Biostatistics (2023) **24**, 1, *pp.* 161–176
doi: 10.1093/biostatistics/kxab011
Advance Access publication on September 14, 2021

Accounting for technical noise in Bayesian graphical models of single-cell RNA-sequencing data

JIHWAN OH, CHANGGEE CHANG, QI LONG*

Department of Biostatistics, Epidemiology and Informatics, Perelman School of Medicine, University of Pennsylvannia, 423 Guardian Drive, Philadelphia, PA 19104, USA

qlong@upenn.edu

Summary

Single-cell RNA-sequencing (scRNAseq) data contain a high level of noise, especially in the form of zero-inflation, that is, the presence of an excessively large number of zeros. This is largely due to dropout events and amplification biases that occur in the preparation stage of single-cell experiments. Recent scRNAseq experiments have been augmented with unique molecular identifiers (UMI) and External RNA Control Consortium (ERCC) molecules which can be used to account for zero-inflation. However, most of the current methods on graphical models are developed under the assumption of the multivariate Gaussian distribution or its variants, and thus they are not able to adequately account for an excessively large number of zeros in scRNAseq data. In this article, we propose a single-cell latent graphical model (scLGM)—a Bayesian hierarchical model for estimating the conditional dependency network among genes using scRNAseq data. Taking advantage of UMI and ERCC data, scLGM explicitly models the two sources of zero-inflation. Our simulation study and real data analysis demonstrate that the proposed approach outperforms several existing methods.

Keywords: Bayesian hierarchical model; Graphical models; Single-cell RNA-sequencing; Zero-inflation.

1. Introduction

Advances in high-throughput sequencing technology have paved the way for utilizing RNA-sequencing data in biomedical research. Especially during the last decade, various statistical methods have been developed to analyze high-dimensional data from the *bulk RNA-sequencing* (bRNAseq) experiments. For example, Chun *and others* (2015) proposed a method to retrieve gene networks from bRNAseq data. However, the bRNAseq technology ignores heterogeneity among individual cells and is not appropriate for analyzing the data with cellular diversity, because the expression levels are summed over different types of input cells in the tissue of interest.

More recently, researchers started to use the *single-cell RNA-sequencing* (scRNAseq) technology (Tang *and others*, 2009), which was nominated to be the “Method of the Year 2013” by Nature Methods (Editorial, 2014). Unlike the traditional bRNAseq, each observation from scRNAseq experiments consists of gene expression levels from each individual cell. This fundamental difference enables scientists to have better views into cell-to-cell heterogeneity, such as subpopulation identification (Buettner *and others*, 2015),

*To whom correspondence should be addressed.

heterogeneity of cell responses (Harari *and others*, 2005), and stochasticity of gene expression (Elowitz *and others*, 2002).

On the other hand, the scRNAseq technology brought us its own problems. Even though scRNAseq data are often structurally indistinguishable from bRNAseq data, the scarcity in starting materials in scRNAseq experiments often results in technical noise (Jia *and others*, 2017)—highly frequent dropout events and severe amplification biases. The dropout event refers to the situation where a transcript expressed in the cell is lost during the library preparation (Gong *and others*, 2018), and the amplification bias happens when the end product of the amplification does not faithfully recapitulate the amount of starting DNA (Islam *and others*, 2014).

In this article, we focus on the problem of inferring the gene conditional dependency network from scRNAseq data by properly addressing the aforementioned special characteristics. Note that most of the available methods are not suitable for scRNAseq data. For example, many existing methods on graphical models (Meinshausen and Bühlmann, 2006; Yuan and Lin, 2007; Banerjee *and others*, 2008; Peng *and others*, 2009; Fan *and others*, 2009; Lam and Fan, 2009; Cai *and others*, 2011; Li *and others*, 2012) including *graphical LASSO* (GLASSO) (Friedman *and others*, 2008) adopted the multivariate Gaussian assumption, as it simplifies the identification of conditional independence between a pair of random variables into the problem of identifying the zero entry in the precision matrix. On these foundations, many researchers went one step further towards more general settings (Fukumizu *and others*, 2007; Liu *and others*, 2009, 2011, 2012; Harris and Drton, 2013; Voorman *and others*, 2013; Li *and others*, 2014; Székely and Rizzo, 2014; Wang *and others*, 2015). However, these methods are not capable of taking into account the scRNAseq related issues.

Recently, several new methods for the analysis of scRNAseq data have been introduced. Oh *and others* (2018) proposed a machine learning method, in which the zero inflations are regarded as outliers. They proposed a method in which two robust statistical methods—support vector regression and Hilbert–Schmidt information criterion—are applied to calculate the partial correlation coefficients among the genes. On the other hand, McDavid *and others* (2019) incorporated the zero inflations into their multivariate Hurdle model using a finite mixture of singular Gaussian distributions. Their model permits inference on statistical independence in zero-inflated, semicontinuous data to learn undirected Markov graphical models. Yet, their model does not take into account where and how the technical noises occur in scRNAseq data.

To overcome such limitations, we propose *single-cell latent graphical model* (scLGM)—a new method that estimates the conditional dependency network among the gene expression levels from scRNAseq data. Unlike Oh *and others* (2018), our model explicitly incorporates the source of technical noise by introducing two types of parameters—cell-specific parameters and gene-specific parameters—in a unified Bayesian hierarchical structure, so that both dropout events and amplification bias can be well explained while estimating the conditional dependency network. Specifically, the cell-specific parameters account for the special characteristics of scRNAseq data while the gene-specific parameters explicate the conditional independence structure between true but unobservable underlying gene expression levels which are assumed to follow the multivariate Gaussian distribution.

The introduction of cell-specific parameters are motivated by two recent technologies, *unique molecular identifiers* (UMIs) and *external RNA controls consortium* (ERCC) spike-in molecules. First, UMIs enable us to accurately identify true *polymerase chain reaction* (PCR) duplicates in high-throughput sequencing experiments (König *and others*, 2010; Kivioja *and others*, 2012; Islam *and others*, 2014; Smith *and others*, 2017). Because the UMIs can distinguish identical copies arising from distinct molecules by attaching a random barcode to each individual fragment during the library preparation step, it establishes a one-to-one mapping between the set of unique UMI barcodes and the set of unique fragments that have been sequenced. Second, each observed cell is augmented with the ERCC spike-in molecules, which are designed to be added to an RNA analysis experiment following a sample isolation step. Adding these external RNA controls enables researchers to follow cell-to-cell variabilities in scRNAseq experiments (Jiang *and others*, 2011; Stegle *and others*, 2015; Bacher and Kendziorski, 2016).

In this article, we propose a new method that infers the conditional dependency network from scRNAseq data. In order to account for the technical noise from the single-cell data, we adopt a similar approach as in Jia *and others* (2017), which uses the aforementioned technologies to analyze differentially expressed genes from single-cell data. Since the cell-specific parameters properly take into account the technical noises which occur in the reverse transcription step and the preamplification step of the current scRNAseq protocols (Hicks *and others*, 2018), our model is able to latently classify each observed zero into either a true zero in the cell or a false zero attributed to the technical noise and incorporates a specific mechanism of zero inflations. In this sense, scLGM is more interpretable than the approach of McDavid *and others* (2019).

The two sets of parameters are estimated individually; the cell-specific parameters are estimated with the Gibbs sampling and the gene-specific parameters are estimated with the variational *expectation–maximization* (EM) algorithm. Simulation studies show that scLGM outperforms other methods in terms of edge selection. A real data application with a mouse scRNAseq data is illustrated and the results are compared to the *Kyoto encyclopedia of genes and genomes* (KEGG) pathway database (Kanehisa *and others*, 2016).

The remainder of this article is organized as follows. We describe scLGM in Section 2. The computational details are provided in Section 2.1, which is comprised of two separate algorithms: one for the estimation of the cell-specific parameters, and the other for the estimation of the gene-specific parameters. It is followed by Section 2.2, which discusses an alternative approach. The results of the simulation studies are summarized in Section 3, and the real data analysis follows in Section 4. We conclude in Section 5.

2. Modeling of scRNAseqdata

In this section, we describe our model which can deal with the technical noise in scRNAseq data. Let $\mathcal{I} = \{1, \ldots, n\}$ be the index set of n cells, and $\mathcal{J} = \{1, \ldots, p\}$ be the index set of p genes. For each pair of $(i, j) \in \mathcal{I} \times \mathcal{J}$ representing the jth gene in the ith cell, x_{ij} denotes the random variable for the unobservable true gene expression level, and y_{ij} denotes the random variable for the observed UMI count. We have auxiliary random variables z_{ij}, which indicate whether the jth gene has been captured via UMI counting during the library preparation step of the ith cell. The statistical variations of our observations are modeled in the following way.

(1) For each $i \in \mathcal{I}$, the p-dimensional random vector $\mathbf{x}_i = \begin{bmatrix} x_{i1} & \cdots & x_{ip} \end{bmatrix}^\top$ representing the unobservable true counts of p genes in the ith cell jointly follows a multivariate log-normal distribution such that

$$\log \mathbf{x}_i = \begin{bmatrix} \log x_{i1} & \cdots & \log x_{ip} \end{bmatrix}^\top \mid \boldsymbol{\mu}, \boldsymbol{\Omega} \sim \text{Normal}_p\left(\boldsymbol{\mu}, \boldsymbol{\Omega}^{-1}\right),$$

where $\boldsymbol{\mu}$ is the p-dimensional mean vector and $\boldsymbol{\Omega}$ is the $p \times p$ inverse covariance matrix. We set $v_{ij} = \log x_{ij}$ and $\mathbf{v}_i = \begin{bmatrix} v_{i1} & \cdots & v_{ip} \end{bmatrix}^\top$ to simplify manipulations of the model.

(2) We have the binary random variable z_{ij}, where $z_{ij} = 1$ indicates that the jth gene has been captured well in the library of the ith cell, whereas $z_{ij} = 0$ indicates that the dropout event has happened. We assume that the probability as to whether a dropout event occurs depends on the unobservable true expression level of the corresponding gene via the probit model (Albert and Chib, 1993) such that

$$z_{ij} \mid v_{ij} \sim \text{Bernoulli}\left(\Phi\left(\psi_i\left(v_{ij}\right)\right)\right),$$

where $\psi_i\left(v_{ij}\right) = \kappa_i + \tau_i v_{ij}$, and $\Phi(\cdot)$ is the *cumulative distribution function* of the standard normal random variable. This reflects the fact that the chance of a gene being captured in the library increases as the true expression level of the gene increases. To facilitate computations, we employ another auxiliary random variables w_{ij} as $z_{ij} = I\left(w_{ij} \geq 0\right)$, where $w_{ij}|v_{ij} \sim \mathrm{N}\left(\psi_i\left(v_{ij}\right), 1\right)$.

$$\Pr\left[z_{ij} = 1 \mid v_{ij}\right] = \Phi\left(\psi_i\left(v_{ij}\right)\right) = \Pr\left[w_{ij} \geq 0 \mid v_{ij}\right].$$

(3) The conditional distribution of the observed count y_{ij} given v_{ij} and z_{ij} is assumed that

$$y_{ij} \mid v_{ij},\ w_{ij} \ \sim \ \begin{cases} \delta_0 & \text{if } w_{ij} < 0; \\ \text{Poisson}\left(\lambda_i\left(v_{ij}\right)\right) & \text{if } w_{ij} \geq 1, \end{cases}$$

where δ_0 is the Dirac measure of y_{ij} concentrated on the singleton $\{0\}$, $\log \lambda_i\left(v_{ij}\right) = \alpha_i + \beta_i v_{ij}$, α_i denotes the capture efficiency of reverse transcription, and β_i reflects the amplification rate. The link function in the distribution reflects the fact that the genes are amplified exponentially, and the difference between β_i and 1 implies the amplification bias.

We define $\{\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\kappa}, \boldsymbol{\tau}\}$ to be the vectors of α_i's, β_i's, κ_i's, and τ_i's, respectively, which we call the cell-specific parameters as opposed to the gene-specific parameters $\{\boldsymbol{\mu}, \boldsymbol{\Omega}\}$.

2.1. *Computation*

Let $\mathbf{V}$, $\mathbf{W}$, and $\mathbf{Y}$ be the matrices with entries v_{ij}, w_{ij}, and y_{ij}, respectively. Our goal is to find the nonzero entries in $\boldsymbol{\Omega}$ as they represent the conditional dependencies among true but un-observable gene expression levels $\mathbf{V}$. For this, the likelihood function of our model is

$$\begin{aligned} & f\left(\mathbf{V}, \mathbf{W}, \mathbf{Y} | \boldsymbol{\mu}, \boldsymbol{\Omega}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\kappa}, \boldsymbol{\tau}\right) \\ & \quad = \prod_{i \in \mathcal{I}} \left[f\left(\mathbf{v}_i | \boldsymbol{\mu}, \boldsymbol{\Omega}\right) \left\{ \prod_{j \in \mathcal{J}} f\left(w_{ij} | v_{ij}, \kappa_i, \tau_i\right) f\left(y_{ij} | v_{ij}, w_{ij}, \alpha_i, \beta_i\right) \right\} \right], \end{aligned}$$

where f is either the *probability density function* (PDF) or the *probability mass function* under the distributions described above. A graphical summary is provided in Figure 1. The following two subsections describe how our method estimates the cell-specific parameters first and use them to estimate the gene-specific parameters.

2.1.1. *Estimation of cell-specific parameters* Let a q-dimensional vector $\mathbf{v}'$ be the log-scaled counts of q predetermined ERCC spike-in molecules, $\mathcal{K} = \{1, \ldots, q\}$ be their index set, and y'_{ik} be the observed UMI count of the kth spike-in in the ith cell. Here, we used the notation $'$ to distinguish fake RNAs from real RNAs. As we mentioned in Section 1, we know the number of ERCC spike-in molecules in each cell. The expression levels of those fake genes serve as a control group against the true but unobservable gene expression levels. We connect these known spike-in counts and corresponding observed UMI counts to estimate the cell-specific parameters $\{\alpha_i, \beta_i, \kappa_i, \tau_i\}$ for each cell $i \in \mathcal{I}$ in the following way.

First, we estimate a pair $\{\alpha_i, \beta_i\}$ for each $i \in \mathcal{I}$ by regressing the log-transformed nonzero UMI counts $\log\left(y'_{ik}\right)$ on the predictor v'_k. This procedure is based on the fact that the conditional expectation of the nonzero UMI counts becomes

$$\log \mathrm{E}\left[y'_{ik} | z'_{ik} = 1; v'_k\right] = \alpha_i + \beta_i v'_k.$$

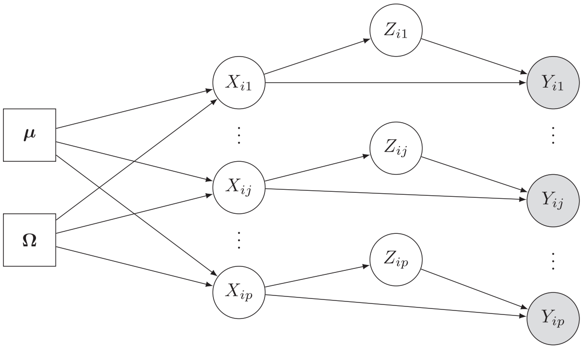

Fig. 1. **Modeling Scheme.** In our model, the unobservable true count of mRNA (X_{ij}) effects both the probability of it being captured in the library (Z_{ij}) and the observed counts (Y_{ij}). Rectangles represent the model parameters, uncolored circles the latent variables, and colored circles the observed variables.

As described in Jia *and others* (2017), the pattern of data missing is not at random, but they also showed in simulation studies that the amounts of biases of these estimators are hardly recognizable, indicating the biases are under control with this estimation procedure.

Next, we use Gibbs sampling method to estimate a pair $\{\kappa_i, \tau_i\}$ for each cell $i \in \mathcal{I}$. Specifically, we use the Gaussian priors for κ_i and τ_i:

$$\kappa_i \sim N(\mu_\kappa, \sigma_\kappa^2) \qquad \text{and} \qquad \tau_i \sim N(\mu_\tau, \sigma_\tau^2).$$

Then, the joint posterior density of the cell-specific parameters, the latent variables, and the observed counts is given by

$$f(\kappa_i, \tau_i, \mathbf{w}_i' | \mathbf{y}_i', \mathbf{v}') \propto \phi(\kappa_i; \mu_\kappa, \sigma_\kappa^2)\phi(\tau_i; \mu_\tau, \sigma_\tau^2) \prod_{k \in \mathcal{K}} f(w_{ik}' | \kappa_i, \tau_i, v_k') f(y_{ik}' | w_{ik}', v_k'),$$

where $f(w_{ik}'|\kappa_i, \tau_i, v_k') = \phi(w_{ik}'; \psi_i(v_k'), 1)$,

$$f\left(y_{ik}'|w_{ik}', v_k'\right) = I\left(w_{ik}' < 0\right) I\left(y_{ik}' = 0\right) + I\left(w_{ik}' \geq 0\right) f_{\text{Poisson}\left(\lambda_i(v_k')\right)}\left(y_{ik}'\right),$$

and $\phi(\cdot; a, b)$ is the PDF of a normal distribution with mean a and variance b. This joint distribution yields Gaussian conditional distributions for all parameters. Their exact forms for Gibbs sampling can be found in Supplementary material available at *Biostatistics* online. We estimate κ_i and τ_i by the averages of the corresponding *Markov chain Monte Carlo* (MCMC) samples.

2.1.2. *Estimation of gene-specific parameters* Recall that our ultimate goal is to estimate the precision matrix $\mathbf{\Omega}$ of the unobservable true gene expression levels x_{ij}'s only with the observed UMI counts y_{ij}'s, while the other latent variables remain unknown. In this subsection, we propose an iterative algorithm for estimating $\mathbf{\Omega}$.

We use the exponential distribution for the prior of the diagonal elements of $\mathbf{\Omega}$ and a Laplace distribution for the prior of its off-diagonal elements to impose sparsity. The location parameter $\boldsymbol{\mu}$ are given noninformative flat priors. The prior distribution can be formulated into

$$f(\boldsymbol{\mu}, \mathbf{\Omega}|\eta, \xi) \propto \left\{\prod_{j \in \mathcal{J}} \eta \exp\left(-\eta \Omega_{jj}\right)\right\} \left\{\prod\prod_{(j_1 < j_2) \in \mathcal{J} \times \mathcal{J}} \xi \exp\left(-\xi \left|\Omega_{j_1 j_2}\right|\right)\right\},$$

where η controls the magnitude of the diagonal elements of $\boldsymbol{\Omega}$, and ξ controls the sparsity of the off-diagonal elements of $\boldsymbol{\Omega}$. The Laplace priors force the off-diagonal elements to shrink towards 0 (Tibshirani, 1996; Park and Casella, 2008).

We propose the *maximum a posteriori* (MAP) estimator for $\boldsymbol{\mu}$ and $\boldsymbol{\Omega}$

$$\begin{aligned}&\log f\left(\boldsymbol{\mu}, \boldsymbol{\Omega}|\mathbf{Y}, \eta, \xi, \widehat{\boldsymbol{\alpha}}, \widehat{\boldsymbol{\beta}}, \widehat{\boldsymbol{\kappa}}, \widehat{\boldsymbol{\tau}}\right)\\ &\quad = C + \log f\left(\boldsymbol{\mu}, \boldsymbol{\Omega}|\eta, \xi\right) + \log \iint f\left(\mathbf{V}, \mathbf{W}, \mathbf{Y}|\boldsymbol{\mu}, \boldsymbol{\Omega}, \widehat{\boldsymbol{\alpha}}, \widehat{\boldsymbol{\beta}}, \widehat{\boldsymbol{\kappa}}, \widehat{\boldsymbol{\tau}}\right) d\mathbf{V} d\mathbf{W},\end{aligned} \tag{2.1}$$

where C is a normalizing constant, and $\left\{\widehat{\boldsymbol{\alpha}}, \widehat{\boldsymbol{\beta}}, \widehat{\boldsymbol{\kappa}}, \widehat{\boldsymbol{\tau}}\right\}$ are the estimate of cell-specific parameters from the previous subsection. Note that, however, this posterior marginal distribution is analytically intractable. Therefore, we augment auxiliary random variables $r_{11}, \ldots, r_{np}$ following Pólya-gamma distributions (Polson *and others*, 2013) as described in Supplementary material available at *Biostatistics* online and use the variational EM approach (Tzikas *and others*, 2008; Blei *and others*, 2017) to find the MAP estimate. Specifically, we maximize the *evidence lower bound* (ELBO), which is a lower bound of the evidence

$$\begin{aligned}&\max_{\boldsymbol{\mu}, \boldsymbol{\Omega}} \log f\left(\boldsymbol{\mu}, \boldsymbol{\Omega} \mid \mathbf{Y}, \eta, \xi, \widehat{\boldsymbol{\alpha}}, \widehat{\boldsymbol{\beta}}, \widehat{\boldsymbol{\kappa}}, \widehat{\boldsymbol{\tau}}\right)\\ &\quad \gtrsim \max_{\boldsymbol{\mu}, \boldsymbol{\Omega}, g} \left\{\log f\left(\boldsymbol{\mu}, \boldsymbol{\Omega}|\eta, \xi\right) + \mathrm{E}_g\left[\log \frac{f\left(\mathbf{V}, \mathbf{W}, \mathbf{R}, \mathbf{Y}|\boldsymbol{\mu}, \boldsymbol{\Omega}, \widehat{\boldsymbol{\alpha}}, \widehat{\boldsymbol{\beta}}, \widehat{\boldsymbol{\kappa}}, \widehat{\boldsymbol{\tau}}\right)}{g\left(\mathbf{V}, \mathbf{W}, \mathbf{R}|\mathbf{Y}\right)}\right]\right\}.\end{aligned} \tag{2.2}$$

Note that g is the variational distribution approximating the posterior distribution, which is defined as

$$g\left(\mathbf{V}, \mathbf{W}, \mathbf{R}|\mathbf{Y}\right) = \prod_{i \in \mathcal{I}} g\left(\mathbf{v}_i\right) \times \prod_{i \in \mathcal{I}} \prod_{j \in \mathcal{J}} g\left(w_{ij}|y_{ij}\right) \times \prod_{i \in \mathcal{I}} \prod_{j \in \mathcal{J}} g\left(r_{ij}\right),$$

where the exact forms of each component are

$$\begin{aligned}g\left(\mathbf{v}_i\right) &\propto \phi\left(\mathbf{v}_i|\boldsymbol{\mu}_{\mathbf{v}_i}, \left(\boldsymbol{\Omega}_{\mathbf{v}_i}\right)^{-1}\right),\\ g\left(w_{ij}|y_{ij}\right) &\propto \phi\left(w_{ij}|\theta_{ij}, 1\right)\left[I\left(w_{ij} < 0\right) I\left(y_{ij} = 0\right) + I\left(w_{ij} \geq 0\right) e^{v_{ij}}\right],\\ g\left(r_{ij}\right) &\propto f_{\mathrm{PG}\left(N, \rho_{ij}\right)}\left(r_{ij}\right),\end{aligned}$$

with $f_{\mathrm{PG}(a, b)}$ being the PDF of the Pólya-gamma distribution with parameters a and b (Polson *and others*, 2013).

While the variational parameters are chosen by maximizing the ELBO given $\boldsymbol{\mu}$ and $\boldsymbol{\Omega}$, we estimate the gene-specific parameters $\boldsymbol{\mu}$ and $\boldsymbol{\Omega}$ with g fixed, for which the solution is given by $\boldsymbol{\mu} = \frac{1}{n}\sum_{i \in \mathcal{I}} \mathrm{E}_g\left[\mathbf{v}_i|\mathbf{y}_i\right]$ and

$$\underset{\boldsymbol{\Omega}}{\operatorname{argmax}} \left\{\sum_j \exp\left(-\eta \Omega_{jj}\right) + \sum_{j_1 < j_2}\sum \exp\left(-\xi\left|\Omega_{j_1 j_2}\right|\right) + \frac{n}{2}\log\left(|\boldsymbol{\Omega}|\right) - \frac{n}{2}\operatorname{tr}\left(\boldsymbol{\Omega}\mathbf{S}^{+}\right)\right\}, \tag{2.3}$$

where

$$\mathbf{S}^{+} = \frac{1}{n}\sum_{i \in \mathcal{I}}\left(\mathrm{E}_g\left[\mathbf{v}_i|\mathbf{y}_i\right] - \boldsymbol{\mu}\right)\left(\mathrm{E}_g\left[\mathbf{v}_i|\mathbf{y}_i\right] - \boldsymbol{\mu}\right)^{\top} + \frac{1}{n}\sum_{i \in \mathcal{I}} \mathrm{Var}_g\left[\mathbf{v}_i|\mathbf{y}_i\right]. \tag{2.4}$$

Note that (2.3) can be seen as a graphical lasso problem with $\mathbf{S}^+$ being the sample covariance matrix, and can be solved by the GLASSO algorithm (Friedman *and others*, 2008).

2.2. *Alternative approach*

The algorithm proposed above is an iteration of two stages until convergence: (i) approximating the posterior marginal distribution of the gene-specific parameters via approximating the conditional distribution of the latent variables by a variational distribution and then (ii) estimating the gene-specific parameters using the approximated marginal distribution. Note that these two steps can be viewed in a different way. The first step can be seen as the imputation of the sample covariance matrix ($\mathbf{S}^+$) of the underlying true gene expression levels and the second step can be seen as finding the sparse precision matrix based on the imputed sample covariance matrix via GLASSO.

From this perspective, we introduce an alternative approach by replacing the second part of the algorithm, GLASSO, with another inverse covariance matrix estimation method, the *constrained* l_1-*minimization for inverse matrix estimation* (CLIME) by Cai *and others* (2011). In particular, we find the optimal solution of

$$\min \|\boldsymbol{\Omega}\|_1 \quad \text{subject to} \quad \left|\mathbf{S}^+\boldsymbol{\Omega} - \mathbf{I}\right|_\infty \leq \lambda_n, \quad \boldsymbol{\Omega} \in \mathbb{R}^{p\times p},$$

where $\mathbf{S}^+$ is given by the first step as described in (2.4). This new algorithm does not find the MAP estimator from the posterior density of our model, but gives scalable computations.

3. Simulation studies

In this section, we present the comparison of the scLGM method against other methods – GLASSO and HurdleNormal. A total of nine different simulation scenarios are considered by combining three different graph structures and three different n/p ratios. We fix $n = 100$ and consider $p \in \{50, 100, 300\}$. For the other two methods, we applied a logarithmic transformation to the UMI counts with a very small number added due to zero counts.

Three different graph structures are described with three corresponding graphs in Figure 2 for the $p = 50$ cases. Specifically, the subfigure on the left represents our first graph structure, in which p vertices are divided into 10 equally sized subsets. Each subset forms a hub structure such that there is one vertex linked to all the rest nine vertices while they are not linked among themselves. The second graph structure is the random sparse graph such that the edge densities are around 0.5% for all three p's. Lastly, we consider *moralized direct acyclic graphs* (MDAGs). We generated a *direct acyclic graph* (DAG) for each scenario with p directional edges out of $p\,(p-1)$ possible combinations. Then, the graphs are moralized by adding undirected edges among parent nodes if they share the same child node, and then converting all the edges to undirected ones.

On these three graph structures, we used existing algorithms to simulate the underlying true unobservable UMI counts of each simulation scenario. For the hub structured graphs and the random sparse graphs, we used "huge" package (Zhao *and others*, 2012) to explicitly generate the inverse covariance matrix $\boldsymbol{\Omega}$, and then simulated the log-valued gene expression levels v_{ij} from the p-dimensional multivariate Gaussian distribution with the mean vector $\boldsymbol{\mu}$ and the covariance matrix $\boldsymbol{\Omega}^{-1}$. In the moralized DAG cases, we used "spacejam" package (Voorman *and others*, 2013) to generate DAGs and to simulate data with p-dimensional multivariate normal distributions. Note that, in all scenarios, each element in $\boldsymbol{\mu}$ is generated from the normal distribution with mean 4 and variance 1, which resulted in around 90% of nonzero data entries in the observed counts y_{ij}.

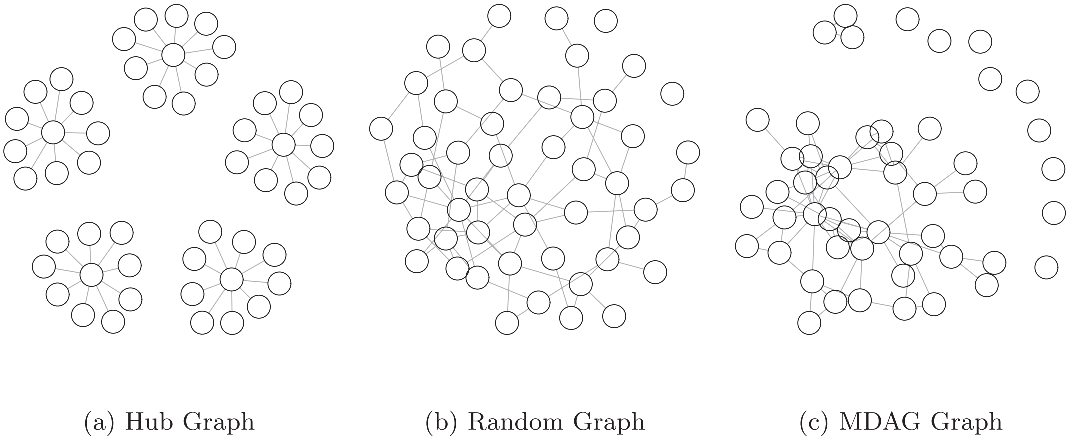

Fig. 2. Shapes of three simulated graphs when $p = 50$.

Given the sample observations of true log-expression levels v_{ij} from the multivariate normal distribution with the gene-specific parameters $\boldsymbol{\mu}$ and $\boldsymbol{\Omega}$, we generate the latent variables z_{ij} from the Bernoulli distribution and y_{ij} from the Poisson distribution as described in the modeling framework. In these simulated library preparation processes and the simulated amplification processes, the four cell-specific parameters for each of the 100 observations are generated from the multivariate Gaussian distributions given below, where the mean vectors and the covariance matrices have been estimated from the cell class "CA1Pyr2" of the real data analyzed in Section 4.

$$\begin{bmatrix} \alpha_i \\ \beta_i \end{bmatrix} \sim^{\text{iid}} \text{Normal}_2\left(\begin{bmatrix} -1.136 \\ 0.900 \end{bmatrix}, \begin{bmatrix} 0.077 & -0.006 \\ -0.006 & 0.002 \end{bmatrix}\right);$$

$$\begin{bmatrix} \kappa_i \\ \tau_i \end{bmatrix} \sim^{\text{iid}} \text{Normal}_2\left(\begin{bmatrix} -0.520 \\ 0.869 \end{bmatrix}, \begin{bmatrix} 0.058 & 0.001 \\ 0.001 & 0.016 \end{bmatrix}\right).$$

The simulation results are summarized in Figure 3 and Table 1. First, Figure 3 consists of nine subfigures containing *receiver operating characteristic* (ROC) curves as to the selection of edges. They consistently exemplify that scLGM outperforms other methods in terms of the area under the ROC curves. Especially in the first two graph structures, scLGM with the GLASSO M-step is the best performing method with respect to MCC. For the MDAG structures, scLGM with the Clime M-step is the best performing, while the performance of scLGM with the GLASSO M-step is comparable. The GLASSO method also works very well on MDAG graphs, while in all cases HurdleNormal is underperforming with respect to the area under the ROC curves.

In addition, we place bullets on the ROC curves indicating the selected graphs. Table 1 show the summary of their qualities. Overall, scLGMs tend to select higher TPRs while keeping FPRs moderate. While our methods show better Matthews correlation coefficients (MCC) for the hub graphs and the random graphs, GLASSO gives slightly better MCC on the moralized DAG graphs. In all scenarios, our methods tend to detect more edges than GLASSO. HurdleNormal has lower MCC in general and shows very inconsistent numbers of detected edges between low p situations and high p situations compared to other methods.

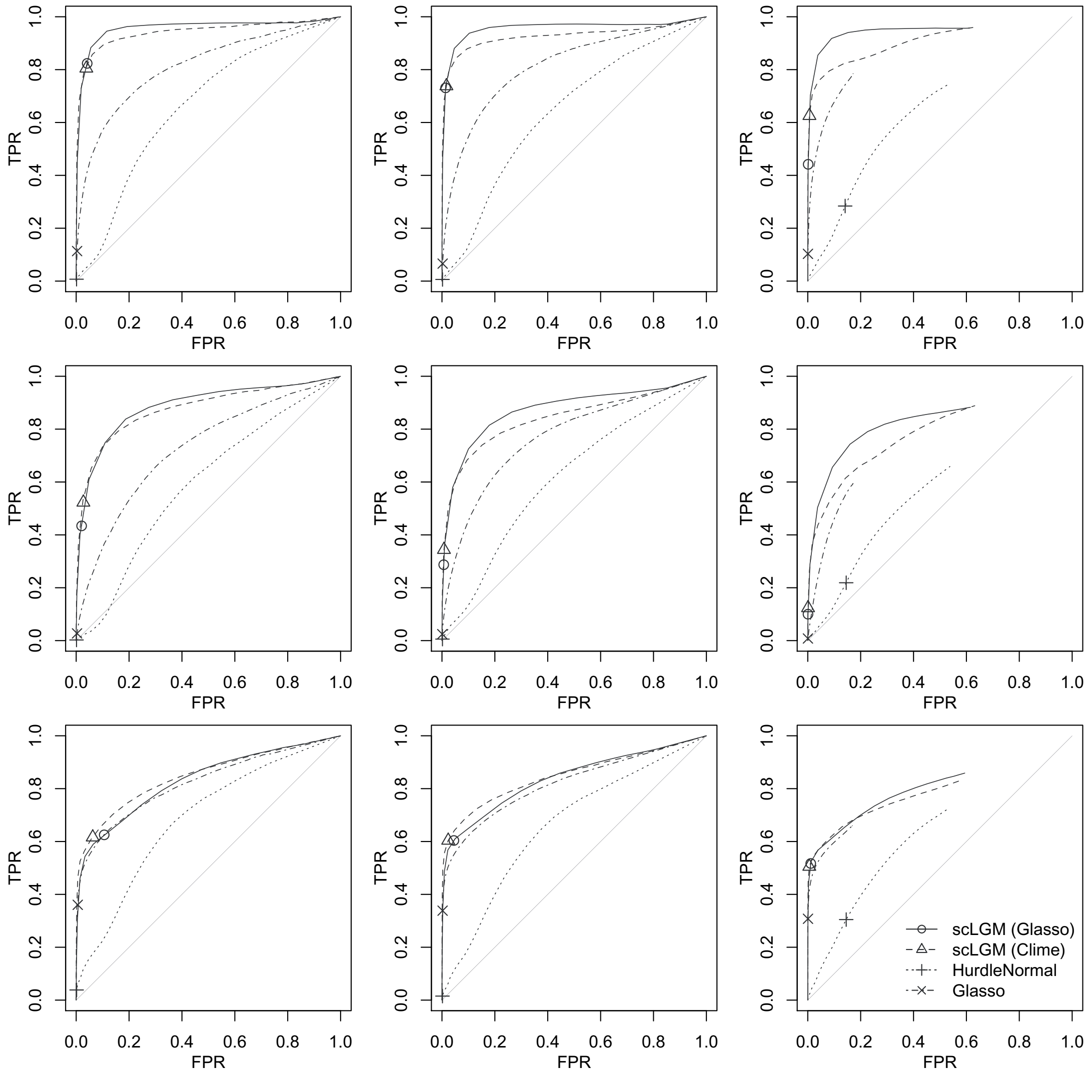

Fig. 3. Simulation results: each subfigure represents a different simulation scenario such that each column represents a corresponding number of simulated genes (50 genes on the left; 100 genes in the middle; 300 genes on the right), while each row represents a corresponding graph structure (hub graph at the top; random graph in the middle; moralized DAG at the bottom). Each bullets on ROC curves represents selected graph with respect to corresponding criteria.

4. Real data analysis

We analyze the data set from Zeisel *and others* (2015), in which large scale scRNAseq data were collected from mouse brain single cells located in the primary somatosensory cortex and the hippocampal CA1 region. In Zeisel *and others* (2015), the authors clustered 3005 cells into 9 level-1 classes, then further decomposed them into 47 level-2 (Table S1 of Supplementary material available at *Biostatistics* online). We focused on the biggest level-1 class "pyramidal CA1" and its biggest level-2 "CA1Pyr2," which have 917 and 447 cells, respectively. We selected 200 genes out of 19 972 genes, where the first 100 genes

Table 1. *Simulation results: we compare scLGM with Glasso M-step, scLGM with Clime M-step, HurdleNormal, and Glasso. Number of detected edges, false positive rates, true positive rates, and Matthews correlation coefficients are reported. Standard errors are in the parentheses.*

Graph	Genes	Method	Edge	FPR	TPR	Matthews
Hub	$p = 50$	scLGM (Glasso)	86.16 (26.159)	0.042 (0.019)	0.824 (0.114)	0.589 (0.057)
		scLGM (Clime)	81.43 (21.627)	0.038 (0.017)	0.805 (0.076)	0.590 (0.064)
		HurdleNormal	1.94 (1.420)	0.001 (0.001)	0.007 (0.012)	0.024 (0.053)
		Glasso	9.18 (3.812)	0.003 (0.002)	0.114 (0.048)	0.240 (0.077)
	$p = 100$	scLGM (Glasso)	131.89 (22.943)	0.014 (0.003)	0.731 (0.086)	0.597 (0.039)
		scLGM (Clime)	148.00 (45.931)	0.017 (0.008)	0.738 (0.074)	0.581 (0.057)
		HurdleNormal	10.58 (3.888)	0.002 (0.001)	0.006 (0.007)	0.012 (0.021)
		Glasso	10.48 (7.711)	0.003 (0.001)	0.066 (0.032)	0.138 (0.056)
	$p = 300$	scLGM (Glasso)	190.67 (68.192)	0.002 (0.001)	0.442 (0.071)	0.529 (0.036)
		scLGM (Clime)	449.78 (15.752)	0.006 (0.000)	0.625 (0.030)	0.481 (0.022)
		HurdleNormal	6380.96 (181.306)	0.141 (0.004)	0.284 (0.024)	0.032 (0.005)
		Glasso	60.41 (15.078)	0.001 (0.000)	0.103 (0.025)	0.216 (0.036)
Random	$p = 50$	scLGM (Glasso)	61.11 (26.922)	0.021 (0.015)	0.434 (0.126)	0.494 (0.060)
		scLGM (Clime)	76.32 (22.362)	0.027 (0.013)	0.523 (0.107)	0.530 (0.059)
		HurdleNormal	1.62 (1.324)	0.001 (0.001)	0.002 (0.005)	0.006 (0.029)
		Glasso	6.01 (3.401)	0.003 (0.002)	0.027 (0.019)	0.087 (0.060)
	$p = 100$	scLGM (Glasso)	76.59 (35.016)	0.007 (0.004)	0.287 (0.108)	0.398 (0.066)
		scLGM (Clime)	89.96 (22.118)	0.008 (0.003)	0.344 (0.053)	0.443 (0.038)
		HurdleNormal	9.68 (2.624)	0.002 (0.000)	0.006 (0.006)	0.014 (0.022)
		Glasso	9.15 (4.293)	0.001 (0.001)	0.024 (0.013)	0.095 (0.043)
	$p = 300$	scLGM (Glasso)	83.63 (18.427)	0.001 (0.000)	0.100 (0.021)	0.226 (0.030)
		scLGM (Clime)	101.63 (36.487)	0.001 (0.001)	0.124 (0.025)	0.255 (0.026)
		HurdleNormal	6550.88 (362.441)	0.145 (0.008)	0.219 (0.022)	0.020 (0.005)
		Glasso	32.54 (13.204)	0.001 (0.000)	0.008 (0.005)	0.027 (0.018)
MDAG	$p = 50$	scLGM (Glasso)	171.92 (30.066)	0.106 (0.025)	0.625 (0.028)	0.376 (0.041)
		scLGM (Clime)	121.45 (21.698)	0.063 (0.017)	0.616 (0.040)	0.465 (0.038)
		HurdleNormal	4.98 (1.803)	0.002 (0.001)	0.038 (0.014)	0.145 (0.046)
		Glasso	35.96 (5.059)	0.006 (0.002)	0.360 (0.040)	0.522 (0.034)
	$p = 100$	scLGM (Glasso)	310.64 (51.376)	0.045 (0.010)	0.604 (0.021)	0.408 (0.036)
		scLGM (Clime)	207.59 (19.071)	0.024 (0.004)	0.604 (0.020)	0.509 (0.024)
		HurdleNormal	11.11 (3.632)	0.002 (0.001)	0.015 (0.009)	0.050 (0.034)
		Glasso	64.52 (7.589)	0.002 (0.001)	0.338 (0.030)	0.519 (0.024)
	$p = 300$	scLGM (Glasso)	760.24 (134.529)	0.011 (0.003)	0.517 (0.022)	0.420 (0.034)
		scLGM (Clime)	514.15 (15.322)	0.006 (0.000)	0.505 (0.011)	0.497 (0.011)
		HurdleNormal	6600.76 (338.319)	0.145 (0.007)	0.305 (0.022)	0.048 (0.005)
		Glasso	194.21 (18.630)	0.001 (0.000)	0.308 (0.019)	0.495 (0.012)

Table 2. *Real data analysis result: we compare scLGM with Glasso M-step, scLGM with Clime M-step, HurdleNormal, and Glasso over the real mouse single-cell data. Among all possible pairs of* 200 *selected genes,* 42.8 *and* 53.8 *edges for each of clusters on average are verified by the KEGG database which is set to be our gold standard for comparison. Each of other possible pairs of genes is either not connected or connected but unknown yet. The number of detected edges for each group is reported.*

Method	**CA1Pyr2**		**Pyramidal CA1**	
	Edge	**Overlap**	**Edge**	**Overlap**
scLGM (Glasso)	3807.0	17.0	4756.4	18.4
scLGM (Clime)	3933.2	18.0	7201.4	26.4
HurdleNormal	352.0	8.0	295.2	6.0
Glasso	4222.0	2.0	4709.2	5.6

Table 3. *Simulation results based on real data analysis: we compare scLGM with Glasso M-step, scLGM with Clime M-step, HurdleNormal, and Glasso on the simulated data using the parameters estimated from the real mouse single-cell data analysis. False positive rates, true positive rates, and Matthews correlation coefficients are reported. Standard errors are in the parentheses.*

Method	**Edge**	**FPR**	**TPR**	**Matthews**
scLGM (Glasso)	1838.13 (63.372)	0.062 (0.003)	0.254 (0.007)	0.243 (0.007)
scLGM (Clime)	1494.30 (272.314)	0.041 (0.011)	0.256 (0.031)	0.300 (0.009)
HurdleNormal	485.73 (27.850)	0.019 (0.001)	0.054 (0.003)	0.084 (0.006)
Glasso	314.55 (37.404)	0.010 (0.001)	0.049 (0.006)	0.114 (0.009)

are the most highly expressed genes, while the other 100 genes are randomly chosen from the rest of the genes with at least 30% of nonzero UMI ratio. The analysis is repeated with five different randomly selected genes. This resulted in 12.3% zeros on average over the five repeated analyses for “CA1Pyr2” cell subclass, and 13.8% zeros for “pyramidal CA1” class.

We compare the edges estimated by scLGM and other methods to the edges retrieved from the KEGG pathway database (Kanehisa *and others*, 2016). A total of 42.8 and 53.8 edges for the 200 genes were collected from mouse-specific KEGG pathways for classes “pyramidal CA1”and “CA1Pyr2,” respectively. Other methods considered are GLASSO and HurdleNormal as in Section 3.

The results are summarized in Table 2 and in Figure 4. Table 2 contains the number of detected edges, their overlaps with our gold standards, and the additionally discovered edges which does not exist in the KEGG database. As in simulations, our methods detected the most edges overlapping with the gold standards. The extra discoveries of all methods beyond the KEGG edges suggest that there can potentially be many unknown relationships between the genes of interests. Figure 4 contains the number of overlapping selected edges across all methods, and the number of overlapping edges that are also included in KEGG. We can see that our methods has the largest overlaps with the two competitors, while GLASSO and HurdleNormal have none or less overlaps. The edges discovered by GLASSO is nearly the subset of scLGM with GLASSO M-step.

To further investigate the performance of scLGM on the real data, we conducted another simulation study which is based on the results of our real data analysis. By using the cell-specific and gene-specific

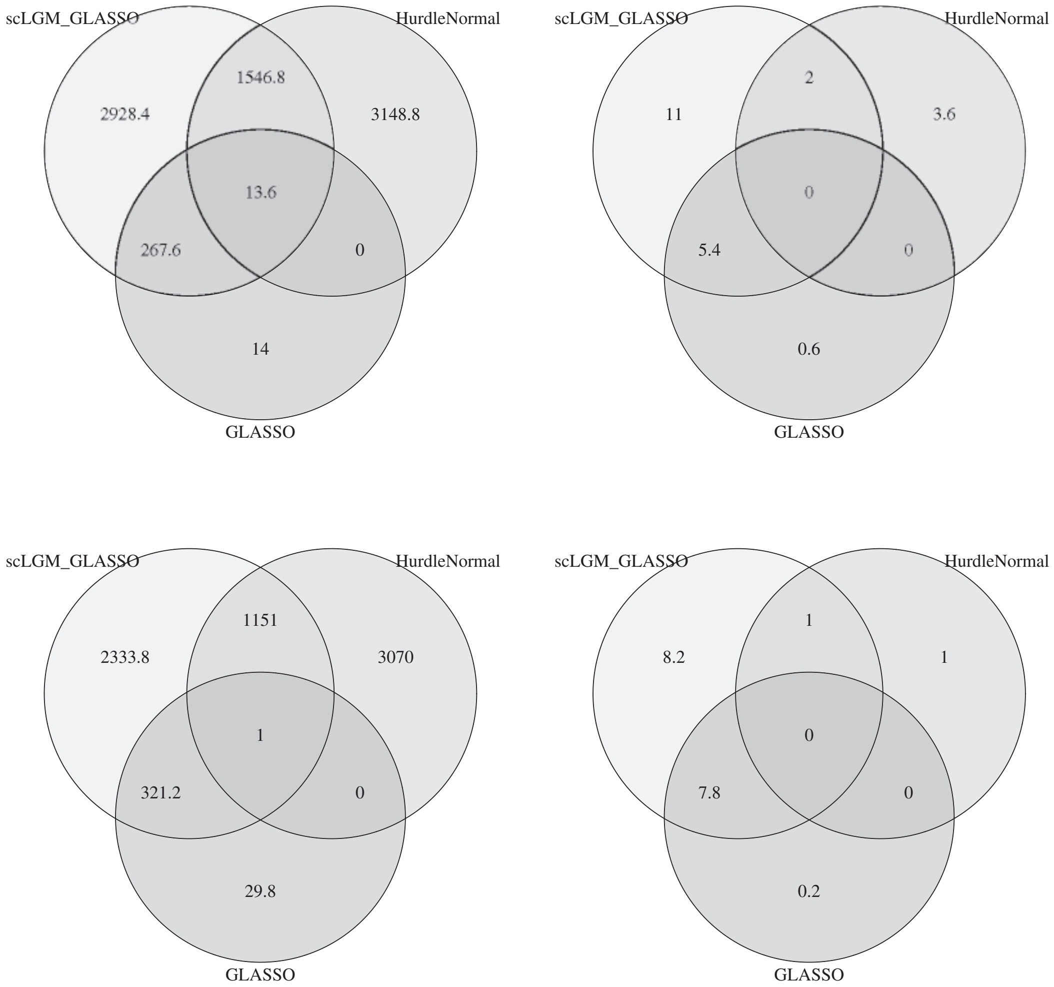

Fig. 4. Comparison of detected edges among methods: each row represents the overlaps from two classes ("pyramidal CA1" on top and "CA1Pyr2" on bottom). The left column represents total number of overlaps among all detected edges, while the right column represents the number of overlaps among ones existing in the KEGG database.

parameters from the real data analysis estimated with GLASSO, we generated 100 synthetic data sets with the same sample size $n = 447$. We conducted the same analysis as in Section 3, and the results are shown in Figure 5 and Table 3. Our two methods still outperform other methods in terms of the area under the ROC curves and achieve the best TPRs with moderate FPRs resulting in the best MCC.

5. Discussion

In this article, we proposed a novel method for graphical modeling using high-dimensional scRNAseq data. Unlike the existing methods that do not account for unique features of scRNAseq data, the proposed approach uses the state-of-art technique of Jia *and others* (2017) to properly explain two source of excessive

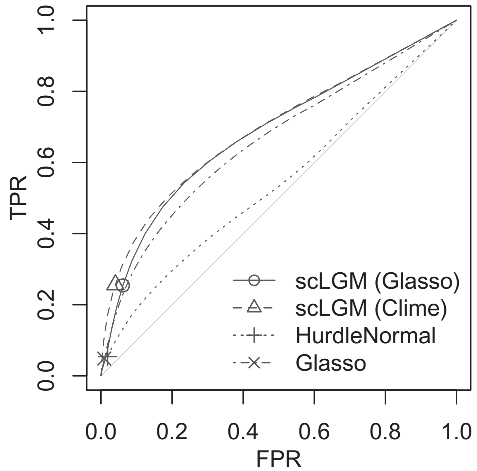

Fig. 5. ROC curves from the simulation results based on real data: the scLGM outperforms other methods in terms of the area under the ROC curves, while compared methods only suggest limited number of edges.

zeros that are present in scRNAseq data and enables estimation of the conditional dependency structure of the unobserved underlying true gene expression values. The simulation results demonstrate the superiority of scLGM, and the real data application shows our method can be useful in practice.

One drawback of scLGM is its scalability. As the algorithms are iterative, our method runs several times slower than the other methods considered in the paper. In our data analysis for "CA1Pyr2" cell class, it took 1 min per tuning parameter on average with $(n,p,q) = (433,200,64)$. We expect that our method will be scalable up to a few thousand genes, depending on the sample size and the sparsity of edges. Developing a more scalable method certainly deserves more attention.

As scLGM enables estimating a gene dependency network, it can be combined with other methods to analyze scRNAseq data where incorporation of graphical information can be valuable. For example, many of supervised and unsupervised learning methods when using scRNAseq data as predictors can be improved by incorporating graph knowledge among the predictors, via sequential or joint estimation.

6. Software

Software in the form of R code is available on https://github.com/jihwan05/scLGM/.

SupplementaryMaterial

Supplementary material is available online at http://biostatistics.oxfordjournals.org.

Acknowledgments

Conflict of Interest: None declared.

Funding

National Institutes of Health (NIH) grants (P30CA016520 and RF1AG063481). The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health.

References

Albert, J. H. and Chib, S. (1993). Bayesian analysis of binary and polychotomous response data. *Journal of the American Statistical Association* **88**, 669–679.

Bacher, R. and Kendziorski, C. (2016). Design and computational analysis of single-cell RNA-sequencing experiments. *Genome Biology* **17**, 63.

Banerjee, O., El Ghaoui, L. and d'Aspremont, A. (2008). Model selection through sparse maximum likelihood estimation for multivariate gaussian or binary data. *The Journal of Machine Learning Research* **9**, 485–516.

Blei, D. M., Kucukelbir, A. and McAuliffe, J. D. (2017). Variational inference: a review for statisticians. *Journal of the American Statistical Association* **112**, 859–877.

Buettner, F., Natarajan, K. N., Casale, F. P., Proserpio, V., Scialdone, A., Theis, F. J., Teichmann, S. A., Marioni, J. C. and Stegle, O. (2015). Computational analysis of cell-to-cell heterogeneity in single-cell RNA-sequencing data reveals hidden subpopulations of cells. *Nature Biotechnology* **33**, 155.

Cai, T., Liu, W. and Luo, X. (2011). A constrained l_1 minimization approach to sparse precision matrix estimation. *Journal of the American Statistical Association* **106**, 594–607.

Chun, H., Zhang, X. and Zhao, H. (2015). Gene regulation network inference with joint sparse gaussian graphical models. *Journal of Computational and Graphical Statistics* **24**, 954–974.

Editorial. (2014). Method of the year 2013. *Nature Methods* **11**.

Elowitz, M. B., Levine, A. J., Siggia, E. D. and Swain, P. S. (2002). Stochastic gene expression in a single cell. *Science* **297**, 1183–1186.

Fan, J., Feng, Y. and Wu, Y. (2009). Network exploration via the adaptive LASSO and SCAD penalties. *The Annals of Applied Statistics* **3**, 521.

Friedman, J., Hastie, T. and Tibshirani, R. (2008). Sparse inverse covariance estimation with the graphical lasso. *Biostatistics* **9**, 432–441.

Fukumizu, K., Gretton, A., Sun, X. and Schölkopf, B. (2007). Kernel measures of conditional dependence. In: *Twenty-First Annual Conference on Neural Information Processing Systems (NIPS 2007).* Curran, Volume 20. pp. 489–496.

Gong, W., Kwak, I.-Y., Pota, P., Koyano-Nakagawa, N. and Garry, D. J. (2018). Drimpute: imputing dropout events in single cell RNA sequencing data. *BMC Bioinformatics* **19**, 220.

Harari, A., Vallelian, F., Meylan, P. R. and Pantaleo, G. (2005). Functional heterogeneity of memory CD4 T cell responses in different conditions of antigen exposure and persistence. *The Journal of Immunology* **174**, 1037–1045.

Harris, N. and Drton, M. (2013). PC algorithm for nonparanormal graphical models. *The Journal of Machine Learning Research* **14**, 3365–3383.

Hicks, S. C., Townes, F. W., Teng, M. and Irizarry, R. A. (2018). Missing data and technical variability in single-cell RNA-sequencing experiments. *Biostatistics* **19**, 562–578.

Islam, S., Zeisel, A., Joost, S., La Manno, G., Zajac, P., Kasper, M., Lönnerberg, P. and Linnarsson, S. (2014). Quantitative single-cell RNA-seq with unique molecular identifiers. *Nature Methods* **11**, 163.

Jia, C., Hu, Y, Kelly, D., Kim, J., Li, M. and Zhang, N. R. (2017). Accounting for technical noise in differential expression analysis of single-cell RNA sequencing data. *Nucleic Acids Research* **45**, 10978–10988.

Jiang, L., Schlesinger, F., Davis, C. A., Zhang, Y., Li, R., Salit, M., Gingeras, T. R. and Oliver, B. (2011). Synthetic spike-in standards for RNA-seq experiments. *Genome Research* **21**, 1543–1551.

Kanehisa, M., Furumichi, M., Tanabe, M., Sato, Y. and Morishima, K. (2016). Kegg: new perspectives on genomes, pathways, diseases and drugs. *Nucleic Acids Research* **45**, D353–D361.

KIVIOJA, T., VÄHÄRAUTIO, A., KARLSSON, K., BONKE, M., ENGE, M., LINNARSSON, S. AND TAIPALE, J. (2012). Counting absolute numbers of molecules using unique molecular identifiers. *Nature Methods* **9**, 72.

KÖNIG, J., ZARNACK, K., ROT, G., CURK, T., Kayikci, M., Zupan, B., Turner, D. J., Luscombe, N. M. and Ule, J. (2010). iCLIP reveals the function of hnRNP particles in splicing at individual nucleotide resolution. *Nature Structural & Molecular Biology* **17**, 909.

LAM, C. AND FAN, J. (2009). Sparsistency and rates of convergence in large covariance matrix estimation. *Annals of Statistics* **37**, 4254.

LI, B., CHUN, H. AND ZHAO, H. (2012). Sparse estimation of conditional graphical models with application to gene networks. *Journal of the American Statistical Association* **107**, 152–167.

LI, B., CHUN, H. AND ZHAO, H. (2014). On an additive semi-graphoid model for statistical networks with application to pathway analysis. *Journal of the American Statistical Association* **109**, 1188–1204.

LIU, H., HAN, F., YUAN, M., LAFFERTY, J., WASSERMAN, L. *and others*. (2012). High-dimensional semiparametric Gaussian copula graphical models. *The Annals of Statistics* **40**, 2293–2326.

LIU, H., LAFFERTY, J. AND WASSERMAN, L. (2009). The nonparanormal: semiparametric estimation of high dimensional undirected graphs. *The Journal of Machine Learning Research* **10**, 2295–2328.

LIU, H., XU, M., GU, H., GUPTA, A., LAFFERTY, J. AND WASSERMAN, L. (2011). Forest density estimation. *The Journal of Machine Learning Research* **12**, 907–951.

MCDAVID, A., GOTTARDO, R., SIMON, N. AND DRTON, M. (2019). Graphical models for zero-inflated single cell gene expression. *The Annals of Applied Statistics* **13**, 848–873.

MEINSHAUSEN, N. AND BÜHLMANN, P. (2006). High-dimensional graphs and variable selection with the lasso. *The Annals of Statistics*, **34**, 1436–1462.

OH, J., ZHENG, F., DOERGE, R. W. AND CHUN, H. (2018). Kernel partial correlation: a novel approach to capturing conditional independence in graphical models for noisy data. *Journal of Applied Statistics*, **45**, 2677–2696.

PARK, T. AND CASELLA, G. (2008). The Bayesian lasso. *Journal of the American Statistical Association* **103**, 681–686.

PENG, J., WANG, P., ZHOU, N. AND ZHU, J. (2009). Partial correlation estimation by joint sparse regression models. *Journal of the American Statistical Association* **104**, 735–746.

POLSON, N. G., SCOTT, J. G. AND WINDLE, J. (2013). Bayesian inference for logistic models using pólya–gamma latent variables. *Journal of the American statistical Association* **108**, 1339–1349.

SMITH, T. S., HEGER, A. AND SUDBERY, I. (2017). Umi-tools: modelling sequencing errors in unique molecular identifiers to improve quantification accuracy. *Genome Research* **27**, 491–499.

STEGLE, O., TEICHMANN, S. A. AND MARIONI, J. C. (2015). Computational and analytical challenges in single-cell transcriptomics. *Nature Reviews Genetics* **16**, 133.

SZÉKELY, G. J. AND RIZZO, M. L. (2014). Partial distance correlation with methods for dissimilarities. *The Annals of Statistics* **42**, 2382–2412.

TANG, F., BARBACIORU, C., WANG, Y., NORDMAN, E., LEE, C., XU, N., WANG, X., BODEAU, J., TUCH, B. B, SIDDIQUI, A. *and others*. (2009). mRNA-Seq whole-transcriptome analysis of a single cell. *Nature Methods* **6**, 377.

TIBSHIRANI, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society Series B (Methodological)* **58**, 267–288.

TZIKAS, D. G., LIKAS, A. C. AND GALATSANOS, N. P. (2008). The variational approximation for Bayesian inference. *IEEE Signal Processing Magazine* **25**, 131–146.

VOORMAN, A., SHOJAIE, A. AND WITTEN, D. (2013). Graph estimation with joint additive models. *Biometrika* **101**, 85–101.

Wang, X., Pan, W, Hu, W., Tian, Y. and Zhang, H. (2015). Conditional distance correlation. *Journal of the American Statistical Association* **110**, 1726–1734.

Yuan, M. and Lin, Y. (2007). Model selection and estimation in the Gaussian graphical model. *Biometrika* **94**, 19–35.

Zeisel, A., Muñoz-manchado, A. B., Codeluppi, S., Lönnerberg, P., La Manno, G., Juréus, A., Marques, S., Munguba, H., He, L., Betsholtz, C. *and others*. (2015). Cell types in the mouse cortex and hippocampus revealed by single-cell RNA-seq. *Science* **347**, 1138–1142.

Zhao, T., Liu, H., Roeder, K., Lafferty, J. and Wasserman, L. (2012). The huge package for high-dimensional undirected graph estimation in R. *Journal of Machine Learning Research* **13**, 1059–1062.

[*Received March 30, 2020; revised March 16, 2021; accepted for publication March 17, 2021*]