neuron system	6
Pollen [Pollen et al. (2014)]	65 cells from human developing cortex (with a high coverage and a low coverage, so 130 samples in total)	8
Usoskin [Usoskin et al. (2015)]	622 cells from mouse primary sensory system	11

4. Application to real single-cell RNA-sequencing data. We apply our method to three datasets where scRNA-seq technology is utilized to identify novel cell subtypes. In addition they span a wide range of sample sizes as well as numbers of cell subtypes, hence representing a variety of scenarios we might come across in many real scRNA-seq analyses. A brief summary of the three datasets is shown in Table 3. In our analysis we treat the biologically validated cell subtypes identified in the original studies as ground truths to evaluate the different clustering methods. For BackSPIN, CIDR, MAST, SIMLR, SNN-Cliq and ZIFA we input log(normalized expression + 1) (+1 to avoid $-\infty$ when taking log; we use TPM, the normalized expression data provided by the original scRNA-seq studies, here), as commonly done for methods that require "true" expression values, while for our methods we input log(raw counts) since it incorporates an RPKM-like normalization procedure. Similar to the simulation studies, we give the correct number of clusters to CIDR, SIMLR, ZIFA and the z-based hierarchical clusterings (BasCluZ, MASTZ and rawZ) for a fair comparison. We also report the results of BackSPIN and SNN-Cliq with the parsimony-controlling parameters set to different values.

Since there are tens of thousands of genes in each dataset, we utilize the protocol of [Lake et al. (2016)] to preselect a few hundred informative genes for clustering purpose. Briefly, in the first step squared coefficient of variation (cv^2) of each gene is fitted against the inverse of mean expression value to obtain a curve with cv^2 as a function of mean expression. Next, an expected cv^2 for each gene is computed based on this estimated function. Finally, genes with cv^2 at least two standard deviations beyond the expected level are considered over dispersed and removed. In addition we add a dropout rate threshold to ensure that only genes with an adequate number of observed measurements are included. Specifically, for the Okaty and Pollen datasets we exclude genes selected from the previous step that have dropout rates greater than 50%, and for the Usoskin dataset we exclude genes with dropout rates greater than 30% as the 50% threshold results in too few genes remaining for analysis. In total we select 199, 136 and 134 genes for the three datasets, respectively. Genes selected for each of the three datasets are provided in the Supplementary Material [Liu, Warren and Zhao (2019)]. Since BackSPIN,

CIDR and ZIFA include a dimension reduction/gene selection module within their frameworks, and SIMLR and SNN-Cliq work directly with a cell-cell expression distance matrix, we also test their performance with all genes supplied (genes with all zero measurements are excluded) to avoid potential performance decline due to information loss in the gene selection procedure.

For BasClu, we run four MCMC chains (each for 1,000,000 iterations on the Okaty dataset, 1,500,000 iterations on the Pollen dataset and 2,000,000 iterations on the Usoskin dataset) with initial values sampled from over-dispersed distributions. The last 200,000 samples of each chain are retained and thinned (keeping every tenth posterior sample). Convergence is assessed through visual inspection of trace plots and with the Gelman–Rubin diagnostic. Mean Gelman–Rubin statistics across all parameters (for binary expression status z we check its mean value across genes and cells) are reported in Table S4. There were no obvious signs of nonconvergence based on both of these tools.

4.1. *Okaty dataset*. We obtain the Okaty dataset [Okaty et al. (2015)], including the raw scRNA-seq read counts, the normalized gene expression values and the cell subtype labels from Supplementary Material accompanying the original publication. Before carrying out the clustering analysis, we examine the dropout patterns across cells and genes. In Figure 1(a) we plot the dropout rate computed for each cell against its log library size, while in Figure 1(b) we plot the dropout rate of each gene against its mean log expression level (normalized, zero expression excluded). The overall dropout rate is negatively correlated to both library size (Spearman correlation -0.38, p-value 0.004) and gene expression level (Spearman correlation -0.88, p-value $< 2.2 \times 10^{-16}$). This observation motivates us to model dropout probability as a function of the "amplified" expression level.

We summarize NMIs of the three methods in Table 4 and ARIs in Table S5 [Liu, Warren and Zhao (2019)]. Both metrics indicate that our method outperforms SIMLR, CIDR, BackSPIN, SNN-Cliq and ZIFA by producing clustering structure more consistent with the ground truth on this dataset. Surprisingly, hierarchical clustering, using the binary expression status defined by whether a count is zero or not, with selected genes only has larger NMI and ARI values than all the other sophisticated clustering methods. We think this is likely due to the strong signals of these genes. Zeros are more likely a result of the genes being truly unexpressed, so that more complex modeling does not necessarily lead to better inference of the expression pattern. However, this may not hold true in other cases (e.g., with all genes as input here). Besides detecting clustering structure of cells, our method also infers the underlying expression status of genes. From the heatmap of the posterior mean of z (Figure S17, cells are reordered according to their true subtypes), we see that some genes display differential expression patterns across subtypes. For example, transcriptional regulator Sox1 shows much higher expression levels in subtypes R1DR, R1MR and R2 (means of z 0.864, 0.807 and 0.816, respectively) than in subtypes R3, R5 and R6P (means of z 0.004, 0.003 and 0.001, respectively), while Meis2 is more likely to express in subtypes R5 and R6P (means

TABLE 4
NMI on three scRNA-seq datasets

Dataset	BasCluS*	BasCluZ	BackSPIN (2)*	BackSPIN (2, all)*	BackSPIN (3)*	BackSPIN (3, all)*
Okaty	0.740 [10]	0.713	0.431 [4]	0.573 [4]	0.604 [8]	0.567 [8]
Pollen#	0.693 [19]	0.723	–	–	0.590 [8]	0.795 [8]
Usoskin	0.689 [12]	0.696	–	–	0.703 [7]	0.647 [8]

Dataset	BackSPIN (4)*	BackSPIN (4, all)*	CIDR0*	CIDR0 (all)*	CIDR	CIDR (all)
Okaty	–	–	0.611 [8]	0.478 [6]	0.604	0.478
Pollen#	–	–	0.561 [6]	0.525 [6]	0.584	0.532
Usoskin	0.606 [13]	0.543 [16]	0.594 [4]	0.643 [4]	0.643	0.568

Dataset	SIMLR	SIMLR (all)	SNN-Cliq (0.3)*	SNN-Cliq (0.3, all)*	SNN-Cliq (0.4)*	SNN-Cliq (0.4, all)*
Okaty	0.700	0.661	0.613 [6]	0.287 [3]	0.656 [8]	0.287 [3]
Pollen#	0.682	0.846	0.354 [7]	0.065 [3]	0.566 [13]	0.526 [13]
Usoskin	0.641	0.588	0.499 [10]	0.068 [3]	0.675 [17]	0.336 [9]

Dataset	SNN-Cliq (0.5)*	SNN-Cliq (0.5, all)*	ZIFA (5)	ZIFA (5, all)	ZIFA (10)	ZIFA (10, all)
Okaty	0.668 [7]	0.475 [4]	0.657	0.266	0.363	0.121
Pollen#	0.549 [18]	0.497 [17]	0.704	0.660	0.469	0.489
Usoskin	0.437 [61]	0.428 [20]	0.630	0.434	0.552	0.373

Dataset	ZIFA (20)	ZIFA (20, all)	rawZ	rawZ (all)	MASTZ	MASTZ (all)
Okaty	0.197	0.144	0.737	0.506	0.501	0.141
Pollen#	0.436	0.494	0.570	0.365	0.354	0.246
Usoskin	0.450	0.089	0.527	0.244	0.702	0.337

In the parentheses following "ZIFA" and "SIMLR", the value denotes the prespecified number of latent factors; in the parentheses following "BackSPIN," the value denotes the number of split cycles; in the parentheses following "SNN-Cliq," the value denotes the quasiclique finding parameter; "all" means all genes are used. *Number of clusters detected by these methods are shown in the brackets following the NMIs. #In Pollen dataset three cells (six samples) in transition states are removed in evaluation due to unclear subtype identity.

of z 0.999 and 0.770, respectively) than in subtypes R1DR, R1MR, R2 and R3 (means of z 0.000, 0.000, 0.001, and 0.007, respectively). Similar patterns were also reported by the original study [Okaty et al. (2015)]. We highlight more examples in Figure S17. Here, we present some prominent cases using the posterior mean of z; a more rigorous statistical testing approach for differential expression gene identification could be developed in future work.

4.2. *Pollen dataset*. The Pollen dataset [Pollen et al. (2014)] has 65 cells, each sequenced at a high coverage and also down-sampled to a low depth in subsequent study to explore the effectiveness of low-coverage sequencing (so 130 samples in total). We obtained it from R package scRNAseq [Risso and Cole (2016)] which includes a subset with all neural cells from the original study [Pollen et al. (2014)]. Two hundred thirty-six more cells from pluripotent, skin and blood were also sequenced there. While cells from different sources are expected to have large transcriptomic differences, cells within a closely related cell type are considered more difficult to separate.

We list the NMI and ARI of each method in Table 4 and Table S5 [Liu, Warren and Zhao (2019)], respectively. Our method outperforms the others in most cases. With all genes as input, SIMLR and BackSPIN perform better than our method; however, with only preselected genes, SIMLR and BackSPIN's performance declines substantially which might indicate some information loss in the gene selection step. Besides checking concordance with known subtypes, we also compare the clustering labels of samples from the same cell. In all cases our method and SIMLR assign such pairs into the same subgroup while BackSPIN, CIDR, SNN-Cliq, ZIFA and hierarchical clustering with raw count and MAST defined z (rawZ, MASTZ) occasionally separate them. With selected genes as input, BackSPIN, CIDR, CIDR0 and SNN-Cliq ($r = 0.3$) separate two pairs, and MASTZ separates seven pairs; with both selected genes and all genes as input, ZIFA ($k = 5$) separates one pair, SNN-Cliq ($r = 0.4, 0.5$) separates more than 10 pairs, and rawZ separates 22 and four pairs. In addition we check the pattern of z posterior mean estimates (Figure S18) and are able to find some potential signatures of different subtypes. For example, gene Ddah1 is expressed at high level in *in vitro* derived neural progenitors and radial glia (mean of z 0.811 and 0.911, respectively) but shows very weak signals in maturing neurons A, maturing neurons B, newborn neurons and interneurons (mean of z 0.005, 0.005, 0.000 and 0.011, respectively); while the pattern of Stmn2 is the exactly opposite on these subtypes (mean of z 0.013, 0.082, 0.998, 0.999, 0.997 and 0.830, respectively), similar to what was discovered previously [Pollen et al. (2014)].

4.3. *Usoskin dataset*. We downloaded the Usoskin dataset [Usoskin et al. (2015)] from the Linnarsonn Lab webpage. Compared to the previous two datasets, this one has a much larger sample size.

As shown in Table 4 and Table S5 [Liu, Warren and Zhao (2019)], with a given number of clusters, our method has larger NMI and ARI than all the others except hierarchical clustering with the binary expression status estimated by MAST (with selected genes only) which performs slightly better. The heatmap based on the posterior mean of z presents a clear checker-board pattern (Figure S19). For example, subtype TH is marked by genes such as Th (mean of z 0.843 versus 0.017 in other subtypes), and subtypes NP1, NP2 and NP3 show high expression levels of genes such as Carhsp1 (mean of z 0.939, 0.830, 0.966, respectively, versus 0.098 in other subtypes). Besides these observations consistent with the original publication [Usoskin et al. (2015)], we notice more heterogeneities in certain subtypes. For example, the large group of TH cells might be further separated into two subgroups and genes such as Apoe (means of z 0.828 and 0.146) show differential expression patterns between the subgroups. More biological experiments might be needed to validate these findings.

5. Discussion. In this paper we developed an innovative method, BasClu, for scRNA-seq clustering. By modeling dropout events explicitly and dichotomizing gene expression status, BasClu can better uncover expression patterns from noisy, single-cell RNA-seq data. In addition by utilizing a DP framework to induce clustering structure, BasClu avoids prespecifying the number of clusters which is very desirable in practice. We note that in one version of BasClu, BasCluS, a posterior similarity matrix is used to automatically determine the number of clusters and assign cluster labels; in the other version, BasCluZ, users can decide how many clusters to call after obtaining posterior estimates of gene expression status, instead of choosing a number at the beginning of the analysis when little information is available.

When applied to our simulated datasets with varying separabilities of expression states and dropout rates, BasClu performs well by accurately predicting the number of clusters (BasCluS) and correctly partitioning the cells (both versions). It outperforms five other methods designed for scRNA-seq data analysis, BackSPIN, CIDR, SIMLR, SNN-Cliq and ZIFA, especially when genes have blurred boundaries between expressed and unexpressed states or the dropout rate is high. We also observed that BasClu can uncover the underlying expression patterns precisely in most cases.

Furthermore, we tested the effectiveness of BasClu on three real scRNA-seq datasets which span a wide range of sample sizes and total number of cell types. In most cases we saw that BasClu has improved performance over BackSPIN, CIDR, SIMLR, SNN-Cliq and ZIFA by producing clustering results more consistent with experimentally validated ones. Besides cell subtype detection, we also presented a few examples to illustrate the idea of using gene expression status inferred by BasClu to pinpoint marker genes for each cell subtype (more rigorous statistical framework for differential expressed gene identification is yet to be developed). In addition we noticed that when applied to these three datasets, BasCluS, unlike

in simulation studies, tends to estimate a greater number of subtypes in each cell population than previously detected. On the Usoskin dataset, for example, it could be the case that there are more heterogeneities; while on the Pollen dataset we observed that some small clusters identified by BasCluS are close to each other in z space. Similar "redundant" cluster phenomena with DP models were also reported by other studies [Xie and Xu (2017), Miller and Harrison (2014)]. Yet, in real data analysis it is generally hard to define cluster boundaries objectively. Therefore, we think it could be a useful practice to take desired parsimony level into account, check the estimates of z and evaluate the necessity of combining "similar" components when using clustering results obtained using BasCluS.

There are several other directions worth further exploration in future works. First, our current model can only handle hundreds of genes with respect to computation, so that the vast majority of genes sequenced have to be excluded from the clustering analysis. Although many of these genes are noninformative (we indeed see performance improvement of BackSPIN, CIDR, SIMLR, SNN-Cliq and ZIFA with preselected genes in most cases), due to the prevalence of dropout and high noise level, it is not always easy to select the most informative ones, and useful signals could be lost by removing too many genes. Therefore, a potential extension of our method could be incorporating a gene selection procedure (based on inferred expression status) within the framework or, alternatively, including some dimension reduction module, so that more genes could be considered in the first stage. Second, our model adopts the idea of RPKM, which means besides the true expression level, we consider gene length (absorbed into the background of each gene in our model) and sequencing depth as the main factors affecting measured counts and the dropout probability. In fact we also test a slightly modified version by letting s_i's in our model be random variables instead of log library sizes; in most cases this version has similar performance with our current model, and the $\boldsymbol{s}$ learned is highly correlated to log library size. More sophisticated models could be developed if experimental evidence provides additional insights about other factors playing a role in the sequencing mechanism.

SUPPLEMENTARY MATERIAL

Supplementary materials for "A hierarchical Bayesian model for single-cell clustering using RNA-sequencing data" (DOI: 10.1214/19-AOAS1250SUPPA; .pdf). Section S1: Sampling algorithm. Section S2: Additional simulation results. Section S3: Additional scRNA-seq data analysis results. Section S4: Data and code availability.

Supplementary data and code (DOI: 10.1214/19-AOAS1250SUPPB; .zip). Code used in this study and genes selected for each of the three datesets.

REFERENCES

BENAGLIA, T., CHAUVEAU, D., HUNTER, D. R. and YOUNG, D. (2009). mixtools: An R package for analyzing finite mixture models. *J. Stat. Softw.* **32** 1–29.

BRENNECKE, P., ANDERS, S., KIM, J. K., KOŁODZIEJCZYK, A. A., ZHANG, X., PROSERPIO, V., BAYING, B., BENES, V., TEICHMANN, S. A., MARIONI, J. C. et al. (2013). Accounting for technical noise in single-cell RNA-seq experiments. *Nat. Methods* **10** 1093–1095.

CALIŃSKI, T. and HARABASZ, J. (1974). A dendrite method for cluster analysis. *Comm. Statist.* **3** 1–27. MR0375641

FERGUSON, T. S. (1973). A Bayesian analysis of some nonparametric problems. *Ann. Statist.* **1** 209–230. MR0350949

FINAK, G., MCDAVID, A., YAJIMA, M., DENG, J., GERSUK, V., SHALEK, A. K., SLICHTER, C. K., MILLER, H. W., MCELRATH, M. J., PRLIC, M. et al. (2015). MAST: A flexible statistical framework for assessing transcriptional changes and characterizing heterogeneity in single-cell RNA sequencing data. *Genome Biol.* **16** 278.

FORGY, E. W. (1965). Cluster analysis of multivariate data: Efficiency versus interpretability of classifications. *Biometrics* **21** 768–769.

FRITSCH, A. and ICKSTADT, K. (2009). Improved criteria for clustering based on the posterior similarity matrix. *Bayesian Anal.* **4** 367–391. MR2507368

GELMAN, A. and RUBIN, D. B. (1992). Inference from iterative simulation using multiple sequences. *Statist. Sci.* **7** 457–472.

HUBERT, L. and ARABIE, P. (1985). Comparing partitions. *J. Classification* **2** 193–218.

LAKE, B. B., AI, R., KAESER, G. E., SALATHIA, N. S., YUNG, Y. C., LIU, R., WILDBERG, A., GAO, D., FUNG, H.-L., CHEN, S. et al. (2016). Neuronal subtypes and diversity revealed by single-nucleus RNA sequencing of the human brain. *Science* **352** 1586–1590.

LIN, P., TROUP, M. and HO, J. W. K. (2017). CIDR: Ultrafast and accurate clustering through imputation for single-cell RNA-seq data. *Genome Biol.* **18** 59.

LIU, Y., WARREN, J. L and ZHAO, H. (2019). Supplement to "A hierarchical Bayesian model for single-cell clustering using RNA-sequencing data." DOI:10.1214/19-AOAS1250SUPPA, DOI:10.1214/19-AOAS1250SUPPB.

MAATEN, L. V. D. and HINTON, G. (2008). Visualizing data using t-SNE. *J. Mach. Learn. Res.* **9** 2579–2605.

MILLER, J. W. and HARRISON, M. T. (2014). Inconsistency of Pitman–Yor process mixtures for the number of components. *J. Mach. Learn. Res.* **15** 3333–3370. MR3277163

MORTAZAVI, A., WILLIAMS, B. A., MCCUE, K., SCHAEFFER, L. and WOLD, B. (2008). Mapping and quantifying mammalian transcriptomes by RNA-Seq. *Nat. Methods* **5** 621–628.

OKATY, B. W., FRERET, M. E., ROOD, B. D., BRUST, R. D., HENNESSY, M. L., KIM, J. C., COOK, M. N., DYMECKI, S. M. et al. (2015). Multi-scale molecular deconstruction of the serotonin neuron system. *Neuron* **88** 774–791.

PATEL, A. P., TIROSH, I., TROMBETTA, J. J., SHALEK, A. K., GILLESPIE, S. M., WAKIMOTO, H., CAHILL, D. P., NAHED, B. V., CURRY, W. T., MARTUZA, R. L. et al. (2014). Single-cell RNA-seq highlights intratumoral heterogeneity in primary glioblastoma. *Science* **344** 1396–1401.

PIERSON, E. and YAU, C. (2015). ZIFA: Dimensionality reduction for zero-inflated single-cell gene expression analysis. *Genome Biol.* **16** 241.

POLLEN, A. A., NOWAKOWSKI, T. J., SHUGA, J., WANG, X., LEYRAT, A. A., LUI, J. H., LI, N., SZPANKOWSKI, L., FOWLER, B., CHEN, P. et al. (2014). Low-coverage single-cell mRNA sequencing reveals cellular heterogeneity and activated signaling pathways in developing cerebral cortex. *Nat. Biotechnol.* **32** 1053–1058.

RISSO, D. and COLE, M. (2016). scRNAseq: A collection of public single-cell RNA-seq datasets. R Package Version 1.4.0.

STEGLE, O., TEICHMANN, S. A. and MARIONI, J. C. (2015). Computational and analytical challenges in single-cell transcriptomics. *Nat. Rev. Genet.* **16** 133–145.

TASIC, B., MENON, V., NGUYEN, T. N., KIM, T. K., JARSKY, T., YAO, Z., LEVI, B., GRAY, L. T., SORENSEN, S. A., DOLBEARE, T. et al. (2016). Adult mouse cortical cell taxonomy revealed by single cell transcriptomics. *Nat. Neurosci.* **19** 335–346.

TSAFRIR, D., TSAFRIR, I., EIN-DOR, L., ZUK, O., NOTTERMAN, D. A. and DOMANY, E. (2005). Sorting points into neighborhoods (SPIN): Data analysis and visualization by ordering distance matrices. *Bioinformatics* **21** 2301–2308.

USOSKIN, D., FURLAN, A., ISLAM, S., ABDO, H., LÖNNERBERG, P., LOU, D., HJERLING-LEFFLER, J., HAEGGSTRÖM, J., KHARCHENKO, O., KHARCHENKO, P. V. et al. (2015). Unbiased classification of sensory neuron types by large-scale single-cell RNA sequencing. *Nat. Neurosci.* **18** 145–153.

VALLEJOS, C. A., RISSO, D., SCIALDONE, A., DUDOIT, S. and MARIONI, J. C. (2017). Normalizing single-cell RNA sequencing data: Challenges and opportunities. *Nat. Methods* **14** 565–571.

VINH, N. X., EPPS, J. and BAILEY, J. (2010). Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance. *J. Mach. Learn. Res.* **11** 2837–2854. MR2738784

WANG, B., ZHU, J., PIERSON, E., RAMAZZOTTI, D. and BATZOGLOU, S. (2017). Visualization and analysis of single-cell RNA-seq data by kernel-based similarity learning. *Nat. Methods* **14** 414–416.

XIE, F. and XU, Y. (2017). Bayesian repulsive Gaussian mixture model. ArXiv Preprint. Available at arXiv:1703.09061.

XU, C. and SU, Z. (2015). Identification of cell types from single-cell transcriptomes using a novel clustering method. *Bioinformatics* **31** 1974–1980.

ZEISEL, A., MUÑOZ-MANCHADO, A. B., CODELUPPI, S., LÖNNERBERG, P., LA MANNO, G., JURÉUS, A., MARQUES, S., MUNGUBA, H., HE, L., BETSHOLTZ, C. et al. (2015). Cell types in the mouse cortex and hippocampus revealed by single-cell RNA-seq. *Science* **347** 1138–1142.

DEPARTMENT OF BIOSTATISTICS
SCHOOL OF PUBLIC HEALTH
YALE UNIVERSITY
NEW HAVEN, CONNECTICUT 06520
USA
E-MAIL: yiyi.liu@yale.edu
joshua.warren@yale.edu
hongyu.zhao@yale.edu