5-2020

Novel Bayesian methodology for the analysis of single-cell RNA sequencing data.

Michael Sekula
University of Louisville

NOVEL BAYESIAN METHODOLOGY FOR THE ANALYSIS OF SINGLE-CELL RNA SEQUENCING DATA

By

Michael Sekula
B.S., Saginaw Valley State University, 2010
M.S., University of Louisville, 2015

A Dissertation
Submitted to the Faculty of the
School of Public Health and Information Sciences
of the University of Louisville
in Partial Fulfillment of the Requirements
for the Degree of

Doctor of Philosophy
in Biostatistics

Department of Bioinformatics and Biostatistics
University of Louisville
Louisville, Kentucky

May 2020

NOVEL BAYESIAN METHODOLOGY FOR THE ANALYSIS OF SINGLE-CELL RNA SEQUENCING DATA

By

Michael Sekula
B.S., Saginaw Valley State University, 2010
M.S., University of Louisville, 2015

A Dissertation Approved on

April 10, 2020

by the following Dissertation Committee:

Jeremy Gaskins, Ph.D., Dissertation Director

Susmita Datta, Ph.D., Dissertation Co-director

K.B. Kulasekera, Ph.D.

Maiying Kong, Ph.D.

Ryan Gill, Ph.D.

ACKNOWLEDGMENTS

I would like to thank my advisors, Dr. Susmita Datta and Dr. Jeremy Gaskins, for their patience, insight, and unwavering support over these past five years. I sincerely appreciate all of their invaluable guidance, advice, and inspiration during this endeavor.

I would also like to thank my dissertation committee members Dr. K.B. Kulasekera, Dr. Maiying Kong, and Dr. Ryan Gill for their time, flexibility, and support throughout this process.

Finally, I thank the students and faculty of the Department of Biostatistics and Bioinformatics for providing support and encouragement during my graduate studies.

ABSTRACT

NOVEL BAYESIAN METHODOLOGY FOR THE ANALYSIS OF SINGLE-CELL RNA SEQUENCING DATA

Michael Sekula

April 10, 2020

With single-cell RNA sequencing (scRNA-seq) technology, researchers are able to gain a better understanding of health and disease through the analysis of gene expression data at the cellular-level; however, scRNA-seq data tend to have high proportions of zero values, increased cell-to-cell variability, and overdispersion due to abnormally large expression counts, which create new statistical problems that need to be addressed. This dissertation includes three research projects that propose Bayesian methodology suitable for scRNA-seq analysis. In the first project, a hurdle model for identifying differentially expressed genes across cell types in scRNA-seq data is presented. This model incorporates a correlated random effects structure based on an initial clustering of cells to capture the cell-to-cell variability within treatment groups but can easily be adapted to an independent random effect structure if needed. A sparse Bayesian factor model is introduced in the second project to uncover network structures associated with genes in scRNA-seq data. Latent factors impact the gene expression values for each cell and provide flexibility to account for the common features of scRNA-seq. The third project expands upon this latent factor model to allow for the comparison of networks across different treatment groups.

TABLE OF CONTENTS

PAGE

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

For over a decade, RNA sequencing technologies have been instrumental in transforming our knowledge and understanding of transcriptomes. The analysis of data generated by these technologies has led to a plethora of novel biological findings ranging from the discovery of new transcripts to the identification of genes associated with specific diseases (Wang et al., 2009; Liu et al., 2015). Currently, RNA sequencing experiments fall into one of two categories: bulk RNA sequencing (RNA-seq) or single-cell RNA sequencing (scRNA-seq). Traditional bulk RNA-seq experiments examine transcript abundance measurements that have been averaged over populations of thousands, or even millions, of cells. In contrast, the more recent scRNA-seq experiments examine gene expressions from individual cells. While bulk RNA-seq studies have a longer history in the literature, scRNA-seq studies are rapidly gaining attention among researchers today.

What makes the future of scRNA-seq so promising is the unprecedented opportunity to thoroughly investigate cellular functionality at the level of a single cell. Diverse gene expression patterns in cell populations that have previously seemed homogeneous are becoming exposed, allowing investigators to uncover solutions to unanswered questions across various fields of biology (Shapiro et al., 2013; Fan et al., 2015; Kanter and Kalisky, 2015). Nevertheless, the information collected from single-cell sequencing does present new computational and statistical challenges. Several intrinsic features of scRNA-seq data are not observed in bulk RNA-seq data (Figure

1.1), and so many of the well-established methods used in bulk RNA-seq studies are not suitable for scRNA-seq studies. Thus, in order to take full advantage of scRNA-seq technologies, new statistical techniques need to be developed (Stegle et al., 2015; Bacher and Kendziorski, 2016).

Perhaps the most distinct characteristic of data generated by scRNA-seq is the abundance of zeros (Figure 1.1A). Specifically, the expression values of transcripts often exhibit bimodal distributions (Figure 1.1B) such that a gene can have high expression values for some cells but not be expressed in others (Shalek et al., 2013; Kharchenko et al., 2014). Although these expression patterns are due in part to technical variation and low concentrations of mRNA, they are also attributed to true biological differences between populations (or even subpopulations) of cells (Macaulay and Voet, 2014; Finak et al., 2015). Unimodal distributions typically used for the analysis of bulk RNA-seq data fail to capture the complex structure of scRNA-seq data, which bolsters the need for developing new methods specific to single-cell analyses.

Another prominent feature of scRNA-seq data is the increased cell-to-cell variability (Figure 1.1C). Since information is being gathered from individual cells, differences in gene expressions across a single cellular population can now be observed. This means variation in scRNA-seq data can exist both between different groups of cells and within the same cellular population (Huang, 2009; Buettner et al., 2015; Korthauer et al., 2016; Tirosh et al., 2016). Traditional bulk RNA-seq experiments often mask this heterogeneity by averaging out the gene expression measurements, and for that reason, bulk RNA-seq methods do not directly take the high cell-to-cell variation associated with scRNA-seq into consideration. Here, the need for new single-cell analysis methods is again highlighted by the inability of bulk RNA-seq methods to appropriately address the unique characteristics of scRNA-seq data.

Many of the tasks performed in bulk RNA-seq studies, such as detecting genes

that are differentially expressed between different populations of cells or constructing biological networks of genes within a population of cells, are also performed in scRNA-seq studies. Moreover, there are some tasks, such as identifying subpopulations of cells, that are unique to scRNA-seq experiments. Since bulk methods are not appropriate for analyzing single-cell data, new analytic tools specific to scRNA-seq are in high demand. The development of novel statistical techniques for the analysis of scRNA-seq data is gaining much attention, and research in this area is moving quickly. Some progress has already been made in this emerging area of research, but numerous opportunities still exist for the development of new single-cell methodology.

1.1 Differential Expression of Single-cell RNA Sequencing Data

A commonly performed task in sequencing analysis is the identification of genes that are differentially expressed (DE) across different populations of cells. Two of the most commonly referenced methods that have been designed specifically for differential expression analysis of scRNA-seq data are Single-Cell Differential Expression (SCDE; Kharchenko et al., 2014) and Model-based Analysis of Single-cell Transcriptomics (MAST; Finak et al., 2015). SCDE generates error models for each gene by using a mixture of a low-level Poisson distribution and a negative binomial distribution. The Poisson distribution is used to capture genes that are undetected across some of the cells, and the negative binomial distribution is used to address overdispersed expression counts commonly observed in sequencing data. MAST uses a two-part generalized linear model (hurdle model) to analyze continuous scRNA-seq expression levels, as opposed to analyzing discrete count values like SCDE. A logistic regression is initially used to model the proportion of cells that express a given gene, thereby addressing the overinflation of zeros observed in scRNA-seq data. Then, if a gene is expressed within a cell, a Gaussian distribution models the transformed expression level.

More recently, several other methods for detecting DE genes in scRNA-seq have also been proposed. The Beta-Poisson Single-Cell (BPSC) method in Vu et al. (2016) uses a generalized linear model framework with a beta-Poisson mixture model to compare mean expression values across cellular groups. Delmans and Hemberg (2016) introduced the Discrete, Distributional method for Differential gene Expression (D^3E), which also uses a beta-Poisson mixture model but compares gene expression distributions using either the Kolmogorov-Smirnov test, the Cramér-von Mises test, or the likelihood ratio test. A zero-inflated negative binomial model is utilized in DEsingle (Miao et al., 2018) to detect DE genes with likelihood ratio tests and estimate proportions of true zeros and dropout zeros.

Surprisingly, despite the variety of methods designed for differential expression analysis of scRNA-seq data, several studies have concluded that these methods do not perform much better than the bulk RNA-seq methods (Jaakkola et al., 2017; Miao and Zhang, 2016; Soneson and Robinson, 2018). Therefore, opportunities still exist for developing new methodology that will significantly outperform the bulk methods. In Chapter 2, a novel statistical model for high dimensional and zero-inflated scRNA-seq count data is introduced to identify differentially expressed genes across cell types. We adopt a hurdle model to address the overabundance of zeros in scRNA-seq data, and employ a correlated random effects structure guided by an initial supervised subpopulation clustering assignment to capture the observed cellular variability within treatment groups of cells.

1.2 Network Inference from Single-cell Gene Expression Data

Another common task in sequencing analysis is the construction of networks of genes with similar biological processes. These networks, which are often classified as either gene co-expression networks (GCNs) or gene regulatory networks (GRNs), provide valuable insight into the functionality and mechanics of biological processes. In

GCNs, the edges that connect nodes (genes) within the network are considered to be "undirected" since they only indicate the relationships or dependencies between the co-expression of genes, not the underlying cause of these associations. This makes GCNs slightly different from GRNs, which connect nodes with directed edges that can be used to infer casual relationships (De Smet and Marchal, 2010). Network analysis is an important tool in the biomedical sciences because genes involved in the same biological pathway or have similar functionality tend to also have similar expression patterns (Eisen et al., 1998; Allocco et al., 2004). By examining GCNs and GRNs, researchers can gain a better understanding of the relationships and interactions between sets of genes during different cellular functions and processes (Wolfe et al., 2005; Hecker et al., 2009; Wang et al., 2016).

Interestingly, research in designing new methodology for scRNA-seq gene networking analysis has only recently been gaining attention in the literature. Lag-based Expression Association for Pseudotime-series (LEAP; Specht and Li, 2016) is a GCN method that determines gene co-expression by taking into account the possible lags in time that can be caused by cells being in different time points of their cell cycles. Introduced by Aibar et al. (2017), Single-Cell rEgulatory Network Inference and Clustering (SCENIC) constructs GRNs by first detecting potential sets of co-expressed genes within a population of cells and then performing a transcript factor enrichment analysis to identify and score significantly enriched gene sets. The Single-Cell Ordinary Differentiation Equation (SCODE) algorithm from Matsumoto et al. (2017) uses linear ordinary differentiation equations to obtain an optimized square matrix that represents the regulatory relationships between transcription factors. Partial Information Decomposition and Context (PIDC; Chan et al., 2017) is an information theory based algorithm that utilizes partial information decomposition to identify GRNs.

Like the DE methods for scRNA-seq, the current network methods for scRNA-

seq do not outperform methods developed for bulk RNA-seq data (Chen and Mar, 2018). Thus, new network methods applicable to scRNA-seq data need to be developed. In Chapter 3, a sparse Bayesian latent factor model is presented to explore the network structure associated with genes in a single population of cells. For a given cell, a set of shared latent factors adjusts the expression value for each gene, thereby accounting for the zero-inflation and overdispersion commonly observed in scRNA-seq data. A network structure is then inferred from the common factors between pairs of genes that impact their expressions.

1.3 Single-cell Differential Network Analysis

Methods for constructing GCNs and GRNs typically assume that the network structure is being explored within one population of cells such as a single tissue type, environmental condition, or disease status. For some biological studies, however, it may be of greater interest to compare structures from different cellular populations. Different types of cells, or the same type of cell in different stages or conditions, may carry out different functions, and by performing differential network analysis between two (or more) gene-gene association or interaction networks, researchers can identify the parts of the network that are affected by these biological differences.

Most methods for examining differences between gene network structures have been developed in the context of microarray and bulk RNA-seq data. To our knowledge, the literature related to methodology developed for scRNA-seq differential network analysis is quite sparse. In fact, Chowdhury et al. (2019) provide an extensive review on differential co-expression analysis of gene expression data that highlights the need for more research in scRNA-seq methodolgy. The statistical framework developed by Gill et al. (2010) for microarray gene expression data was utilized in Wang et al. (2017) to present proof-of-concept analyses for comparing network structures constructed from scRNA-seq data. Chiu et al. (2018) introduced the scRNA-seq-based

differential network (scdNet) analysis method to determine a sample size corrected gene-gene correlation matrix for each cellular state and identify gene-gene pairs that have significant changes between these states. The authors claim that scdNet is the first tool for differential network analysis of scRNA-seq data.

In Chapter 4, we expand upon the network model proposed in Chapter 3 to examine differences in the underlying networks across two separate cellular populations. Under this model, the parameters that influence the latent factors are treatment-dependent to allow gene-gene co-expression calculations within each group of cells. The gene network structures can then be compared by analyzing credible intervals of the differences between the co-expressions of each group.

1.4 Figures

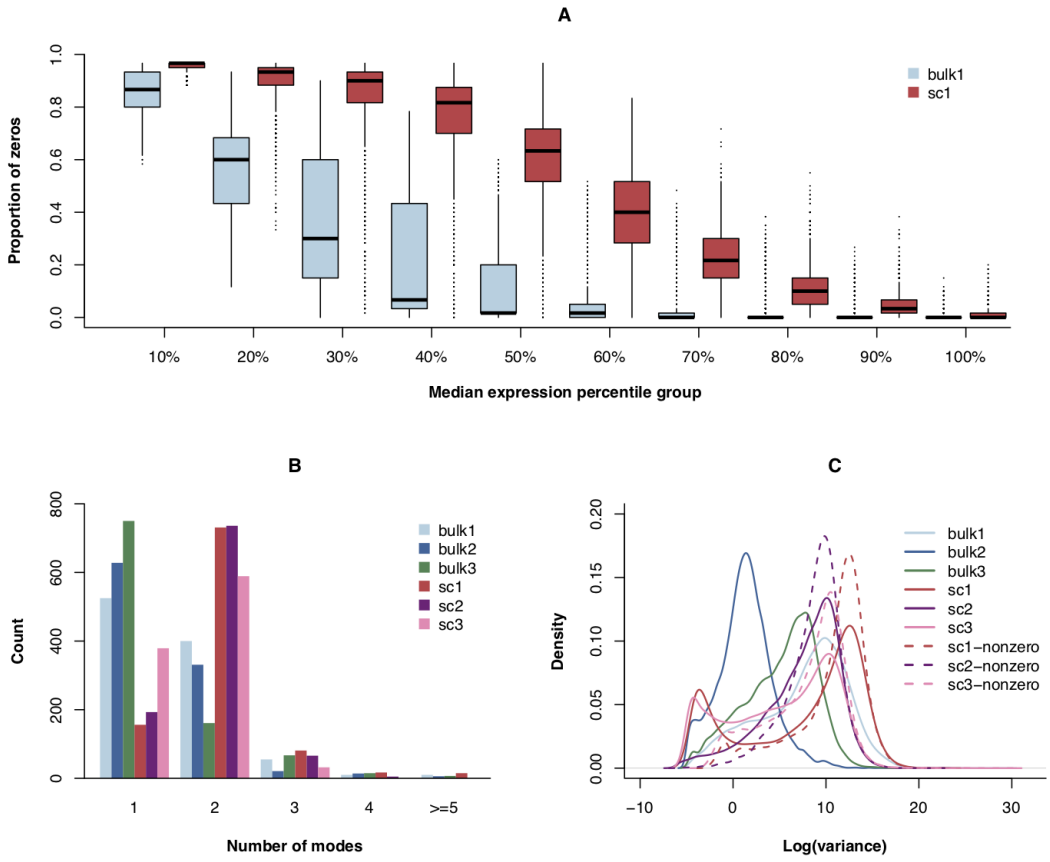

Figure 1.1: Comparisons between bulk RNA-seq and scRNA-seq data. (A) Proportion of zeros boxplots from a bulk (*bulk*1) and a single-cell (*sc*1) dataset. For each type of RNA-seq data, genes were sorted by their median expression values and groups were formed based on percentiles. (B) Estimated number of modes for the expression distributions of 1,000 randomly selected genes from three bulk and three scRNA-seq datasets. (C) Log variance density plots for all of the genes in the datasets from B. Densities were also created for the log variance of the scRNA-seq datasets when zeros were removed to illustrate the variation across the non-zero expression values.

CHAPTER 2

DETECTION OF DIFFERENTIALLY EXPRESSED GENES IN DISCRETE SINGLE-CELL RNA SEQUENCING DATA USING A HURDLE MODEL WITH CORRELATED RANDOM EFFECTS[1]

2.1 Introduction

Rapidly emerging advances in next-generation technology have pushed single-cell analysis to the forefront of gene expression profiling experiments. Traditionally, transcriptomic studies have examined transcript abundance measurements averaged over bulk populations of thousands of cells. While bulk RNA sequencing (RNA-seq) measurements have been valuable in countless studies, they often conceal cell-specific heterogeneity in expression signals that may be paramount to new biological findings. Fortunately, with single-cell RNA sequencing (scRNA-seq), transcriptome data from individual cells are now accessible, providing opportunities to investigate functional states of cells, identify rare cell populations, and uncover diverse gene expression patterns in seemingly homogeneous cell populations (Huang, 2009; Shapiro et al., 2013; Buettner et al., 2015).

One of the most commonly performed tasks in transcriptome expression profiling is the identification of genes that are differentially expressed (DE) across different biological conditions, treatment groups, or cell types. For consistency in this

[1]Reproduced with permission from "Detection of differentially expressed genes in discrete single-cell RNA sequencing data using a hurdle model with correlated random effects" by Michael Sekula, Jeremy Gaskins, and Susmita Datta, 2019. *Biometrics*. DOI:10.1111/biom.13074.

manuscript, we will refer to the populations of cells being compared in a differential expression analysis as treatment groups. Several popular methods for differential expression analysis of traditional bulk RNA-seq datasets currently exist, but these methods fail to capture the intrinsic characteristics that differentiate scRNA-seq data.

The most prominent attribute of scRNA-seq is that a transcript can be moderately or highly expressed in some of the individual cells but not detected in others, resulting in a bimodal distribution of expression values (Shalek et al., 2013; Kharchenko et al., 2014). This expression pattern is caused by the low starting amounts of mRNA within each individual cell in combination with variation from biological and technical sources (Macaulay and Voet, 2014; Finak et al., 2015). The unimodal distributions used for the traditional differential expression analysis of RNA-seq data do not properly model this inherent bimodal structure of scRNA-seq data. In addition, cell-to-cell variability in scRNA-seq has been shown to exist not only between different cellular populations but also within the same population of cells (Huang, 2009; Buettner et al., 2015). This observed heterogeneity is not directly addressed in traditional RNA-seq differential expression methods.

Because scRNA-seq datasets exhibit properties different from bulk RNA-seq datasets, new techniques for identifying DE genes specific to scRNA-seq data need to be developed (Stegle et al., 2015; Bacher and Kendziorski, 2016). Two commonly used methods that have been proposed to identify DE genes while taking into consideration the intricate nature of scRNA-seq data are SCDE (Kharchenko et al., 2014) and MAST (Finak et al., 2015). With the SCDE method, error models for each gene are first modeled using a mixture of a negative binomial distribution (to account for overdispersed expression counts from detected transcripts) and a low-level Poisson distribution (to accommodate genes that are undetected across some of the cells). Posterior probabilities of a given fold expression difference are then calculated to test genes for differential expression between two subgroups of cells. MAST is a

two-part generalized linear model (hurdle model) that analyzes continuous scRNA-seq expression levels, rather than discrete count values. A logistic regression first models the gene expression rate, and, conditioning on a cell expressing the given gene, a Gaussian distribution models the transformed expression level. Differential expression can then be tested using a likelihood ratio test.

In our view, a hurdle model structure (like MAST) is the best way to model such data with an overabundance of zeros. The components of a hurdle model are regression models (one for zero counts and one for expression values), which makes parameter estimation fairly straightforward and computationally simple compared to other types of methods. With a two component model, one can distinguish whether differences between treatment groups come from differences in the proportion of zeros, differences in actual expression, or both. Moreover, hurdle models are flexible and can adjust for potential experimental bias, such as dropout or cell size, with the addition of biological and/or technical covariates.

Rather than transforming the scRNA expression counts to continuous variables (as required in MAST), we adopt the hurdle model approach to directly model the discrete data. Consequently, we propose a mixed effect hurdle model for discrete scRNA-seq gene expression counts to detect genes that are DE between different treatment groups. The expression rate for a particular gene is first modeled with logistic regression to account for the high proportion of zeros in scRNA-seq data, and the expression count is then modeled with a zero-truncated negative binomial regression, conditional on the gene actually being expressed. Besides using discrete count data, another key difference between MAST and our proposed methodology is the incorporation of cell heterogeneity in a supervised manner. We utilize a random effects structure, guided by subpopulations of cells, to provide dependence across genes within a cell and across cells of a subpopulation. Finally, the third major difference between our method and MAST is that we implement a Bayesian approach

to estimate model parameters.

This manuscript is organized as follows. We define the hurdle model and introduce the structure of the correlated random effects (CRE) in Section 2.2. In Section 2.3, we present the methods for estimating model parameters and determining DE genes. Our proposed methodology is applied to both simulated and real data in Section 2.4, where we also compare the performance of our methodology to the performance of other commonly used methods for detecting DE genes. Finally, we conclude with a brief discussion of our results in Section 2.5.

2.2 Methodology

2.2.1 Model Structure

Let Y_{gi} be the expression count of gene g ($g = 1, ..., G$) in cell i ($i = 1, ..., N$), and Z_{gi} indicate whether the gene is expressed within the cell. With this definition, $Z_{gi} = 1$ when $Y_{gi} > 0$, and $Z_{gi} = 0$ when $Y_{gi} = 0$. Defining $\theta_{gi} = P(Z_{gi} = 1)$, the indicator variable Z_{gi} follows Bernoulli(θ_{gi}), and the logistic model is defined as

$$logit(\theta_{gi}) = \beta_{0g}^{L} + X_i \boldsymbol{\beta_g^L} + \omega_i \zeta_g^L \,, \tag{2.1}$$

where X_i is a row from the design matrix consisting of a treatment group indicator and any other covariates of interest, such as cell size or estimated dropout rate. In Equation (2.1), we present the general case where X_i has more than one element, hence $\boldsymbol{\beta_g^L}$ is a vector of regression coefficients. Also, we use the superscript L on the coefficients from the logistic model to distinguish them from the coefficients in the zero-truncated negative binomial regression.

The random effect ω_i for cell i is included in the model to account for additional variability between cells and to induce correlation across genes. Depending on the particular gene, cellular random effects may have more or less of a predictive influence,

thus, we introduce a coefficient for the random effects ζ_g^L to represent a gene specific scaling factor for the random effects within the logistic regression component.

For the conditional expression counts, we define the negative binomial distribution as

$$P(Y_{gi} = y) = \frac{\Gamma(y + \phi_g)}{y!\Gamma(\phi_g)} \left(\frac{\mu_{gi}}{\mu_{gi} + \phi_g} \right)^y \left(\frac{\phi_g}{\mu_{gi} + \phi_g} \right)^{\phi_g}, \quad y = 0, 1, \ldots, \tag{2.2}$$

where μ_{gi}, $\phi_g > 0$. Under this set-up, $E(Y) = \mu_{gi}$ and $Var(Y) = \mu_{gi} + \frac{\mu_{gi}^2}{\phi_g}$, making ϕ_g the overdispersion parameter. A modification to Equation (2.2) is needed to account for conditioning on a non-zero expression count ($Y_{gi} > 0$); therefore, the zero-truncated negative binomial distribution is defined as

$$P(Y_{gi} = y | Z_{gi} = 1) = \frac{\frac{\Gamma(y+\phi_g)}{y!\Gamma(\phi_g)} \left(\frac{\mu_{gi}}{\mu_{gi}+\phi_g} \right)^y \left(\frac{\phi_g}{\mu_{gi}+\phi_g} \right)^{\phi_g}}{1 - \left(\frac{\phi_g}{\mu_{gi}+\phi_g} \right)^{\phi_g}}, \quad y = 1, 2, \ldots, . \tag{2.3}$$

Using the distribution in (2.3), we have the following regression model for conditional expression counts:

$$log(\mu_{gi}) = \beta_{0g}^C + X_i \boldsymbol{\beta_g^C} + \omega_i \zeta_g^C \, . \tag{2.4}$$

Here, the superscript C indicates the coefficients in the count model (zero-truncated negative binomial regression). With this hurdle component, the regression coefficients in (2.4) can be interpreted as approximately representing a multiplicative effect on the expression count. Thus, if X_{i1} is a dummy variable for the treatment indicator, β_{1g}^C (the first element of vector $\boldsymbol{\beta_g^C}$) would approximately represent the log-fold change. The same random effect (ω_i) used in the logistic model is also used here to control dependence across genes and dependence with the logistic model. The coefficient of the random effects ζ_g^C is representative of a scaling factor for ω_i per gene within the

truncated negative binomial regression component.

2.2.2 Correlated Random Effects

It has been observed that within defined treatment groups of scRNA-seq experiments there exist subpopulations of cells with different expression patterns across different genes (Huang, 2009; Buettner et al., 2015). In order to account for this observation, we assume that the random effects of cells within a subpopulation are positively correlated, but between subpopulations, the random effects are independent. We refer to this model as CRE.

Before utilizing our CRE model, cells within each treatment group need to be clustered separately to form K_0 subpopulations in the control group and K_1 subpopulations in the treatment group. These subpopulation clusters can be identified with a suitable scRNA-seq clustering algorithm and then applied to our model structure. We must emphasize that our focus is not on how to perform a cluster analysis, but rather on how the results from a cluster analysis are incorporated into our model. Therefore, we assume that best practices (e.g., normalization, batch effect adjustments, etc.) have been followed before clustering to avoid additional influence of any biological and/or technical bias on the differential expression results.

Letting $k_t(i)$ indicate the cluster assignment for cell i in treatment t, we have $k_0(i)$ and $k_1(i)$ representing the clusters/subpopulations within the control group and treatment group, respectively. Using this notation, each cellular random effect ω_i is defined as the sum of two separate components: $\omega_i = \gamma_{t,k_t(i)} + \omega_i^*$. Here, $\gamma_{t,k}$ represents the average random effect for the subpopulation k within treatment t, and ω_i^* is the individual cellular adjustment for cell i within the subpopulation. With each $\gamma_{t,k}$ following an independent and identically distributed (i.i.d.) Normal$(0, \sigma_t^2)$ and each ω_i^* following an i.i.d. Normal$(0, \sigma_*^2)$, the correlation, ρ_t, between cells within the same subpopulation is then $\rho_t = \frac{\sigma_t^2}{\sigma_t^2 + \sigma_*^2}$.

We note that the random effect ω_i enters into the model through the terms $\omega_i\zeta_g^L$ and $\omega_i\zeta_g^C$. As ω_i, ζ_g^L, and ζ_g^C are all estimated parameters, the individual ω_i's are, therefore, scale-unidentified. However, the relative contribution of $\gamma_{t,k_t(i)}$ to ω_i is identifiable, as will be the correlation ρ_t. To facilitate interpretation, the estimated ω_i's can be post hoc rescaled to have variance one.

For special cases, such as datasets with large numbers of cells, or situations when an initial clustering is not preferred, the correlations between random effects can be removed and one can simply assume that all of the random effects are independent of each other. Under this assumption, each ω_i simply follows an i.i.d. Normal$(0, \sigma^2)$. We refer to this model choice as independent random effects (IRE).

2.3 Model Inference

2.3.1 Parameter Estimation

A Bayesian approach is utilized to estimate the parameters of our proposed model. While a seemingly straight-forward technique for obtaining these parameter estimates is Markov chain Monte Carlo (MCMC) sampling, it may take days of computational time for this iterative process to generate enough samples to reasonably estimate our model parameters on large scRNA-seq datasets. This makes an MCMC approach impractical compared to the computational time of methods currently available for identifying DE genes. If time is not a factor, researchers may still choose to utilize full MCMC to obtain parameter estimates and model inference.

Instead of time-consuming MCMC, we use variational inference (VI) to approximate the posterior distribution and obtain parameter estimates more quickly. VI has been recently proposed as a computationally faster alternative to MCMC for solving Bayesian problems involving large data (Blei et al., 2017). Briefly, mean-field variational Bayes approximates the usual posterior distribution $p(\Theta|y)$ with a distri-

bution $q(\Theta)$ that assumes all components of Θ are independent, $q(\Theta) = \prod_j q_j(\Theta_j)$. This leads to an optimization problem of finding the $q(\Theta) = \prod_j q_j(\Theta_j)$ that is closest in Kullback-Leibler divergence to the true posterior.

Introduced by Kucukelbir et al. (2015), automatic differentiation variational inference (ADVI) is a user-friendly method that automatically generates an algorithm to solve this optimization problem. In essence, each $q_j(\cdot)$ is assumed to be a normal distribution on a suitable transformation of Θ_j. Optimizing the parameters of these normal distributions is accomplished using a stochastic gradient ascent algorithm to maximize the evidence lower bound (ELBO). Monte Carlo integration approximates the expectations of the ELBO and automatic differentiation computes the gradients that are maximized. We implement the mean-field algorithm of ADVI in R (R Core Team, 2018) through the package rstan (Stan Development Team, 2018). After achieving convergence to the approximate posterior with rstan's ADVI algorithm, parameter samples are drawn independently from $q(\Theta)$ and provided to the user as approximate posterior samples from $p(\Theta|y)$.

To complete the specification of our Bayesian model, we need prior distributions for the remaining parameters. The regression coefficients in both the logistic regression and the zero-truncated negative binomial regression are given weakly informative Cauchy priors (Gelman et al., 2008). The intercept terms have Cauchy$(0, 10)$ priors, while the remaining coefficients have Cauchy$(0, 2.5)$ priors. The lognormal distribution is used as the prior for the overdispersion parameter ϕ_g, with the hyperparameters $\lambda_1$and λ_2 defined as

$$\phi_g \sim Lognormal(\lambda_1, \lambda_2),$$

$$\lambda_1 \sim Cauchy(0, 10),$$

$$\lambda_2 \sim Inverse\ Gamma(0.001, 0.001).$$

Sensitivity analysis with the case study data (see Appendix D) indicates replacing the λ_2 prior with Inverse Gamma$(1, 1)$ has little effect on the final differential expression results. In addition, the variance parameters for the cellular random effects, σ_t^2 and σ_*^2, have Inverse Gamma$(1, 1)$ priors. By choosing this prior for the variance parameters, the prior for ρ_t, the correlation between cells within the same cluster, is Beta$(1, 1)$.

2.3.2 Testing for Differential Expression

Typically, the target of inference in scRNA-seq studies is to determine if genes from two treatments are "differentially expressed" (DE), that is, they follow different distributions. In our modeling framework, the difference in the distributions between treatments is controlled by the pair of regression coefficients β_{1g}^L and β_{1g}^C of the treatment group indicator. A gene is considered to be DE if at least one of these parameters is non-zero. That is, we need to test $H_0 : \beta_{1g}^L = \beta_{1g}^C = 0$ against the alternative. While we develop our model and estimate parameters under the Bayesian paradigm, we choose to perform hypothesis testing under the standard frequentist framework as most researchers are more familiar with this approach.

We define $\widehat{\boldsymbol{B_g}}$ to be the two-dimensional vector consisting of the point estimates of $\widehat{\beta_{1g}^L}$ and $\widehat{\beta_{1g}^C}$ (the treatment effect coefficients from the logistic and count models for gene g), and let $\boldsymbol{V_g}$ represent the estimate of the covariance matrix, as determined empirically from the posterior samples (provided as output by Stan) of these two coefficients. Wang and Blei (2019) have established that the variational posterior $q(\Theta)$ is asymptotically normal with a random mean centered at the true parameter value. Thus, under the null hypothesis that $\beta_{1g}^L = \beta_{1g}^C = 0$, the test statistic $W_g = \widehat{\boldsymbol{B_g}}^T \boldsymbol{V_g}^{-1} \widehat{\boldsymbol{B_g}}$ will asymptotically follow a chi-square distribution with two degrees of freedom. If W_g is larger than the appropriate critical value, we reject H_0 and conclude that gene g is differentially expressed; a p-value can also be obtained.

2.4 Applications

2.4.1 Simulation Studies

To evaluate performance, we applied our method to simulated data generated from our proposed model. We considered scenarios when the true subpopulations had equal sizes (same number of cells per subpopulation) and unequal sizes (different number of cells per subpopulation). An additional simulation was run using data generated with the Splat simulation design (Zappia et al., 2017) to evaluate the performance of our models on data simulated from a structure that differs from our proposed methodology. Small datasets (100 cells and 10,000 genes) were analyzed to illustrate the feasibility of our methods, while larger datasets (1,000 cells and 10,000 genes) were analyzed to demonstrate their practical utility. Details on these simulation designs are available in the Appendix.

Two versions of the proposed hurdle model, the CRE model and the IRE model, were evaluated under these different simulation scenarios. In our model matrix, we included the treatment group indicator and the cellular detection rate (CDR) as covariates. The CDR for cell i is the sample proportion of genes that have a non-zero count,

$$CDR_i = \frac{1}{G}\sum_{g=1}^{G} Z_{gi} \,, \tag{2.5}$$

and has been presented by Finak et al. (2015) as an important source of variability that captures variation due to biological and technical factors, such as cell volume and dropout. Since our model does not inherently distinguish between true biological zeros and technical zeros, including CDR as a covariate will help control for expression differences due to these unwanted sources of variation. To demonstrate that the addition of random effects actually improves upon a model that includes CDR, we also tested our proposed methodology using only fixed effects (i.e., removing the $\omega_i\zeta_g$

terms from (2.1) and (2.4)). We refer to this model as no random effects (NRE).

When applying the CRE model, clusters within each treatment group were assigned by the SNN-Cliq algorithm (Xu and Su, 2015) and the SC3 algorithm (Kiselev et al., 2017). SNN-Cliq requires the number of nearest neighbors to find a clustering structure, and so the number of nearest neighbors was set to the default value of three. We also set the number of nearest neighbors to seven in the hurdle model simulations to obtain fewer numbers of clusters within each treatment group. Since the Splat simulation does not inherently create subpopulations, we only considered three nearest neighbors. The SC3 algorithm, on the other hand, requires the number of clusters to be specified. Therefore, we utilized SC3's option to estimate the optimal number of clusters for each treatment group in each simulated dataset. As an alternative clustering option in the hurdle model simulations, we also estimated the model under the true data-generating subpopulation assignments to gauge performance when the actual clustering structure is known.

In addition to evaluating the performance of our methodology, we also compare our proposed models to methods commonly used in the literature. The methods for scRNA-seq differential expression of MAST and SCDE were examined in these studies along with two methods designed for bulk RNA-seq differential expression: edgeR (Robinson and Smyth, 2007) and DESeq2 (Love et al., 2014). The covariate of CDR was also implimented in the model matrix for MAST as described in the MAST package vignette for the MAIT data analysis (McDavid et al., 2019).

Instead of utilizing raw discrete counts, the two clustering algorithms and MAST require continuous expression data. To that end, a trimmed mean of M-values normalization was first applied to the simulated datasets to account for between-sample bias, and the adjusted count values were then scaled to counts per million with the edgeR package (2019), thereby accounting for differences in library size.

In this simulation analysis, 100 datasets were generated for all scenarios, and

DE genes were determined at a false discovery rate (FDR) of 0.05 for each method. We used the measures of true positive rate (TPR), false positive rate (FPR), observed FDR, area under the receiver operating characteristic curve (AUC), and the number of identified DE genes to compare methods. Summaries of these measures for all simulations are provided in Table 2.1.

In all hurdle model simulations, the NRE model was unable to control FDR at a nominal rate and had approximately twice the FDR as both CRE and IRE. This result was observed even when the "true" subpopulation structures were simulated with different numbers of clusters and with higher within-cluster correlations (Appendix A). Therefore, the addition of random effects to our methodology helps control the FDR when underlying subpopulations exist. When comparing the random effects models, CRE and IRE have similar performances in the smaller hurdle model simulations, with CRE having slightly lower FDRs. However, this difference becomes more evident in the larger datasets. Hence, the correlated random effects do a better job at controlling the FDR to a nominal level than the independent random effects. Moreover, the correlation structure of CRE is quite robust as the performance of this method is generally unaffected by the initial clustering of cells within each treatment group.

Even though the Splat design does not simulate a subpopulation structure, CRE and IRE still obtain higher TPRs and larger numbers of detected DE genes compared to NRE. Additionally, the Splat simulations demonstrate that bias is not introduced if a clustering structure is input into CRE when the dataset does not inherently have "true" subpopulations. In fact, the results from IRE and CRE are nearly identical in the larger Splat simulation.

When comparing our methodology to the other methods for detecting DE genes, CRE and IRE consistently identified large numbers of DE genes with high power (TPR), and detected more DE genes than MAST across all simulations. The

FDR of our models is well maintained at the nominal level (as is MAST), but SCDE and the bulk methods consistently fail to control FDR. Regarding the AUC, our method outperforms the competing approaches (both bulk and scRNA) across all scenarios. Because it also relies on a hurdle model specification, we do note that MAST is somewhat competitive in AUC when the data-generating mechanism is our hurdle model. Nevertheless, MAST performed poorly in the Splat simulations as it had the lowest AUC out of all the considered methods. These overall trends were also observed when different subpopulation structures were simulated (Appendix A).

2.4.2 Case Studies

To further illustrate our proposed methods, we analyzed the mouse embryonic cell (MEC) dataset (Islam et al., 2011), which contains expression counts of 92 single-cells generated from two different cell types: 48 stem cells and 44 fibroblast cells. This dataset was obtained from the Gene Expression Omnibus (GEO) database under accession number GSE29087. Genes not expressed in at least 20% of the cells were removed, leaving 7,912 genes in the analysis. SNN-Cliq with five nearest neighbors was used to generate cluster assignments within each treatment group to form three clusters within the stem cells and two clusters within the fibroblast cells for input into our CRE model. It has been noted that if the number of nearest neighbors is too large, clusters formed by SNN-Cliq may not be thoroughly separated, and if the number of nearest neighbors is too small, a true cluster may be split into multiple parts (Xu and Su, 2015). Based on visual inspections of different SNN-Cliq clustering assignments, we determined that five nearest neighbors was a reasonable choice (not too small, not too large) for clustering this particular dataset.

We also analyzed a Drop-seq dataset containing single-cell expressions of 2,000 human mammary epithelial cells (HMEC) expressing either exogenous wild type or mutant histone H2B to demonstrate the utility and scalability of our methods on big

data. Details of the Drop-seq procedure are provided in Appendix C. This dataset is quite sparse, so we chose to filter out genes not expressed in at least 50 cells (2.5% of the total cells), leaving a total of 3,139 genes in the analysis. Four cells from the dataset were also removed because they had a library size of zero after gene filtering. To define the random effect structure of CRE, we utilized SC3 to cluster the 999 wild type cells into seven subpopulations and the 997 mutant cells into nine subpopulations. These clustering results were taken from a previous cluster analysis performed on this data by our research group (see Appendix C).

The differential expression methods used in the simulation studies, excluding NRE, were run in R independently using a single core of a system with an Intel Core i7 processor (3.5 GHz) and 8 GB of RAM. Based on the results in Table 2.2, the computational time required for our methods is very reasonable for a Bayesian analysis on datasets of these sizes. MCMC sampling would take days of running time before obtaining enough samples for an appropriate analysis. While both CRE and IRE take longer than the other methods on the smaller MEC dataset, they do scale better, and are also faster, than SCDE and DESeq2 on the larger HMEC data.

In terms of the number of DE genes detected, CRE and IRE performed similarly in the case studies, which is consistent with the simulation results. CRE identified 4,927 and 1,698 DE genes in the MEC and HMEC datasets, respectively, while IRE identified 4,947 and 1,696 genes. The overlap of genes was also very high in the case studies for these methods (4,808 in the MEC data, and 1,640 in the HMEC data). Nevertheless, the simulation studies do show that CRE outperforms IRE in terms of FDR, especially in larger datasets. For that reason, we focus the rest of our discussion in this section on interpreting the CRE results.

Figure 2.1 displays the UpSet plots (Lex et al., 2014) for the intersection of DE genes identified by the different methods in the MEC and HMEC analyses. Just like the simulation studies, our method detects a larger number of DE genes than

most methods in these case studies. In the MEC data, CRE detected the most DE genes, and surprisingly, the other two scRNA-seq methods detected fewest DE genes. CRE was second, only to edgeR, in the number of identified DE genes in the HMEC dataset, while DESeq2 detected very few DE genes compared to the other methods.

As CRE identified a large number of unique DE genes in the MEC analysis relative to the competing models, we further examined the 1,335 genes uniquely identified as DE by CRE to determine if they have any biological relevancy. For comparison purposes, we also examined the 293 genes identified only by edgeR and the 172 genes detected only by DESeq2 (Appendix D). We found that the subset of genes detected by CRE are associated with more clusters of enriched gene ontology (GO) categories than the subsets of genes detected by the other two methods. Thus, not only is CRE able to identify a larger number of DE genes, but these genes also have roles in similar biological functions and processes.

From a statistical standpoint, our methodology determines DE genes by taking into account the difference in the proportion of zeros between the two treatments as well as the difference in the average counts conditional on the gene being expressed. This is why the CRE model detects different genes than the other methods, particularly in the MEC analysis. In Figure 2.2, we present the log2 fold change (log2FC) against the log2 ratio of the proportion of zeros (log2PZ) for the top 500 genes determined by each method in the MEC data.

Figure 2.2 highlights the ability of our methodology to incorporate both components of the hurdle model (zero counts and expression values) when identifying DE genes. Out of the top 500 genes identified by CRE, 433 of them had notable differences in the number of zeros across groups as indicated by the absolute value of the log2PZ being greater than one (i.e., one treatment group has more than twice the number of zeros than the other). In addition, all but one of those genes also had an absolute value of log2FC greater than one. Therefore, most of the top DE genes

identified by CRE were not only different in terms of the number of zeros between the stem and fibroblast cells, but also in terms of the expression values between the treatment groups.

The other methods did not detect as many DE genes with notable differences in the proportion of zeros between treatments. Out of all of the methods, MAST detected the fewest number of genes with an absolute value of log2PZ greater than one. This showcases the superiority of our model in the MEC analysis since MAST also takes into account the differences in the proportion of zeros when determining DE genes. Only 238 out of the top 500 genes identified by MAST had more than a twofold difference in the proportion of zeros between treatments, whereas DESeq2, edgeR, and SCDE identified 342, 331, and 290 genes that satisfied this criterion, respectively.

Lastly, to demonstrate the role of the random effects in our methodology, we ran an additional analysis of the MEC data with a model matrix that only included the treatment group indicator covariate. This produced an estimate of ω_i that is free from potential feedback from the CDR covariate. We see from Figure 2.3A that the estimates of the random effects within the same treatment subpopulation tend to have similar values when CDR was not included in the model matrix. The estimated within subpopulation variance was 0.40 while the between subpopulation variance was 14.13. A majority of the random effect estimates for Subpopulation 1 in the stem cells and Subpopulation 1 in the fibroblast cells were negative, whereas the estimates of the random effects for the other subpopulations were mostly positive. When CDR was included in the model (Figure 2.3B), there appeared to be less separation between cluster/subpopulation means, as this information is now accounted for through CDR. For these estimates, the within subpopulation variance was 0.91 and the between subpopulation variance was 2.84.

In Figures 2.3C and 2.3D, the normalized random effects were plotted against

the normalized CDR to display the relationship between these two terms. With CDR not included in the model matrix, the random effect estimates tended to be fairly similar to their corresponding CDR counterparts. The linear association between these terms suggests that they are able to capture similar cellular variability. When CDR was included in the model, there is no longer a discernible trend between the random effects and CDR, indicating that the random effects terms are accounting for some secondary source of cellular variation beyond the fraction of genes expressed.

2.5 Discussion

In this manuscript, we have introduced a mixed effects hurdle model for detecting genes that are DE between treatment groups of cells in discrete scRNA-seq data. The hurdle model structure handles the abundance of zero counts typical of scRNA-seq, while the CRE help account for the cell-to-cell heterogeneity that has been frequently observed within treatment groups of cells. One may also choose to use our proposed hurdle model with independent random effects for situations where clustering may not be suitable. Both of our proposed models (CRE and IRE) outperformed two methods developed for detecting DE genes in scRNA-seq data (MAST and SCDE) and two methods designed for bulk RNA-seq data (edgeR and DESeq2) in the simulation studies. We recommend using CRE over IRE when possible as it tends to have lower FDRs.

Our proposed methodology is comparable in structure to that of MAST, developed by Finak et al. (2015), which is also a hurdle model but for continuous data. We likewise incorporate the covariate of CDR (Equation (2.5)) into our model matrix to help control for the expression differences due to unwanted sources of variation. Nevertheless, our methodology is unique because it (1) analyzes discrete count data rather than continuous data that has already been transformed, (2) incorporates a novel correlated random effect structure to capture additional sources of variation, and

(3) utilizes a Bayesian approach to parameter estimation. These three key differences lead to the detection of more DE genes and higher TPRs and AUCs than MAST, as illustrated in the simulation studies. The MEC data analysis also demonstrates that our model can be more sensitive when detecting differences in the proportion of zeros. Therefore, despite the similarities in motivation, our methodology demonstrates superior performance over MAST.

We additionally note that we utilize a VI technique to quickly obtain samples of parameter estimates from an approximate posterior distribution rather than a typical MCMC sampling on the true posterior. However, there has been discussion in the literature regarding the accuracy of variance estimates under mean-field variational Bayes. Recently, Wang and Blei (2019) show that VI can recover the diagonal of the concentration (inverse covariance) matrix, but since the off-diagonal elements of concentration are set to zero, the marginal variances may be underestimated.

For this reason, some have argued that mean-field variation inference may not produce appropriate testing conclusions (e.g., Kucukelbir et al., 2017). Alternative approaches include running full MCMC or estimating parameter variances by bootstrapping treatment assignments (Chen et al., 2018), although both would be enormously computationally expensive. However, based on our empirical work in Section 2.4, we emphasize that we are able to accurately estimate the regression parameters in this context and that our hypothesis tests maintain the required FDR and achieve higher power than the competing scRNA methods. Thus, we conclude that our proposed methodology for detecting DE genes from single-cell RNA represents a new and powerful strategy for this biologically important problem.

2.5.1 Software Availability

The R package that implements our proposed methodology is maintained at github.com/mnsekula/scREhurdle.

2.6 Tables and Figures

	Hurdle model: Equal clusters ($N = 100$)					Hurdle model: Equal clusters ($N = 1000$)				
	TPR	**FPR**	**FDR**	**AUC**	**DE Genes**	**TPR**	**FPR**	**FDR**	**AUC**	**DE Genes**
CRE, SC3	0.682	0.010	0.046	0.958	1548	0.751	0.016	0.056	0.965	1746
CRE, NN=3	0.682	0.009	0.045	0.959	1547	0.752	0.015	0.058	0.966	1739
CRE, NN=7	0.682	0.010	0.046	0.958	1550	0.751	0.015	0.058	0.965	1741
CRE, TRUE	0.682	0.010	0.047	0.958	1551	0.752	0.016	0.059	0.965	1749
IRE	0.686	0.011	0.050	0.958	1564	0.754	0.026	0.078	0.960	1830
NRE	0.669	0.019	0.086	0.947	1591	0.745	0.047	0.168	0.942	1965
MAST	0.594	0.007	0.038	0.948	1337	0.693	0.008	0.039	0.962	1563
SCDE	0.126	0.094	0.200	0.646	974	0.308	0.280	0.396	0.601	2756
DESeq2	0.402	0.058	0.321	0.777	1304	0.426	0.084	0.394	0.767	1549
edgeR	0.396	0.073	0.377	0.764	1400	0.488	0.148	0.504	0.740	2161
	Hurdle model: Unequal clusters ($N = 100$)					**Hurdle model: Unequal clusters** ($N = 1000$)				
	TPR	**FPR**	**FDR**	**AUC**	**DE Genes**	**TPR**	**FPR**	**FDR**	**AUC**	**DE Genes**
CRE, SC3	0.681	0.010	0.049	0.957	1550	0.572	0.017	0.074	0.924	1367
CRE, NN=3	0.680	0.010	0.049	0.957	1547	0.573	0.018	0.077	0.923	1372
CRE, NN=7	0.681	0.010	0.049	0.957	1550	0.573	0.018	0.079	0.923	1374
CRE, TRUE	0.681	0.010	0.049	0.957	1549	0.573	0.017	0.076	0.924	1365
IRE	0.684	0.011	0.051	0.957	1560	0.579	0.026	0.097	0.919	1444
NRE	0.669	0.023	0.097	0.945	1616	0.573	0.050	0.209	0.894	1610
MAST	0.587	0.007	0.038	0.947	1320	0.468	0.007	0.048	0.914	1066
SCDE	0.165	0.130	0.257	0.631	1353	0.364	0.343	0.554	0.545	3340
DESeq2	0.411	0.074	0.361	0.766	1442	0.378	0.151	0.549	0.684	1941
edgeR	0.407	0.087	0.406	0.754	1525	0.439	0.214	0.607	0.667	2543
	Splat ($N = 100$)					**Splat** ($N = 1000$)				
	TPR	**FPR**	**FDR**	**AUC**	**DE Genes**	**TPR**	**FPR**	**FDR**	**AUC**	**DE Genes**
CRE, SC3	0.475	0.006	0.051	0.924	576	0.434	0.004	0.041	0.840	539
CRE, NN=3	0.475	0.006	0.052	0.923	576	0.434	0.004	0.041	0.841	540
IRE	0.501	0.008	0.063	0.923	615	0.434	0.004	0.041	0.841	540
NRE	0.404	0.007	0.063	0.910	497	0.373	0.003	0.031	0.828	459
MAST	0.220	0.002	0.032	0.879	274	0.287	0.003	0.037	0.795	355
SCDE	0.254	0.002	0.028	0.917	303	0.287	0.001	0.010	0.826	294
DESeq2	0.601	0.042	0.215	0.887	892	0.450	0.019	0.149	0.814	634
edgeR	0.740	0.076	0.297	0.911	1219	0.563	0.062	0.317	0.818	987

Table 2.1: Results of performance measures from simulation studies. The subpopulation structure for the CRE model was input using either SC3, SNN-Cliq with 3 nearest neighbors (NN=3), 7 nearest neighbors (NN=7), or the true simulated subpopulation assignment (TRUE).

	CRE	**IRE**	**MAST**	**SCDE**	**DESeq2**	**edgeR**
MEC data	38.4	40.1	1.2	23.6	0.8	0.1
HMEC data	77.6	69.7	2.3	107.9	392.8	0.9

Table 2.2: Running times (in minutes) of each method for each case study.

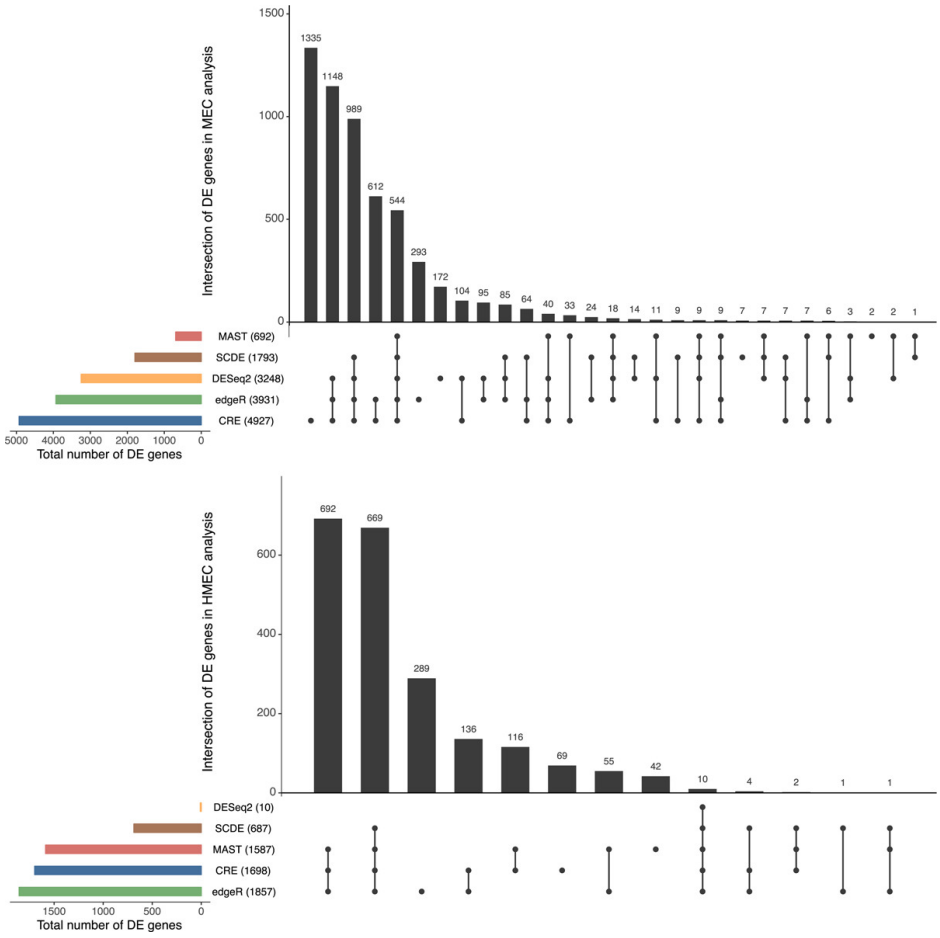

Figure 2.1: UpSet plots of DE genes as determined by five different methods for both the MEC and HMEC datasets. Numbers in parentheses represent the total number of DE genes identified by the corresponding method. This figure appears in color in the electronic version of this article.

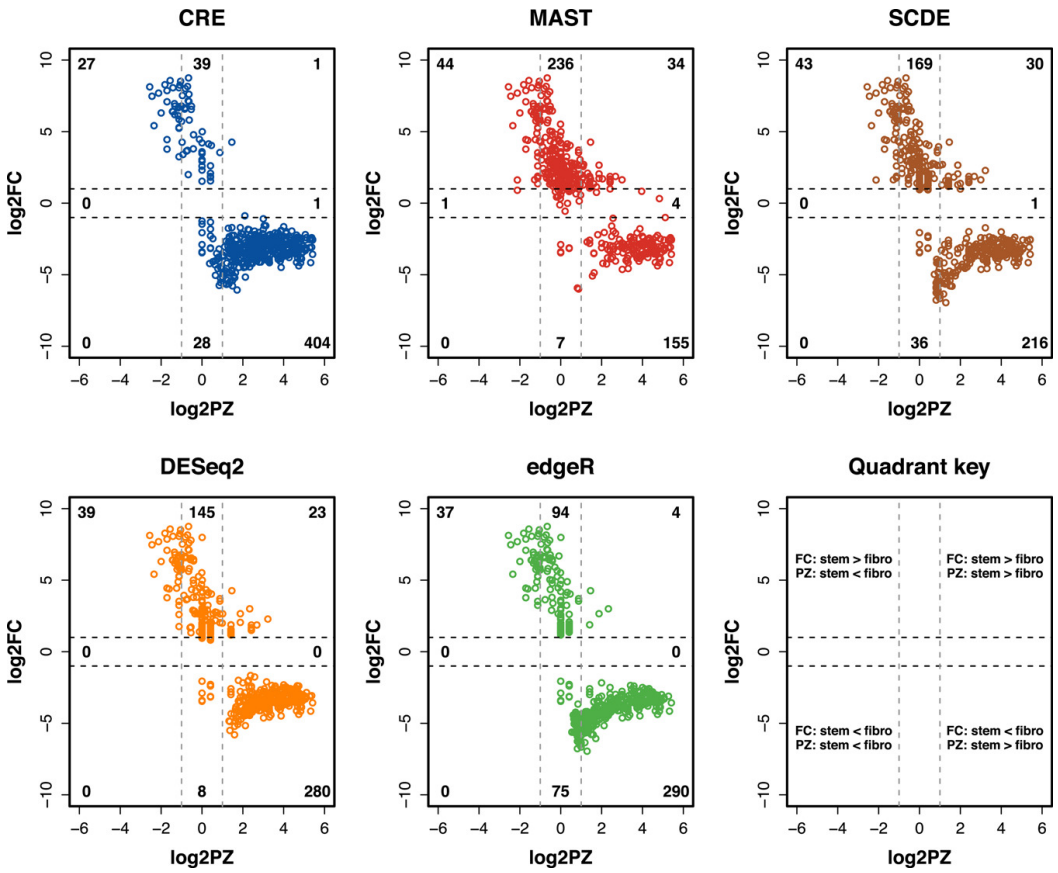

Figure 2.2: Scatterplots of the log2 proportion of zeros ratio (log2PZ) on the x-axis and the log2 fold change (log2FC) on the y-axis for the top 500 most DE genes in the MEC dataset as determined by the five different methods for detecting DE genes. Ratios compare stem cells to fibroblast cells and the labels in each section of the plot represent the number of genes in that section. This figure appears in color in the electronic version of this article.

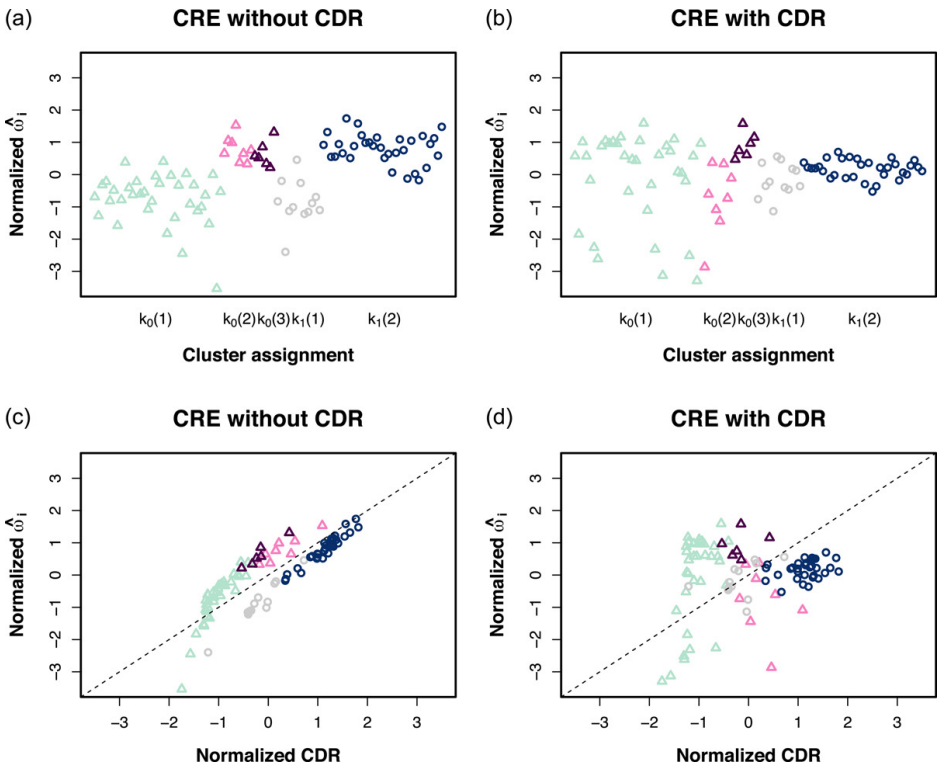

Figure 2.3: Random effect estimates, $\widehat{\omega}_i$, for the cells in the MEC dataset with points colored according to cluster assignments determined by SNN-Cliq. (A, B) Plots of normalized $\widehat{\omega}_i$ estimates by subpopulation $k_t(i)$ where $t = 0$ for stem cells (represented by triangles) and $t = 1$ for fibroblast cells (represented by circles). (C, D) Plots of normalized $\widehat{\omega}_i$ estimates vs. normalized CDR. This figure appears in color in the electronic version of this article, and color refers to that version.

CHAPTER 3

A SPARSE BAYESIAN FACTOR MODEL FOR THE CONSTRUCTION OF GENE CO-EXPRESSION NETWORKS FROM SINGLE-CELL RNA SEQUENCING COUNT DATA

3.1 Background

Deriving co-expression networks from gene expression data is a primary goal in numerous biological studies. These networks, which are commonly referred to as gene co-expression networks (GCNs), are constructed by identifying pairs of genes that have significant associations between their expression profiles across samples. Genes are represented by nodes in GCNs and co-expression values are represented by edges that connect pairs of nodes. These edges are undirected to indicate the relationships or dependencies between genes, not the underlying cause of these associations. This makes GCNs different from gene regulatory networks, which have directed edges to infer casual relationships (De Smet and Marchal, 2010). As demonstrated in Eisen et al. (1998), genes with similar expression patterns tend to be involved in similar cellular processes and functions. Therefore, researchers are able to identify novel interactions and relationships between genes by exploring GCNs (Wolfe et al., 2005; Wang et al., 2016).

Many of the statistical methods for building GCNs have been developed for analyzing data consisting of expression values averaged over bulk populations of cells, such as microarray or bulk RNA sequencing; however, advancements in technology

now allow researchers to obtain expressions at the level of a single cell. By gathering information from individual cells, new opportunities to study cellular heterogeneity are presented. This is of particular interest in GCNs since mapping gene expressions across different states of cells can lead to a better understanding of the biological mechanisms behind this heterogeneity (Fiers et al., 2018). Single-cell RNA sequencing (scRNA-seq) provides new and exciting opportunities to examine biological processes at a high resolution, yet at the same time, this data presents new statistical and computational challenges (e.g., zero-inflation, high cell-to-cell variability, multimodality) that have not been previously faced with bulk sample data (Bacher and Kendziorski, 2016). Therefore, network algorithms initially developed for bulk samples are often not suitable for single cell analysis (Blencowe et al., 2019).

Some algorithms for network analysis in scRNA-seq data have been recently proposed, but these methods fail to outperform general methods developed for bulk sample data Chen and Mar (2018). To that end, we present a sparse hierarchical Bayesian factor model to explore the network structure associated with genes. The latent factors in our model adjust the gene expressions for each cell to help accommodate for the zero-inflated and overdispersed attributes of scRNA-seq data, and a GCN structure is constructed by examining the shared factors between pairs of genes.

This manuscript is organized as follows. We define our proposed model and GCN inference in Section 3.2. In Section 3.3, we apply our method to both simulated and real data and also compare the performance of our methodology to the performance of other network methods. Finally, we conclude with a brief summary in the Section 3.4.

3.2 Methods

3.2.1 Hierarchical Bayesian Factor Model

Let Y_{gi} be the (count) expression for gene g ($g = 1, \ldots, G$) in cell i ($i = 1, \ldots, N$). We assume each expression comes from the Poisson(μ_{gi}) distribution, where the mean μ_{gi} is modeled through the representation

$$\mu_{gi} = \beta_g \prod_{f=1}^{F} exp\Big\{ -\frac{\phi_f}{2}|\alpha_{gf}|\Big\}\lambda_{if}^{\alpha_{gf}}. \tag{3.1}$$

Here, the parameter β_g denotes the average expression for gene g. For each cell i, there are F associated factors $\boldsymbol{\lambda_i} = \{\lambda_{i1}, \ldots, \lambda_{iF}\}$ that impact the expression. These factors are strictly positive and come from a Lognormal$(0, \phi_f)$ distribution. We can think of each factor as representing a distinct attribute (e.g., cell stage, pseudotime point) that will only influence a specific set of related gene expressions. The exponent of the fth factor λ_{if} is $\alpha_{gf} \in \{-1, 0, 1\}$, and by using this set of discrete exponents for the factors, the expression for gene g is impacted only by the factors with $\alpha_{gf} = -1$ or 1. The adjustment term of $exp\{-\frac{\phi_f}{2}|\alpha_{gf}|\}$ is included in Equation (3.1) to ensure that $E(Y_{gi})$ is equal to β_g (after marginalizing out $\boldsymbol{\lambda_i}$) regardless of the α_{gf} values.

Our defined factor structure provides the flexibility required to account for the typical cell-to-cell variability of scRNA-seq data. For a given f, λ_{if} is unique to each cell and is only activated for a particular gene when $\alpha_{gf} \neq 0$. If the activated factors $\lambda_{if}^{\alpha_{gf}}$ for a given gene are much smaller than 1 (near zero), then μ_{gi} will be very small and account for the high proportion of zeros typical of this data. Conversely, very large values of the factors will increase μ_{gi} (relative to the baseline β_g) and accommodate the occasional extremely large count. We note here that Y_{gi} follows a Poisson distribution conditional on the $\boldsymbol{\lambda_i}$ terms. However, the variance of Y_{gi}, marginal on $\boldsymbol{\lambda_i}$, is equal to $\beta_g + \beta_g^2\big(exp\{-\phi_f|\alpha_{gf}|\} - 1\big)$. Thus, Y_{gi} is conditionally Poisson but marginally

overdispersed. So, despite the choice of Poisson for the distribution of the count, our model is able to capture the high proportion of zeros and large variance typical of scRNA data.

To finish specification of our Bayesian model, prior distributions for the remaining parameters must be defined. We use a conditionally conjugate, non-informative prior for the average expression of gene g, $\beta_g \sim Gamma(0.001, 0.001)$. The hierarchical prior structure for the scale parameter of the factors is $\phi_f \sim Lognormal(h_1, h_2)$, where $h_1 \sim Normal(0, 100)$ and $h_2 \sim Inverse\ Gamma(1, 1)$. For the exponent parameters, the prior is $|\alpha_{gf}| \sim Bernoulli(\theta_f)$ with $\theta_f \sim Beta(1, 1)$. Here, we define $P(\alpha_{gf} = 1) = P(\alpha_{gf} = -1) = \frac{\theta_f}{2}$. Consequently, $P(\alpha_{gf} = 0) = 1 - \theta_f$. The number of associated factors F is often unknown, but one can fit multiple models with different numbers of factors and choose the most suitable model based on a comparison of a model selection statistic such as the Deviance Information Criterion (DIC) described in Gelman et al. (2004). Throughout the manuscript, we will refer to our hierarchical Bayesian factor model as HBFM.

3.2.2 Network Structure

Posterior samples for model parameters are obtained with the Markov chain Monte Carlo (MCMC) algorithm defined later in Section 3.2.3. At each iteration of the MCMC, a correlation matrix is computed based on the current set of parameters, and we infer a GCN by examining the posterior distribution of this correlation matrix. Under our proposed model, the sparse $\boldsymbol{\alpha} = \{\alpha_{gf}\}_{(g,f)}$ matrix imposes a crude network structure on the gene expressions. Consider two genes g and g', where $g \neq g'$. If $\alpha_{gf}\alpha_{g'f} \neq 0$ for some f, the expressions Y_{gi} and $Y_{g'i}$ are both impacted by the shared factor λ_{if}. Conversely, if genes g and g' have no shared factors ($\alpha_{gf}\alpha_{g'f} = 0$ for all f), these genes are conditionally independent. To quantify the association between gene g and gene g', we examine the correlation (after marginalizing out $\boldsymbol{\lambda_i}$) between

the values of $log(\mu_{gi})$ and $log(\mu_{g'i})$.

We motivate our decision to use this specific correlation structure by considering the matrix $\tilde{\boldsymbol{A}} = \boldsymbol{\alpha}\boldsymbol{\alpha}^{\boldsymbol{T}}$. The (g, g') element of this $G \times G$ matrix provides of a summation of the associated factors that are active in both genes g and g' since $\tilde{a}_{g,g'} = \sum_{f=1}^{F} \alpha_{gf}\alpha_{g'f}$. When $\tilde{a}_{g,g'} > 0$, the two genes have more factors with the same association (i.e., $\alpha_{gf} = \alpha_{g'f} = 1$ or $\alpha_{gf} = \alpha_{g'f} = -1$) than factors with opposite associations (i.e., $\alpha_{gf} = 1$ and $\alpha_{g'f} = -1$ or vice versa). Conversely, when $\tilde{a}_{g,g'} < 0$, the genes have more factors with opposite associations than factors with the same association. If $\tilde{a}_{g,g'} = 0$, then either no factors are in common between the genes or the number of factors with the same association is equal to the number of factors with opposite associations for those genes.

By recognizing that factors with a larger variance ϕ_f will have a greater influence on the joint expression, we can weigh the shared factors by their variance. In fact, this weighted expression is exactly equal to the the covariance (marginally over $\boldsymbol{\lambda_i}$) between $log(\mu_{gi})$ and $log(\mu_{g'i})$,

$$Cov\big[log(\mu_{gi}), log(\mu_{g'i})\big] = \sum_{f=1}^{F} \phi_f \alpha_{gf}\alpha_{g'f} \ .$$

The active factors also increase the variance for $log(\mu_{gi})$,

$$Var\big[log(\mu_{gi})\big] = \sum_{f=1}^{F} \phi_f \alpha_{gf}^2 \ ,$$

which is important when addressing the zeros and overdispersion of scRNA-seq data. From these covariance and variance expressions, the correlation between $log(\mu_{gi})$ and $log(\mu_{g'i})$ is defined as

$$Corr\big[log(\mu_{gi}), log(\mu_{g'i})\big] = \rho_{gg'} = \frac{\sum_{f=1}^{F} \phi_f \alpha_{gf} \alpha_{g'f}}{\sqrt{\Big(\sum_{f=1}^{F} \phi_f \alpha_{gf}^2\Big)\Big(\sum_{f=1}^{F} \phi_f \alpha_{g'f}^2\Big)}}. \tag{3.2}$$

We illustrate the mechanics of this correlation structure by considering just one factor f. If gene g and gene g' have the same association with this given factor, the correlation between $log(\mu_{gi})$ and $log(\mu_{g'i})$ is 1. When gene g has a positive association with factor f and gene g' has a negative association with factor f, the correlation is -1. Additionally, if factor f is inactive for either of the genes, the correlation is 0. The significance of each correlation is determined by analyzing the the credible interval (CI) of $\rho_{gg'}$ in the posterior distribution, as described in Section 3.2.4.

We note that each gene must have at least one active factor for our correlation structure in Equation (3.2) to be defined since $Var\big[log(\mu_{gi})\big]$ is equal to 0 if all of the factors are inactive. Utilizing the correlation structure (after marginalizing out $\boldsymbol{\lambda_i}$) between Y_{gi} and $Y_{g'i}$ would avoid this issue, but the additional β_g term in the variance leads to a correlation structure dependent on the average expression for each gene. For this reason, we do not focus on the correlation structure between Y_{gi} and $Y_{g'i}$. Throughout, if (3.2) is $\frac{0}{0}$, we define this correlation as zero to match the zero value for $Corr(Y_{gi}, Y_{g'i})$.

3.2.3 Model Inference

The posterior distribution for our hierarchical Bayesian model is complex, and so MCMC is required for inference. For simplicity in our posterior distribution notations, let $\psi_{gif} = \prod_{f' \neq f} exp\big\{ -\frac{\phi_{f'}}{2}|\alpha_{gf'}|\big\}\lambda_{if'}^{\alpha_g f'}$. We utilize an MCMC sampler that iterates through the following steps:

1. For $g = 1, \ldots, G$, update

$$\beta_g \sim Gamma\big(0.001 + \textstyle\sum_{i=1}^{N} y_{gi}\ ,\ 0.001 + \sum_{i=1}^{N} \prod_{f=1}^{F} exp\big\{ -\frac{\phi_f}{2}|\alpha_{gf}|\big\}\lambda_{if}^{\alpha_g f}\ \big).$$

2. For $f = 1, \ldots, F$, update $\theta_f \sim Beta\big(1 + \sum_{g=1}^{G} |\alpha_{gf}| \ , \ 1 + G - \sum_{g=1}^{G} |\alpha_{gf}|\big)$.

3. For all g, f, sample α_{gf} from a multinomial distribution with

   $p(\alpha_{gf} = 0 | \cdots) = \frac{A}{A+B+C}$,

   $p(\alpha_{gf} = 1 | \cdots) = \frac{B}{A+B+C}$,

   $p(\alpha_{gf} = -1 | \cdots) = \frac{C}{A+B+C}$.

   Here, A, B, and C are defined as

   $A = (1 - \theta_f) exp\big\{ - \beta_g \sum_{i=1}^{N} \psi_{gif} \big\}$,

   $B = \big(\frac{\theta_f}{2}\big) exp\big\{ - \beta_g \sum_{i=1}^{N} exp\big\{ - \frac{\phi_f}{2} \big\} \lambda_{if} \psi_{gif} \big\}$,

   $C = \big(\frac{\theta_f}{2}\big) exp\big\{ - \beta_g \sum_{i=1}^{N} \frac{exp\{-\frac{\phi_f}{2}\}}{\lambda_{if}} \psi_{gif} \big\}$.

4. Update $h_1 \sim Normal(\frac{1/h_2}{1/100 + F/h_2} * \sum_{f=1}^{F} log(\phi_f), (1/100 + F/h_2)^{-1})$.

5. Update $h_2 \sim Inverse\ Gamma(\frac{F}{2} + 1, \frac{\sum_{f=1}^{F} (log(\phi_f) - h_1)^2}{2} + 1)$.

6. For $f = 1, \ldots, F$, use a Metropolis-Hastings step to update ϕ_f. The posterior distribution for ϕ_f is

   $$p(\phi_f | \cdots) \propto \phi_f^{-\frac{N}{2} - 1} exp\Big\{ - \Big(\frac{\phi_f}{2} \sum_{g=1}^{G} \sum_{i=1}^{N} |\alpha_{gf}| y_{gi} + \frac{\sum_{i=1}^{N} log(\lambda_{if})^2}{2\phi_f} + \frac{(log(\phi_f) - h_1)^2}{2h_2} + \sum_{g=1}^{G} \beta_g exp\big\{ - \frac{\phi_f}{2} |\alpha_{gf}| \big\} \sum_{i=1}^{N} \lambda_{if}^{\alpha_{gf}} \ \psi_{gif} \Big) \Big\}.$$

   We propose a candidate value for $\phi_f^{(c)}$ through a pseudo-random walk from Lognormal(ϕ_f, σ^2) and accept this value with the usual Metropolis-Hastings ratio. If factor f is not active for any gene (i.e., $\sum_{g=1}^{G} |\alpha_{gf}| = 0$), then update ϕ_f from the Lognormal(h_1, h_2) prior.

7. For all i, f, use a Metropolis-Hastings step to update λ_{if}. By defining

   $\kappa = \sum_{g=1}^{G} y_{gi} \alpha_{gf}$,

   $\tau = 2 \sum_{g=1}^{G} I(\alpha_{gf} = 1) \beta_g \ exp\big\{ - \frac{\phi_f}{2} \big\} \psi_{gif}$,

   $\chi = 2 \sum_{g=1}^{G} I(\alpha_{gf} = -1) \beta_g \ exp\big\{ - \frac{\phi_f}{2} \big\} \psi_{gif}$,

where $I(\cdot)$ represents an indicator variable, the posterior distribution for λ_{if} is $p(\lambda_{if}|\cdots) \propto \lambda_{if}^{\kappa-1} exp\Big\{ -\frac{1}{2}\Big(\tau\lambda_{if} + \frac{\chi}{\lambda_{if}} + \frac{log(\lambda_{if})^2}{\phi_f}\Big)\Big\}$.

This posterior has a similar appearance to a generalized inverse Gaussian (GIG) distribution with an extra exponential term $\big(\frac{log(\lambda_{if})^2}{\phi_f}\big)$. To that end, we propose a candidate value for $\lambda_{if}^{(c)}$ from GIG$(\kappa, b\tau, b\chi)$, where the multiplicative factor of b on τ and χ is used to create thicker tails in the proposal distribution. For our sampling scheme, we set b to 0.9. Acceptance of the candidate value is determined by the typical Metropolis-Hastings rules. If $\tau = \chi = 0$, factor f is not active and we update λ_{if} from the Lognormal$(0, \phi_f)$ prior.

Due to the large number of model parameters and complexity of the posterior distribution, it is possible for the MCMC sampler to get stuck exploring a local mode of the posterior rather than exploring the entire posterior distribution. This is particularly an issue with the one-at-a-time sampling for $\boldsymbol{\alpha}$, which does not allow for large scale moves such as splitting or combining factors. To address this sampling problem, we implement a stochastic EM approach (Celeux et al., 1996; Bhattacharya and Dunson, 2011) to obtain initial values for our MCMC algorithm.

For the stochastic EM approach, we run the usual MCMC sampler but replace sampling with optimization in several of the steps. Specifically, we optimize the following steps of the sampler:

1. For $g = 1, \ldots, G$, update β_g to its conditional posterior mode.

3. For all g, f, select the value of α_{gf} with the highest probability: $p(\alpha_{gf} = 0|\cdots)$, $p(\alpha_{gf} = 1|\cdots)$, or $p(\alpha_{gf} = -1|\cdots)$.

6. For $f = 1, \ldots, F$, find ϕ_f that optimizes its respective conditional posterior distribution. In this step, we utilize the *optimize* function from the base packages in R (R Core Team, 2018).

After randomly selecting starting values and running an initial MCMC sampling warm-up period, the stochastic EM approach is implemented for a number of iterations (e.g., 2000 iterations) to ensure stabilization. Parameter estimates are then calculated by averaging the samples generated from a final set of iterations (e.g., the samples from the last 200 iterations). In the case of the discrete α_{gf} parameters, we select the value (either -1, 0, or 1) that has the highest frequency. The parameter estimates from this stochastic EM approach are then input as the initial starting values of our MCMC sampler. We choose to run a number of MCMC chains (in parallel) and implement the stochastic EM approach individually for each chain to produce different initial starting values. For final parameter inference, the lowest performing chains (i.e., the chains with the lowest marginal likelihoods) are discarded from analysis.

3.2.4 Network Inference

The association level network structure $\tilde{\boldsymbol{N}} = \{\tilde{n}_{gg'}\}_{(g,g')}$ between genes is obtained by analyzing the posterior of the correlation matrix defined in Equation (3.2). For each (g, g') element in the correlation matrix, M samples are used to calculate the posterior mean $\widehat{\rho}_{gg'} = \frac{1}{M}\sum_{m=1}^{M} \rho_{gg'}^{(m)}$. This estimate provides a quantifiable value of association between genes g and g'.

Since we are working in the Bayesian paradigm, we can examine the CI of the posterior to determine whether or not genes g and g' are associated with one another. By choosing an appropriate level of significance α^*, two genes have a significant association when zero is excluded from the $100(1-\alpha^*)\%$ CI. A second method to determine significant associations from the posterior samples of $\rho_{gg'}$ is to find the smallest $100(1-\mathrm{a}^*)\%$ CI that includes 0. The corresponding a^* value would indicate the proportion of the posterior distribution outside of the smallest CI that includes 0. Hence, we can think of a^* as an approximate "p-value" that can be used to rank

correlations by significance.

3.3 Results

3.3.1 Datasets

To demonstrate the feasibility of our methodology, we generated simulated datasets consistent with our proposed methodology structure. Each Y_{gi} count was sampled from Poisson(μ_{gi}), with μ_{gi} modeled from Equation (3.1). The β_g parameters were randomly sampled from Gamma(3,0.5) and the λ_{if} parameters were randomly sampled from Lognormal$(0, \phi_f)$.

For the network structures, we fixed the values of the $\boldsymbol{\alpha}$ matrix. In each dataset, we considered $G = 50$ genes and sorted them into ten groups of five (e.g., Group 1 consisted of genes 1 - 5, Group 2 consisted of genes 6 - 10), and all genes within each factor group were assigned the same α_{gf} values. In three of the datasets, we considered the same network structure (Figure 3.1A) consisting of 350 "true" edges using $F_{sim} = 10$ factors and varied the number of cells to be either $N = 125$ (Sim 1), $N = 500$ (Sim 3), or $N = 1,000$ (Sim 5). In the other three datasets, we utilized a network structure of $F_{sim} = 15$ factors to simulate expression values, which created a network structure with 425 "true" edges (Figure 3.1C). Again, the number of cells were set to either $N = 125$ (Sim 2), $N = 500$ (Sim 4), or $N = 1,000$ (Sim 6). In order to define the correlation structures, the values of ϕ_f were fixed to be either 0.2, 0.35, 0.5, 0.65, or 0.8. In the simulations with $F_{sim} = 10$, each fixed value of ϕ_f was used twice (e.g., $\phi_1 = \phi_2 = 0.2$, $\phi_3 = \phi_4 = 0.35$) and in the simulation with $F_{sim} = 15$, each fixed value was used three times (e.g., $\phi_1 = \phi_2 = \phi_{11} = 0.2$, $\phi_3 = \phi_4 = \phi_{12} = 0.35$).

We also ran analyses on two real datasets to demonstrate the utility of our method on real data. The expression counts for the mouse brain single-cell (MBSC)

dataset from Zeisel et al. (2015) were downloaded from the Gene Expression Omnibus (GEO) database under accession number GSE60361. For this analysis, we selected the $G = 48$ known and novel genetic markers displayed in Figure S6 of the supplementary materials of Zeisel et al. (2015) and cells with a library size of zero were removed, leaving a total of $N = 2,946$ cells in the dataset. The second dataset was obtained from the GEO database under accession number GSE90975 and contains the gene expressions from single-cell analysis of neurodegeneration in microglia cells of mice (Tay et al., 2018). We considered all $N = 944$ cells and analyzed the $G = 101$ differentially expressed genes from Figure S1 of Tay et al. (2018). This second real dataset is referred to as the mouse microglia cell (MMC) data.

3.3.2 Simulation Studies

Using the simulated data, we fit our proposed model (HBFM) by running the MCMC sampling algorithm described in Section 3.2.3. The stochastic EM approach was run for $2,000$ iterations, after an initial warm-up period of 100 iterations, and samples from the last 200 iterations of this approach were used to obtain starting parameter values for the MCMC sampler. We ran the MCMC sampler for $4,000$ iterations and used the last $1,000$ iterations for inference.

Nine runs of HBFM were considered by selecting nine different choices for the number of factors: $F = 5$, 8, 10, 12, 15, 18, 20, 22, and 25. For each choice of F, we ran eight separate MCMC sampling chains in R (R Core Team, 2018), and used only the samples from the five chains with the highest average marginal likelihood for inference. DIC was calculated using half the posterior variance of the deviance to estimate the effective number of parameters Gelman et al. (2004), and the number of factors F with the lowest DIC was selected as the "best" model choice. In the cases where $F = 25$ was chosen as the "best" model, we ran an additional model with $F = 28$ factors to ensure that the upper bound of our considered set was also

the optimal choice for the number of factors. For each pair of genes g and g' in the "best" model, we tested for a significant relationship by using a 95% CI for $\rho_{gg'}$.

To evaluate the performance of our model against other gene network methods, we ran the single-cell co-expression model LEAP (Specht and Li, 2016) and the single-cell regulatory network models of PIDC (Chan et al., 2017) and SCODE (Matsumoto et al., 2017) on the simulated data. After creating a symmetric correlation matrix with the LEAP package in R, a permutation analysis was then performed with this package using a false discovery rate (FDR) of 5% to determine a cutoff for significant correlation values. PIDC was implemented in Julia (Bezanson et al., 2017) using the basic usage code available at https://github.com/Tchanders/NetworkInference.jl. For SCODE, we ran the R code available at https://github.com/hmatsu1226/SCODE and averaged the results of 50 separate trials using the same parameters as the example code provided on the GitHub page. The methods of LEAP and SCODE utilize a pseudotime estimation of the cells and the R package monocle (Trapnell et al., 2014) was used for this estimation.

We also included three popular network methods originally developed for bulk data in our simulation studies: partial correlation, Bayesian networks, and GENIE3 (Huynh-Thu et al., 2010). Partial correlation (PCORR) was implemented with the R package ppcor (Kim, 2015) using the Spearman partial correlation coefficient. We performed the Benjamini-Hochberg (Benjamini and Hochberg, 1995) procedure to control for FDR and defined 5% as the threshold for significant correlation values. Bayesian networks (BN) were constructed in R with the bnlearn package (Scutari, 2010). After learning a set of 1,000 bootstrap replicates with the hill-climbing algorithm, the optimal network was created using model averaging (Scutari, 2010). The analysis for GENIE3 was performed in R with the GENIE3 package using default parameters.

The methods of PIDC, SCODE, and GENIE3 output a matrix of scores/weights

to indicate how likely each gene-gene regulatory link is, but these methods do not determine a cutoff score/weight for identifying significant associations. To be consistent across the simulated datasets, we chose the threshold for PIDC, SCODE, and GENIE3 such that the number of edges in the constructed network was equal to the number of edges determined by our HBFM method. By matching the number of edges to our method, we provide a direct comparison between these methods and HBFM. In addition, SCODE and GENIE3 provide different scores/weights for the different directions of edges in the network; therefore, we selected the directed edges with the higher magnitude to quantify the strengths of the gene-gene associations for these methods.

For each simulated dataset, we compared the significant gene-gene associations identified by each method to the "true" gene-gene associations created by the simulated network structure. The measures of true positive rate (TPR), FDR, area under the receiver operating characteristic curve (AUC), and number of significant edges in the estimated network were used to compare methods. When calculating the AUC, the regulatory link score/weight was used for PIDC, SCODE, and GENIE3, the association strength was considered for BN, and for the remaining methods the inverse of the adjusted p-value (inverse of the approximate "p-value" in HBFM) was utilized. We note that selecting a different threshold for PIDC, SCODE, and GENIE3 may impact the TPR and FDR results since the number of edges in the constructed network will change; however, the AUC results will remain unchanged by the threshold choice. We found that the FDRs for SCODE and GENIE3 tend to remain fairly stable across different threshold choices, and the FDR of PIDC tends to increase as the threshold increases. The performances of the different network methods are summarized in Table 3.1.

From the simulation results, we see that our methodology performs quite well across the different scenarios, as HBFM has consistently high power and low FDRs.

The heatmaps of estimated correlation structures produced by HBFM resemble the "true" structure of the simulated datasets (Figure 3.1). Our model outperforms the other methods across the TPR and AUC performance measures in these simulation studies. In Sim 6, HBFM and PIDC perform very comparably when the number of edges is the same. While PIDC has a slightly higher TPR and lower FDR at this threshold, HBFM does have the higher AUC. The FDR of our method is also reasonably controlled to a nominal level, especially compared to the FDRs of LEAP, SCODE, GENIE3, and PCORR. While BN had lower FDRs than HBFM in some of the simulations, it also identified the fewest number of edges and had lower TPR and AUC than HBFM.

When using DIC as the criterion for our model selection, the best-fitting model often contains more factors than the "true" simulated structure in the examples we've considered so far. However, we note that the additional factors provide more opportunities to explore different factor structures within the model during MCMC sampling. For example, a single factor from a model with $F = 10$ may be split into several factors when using a model with $F = 20$. Therefore, it is not surprising that the "best" model choices contain more factors than the "true" number of factors, F_{sim}, as these models are more likely to explore the high regions of the posterior as they are less likely to get stuck during sampling.

3.3.3 Case Studies

The same network methods described in Section 3.3.2 were applied to the two real datasets. For each method, we constructed a network and obtained the top 100 most significant gene-gene pairs, out of the 1,128 possible pairs, for comparison in the MBSC analysis and the top 500 most significant pairs, out of the 5,050 possible pairs, for comparison in the MMC analysis (approximately the top 10% associations for each dataset). From the nine different numbers of factors considered for HBFM,

we selected $F = 25$ factors as the "best" choice for both the MBSC and MMC data because this factor choice had the lowest DIC.

Since the "true" network structure of this real data is unknown, we constructed three reference protein-protein interaction networks with the STRING database (Szklarczyk et al., 2014) for each dataset to compare across the different methods. These reference networks were created by adjusting the threshold for the minimum required interaction score between pairs of proteins: high confidence (minimum score of 0.700), medium confidence (minimum score of 0.400), and low confidence (minimum score of 0.150). STRING computes these scores by combining the probabilities of different evidence sources (e.g., text mining, experiments, databases) and correcting for the probability of observing the interactions by random chance (von Mering et al., 2005). This is, of course, an imperfect reference as any method may detect novel interactions that have not been previously published. Likewise, some entries in STRING may represent published false positives. However, on average, the method producing the GCN most similar to the STRING reference set should be considered as the network most consistent with biological literature.

The UpSet plots (Lex et al., 2014) for the intersection between the the top 100 associations in the MBSC dataset and the top 500 associations in the MMC dataset identified by each network method is displayed in Figure 3.2. Interestingly, each method identifies a number of unique associations with only 3 and 10 associations in common among all seven methods in the MBSC and MMC datasets, respectively. Table 3.2 displays the comparisons of the top associations from each method to the reference networks. We see that HBFM has the highest number of associations in common with each STRING reference network. This is particularly apparent in the MMC dataset, where over 80% of the top 500 genes pairs detected by our method matched with the STRING reference sets. Less than half of the top 500 gene pairs from the other methods overlapped with the high and medium MMC reference sets.

When comparing with the low MMC reference set, HBFM matched 94.8% of the top 500 identified gene pairs. PIDC had the second highest overlap with the low MMC reference set matching only 64.8% of its top 500 gene pairs. Based on these results, our methodology is able to identify more known protein-protein interactions in these real datasets than the other network methods.

We also evaluated our HBFM model by creating 100 posterior predictive datasets (PPDs; Gelman et al., 2004) from each chain of the MMC analysis (500 PPDs in total) and comparing the overdispersion and proportion of zeros in these datasets to the overdispersion and proportion of zeros in the MMC dataset. Each count Y_{gi} of the PPDs was generated from Poisson(μ_{gi}), with μ_{gi} modeled from Equation (3.1) using parameter estimates (with the exception of the $\boldsymbol{\lambda_i}$ parameters) from different iterations of the MCMC sampler. The $\boldsymbol{\lambda_i}$ values were drawn randomly from Lognormal$(0, \phi_f)$.

In Figure 3.3A, the log(variance) is plotted against the log(mean) across all $G = 101$ genes for the real expressions in the MMC dataset and the estimated expressions from a single representative PPD. Both datasets display high cell-to-cell variability, as expected of scRNA-seq data. In fact, even with the choice of Poisson for the (conditional) distribution of the counts, the PPDs generated from the parameters estimated from the MMC dataset tend to generate variability that is comparable to the variability observed in the real data. We can see that many genes from the PPD are overdispersed, especially those with log(means) greater than 1, as in the true MMC data. From Figure 3.3B, the gene expression in the MMC data is zero-inflated as the proportion of zero values for each gene ranged between 0 and 0.99. In the PPD, the proportion of zeros for each gene tended to be only slightly lower than what was observed in the real dataset. Nevertheless, the proportion of zero expressions were still quite high and variable across the genes in the PPD.

To further evaluate the PPDs generated by HBFM, we selected nine genes from

the MMC dataset that represent the 10th through 90th percentiles of average gene expression and examined the log(variance/mean) and proportion of zeros of these genes across all PPDs. Figure 3.4A illustrates that across the PPDs, the estimated log(variance/mean) for most of the genes is greater than 0, indicating variances that are larger than their corresponding means. Also, for a majority of these genes, the true log(variance/mean) value is captured across the PPD estimates. The estimated proportion of zeros for these genes across the PPDs also capture the true proportion of zeros from the MMC dataset, as displayed in Figure 3.4B.

3.4 Discussion

In this manuscript, we have presented a hierarchical Bayesian factor model (which we have referred to as HBFM) for constructing GCNs from scRNA-seq data. The results from our simulation studies demonstrate that HBFM is able to identify true co-expressions while maintaining a nominal FDR across different numbers of cells and different network structures. Our case study analyses with the MBSC and MMC datasets also demonstrate the practical use of HBFM for determining significant gene-gene associations, as our model was able to detect more known protein-protein interactions than the other network methods.

The number of genes (G) in the simulated and real datasets presented in this manuscript is smaller than what is often considered for other scRNA-seq data problems, such as clustering cells/genes and detection of differentially expressed genes. However, the use of a smaller pre-screened set of genes is common among other complex network methods (Fiers et al., 2018; Delgado and Gómez-Vela, 2018). In part, this is due to the GCN being determined by $G * (G-1)/2$ correlations, a quadratic number of parameters, making it difficult to numerically and graphically communicate results for large G. While constructing a GCN as an exploratory analysis from an entire dataset is possible with our method, it may not be computationally practical.

HBFM performs Bayesian inference via iterative MCMC, which can become computationally expensive as the number of genes (G) and number of cells (N) increase. In light of these computational considerations, we typically recommend the user consider some initial analysis such as clustering or differential expression to determine a smaller set of genes, generally 100 or fewer, before using HBFM to estimate the GCN. On a system with an Intel Core i7 processor (3.5 GHz) and 8 GB of RAM, the running time for a single chain of HBFM with $F = 25$ factors was 32.7 hours for the MBSC data ($G = 48, N = 2,946$) and 22.1 hours for the MMC data ($G = 101, N = 944$).

In our methodology, the distribution of count values is defined to follow a Poisson distribution, conditional on the latent factors $\boldsymbol{\lambda_i}$. While we acknowledge that the Negative Binomial distribution tends to be the preferred choice for modeling overdispersed data, the latent factors of HBFM are random effects that help account for the additional variability across samples. After marginalizing out $\boldsymbol{\lambda_i}$, $E(Y_{gi}) = \beta_g$ and $Var(Y_{gi}) = \beta_g + \beta_g^2\big(exp\{-\phi_f|\alpha_{gf}|\} - 1\big)$. As illustrated in the PPDs generated from the real MMC data, HBFM is able to generate overdispersed and zero-inflated data that is consistent with the features of the real data. Hence, the use of a Poisson distribution is not a meaningful drawback.

We also note that the high resolution of scRNA-seq technology allows researchers the opportunity to estimate "pseudotime" and obtain a temporal ordering of cells (Trapnell et al., 2014; Street et al., 2018). The general idea is that at any given time, a cell population will consist of cells that are at different stages of their cell cycles, and cells in different stages will express different sets of genes. Our method does not directly take pseudotime into account, but the latent factors ($\boldsymbol{\lambda_i}$'s) are likely to adapt and capture this contribution on the gene expression.

3.4.1 Software Availability

The source code for implementing the HBFM model is available as an R package at https://github.com/mnsekula/hbfm.

3.5 Tables and Figures

	Sim 1: N=125, F_{sim}=10					Sim 2: N=125, F_{sim}=15			
	TPR	**FDR**	**AUC**	**Edges**		**TPR**	**FDR**	**AUC**	**Edges**
HBFM, F = 15	0.760	0.153	0.927	314	**HBFM, F = 15**	0.640	0.111	0.820	306
LEAP	0.386	0.378	0.705	217	**LEAP**	0.341	0.275	0.665	200
PIDC	0.634	0.293	0.821	314*	**PIDC**	0.506	0.297	0.742	306*
SCODE	0.229	0.745	0.550	314*	**SCODE**	0.249	0.654	0.504	306*
BN	0.206	0.077	0.682	78	**BN**	0.186	0.037	0.672	82
GENIE3	0.540	0.398	0.746	314*	**GENIE3**	0.468	0.350	0.711	306*
PCORR	0.123	0.566	0.599	99	**PCORR**	0.148	0.442	0.602	113
	Sim 3: N=500, F_{sim}=10					**Sim 4: N=500, F_{sim}=15**			
	TPR	**FDR**	**AUC**	**Edges**		**TPR**	**FDR**	**AUC**	**Edges**
HBFM, F = 25	0.889	0.034	0.984	322	**HBFM, F = 25**	0.704	0.029	0.929	308
LEAP	0.743	0.608	0.741	664	**LEAP**	0.402	0.305	0.696	246
PIDC	0.794	0.137	0.915	322*	**PIDC**	0.621	0.143	0.866	308*
SCODE	0.246	0.733	0.503	322*	**SCODE**	0.226	0.688	0.575	308*
BN	0.277	0.040	0.751	101	**BN**	0.212	0.032	0.716	93
GENIE3	0.554	0.398	0.754	322*	**GENIE3**	0.466	0.357	0.729	308*
PCORR	0.300	0.266	0.683	143	**PCORR**	0.261	0.327	0.624	165
	Sim 5: N=1000, F_{sim}=10					**Sim 6: N=1000, F_{sim}=15**			
	TPR	**FDR**	**AUC**	**Edges**		**TPR**	**FDR**	**AUC**	**Edges**
HBFM, F = 20	0.909	0.076	0.973	344	**HBFM, F = 25**	0.624	0.070	0.904	285
LEAP	0.780	0.550	0.804	606	**LEAP**	0.591	0.541	0.680	547
PIDC	0.857	0.128	0.954	344*	**PIDC**	0.633	0.056	0.889	285*
SCODE	0.269	0.727	0.496	344*	**SCODE**	0.221	0.670	0.510	285*
BN	0.323	0.050	0.793	119	**BN**	0.247	0.037	0.710	109
GENIE3	0.603	0.387	0.764	344*	**GENIE3**	0.440	0.344	0.700	285*
PCORR	0.403	0.291	0.720	199	**PCORR**	0.294	0.251	0.669	167

* Number of edges fixed to match HBFM.

Table 3.1: Results from simulation studies. The value of F for HBFM represents the number of factors in the "best" model choice, as determined by DIC.

	MBSC reference set			MMC reference set		
	High	**Medium**	**Low**	**High**	**Medium**	**Low**
HBFM, F=25	10	23	39	416	446	474
LEAP	9	20	34	78	121	252
PIDC	7	17	38	151	197	324
SCODE	5	14	31	42	75	165
BN	3	16	36	162	208	319
GENIE3	4	14	37	99	142	290
PCORR	6	15	32	116	154	236
Reference total	42	116	322	697	897	1600

Table 3.2: The overlap between the top 100 gene-gene associations in the MBSC dataset and the top 500 gene-gene associations in the MMC dataset for each network method. Reference networks were created by the STRING database.

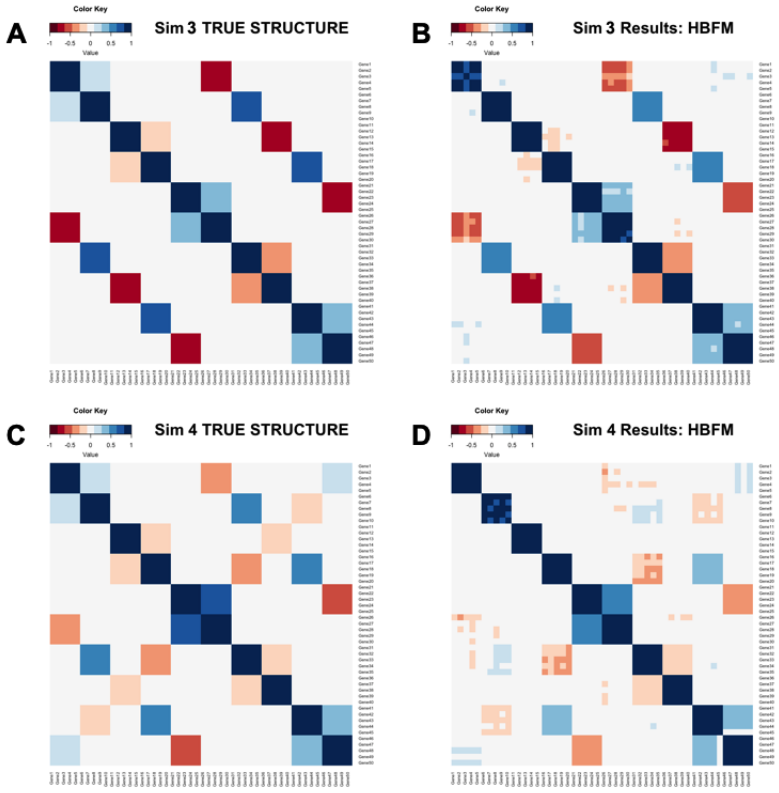

Figure 3.1: (A) Heatmap of the "true" correlation structure in Sim 3 ($F = 10, N = 500$). (B) Heatmap of the estimated correlation structure in Sim 3 by HBFM and $F = 25$ factors. (C) Heatmap of the "true" correlation structure in Sim 4 ($F = 15, N = 500$). (D) Heatmap of the estimated correlation structure in Sim 4 by HBFM and $F = 25$ factors.

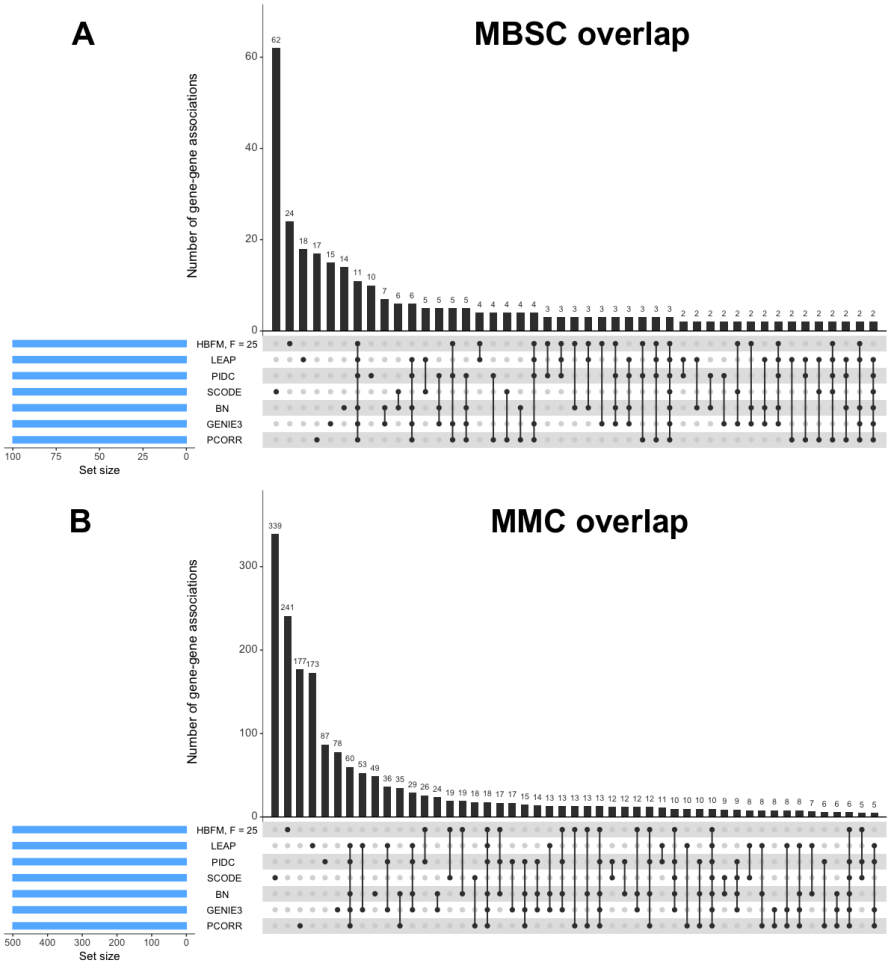

Figure 3.2: (A) UpSet plot of the top 100 gene-gene associations as determined by seven different methods for the MBSC dataset. (B) UpSet plot of the top 500 gene-gene associations as determined by seven different methods for the MMC dataset.

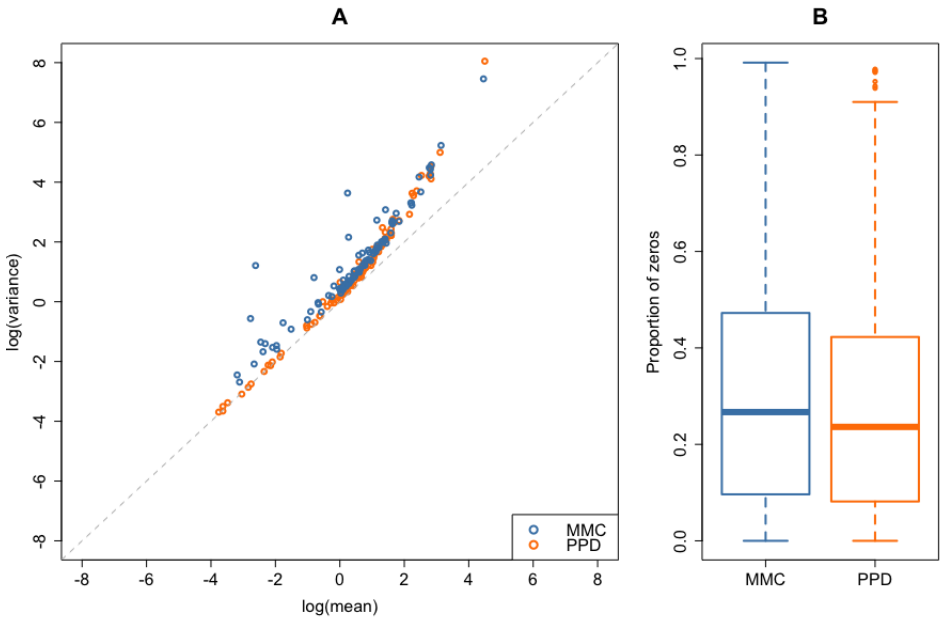

Figure 3.3: Example comparison between the MMC dataset and one representative PPD generated by HBFM. (A) The log(variance) vs. log(mean) scatterplot for each gene. (B) Boxplots of gene-specific proportion of zeros.

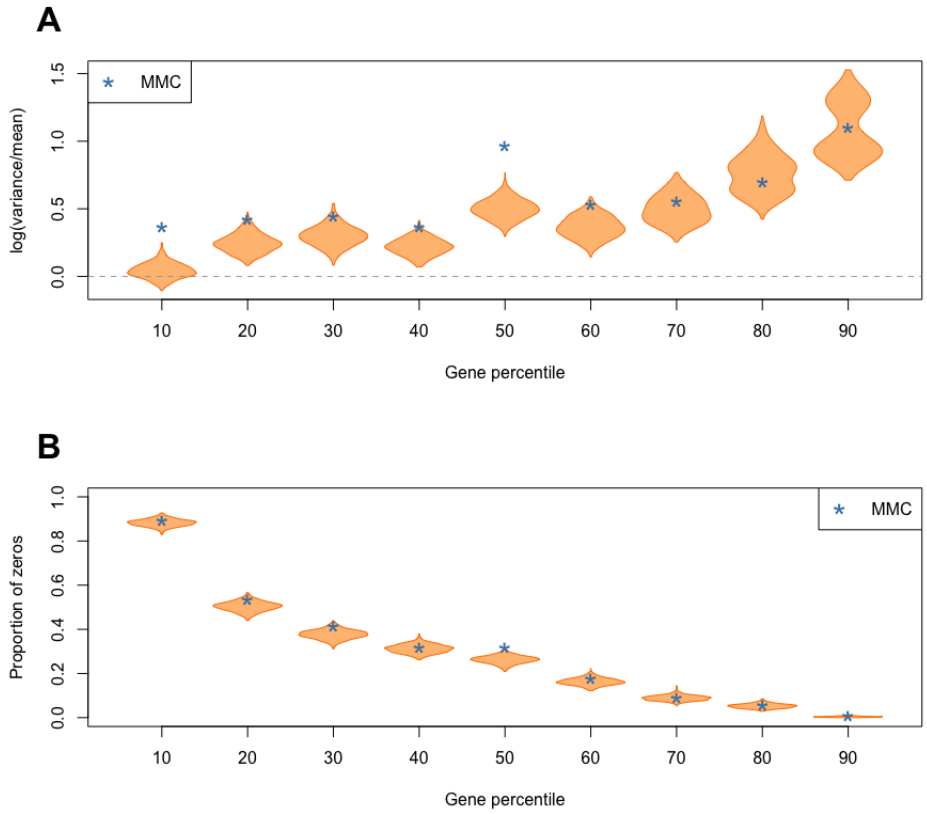

Figure 3.4: Properties of PPD estimates from a sample of nine genes in the MMC dataset. Genes were selected based on percentiles (10th through 90th) of average gene expression. (A) Violin plots of estimated log(variance/mean) for each gene across all PPDs. (B) Violin plots of estimated gene-specific proportion of zeros across all PPDs. The blue stars represent the true values from the MMC dataset.

CHAPTER 4

SINGLE-CELL DIFFERENTIAL NETWORK ANALYSIS WITH SPARSE BAYESIAN FACTOR MODELS

4.1 Introduction

Gene network modeling has become essential to the understanding of complex biological systems related to health and disease. These networks allow researchers to uncover and interpret relationships and interactions between genes during different biological processes (Blencowe et al., 2019). There are several popular methods for constructing gene networks from microarray and bulk RNA sequencing data (Margolin et al., 2006; Langfelder and Horvath, 2008; Huynh-Thu et al., 2010), and more recently, methods for identifying gene networks from single-cell RNA sequencing (scRNA-seq) data have also been proposed (Specht and Li, 2016; Chan et al., 2017; Matsumoto et al., 2017) including our methodology for scRNA-seq gene network inference from Chapter 3. Interestingly, the vast majority of these methods have focused only on analyzing gene expressions from one cellular population, such as a single tissue type, disease, or environmental condition.

Since biological systems are highly dynamic, there is also great interest in performing differential network analysis to examine the changes in network structure under different biological settings. In the context of bulk population data (i.e., microarray and bulk RNA sequencing), efforts have been made to develop different strategies for identifying differences between gene-gene networks. Some approaches propose qualitative analyses through visual inspection of different network topologies

(Caldana et al., 2011; Weston et al., 2011), while others rely on statistical tests to determine differences across conditions (Choi and Kendziorski, 2009; Gill et al., 2010; Fukushima, 2013). For scRNA-seq data, however, there has been very little research in developing methods to compare gene networks from two (or more) biological conditions. Wang et al. (2017) present several proof-of-concept analyses for comparing network structures constructed from scRNA-seq data by utilizing a differential connectivity test that was originally developed by Gill et al. (2010) for microarray gene expression data. In Chiu et al. (2018), the scRNA-seq-based differential network (scdNet) analysis method is proposed to first determine a sample size corrected gene-gene correlation matrix for each cellular state and then identify differential gene-gene pairs across these states. The developers of scdNet state that, to their knowledge, their method is the first tool for differential network analysis of scRNA-seq data.

In this work, we adapt our hierarchical Bayesian factor methodology for constructing gene co-expression networks (GCNs) from scRNA-seq data to explore differences in the network structure across various cell groups due to different biological conditions, cell types, cell stages, or other group choice. The key adjustment in this new model is that the parameters that determine which factors are activated in a gene are now treatment-dependent to allow for the calculation of gene-gene co-expression within each treatment group. For simplicity, we consider a two-group setting and refer to these groups as treatment and control, but our model can easily be extended to a multiple group scenario, if necessary.

The rest of this manuscript is organized as follows. We define our proposed model and inference for differential network analysis in Section 4.2. Results from simulation studies are presented in Section 4.3 to demonstrate the performance of our methodology. In Section 4.4, we conclude with a brief discussion on our results.

4.2 Methods

4.2.1 Hierarchical Bayesian Factor Model for Two Treatment Groups

Let Y_{gi} be the expression count of gene g ($g = 1, \ldots, G$) in cell i ($i = 1, \ldots, N$) for treatment $t \in \{0, 1\}$. We define $t = 0$ as the control (reference) group and $t = 1$ as the treatment group. Like our model in Chapter 3, we assume that each expression comes from the Poisson(μ_{gi}) distribution, but here, we model the log-mean $log(\mu_{gi})$ through the representation

$$log(\mu_{gi}) = \beta_g + t_i\delta_g + \sum_{f=1}^{F} \lambda_{if}\alpha_{gf;t_i} - \Big\{ \sum_{f=1}^{F} \frac{\alpha_{gf;t_i}^2}{2} \Big\}. \tag{4.1}$$

For notation purposes in Equation (4.1), t_i indicates the treatment group ($t_i \in \{0, 1\}$) for cell i. Marginally over $\boldsymbol{\lambda_i}$, the parameter β_g denotes the log-mean expression for gene g in the control group and $\beta_g + \delta_g$ is the log-mean expression for gene g in the treatment group. Hence, δ_g represents the log-fold change in the expression for gene g. For each cell i, there are F associated factors $\boldsymbol{\lambda_i} = \{\lambda_{i1}, \ldots, \lambda_{iF}\}$ that impact the expression. Each factor can be thought of as some unique cellular attribute (e.g., cell stage, pseudotime point) that will only affect a specific set of related gene expressions. Since we are defining our model on the log scale, we assume these factors come from a Normal$(0, 1)$ distribution.

The magnitude of the impact by factor f on gene g in treatment t is influenced by the parameter $\alpha_{gf;t} \in \mathbb{R}$. With this setup, the expression for gene g in treatment t is minimally impacted by factors with $\alpha_{gf;t}$ values close to 0 and greatly impacted by factors with absolute values of $\alpha_{gf;t}$ much greater than 0. It is important to note that the $\alpha_{gf;t}$'s are treatment dependent to allow factors to impact the gene expressions differently across the treatments. Clearly if $\alpha_{gf;0}$ and $\alpha_{gf;1}$ have similar values, then factor f has a similar influence on the gene expression in both treatments. However,

the more interesting case is when $\alpha_{gf;0}$ and $\alpha_{gf;1}$ have very different values, which indicates a difference in the impact of factor f on gene g between the groups. By examining the differences between the $\boldsymbol{\alpha_t} = \{\alpha_{gf;t}\}_{(g,f)}$ matrices, we can identify differences between the gene networks of the treatment groups.

For most factors, we assume that the values of $\alpha_{gf;0}$ and $\alpha_{gf;1}$ in our model will be similar. In other words, we anticipate that most factors will have a similar impact on the genes within both groups. We also expect each factor f to impact only a small number of genes, and so the $\boldsymbol{\alpha_t}$ matrices will be sparse. To that end, we define the following hierarchy on the $\alpha_{gf;t}$ parameters:

$$\alpha_{gf;t} \sim Normal(\widetilde{\alpha}_{gf}, \kappa^2_{gf;t}\tau^2_f) \ , \tag{4.2}$$

$$\kappa_{gf;t} \sim \textit{half-Cauchy}(0,1) \ ,$$

$$\tau_f \sim \textit{half-Cauchy}(0,1) \ ,$$

$$\widetilde{\alpha}_{gf} \sim Normal(0, \zeta^2) \ , \tag{4.3}$$

$$\zeta \sim \textit{half-Cauchy}(0,1) \ .$$

Under this scheme, the horseshoe prior (Carvalho et al., 2009) placed on each $\alpha_{gf;t}$ in Equation (4.2) will help shrink the values of $\alpha_{gf;0}$ and $\alpha_{gf;1}$ together. For a given factor f, we define τ_f as the global shrinkage parameter and the $\kappa_{gf;t}$'s as the local shrinkage parameters. The global shrinkage parameter will pull the values of $\alpha_{gf;0}$ and $\alpha_{gf;1}$ towards $\widetilde{\alpha}_{gf}$, while the treatment-dependent local shrinkage parameters will allow some values to be much different than $\widetilde{\alpha}_{gf}$. Thus, the $\kappa_{gf;t}$'s can account for any variability between the groups. Our model favors borrowing information across treatments, so it should be efficient for factor-gene effects that are common. Nevertheless, the horseshoe priors allow big differences to accommodate differences between treatments.

To achieve more sparsity, a horseshoe prior could also be placed on the $\widetilde{\alpha}_{gf}$ parameters in Equation (4.3) to help shrink most of these values close to 0:

$$\widetilde{\alpha}_{gf} \sim Normal(0, \omega_{gf}^2 \zeta^2) \ , \tag{4.4}$$

$$\omega_{gf} \sim \textit{half-Cauchy}(0, 1) \ ,$$

$$\zeta \sim \textit{half-Cauchy}(0, 1) \ .$$

Here, ζ is a global shrinkage parameter that will pull the values of $\widetilde{\alpha}_{gf}$ towards 0. In Equation (4.4), we introduce local shrinkage parameters (ω_{gf}'s) to allow some of the $\widetilde{\alpha}_{gf}$ values to be much different than 0. Therefore, the horseshoe priors on the $\alpha_{gf;t}$ parameters (Equation (4.2)) will promote sparsity in the treatment difference and the horseshoe priors on the $\widetilde{\alpha}_{gf}$ parameters (Equation (4.4)) will promote sparsity in the underlying network.

The flexibility of our defined factor structure allows for the zero-inflation and high cell-to-cell variability of scRNA-seq data. For a given f, λ_{if} is unique to each cell and only affects a particular gene within a treatment when $\alpha_{gf;t} \neq 0$. If the activated factors $\lambda_{if}\alpha_{gf;t}$ for a given gene are highly negative, then μ_{gi} will be very small and account for the high proportion of zeros typical of this data. Conversely, large positive values of the factors will increase μ_{gi} (relative to the baseline of either $exp\{\beta_g\}$ for the control group or $exp\{\beta_g + \delta_g\}$ for the treatment group) and accommodate extremely large counts. In Equation (4.1), the adjustment term of $-\big\{\sum_{f=1}^{F} \frac{\alpha_{gf;t_i}^2}{2}\big\}$ is included in our model to ensure that $E(Y_{gi})$ in the control group is equal to $exp\{\beta_g\}$ and $E(Y_{gi})$ is equal to $exp\{\beta_g + \delta_g\}$ for the treatment group (after marginalizing out $\boldsymbol{\lambda_i}$) regardless of the $\alpha_{gf;t}$ values. While we choose to let Y_{gi} follow a Poisson distribution conditional on the $\boldsymbol{\lambda_i}$ terms, the variance of Y_{gi}, marginal on $\boldsymbol{\lambda_i}$, is

$$Var[Y_{gi}] = exp\{\beta_g + t_i\delta_g\}\big[1 + exp\{\beta_g + t_i\delta_g\}\prod_{f=1}^{F}\big(exp\{\alpha_{gf;t}^2\} - 1\big)\big] \ . \tag{4.5}$$

Hence, Y_{gi} is conditionally Poisson but marginally overdispersed.

To complete the specification of our Bayesian model, we define priors for the average gene expression parameters as $\beta_g \sim Normal(0, \sigma_\beta^2)$ and $\delta_g \sim Normal(0, \sigma_\delta^2)$, with standard deviation hyperparameters σ_β and σ_δ from half-Cauchy$(0, 1)$. Our methodology does rely on fixed number of latent factors F, which unfortunately is often unknown. Nevertheless, one can fit multiple models with different numbers of factors and choose the most suitable model based on a comparison of a model selection statistic such as the Deviance Information Criterion (DIC) described in Gelman et al. (2004) or the Watanabe-Akaike Information Criterion (WAIC; Watanabe, 2010).

4.2.2 Network Structure and Inference

We use Hamiltonian Monte Carlo (Neal, 2011) from Stan (Stan Development Team, 2018) to generate samples from the posterior distributions for inference. At each iteration, a co-expression matrix for each treatment group is calculated based on the current set of parameters. While the $\boldsymbol{\alpha_t}$ matrices in our model do impose a crude network structure on the gene expressions for each treatment, the individual $\alpha_{gf;t}$ parameters are non-identifiable, and so we cannot perform inference about these parameters directly. To that end, we consider the matrices $\boldsymbol{A_t} = \boldsymbol{\alpha_t}\boldsymbol{\alpha_t^T}$ whose elements are identifiable.

For a given treatment t, the (g, g') element, where $g \neq g'$, of the $G \times G$ matrix $\boldsymbol{A_t}$ provides a summation of impact by the associated factors that are active in both genes g and g' since $A_{t(g,g')} = \sum_{f=1}^{F} \alpha_{gf;t}\alpha_{g'f;t}$. This expression also happens to be equal to the covariance (after marginalizing out $\boldsymbol{\lambda_i}$) between the values of $log(\mu_{gi})$ and $log(\mu_{g'i})$ in treatment t,

$$Cov\big[log(\mu_{gi}), log(\mu_{g'i})\big] = \sum_{f=1}^{F} \alpha_{gf;t}\alpha_{g'f;t} \ .$$

With the marginal variance for $log(\mu_{gi})$ being

$$Var\big[log(\mu_{gi})\big] = \sum_{f=1}^{F} \alpha_{gf;t}^2 \ ,$$

the correlation between $log(\mu_{gi})$ and $log(\mu_{g'i})$ is defined as

$$Corr\big[log(\mu_{gi}), log(\mu_{g'i})\big] = \rho_{gg';t} = \frac{\sum_{f=1}^{F} \alpha_{gf;t}\alpha_{g'f;t}}{\sqrt{\Big(\sum_{f=1}^{F} \alpha_{gf;t}^2\Big)\Big(\sum_{f=1}^{F} \alpha_{g'f;t}^2\Big)}}. \tag{4.6}$$

We focus our interest on the marginal correlation of the log-means due to the simplistic nature of the correlation structure and its reliance on only the $\alpha_{gf;t}$ parameters. As displayed in Equation (4.5), the variance expression of Y_{gi} includes a set of β_g and δ_g parameters that cannot be factored out, which means the correlation structure between Y_{gi} and $Y_{g'i}$ will depend on the average expression for each gene in each treatment. For this reason, we do not utilize the correlation structure between Y_{gi} and $Y_{g'i}$.

The gene-gene network structure $\tilde{\boldsymbol{N}}_{\boldsymbol{t}} = \{\tilde{n}_{gg';t}\}_{(g,g')}$ within each treatment group is obtained by analyzing the posterior of the correlation matrix defined in Equation (4.6). To provide a quantifiable value of association between genes g and g' within treatment t, M samples of each (g, g') element in the correlation matrix are used to calculate the posterior mean $\widehat{\rho}_{gg';t} = \frac{1}{M}\sum_{m=1}^{M} \rho_{gg';t}^{(m)}$. We can also examine the credible interval (CI) of the posterior to determine whether or not genes g and g' are associated with one another within each treatment group, separately. For a given level of significance α^*, two genes will have a significant association when zero is excluded from the $100(1-\alpha^*)\%$ CI. To rank correlations by significance within each treatment group, we determine the smallest $100(1-\mathrm{a}^*)\%$ CI that includes 0 for each gene-gene pair. The corresponding a^* value indicates the proportion of the posterior distribution outside of the smallest CI that includes 0, which can be viewed as an approximate "p-value".

When performing differential network analysis, we examine the CIs of the difference between $\rho_{gg';0}$ and $\rho_{gg';1}$. If zero is excluded from the $100(1-\alpha^*)\%$ CI, the difference between the treatment correlations for gene g and gene g' is significant. An approximate "p-value" can also be determined and used to rank the differences in correlation between the treatment groups.

4.3 Results

In our simulation studies, we generated data consistent with our proposed factor model structure for two treatment groups. Within each treatment t, the Y_{gi} count was sampled from Poisson(μ_{gi}), with $log(\mu_{gi})$ modeled from Equation (4.1). The parameters of β_g, δ_g, and λ_{if} were all randomly sampled from Normal$(0, 1)$, and we fixed the values of $\alpha_{gf;t}$ to create different correlation structures for each treatment (see Figure 4.1).

We considered $G = 50$ genes for each dataset, sorted them into ten groups of five (e.g., Group 1 consisted of genes 1 - 5, Group 2 consisted of genes 6 - 10), and assigned the same $\alpha_{gf;t}$ values to all of the genes within each gene group. In two of the datasets (Sim 1 and Sim 2), we generated $N_0 = 250$ cells in the control group ($t = 0$) and $N_1 = 250$ cells in the treatment group ($t = 1$) for a total of $N = 500$ cells, and in the other two datasets (Sim 3 and Sim 4), we doubled the number of cells within each group for a total of $N = 1,000$ cells. For the network structures, $F_{sim} = 10$ factors were used to create 350 "true" edges in the control network and 325 "true" edges in the treatment network. In Sim 1 and Sim 3, a total of 250 edges are considered to be different between the two groups and all of the common edges have the same direction of correlation (SDC) between the genes pairs (Figures 4.1A and 4.1B). In Sim 2 and Sim 4, some of the common edges have opposite directions of correlation (ODC), which increases the "true" number of different edges between the two groups to be 325 (Figures 4.1C and 4.1D).

Two versions of our sparse Bayesian factor methodology were investigated in the simulation studies. In our "base" model version, the priors defined in Equation (4.2) and Equation (4.3) are placed on the $\alpha_{gf;t}$ and $\widetilde{\alpha}_{gf}$ parameters, respectively (i.e., we use horseshoe priors on the $\alpha_{gf;t}$'s but not on the $\widetilde{\alpha}_{gf}$'s). We refer to this model as Sparse Factor Model - Base Version (SFM-BV). For the second model version, the prior on each $\widetilde{\alpha}_{gf}$ parameter in Equation (4.3) is replaced with the horseshoe prior defined in Equation (4.4). We refer to this second model as Sparse Factor Model - Additional HorseShoe (SFM-AHS).

Using the simulated data, we ran our proposed models in R (R Core Team, 2018) with the package rstan (Stan Development Team, 2018). Inference was performed after combining the posterior samples from four parallel chains that were run for a total of 2,000 iterations each, with a burn-in of 1,000 iterations. To investigate whether the number of factors makes any impact on model performance, we ran both models four separate times and input a different number of factors for each run: $F =$ 5, 10 (the true number of factors), 15, and 20.

For each simulated dataset, we first used our models to test for a significant relationship between each gene-gene pair in each treatment by using a 95% CI for $\rho_{gg';t}$. The significant gene-gene associations identified by SFM-BV and SFM-AHS were compared to the "true" gene-gene associations, and the measures of true positive rate (TPR), false discovery rate (FDR), and area under the receiver operating characteristic curve (AUC) were determined for the control network ($t = 0$) and the treatment network ($t = 1$), separately. The results from the four considered choices of F for our two models are presented in Table 4.1. In all of the simulated datasets, both SFM-BV and SFM-AHS have high TPRs and AUCs when detecting significant gene-gene associations within each treatment group. SFM-BV seems to achieve slightly higher TPRs than SFM-AHS, particularly in the smaller datasets with $N = 500$ cells (Sim 1 and Sim 2). When the input number of factors is greater than or equal to

the true number of factors (i.e., $F = 10, 15$, or 20), the performances of both models are relatively consistent across the different performance measures, and both models control FDRs below 5%. While the choice of $F = 5$ factors tends to have the highest TPRs, this choice of F also tends to have the highest FDRs.

To evaluate the performance of differential network analysis with SFM-BV and SFM-AHS, we examined the 95% CIs of the difference between each $\rho_{gg';0}$ and $\rho_{gg';1}$ pair for each dataset. We also ran analyses with scdNet (Chiu et al., 2018) and the R package DiffCorr (Fukushima, 2013) to compare our models against other methods. In scdNet, a sample size adjustment transformation is first applied to the correlation coefficients within each cellular group and then statistical inference is performed on the differences in the transformed correlations across groups. The scdNet model is, to the best of our knowledge, the only other differential network analysis tool currently available for scRNA-seq. DiffCorr implements Fisher's Z transformation to compare correlations between two experimental conditions in the context of bulk population data. Both scdNet and DiffCorr provide p-values to represent differential results for each gene-gene pair. To control the FDR, DiffCorr utilizes the local false-discovery rate approach from Strimmer (2008). For scdNet, we controlled the FDR with the Benjamini-Hochberg (Benjamini and Hochberg, 1995) procedure. A threshold of 5% was used to indicate significant differences with these methods.

We compared the significant differences between networks that were identified by each method to the "true" differences between networks. The measures of TPR, FDR, AUC, and the number of edges that were classified as significantly different between networks by each method are displayed in Table 4.2. In addition, we provide heatmaps to visually represent the "true" differences between treatment groups and the significant differences detected by each method for Sim 3 (Figure 4.2) and Sim 4 (Figure 4.3). For our methodology, we found that WAIC was better at identifying top performing models than DIC. Therefore, the number of factors F with the lowest

WAIC was selected as the "best" model choice for each of our models and only the results of the "best" model choice for both SFM-BV and SFM-AHS are presented for each dataset.

From Table 4.2, we see that our differential network methodology performs quite well in comparison to the other considered methods. Both SFM-BV and SFM-AHS outperform DiffCorr in terms of all performance measures across all simulated datasets. While scdNet tends to detect more significant edges between treatment groups than the other methods, it also has the highest FDRs. Despite the higher number of significant edges detected by scdNet, SFM-BV is still able to obtain higher TPRs than scdNet in three out of the four simulations, while SFM-AHS obtains higher TPRs than scdNet in two out of the four simulations. In all cases, both SFM-BV and SFM-AHS have higher AUCs than scdNet while controlling the FDRs to a nominal level. Also, the heatmaps in Figures 4.2 and 4.3 visually reinforce that our methods can identify the "true" differential network structure more accurately than the other two considered methods.

We note that the performances of SFM-BV and SFM-AHS are quite similar across all performance measures in the larger simulated datasets with $N = 1,000$ cells (Sim 3 and Sim 4). However, for the datasets with $N = 500$ cells (Sim 1 and Sim 2), SFM-BV tends to identify more significantly different edges and achieve higher TPRs than SFM-AHS. Therefore, based on these simulation results, SFM-BV seems to be the better model choice for our proposed differential network methodology.

4.4 Discussion

In this chapter, we have presented a two-group hierarchical Bayesian factor model to perform differential network analysis from scRNA-seq data. This work extends our hierarchical Bayesian factor model for constructing GCNs (Chapter 3) to include continuous treatment-dependent parameters that determine the impact of the factors

for each gene. In Chapter 3, our GCN model utilizes a single-treatment $\boldsymbol{\alpha}$ matrix and the elements of this matrix only take the values of -1, 0, or 1. While this definition is useful for the interpretation of the factor-gene relationships (i.e., the expression for gene g is impacted only by factors with $|\alpha_{gf}| = 1$ and not impacted by factors with $\alpha_{gf} = 0$), we are not able to utilize Stan for parameter sampling because Stan cannot support the use of discrete parameters. Also, the one-at-a-time sampling for the $\boldsymbol{\alpha}$ matrix does not allow for large scale moves such as splitting or combining factors and so it is possible for our Markov chain Monte Carlo (MCMC) sampler to get stuck exploring a local mode of the posterior.

Since the elements of the treatment-dependent $\boldsymbol{\alpha_t}$ matrices in our proposed model for differential network analysis are continuous, we are able to use Stan for sampling and avoid the problem of getting stuck exploring local modes. The use of the Stan framework is particularly beneficial because Hamiltonian Monte Carlo is more accurate and computationally efficient than other MCMC methods (Betancourt, 2017) and the core code for Stan is written in the C++ language, making it computationally faster than running MCMC sampling code completely in R. As a future extension, we could also consider using Stan to perform variational inference (like we did in Chapter 2), since variational inference tends to be much faster than traditional MCMC techniques. We do note that we are not able to achieve exact sparsity in the $\boldsymbol{\alpha_t}$ matrices because the $\alpha_{gf;t}$ parameters are continuous. However, we have adopted a shrinkage approach and use horseshoe priors on the $\alpha_{gf;t}$'s to handle the expected sparsity.

For simplicity purposes, our methodology has been defined and examined under a two-group situation, but it can be adjusted to fit a multiple group scenario. In the general case, we can consider T number of treatments and represent the $log(\mu_{gi})$

from Equation(4.1) in the general form:

$$log(\mu_{gi}) = \beta_g + \sum_{t=1}^{T-1} I(t_i = t)\delta_{g;t} + \sum_{f=1}^{F} \lambda_{if}\alpha_{gf;t_i} - \Big\{ \sum_{f=1}^{F} \frac{\alpha_{gf;t_i}^2}{2} \Big\}.$$

Here, the $\delta_{g;t}$ parameters depend on the treatment groups $t \in \{1, \ldots, T-1\}$ and $I(t_i = t)$ is the indicator variable for cell i being in treatment group t. The construction of gene-gene correlation structures will remain the same, but there will be T sets of $\alpha_{gf;t}$ parameters that create T different networks to compare. When performing differential network analysis, one can examine the CIs of the difference between $\rho_{gg';t}$ and $\rho_{gg';t'}$ for each pair of treatments t and t' $(t \neq t')$.

The simulation studies in this manuscript demonstrate that our proposed methodology is able to accurately identify true co-expression structures with two treatment groups and detect differences between them. Both SFM-BV and SFM-AHS outperform competing methods across these simulation studies, but we recommend using SFM-BV because this model version tends to have higher TPRs when identifying significantly different edges between networks. Based on these results and the dearth of research in this area, we feel that our methodology serves as one of the first steps in the development of approaches suitable for scRNA-seq differential network analysis.

4.5 Tables and Figures

		Sim 1: SDC, N = 500					
		$\mathbf{TPR_0}$	$\mathbf{FDR_0}$	$\mathbf{AUC_0}$	$\mathbf{TPR_1}$	$\mathbf{FDR_1}$	$\mathbf{AUC_1}$
SFM-BV	**F = 5**	0.977	0.132	0.986	0.994	0.146	0.999
	F = 10	0.871	0.019	0.971	0.886	0.034	0.976
	F = 15	0.871	0.007	0.973	0.874	0.027	0.974
	F = 20	0.849	0.007	0.969	0.874	0.027	0.976
SFM-AHS	**F = 5**	0.954	0.100	0.992	0.985	0.221	0.980
	F = 10	0.823	0.014	0.966	0.855	0.007	0.972
	F = 15	0.797	0.000	0.950	0.883	0.007	0.971
	F = 20	0.757	0.004	0.960	0.837	0.000	0.974
		Sim 2: ODC, N = 500					
		$\mathbf{TPR_0}$	$\mathbf{FDR_0}$	$\mathbf{AUC_0}$	$\mathbf{TPR_1}$	$\mathbf{FDR_1}$	$\mathbf{AUC_1}$
SFM-BV	**F = 5**	0.977	0.132	0.990	1.000	0.092	0.997
	F = 10	0.889	0.019	0.988	0.831	0.022	0.987
	F = 15	0.869	0.007	0.970	0.806	0.022	0.978
	F = 20	0.863	0.007	0.971	0.797	0.023	0.975
SFM-AHS	**F = 5**	0.980	0.159	0.988	1.000	0.100	0.999
	F = 10	0.786	0.018	0.948	0.775	0.016	0.971
	F = 15	0.843	0.013	0.964	0.769	0.020	0.972
	F = 20	0.786	0.000	0.956	0.735	0.016	0.965
		Sim 3: SDC, N = 1,000					
		$\mathbf{TPR_0}$	$\mathbf{FDR_0}$	$\mathbf{AUC_0}$	$\mathbf{TPR_1}$	$\mathbf{FDR_1}$	$\mathbf{AUC_1}$
SFM-BV	**F = 5**	0.940	0.466	0.900	1.000	0.260	0.994
	F = 10	0.949	0.035	0.988	0.975	0.019	0.999
	F = 15	0.934	0.027	0.986	0.972	0.006	0.999
	F = 20	0.943	0.029	0.988	0.960	0.006	0.999
SFM-AHS	**F = 5**	0.937	0.438	0.916	1.000	0.162	1.000
	F = 10	0.949	0.012	0.989	0.938	0.000	0.998
	F = 15	0.929	0.050	0.979	0.982	0.000	1.000
	F = 20	0.929	0.009	0.984	0.948	0.003	0.998
		Sim 4: ODC, N = 1,000					
		$\mathbf{TPR_0}$	$\mathbf{FDR_0}$	$\mathbf{AUC_0}$	$\mathbf{TPR_1}$	$\mathbf{FDR_1}$	$\mathbf{AUC_1}$
SFM-BV	**F = 5**	0.937	0.407	0.920	1.000	0.024	1.000
	F = 10	0.954	0.023	0.993	0.972	0.009	0.999
	F = 15	0.949	0.021	0.991	0.923	0.007	0.996
	F = 20	0.943	0.021	0.988	0.920	0.007	0.998
SFM-AHS	**F = 5**	0.937	0.417	0.920	1.000	0.033	1.000
	F = 10	0.937	0.018	0.981	0.935	0.000	0.997
	F = 15	0.931	0.015	0.984	0.932	0.000	0.999
	F = 20	0.934	0.015	0.988	0.914	0.000	0.993

Table 4.1: Performance measures for the identification of significant gene-gene associations by our proposed differential network methods (SFM-BV and SFM-AHS) with different numbers of factors (F) across four simulated datasets. TPR, FDR, and AUC were calculated separately for each treatment group and the subscripts denote the treatment group of the corresponding network: control = 0, treatment = 1.

	Sim 1: SDC, N = 500, Edges_{TrueD} = 250			
	TPR_D	FDR_D	AUC_D	Edges_D
SFM-BV, F = 20	0.452	0.017	0.814	115
SFM-AHS, F = 20	0.096	0.000	0.758	24
DiffCorr	0.020	0.167	0.693	6
scdNet	0.206	0.518	0.641	139
	Sim 2: ODC, N = 500, Edges_{TrueD} = 325			
	TPR_D	FDR_D	AUC_D	Edges_D
SFM-BV, F = 15	0.671	0.022	0.926	223
SFM-AHS, F = 20	0.584	0.010	0.910	192
DiffCorr	0.538	0.038	0.900	182
scdNet	0.462	0.407	0.758	253
	Sim 3: SDC, N = 1,000, Edges_{TrueD} = 250			
	TPR_D	FDR_D	AUC_D	Edges_D
SFM-BV, F = 15	0.552	0.021	0.943	141
SFM-AHS, F = 20	0.532	0.036	0.901	138
DiffCorr	0.268	0.163	0.836	80
scdNet	0.560	0.378	0.819	225
	Sim 4: ODC, N = 1,000, Edges_{TrueD} = 325			
	TPR_D	FDR_D	AUC_D	Edges_D
SFM-BV, F = 10	0.806	0.022	0.988	268
SFM-AHS, F = 20	0.843	0.011	0.976	277
DiffCorr	0.646	0.041	0.957	219
scdNet	0.717	0.272	0.850	320

Table 4.2: Comparison of the "true" differences between networks and the estimated differences between networks in the simulation studies for each differential network method. The subscript of "D" is used to denote network differences.

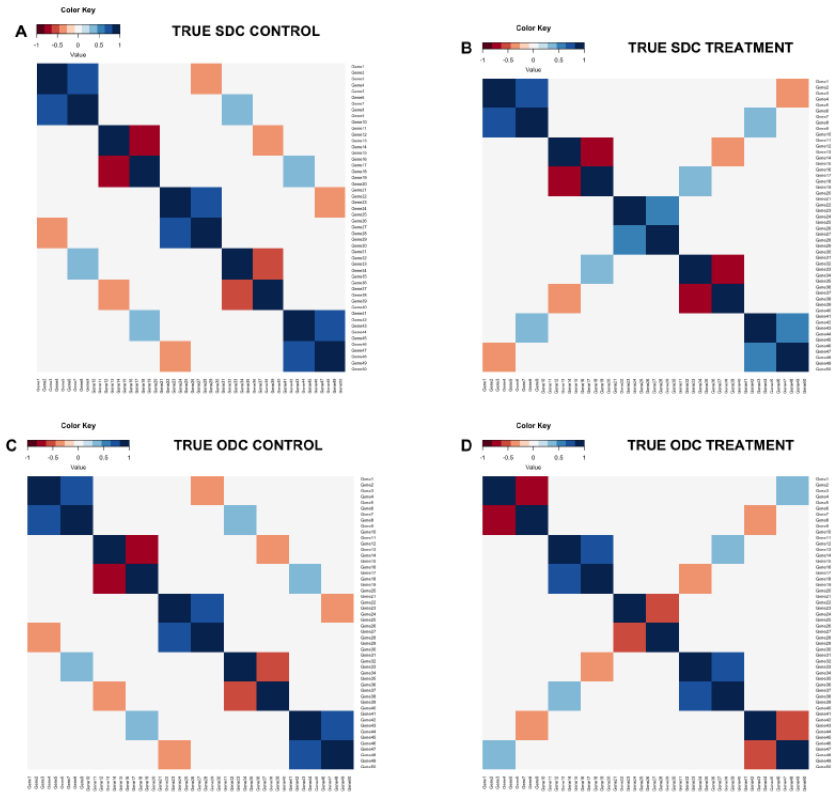

Figure 4.1: Heatmaps of "true" correlation structures for treatment and control groups in simulation studies. (A,B) In Sim 1 and Sim 3, all shared edges between the two groups have the same direction of correlation (SDC). (C,D) In Sim 2 and Sim 4, some shared edges between the two groups have opposite directions of correlation (ODC).

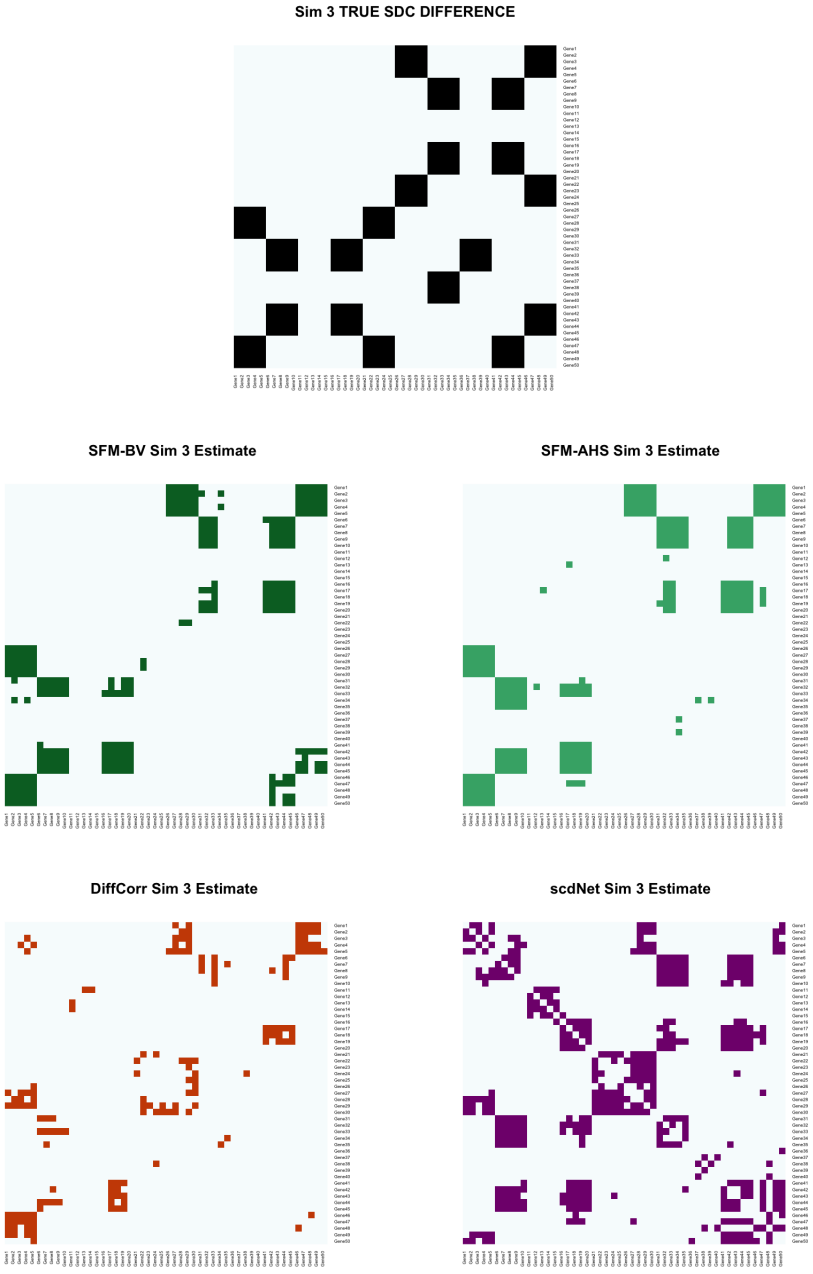

Figure 4.2: Heatmaps of the "true" differences between treatment correlation structures in Sim 3 (top) and the significant treatment differences identified by SFM-BV with $F = 15$, SFM-AHS with $F = 20$, DiffCorr, and scdNet. The colored cells indicate differences in gene-gene associations across the treatment and control groups.

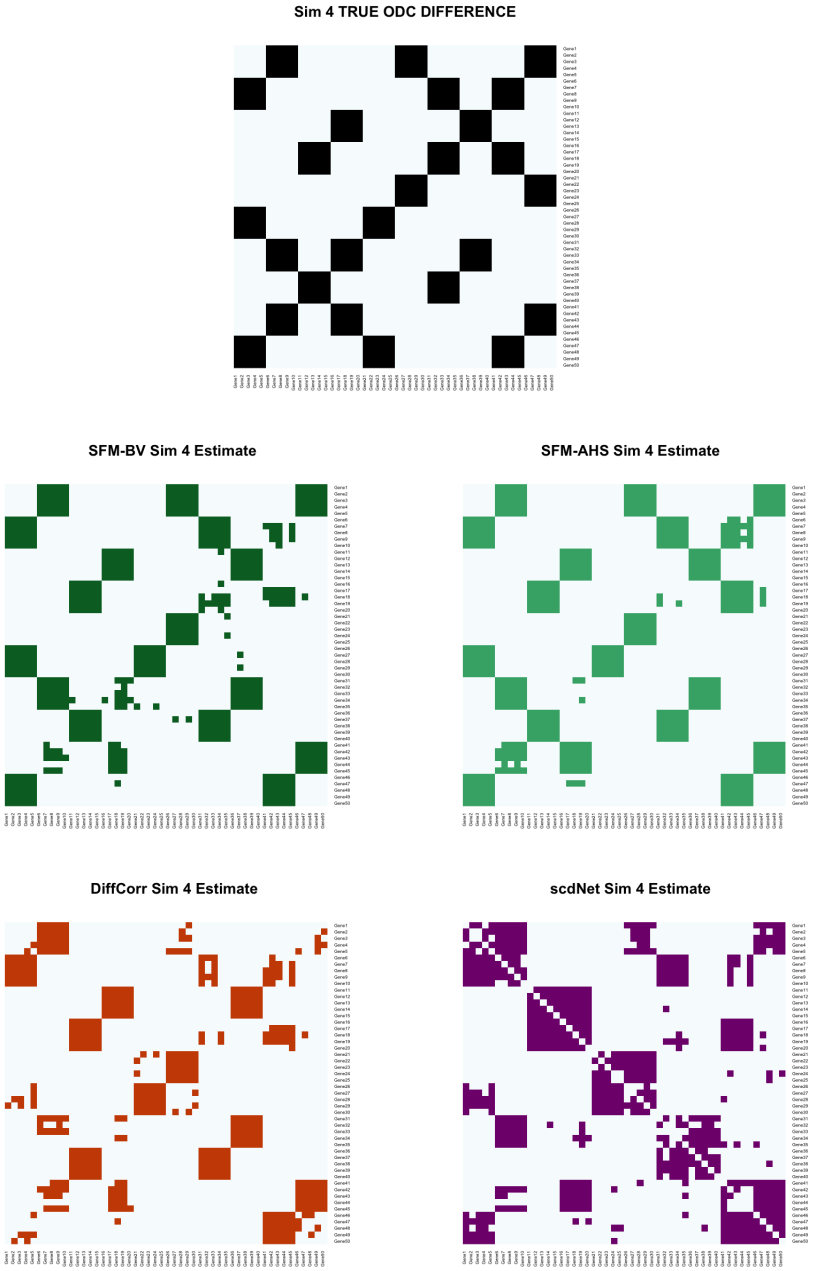

Figure 4.3: Heatmaps of the "true" differences between treatment correlation structures in Sim 4 (top) and the significant treatment differences identified by SFM-BV with $F = 10$, SFM-AHS with $F = 20$, DiffCorr, and scdNet. The colored cells indicate differences in gene-gene associations across the treatment and control groups.

CHAPTER 5

SUMMARY AND FURTHER EXTENSIONS

In this dissertation, we have presented three Bayesian approaches developed specifically for single-cell RNA sequencing (scRNA-seq) analyses. We first defined a hurdle model for identifying differentially expressed genes across cell types in scRNA-seq data in Chapter 2. Through several different analyses with both simulated data and real data, we were able to demonstrate the feasibility and practical utility of our approach in comparison to alternative methods. Then, in Chapter 3, a sparse Bayesian factor model was introduced to detect gene-gene network structures from scRNA-seq data. We highlighted our model's ability to identify true co-expressions while maintaining nominal false discovery rates across different numbers of cells and different network structures in both simulated and real data analyses. Finally, Chapter 4 expanded upon our sparse Bayesian factor model to examine the differences between networks of different treatment groups. Again, we were able to establish our methodology's superiority over other comparable methods through simulation studies.

These research projects provide significant steps toward the application of Bayesian solutions to the statistical challenges posed by the characteristics of scRNA-seq data. With that being said, we note several further extensions and developments that are worth considering. For our network methodology in Chapters 3 and 4, a natural extension would be to place a shrinkage prior on the factor loadings, similar to the approach used in Bhattacharya and Dunson (2011). This would ultimately remove the guesswork involved in choosing the number of factors F by allowing, in theory,

the use of infinitely many factors in our models. Factors with a higher index f will tend to have smaller variances and, therefore, have smaller overall effects. As another extension, we could combine differential expression (DE) analysis into our differential network model from Chapter 4 by performing posterior inference on the log-fold change parameter δ_g. Since this model framework utilizes covariance information between groups, we anticipate achieving higher power and greater efficiency when detecting differentially expressed genes.

Scalability is perhaps one of the biggest obstacles to overcome when using Bayesian methodology for high-dimensional scRNA-seq data. Incorporating computationally efficient and scalable algorithms from the existing literature to our proposed network methodology, like we have done with variational inference for our DE model in Chapter 2, will allow for analyses of larger gene sets. This will be particularly useful because Lichtblau et al. (2017) have demonstrated that in the context of bulk population data, differential network algorithms can outperform differential expression methods in terms of identifying the genes that play key roles in biological processes. Utilizing a faster programming language (e.g., MATLAB or Julia) could also help reduce the amount of running time required for computationally expensive techniques like Markov chain Monte Carlo sampling.

REFERENCES

Aibar, S., González-Blas, C. B., Moerman, T., Imrichova, H., Hulselmans, G., Rambow, F., et al. (2017). SCENIC: Single-cell regulatory network inference and clustering. *Nature Methods*, 14(11):1083–1086.

Allocco, D. J., Kohane, I. S., and Butte, A. J. (2004). Quantifying the relationship between co-expression, co-regulation and gene function. *BMC Bioinformatics*, 5(1):18.

Bacher, R. and Kendziorski, C. (2016). Design and computational analysis of single-cell RNA-sequencing experiments. *Genome Biology*, 17(1):63.

Benjamini, Y. and Hochberg, Y. (1995). Controlling the false discovery rate: A practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society: Series B*, 57(1):289–300.

Betancourt, M. (2017). A conceptual introduction to Hamiltonian Monte Carlo. *arXiv preprint arXiv:1701.02434*.

Bezanson, J., Edelman, A., Karpinski, S., and Shah, V. B. (2017). Julia: A fresh approach to numerical computing. *SIAM Review*, 59(1):65–98.

Bhattacharya, A. and Dunson, D. B. (2011). Sparse Bayesian infinite factor models. *Biometrika*, 98(2):291–306.

Blei, D. M., Kucukelbir, A., and McAuliffe, J. D. (2017). Variational infer-

ence: A review for statisticians. *Journal of the American Statistical Association*, 112(518):859–877.

Blencowe, M., Arneson, D., Ding, J., Chen, Y. W., Saleem, Z., and Yang, X. (2019). Network modeling of single-cell omics data: Challenges, opportunities, and progresses. *Emerging Topics in Life Sciences*.

Buettner, F., Natarajan, K. N., Casale, F. P., Proserpio, V., Scialdone, A., Theis, F. J., et al. (2015). Computational analysis of cell-to-cell heterogeneity in single-cell RNA-sequencing data reveals hidden subpopulations of cells. *Nature Biotechnology*, 33(2):155.

Caldana, C., Degenkolbe, T., Cuadros-Inostroza, A., Klie, S., Sulpice, R., Leisse, A., et al. (2011). High-density kinetic analysis of the metabolomic and transcriptomic response of Arabidopsis to eight environmental conditions. *The Plant Journal*, 67(5):869–884.

Carvalho, C. M., Polson, N. G., and Scott, J. G. (2009). Handling sparsity via the horseshoe. In *Artificial Intelligence and Statistics*, pages 73–80.

Celeux, G., Chauveau, D., and Diebolt, J. (1996). Stochastic versions of the EM algorithm: An experimental study in the mixture case. *Journal of Statistical Computation and Simulation*, 55(4):287–314.

Chan, T. E., Stumpf, M. P., and Babtie, A. C. (2017). Gene regulatory network inference from single-cell data using multivariate information measures. *Cell Systems*, 5(3):251–267.

Chen, S. and Mar, J. C. (2018). Evaluating methods of inferring gene regulatory networks highlights their lack of performance for single cell gene expression data. *BMC Bioinformatics*, 19(1):232.

Chen, Y. C., Wang, Y. S., and Erosheva, E. A. (2018). On the use of bootstrap with variational inference: Theory, interpretation, and a two-sample test example. *The Annals of Applied Statistics*, 12(2):846–876.

Chiu, Y. C., Hsiao, T. H., Wang, L. J., Chen, Y., and Shao, Y.-H. J. (2018). scdNet: A computational tool for single-cell differential network analysis. *BMC Systems Biology*, 12(8):124.

Choi, Y. and Kendziorski, C. (2009). Statistical methods for gene set co-expression analysis. *Bioinformatics*, 25(21):2780–2786.

Chowdhury, H. A., Bhattacharyya, D. K., and Kalita, J. K. (2019). (Differential) co-expression analysis of gene expression: A survey of best practices. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*.

De Smet, R. and Marchal, K. (2010). Advantages and limitations of current network inference methods. *Nature Reviews Microbiology*, 8(10):717.

Delgado, F. M. and Gómez-Vela, F. (2018). Computational methods for gene regulatory networks reconstruction and analysis: A review. *Artificial Intelligence in Medicine*.

Delmans, M. and Hemberg, M. (2016). Discrete distributional differential expression (D^3E) – a tool for gene expression analysis of single-cell RNA-seq data. *BMC Bioinformatics*, 17(1):110.

edgeR package (2019). Bioconductor. https://bioconductor.org/packages/release/bioc/html/edgeR.html (accessed February 24, 2019).

Eisen, M. B., Spellman, P. T., Brown, P. O., and Botstein, D. (1998). Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Sciences*, 95(25):14863–14868.

Fan, H. C., Fu, G. K., and Fodor, S. P. (2015). Combinatorial labeling of single cells for gene expression cytometry. *Science*, 347(6222):1258367.

Fiers, M. W., Minnoye, L., Aibar, S., Bravo González-Blas, C., Kalender Atak, Z., and Aerts, S. (2018). Mapping gene regulatory networks from single-cell omics data. *Briefings in Functional Genomics*, 17(4):246–254.

Finak, G., McDavid, A., Yajima, M., Deng, J., Gersuk, V., Shalek, A. K., et al. (2015). Mast: A flexible statistical framework for assessing transcriptional changes and characterizing heterogeneity in single-cell RNA sequencing data. *Genome Biology*, 16(1):278.

Fukushima, A. (2013). DiffCorr: An R package to analyze and visualize differential correlations in biological networks. *Gene*, 518(1):209–214.

Gelman, A., Carlin, J. B., Stern, H. S., and Rubin, D. B. (2004). *Bayesian Data Analysis*. Chapman and Hall/CRC, Boca Raton, 2nd edition.

Gelman, A., Jakulin, A., Pittau, M. G., and Su, Y. S. (2008). A weakly informative default prior distribution for logistic and other regression models. *The Annals of Applied Statistics*, 2(4):1360–1383.

Gill, R., Datta, S., and Datta, S. (2010). A statistical framework for differential network analysis from microarray data. *BMC Bioinformatics*, 11(1):95.

Haghverdi, L., Lun, A. T., Morgan, M. D., and Marioni, J. C. (2018). Batch effects in single-cell RNA-sequencing data are corrected by matching mutual nearest neighbors. *Nature Biotechnology*, 36(5):421–427.

Hecker, M., Lambeck, S., Toepfer, S., Van Someren, E., and Guthke, R. (2009). Gene regulatory network inference: Data integration in dynamic modelsa review. *Biosystems*, 96(1):86–103.

Holm, S. (1979). A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, pages 65–70.

Huang, D. W., Sherman, B. T., and Lempicki, R. A. (2009a). Bioinformatics enrichment tools: Paths toward the comprehensive functional analysis of large gene lists. *Nucleic Acids Research*, 37(1):1–13.

Huang, D. W., Sherman, B. T., and Lempicki, R. A. (2009b). Systematic and integrative analysis of large gene lists using DAVID Bioinformatics resources. *Nature Protocols*, 4(1):44.

Huang, S. (2009). Non-genetic heterogeneity of cells in development: More than just noise. *Development*, 136(23):3853–3862.

Huynh-Thu, V., Irrthum, A., Wehenkel, L., and Geurts, P. (2010). Inferring regulatory networks from expression data using tree-based methods. *PLoS One*, 5(9).

Islam, S., Kjällquist, U., Moliner, A., Zajac, P., Fan, J. B., Lönnerberg, P., and Linnarsson, S. (2011). Characterization of the single-cell transcriptional landscape by highly multiplex RNA-seq. *Genome Research*, 21(7):1160–1167.

Jaakkola, M. K., Seyednasrollah, F., Mehmood, A., and Elo, L. L. (2017). Comparison of methods to detect differentially expressed genes between single-cell populations. *Briefings in Bioinformatics*, 18(5):735–743.

Kanter, I. and Kalisky, T. (2015). Single cell transcriptomics: Methods and applications. *Frontiers in Oncology*, 5:53.

Kharchenko, P. V., Silberstein, L., and Scadden, D. T. (2014). Bayesian approach to single-cell differential expression analysis. *Nature Methods*, 11(7):740.

Kim, S. (2015). ppcor: An R package for a fast calculation to semi-partial correlation coefficients. *Communications for Statistical Applications and Methods*, 22(6):665.

Kiselev, V. Y., Kirschner, K., Schaub, M. T., Andrews, T., Yiu, A., Chandra, T., et al. (2017). SC3: Consensus clustering of single-cell RNA-seq data. *Nature Methods*, 14(5):483–486.

Korthauer, K. D., Chu, L. F., Newton, M. A., Li, Y., Thomson, J., Stewart, R., and Kendziorski, C. (2016). A statistical approach for identifying differential distributions in single-cell RNA-seq experiments. *Genome Biology*, 17(1):222.

Kucukelbir, A., Ranganath, R., Gelman, A., and Blei, D. (2015). Automatic variational inference in Stan. In *Advances in Neural Information Processing Systems*, pages 568–576.

Kucukelbir, A., Tran, D., Ranganath, R., Gelman, A., and Blei, D. M. (2017). Automatic differentiation variational inference. *Journal of Machine Learning Research*, 18(1):430–474.

Langfelder, P. and Horvath, S. (2008). WGCNA: An R package for weighted correlation network analysis. *BMC Bioinformatics*, 9(1):559.

Lex, A., Gehlenborg, N., Strobelt, H., Vuillemot, R., and Pfister, H. (2014). UpSet: Visualization of intersecting sets. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):1983–1992.

Lichtblau, Y., Zimmermann, K., Haldemann, B., Lenze, D., Hummel, M., and Leser, U. (2017). Comparative assessment of differential network analysis methods. *Briefings in Bioinformatics*, 18(5):837–850.

Liu, Y., Morley, M., Brandimarto, J., Hannenhalli, S., Hu, Y., Ashley, E. A., et al. (2015). RNA-seq identifies novel myocardial gene expression signatures of heart failure. *Genomics*, 105(2):83–89.

Love, M. I., Huber, W., and Anders, S. (2014). Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. *Genome Biology*, 15(12):550.

Lun, A. T., McCarthy, D. J., and Marioni, J. C. (2016). A step-by-step workflow for low-level analysis of single-cell RNA-seq data with Bioconductor. *F1000Research*, 5.

Macaulay, I. C. and Voet, T. (2014). Single cell genomics: Advances and future perspectives. *PLoS Genetics*, 10(1).

Macosko, E. Z., Basu, A., Satija, R., Nemesh, J., Shekhar, K., Goldman, M., et al. (2015). Highly parallel genome-wide expression profiling of individual cells using nanoliter droplets. *Cell*, 161(5):1202–1214.

Margolin, A. A., Nemenman, I., Basso, K., Wiggins, C., Stolovitzky, G., Dalla Favera, R., and Califano, A. (2006). ARACNE: An algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. *BMC Bioinformatics*, 7(S1):S7.

Matsumoto, H., Kiryu, H., Furusawa, C., Ko, M. S., Ko, S. B., Gouda, N., et al. (2017). SCODE: an efficient regulatory network inference algorithm from single-cell RNA-seq during differentiation. *Bioinformatics*, 33(15):2314–2321.

McDavid, A., Finak, G., and Yajima, M. (2019). *MAST: Model-based Analysis of Single Cell Transcriptomics*. R package version 1.8.2. https://github.com/RGLab/MAST/ (accessed February 24, 2019).

Miao, Z., Deng, K., Wang, X., and Zhang, X. (2018). DEsingle for detecting three types of differential expression in single-cell RNA-seq data. *Bioinformatics*, 34(18):3223–3224.

Miao, Z. and Zhang, X. (2016). Differential expression analyses for single-cell RNA-seq: Old questions on new data. *Quantitative Biology*, 4(4):243–260.

Neal, R. M. (2011). MCMC using Hamiltonian dynamics. *Handbook of Markov Chain Monte Carlo*, 2(11):2.

Patel, A. P., Tirosh, I., Trombetta, J. J., Shalek, A. K., Gillespie, S. M., Wakimoto, H., et al. (2014). Single-cell RNA-seq highlights intratumoral heterogeneity in primary glioblastoma. *Science*, 344(6190):1396–1401.

R Core Team (2018). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. https://www.R-project.org/.

Robinson, M. D. and Smyth, G. K. (2007). Moderated statistical tests for assessing differences in tag abundance. *Bioinformatics*, 23(21):2881–2887.

Scutari, M. (2010). Learning bayesian networks with the bnlearn R package. *Journal of Statistical Software*, 35(3):1–22.

Shalek, A. K., Satija, R., Adiconis, X., Gertner, R. S., Gaublomme, J. T., Raychowdhury, R., et al. (2013). Single-cell transcriptomics reveals bimodality in expression and splicing in immune cells. *Nature*, 498(7453):236–240.

Shapiro, E., Biezuner, T., and Linnarsson, S. (2013). Single-cell sequencing-based technologies will revolutionize whole-organism science. *Nature Reviews Genetics*, 14(9):618–630.

Soneson, C. and Robinson, M. D. (2018). Bias, robustness and scalability in single-cell differential expression analysis. *Nature Methods*, 15(4):255.

Specht, A. T. and Li, J. (2016). LEAP: Constructing gene co-expression networks

for single-cell RNA-sequencing data using pseudotime ordering. *Bioinformatics*, 33(5):764–766.

Stan Development Team (2018). *RStan: The R interface to Stan*. R package version 2.18.2. http://mc-stan.org (accessed February 24, 2019).

Stegle, O., Teichmann, S. A., and Marioni, J. C. (2015). Computational and analytical challenges in single-cell transcriptomics. *Nature Reviews Genetics*, 16(3):133–145.

Street, K., Risso, D., Fletcher, R. B., Das, D., Ngai, J., Yosef, N., et al. (2018). Slingshot: Cell lineage and pseudotime inference for single-cell transcriptomics. *BMC Genomics*, 19(1):477.

Strimmer, K. (2008). A unified approach to false discovery rate estimation. *BMC Bioinformatics*, 9(1):303.

Szklarczyk, D., Franceschini, A., Wyder, S., Forslund, K., Heller, D., Huerta-Cepas, J., et al. (2014). STRING v10: Protein–protein interaction networks, integrated over the tree of life. *Nucleic Acids Research*, 43(D1):D447–D452.

Tay, T. L., Dautzenberg, J., Grün, D., and Prinz, M. (2018). Unique microglia recovery population revealed by single-cell RNAseq following neurodegeneration. *Acta Neuropathologica Communications*, 6(1):87.

Tirosh, I., Izar, B., Prakadan, S. M., Wadsworth, M. H., Treacy, D., Trombetta, J. J., et al. (2016). Dissecting the multicellular ecosystem of metastatic melanoma by single-cell RNA-seq. *Science*, 352(6282):189–196.

Trapnell, C., Cacchiarelli, D., Grimsby, J., Pokharel, P., Li, S., Morse, M., et al. (2014). The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering of single cells. *Nature Biotechnology*, 32(4):381.

von Mering, C., Jensen, L. J., Snel, B., Hooper, S. D., Krupp, M., Foglierini, M., et al. (2005). STRING: Known and predicted protein–protein associations, integrated and transferred across organisms. *Nucleic Acids Research*, 33:D433–D437.

Vu, T. N., Wills, Q. F., Kalari, K. R., Niu, N., Wang, L., Rantalainen, M., and Pawitan, Y. (2016). Beta-Poisson model for single-cell RNA-seq data analyses. *Bioinformatics*, 32(14):2128–2135.

Wang, J., Xia, S., Arand, B., Zhu, H., Machiraju, R., Huang, K., et al. (2016). Single-cell co-expression analysis reveals distinct functional modules, co-regulation mechanisms and clinical outcomes. *PLoS Computational Biology*, 12(4):e1004892.

Wang, Y. and Blei, D. M. (2019). Frequentist consistency of variational Bayes. *Journal of the American Statistical Association*, 114(527):1147–1161.

Wang, Y., Wu, H., and Yu, T. (2017). Differential gene network analysis from single cell RNA-seq. *Journal of Genetics and Genomics*, 44(6):331.

Wang, Z., Gerstein, M., and Snyder, M. (2009). RNA-seq: A revolutionary tool for transcriptomics. *Nature Reviews Genetics*, 10(1):57–63.

Watanabe, S. (2010). Asymptotic equivalence of Bayes cross validation and widely applicable information criterion in singular learning theory. *Journal of Machine Learning Research*, 11(Dec):3571–3594.

Weston, D. J., Karve, A. A., Gunter, L. E., Jawdy, S. S., Yang, X., Allen, S. M., and Wullschleger, S. D. (2011). Comparative physiology and transcriptional networks underlying the heat shock response in Populus trichocarpa, Arabidopsis thaliana and Glycine max. *Plant, Cell & Environment*, 34(9):1488–1506.

Wolfe, C. J., Kohane, I. S., and Butte, A. J. (2005). Systematic survey reveals

general applicability of "guilt-by-association" within gene coexpression networks. *BMC Bioinformatics*, 6(1):227.

Xu, C. and Su, Z. (2015). Identification of cell types from single-cell transcriptomes using a novel clustering method. *Bioinformatics*, 31(12):1974–1980.

Zappia, L., Phipson, B., and Oshlack, A. (2017). Splatter: Simulation of single-cell RNA sequencing data. *Genome Biology*, 18(1):174.

Zeisel, A., Muñoz-Manchado, A. B., Codeluppi, S., Lönnerberg, P., La Manno, G., Juréus, A., et al. (2015). Cell types in the mouse cortex and hippocampus revealed by single-cell RNA-seq. *Science*, 347(6226):1138–1142.

APPENDIX A

HURDLE MODEL SIMULATION DETAILS

Simulated datasets were generated from the hurdle model set-up described in the Methodology section of the main manuscript. For each small simulation, $N = 100$ cells were assigned to two different treatment groups: 50 cells allocated to treatment 0 (control) and 50 cells allocated to treatment 1. The cells were further clustered into ten total subpopulations, such that the cells within each treatment group were evenly divided into five subpopulations (ten cells in each subpopulation). In the unequal subpopulation scenario, the five subpopulations consisted of 15, 12, 10, 8, and 5 cells, respectively. We chose the number of clusters within each treatment to reflect the number of clusters observed in previous scRNA-seq studies (Patel et al., 2014; Tirosh et al., 2016) while also including a sufficient number of cells per cluster to reasonably estimate the correlated random effect parameters within the model. For the larger simulations, $N = 1{,}000$ cells were considered and the number of cells within each treatment and subpopulation were scaled by a factor of 10.

Letting $k_t(i)$ represent the subpopulation of cell i within treatment t, the random effect ω_i for each cell was determined by $\omega_i = \gamma_{t,k_t(i)} + \omega_i^*$. The ω_i^*'s were randomly generated from $Normal(0, \sigma_*^2)$, with $\sigma_*^2 = 0.7$. For the unequal simulations, the ω_i^*'s from the smallest subpopulation (5 or 50 cells) were drawn from a normal distribution with a smaller variance ($\sigma_*^2 = 0.6$) and the ω_i^*'s from the largest subpopulation (15 or 150 cells) were were drawn from a normal distribution with a larger variance ($\sigma_*^2 = 0.8$). To enforce some separation of the random effects between subpopulation

clusters, the $\gamma_{t,k}$'s within each treatment were randomly assigned one of five values without replacement: $-2\sigma_t$, $-\sigma_t$, 0, σ_t, and $2\sigma_t$. In this notation, the true value of σ_t represents a scale parameter (not the standard deviation) that determines the separation between subpopulations. Table A.1 lists the values for σ_t^2 and σ_*^2 used in the simulations.

In the simulation scheme described above, a data-generating design matrix X consisting only of the treatment indicator was used. The first 1,250 genes, out of G = 10,000 genes, were set to be significantly different for the logistic regression part of the hurdle model (β_{1g}^L was either -1.5 or 1.5 in the small simulations and either -0.5 or 0.5 in the large simulations), while the remaining genes had $\beta_{1g}^L = 0$. The other coefficients in the model (β_{0g}^L and ζ_g^L) were simulated based on the results from the mouse embryonic cell (MEC) data (Islam et al., 2011). In the large simulations, some methods had TPRs and AUCs near 1 when the values of the significant β_{1g}^L's were set to ± 1.5. Therefore, we decreased the treatment effect to help expose the differences between methods.

An overlap of 250 differentially expressed (DE) genes between the logistic regression and the truncated negative binomial regression were considered. Thus, genes numbered $1{,}000 - 2{,}250$ in the dataset were set to be significantly different for the zero-truncated negative binomial part of the hurdle model (β_{1g}^C was either -1.5 or 1.5 in the small simulations and either -0.5 or 0.5 in the large simulations), while the remaining genes had $\beta_{1g}^C = 0$. The other coefficients in the model (β_{0g}^C and ζ_g^C) were again simulated based on the results from the MEC data.

Following our proposed methodology, a $G \times N$ zero-one matrix Z was first generated from Bernoulli(θ_{gi}) for each combination of g and i. The probability of success θ_{gi} was determined from Equation (2.1) in the main manuscript using the coefficients, random effects, and model matrix described above. A $G \times N$ count matrix Y was then generated, such that $Y_{gi} = 0$ if $Z_{gi} = 0$, and $Y_{gi} > 0$ if $Z_{gi} = 1$. The count

values for $Y_{gi} > 0$ were simulated from Truncated Negative Binomial(μ_{gi}, ϕ_g) defined in Equation (2.3) of the main manuscript. Here, μ_{gi} was calculated from Equation (2.4) using the coefficients, random effects, and model matrix described above. The overdispersion parameter ϕ_g was generated from Lognormal(λ_1, λ_2) with the estimates of λ_1 and λ_2 coming from the MEC data. Finally, genes expressed in less than 20% of all cells were removed from analysis. After data generation, CDR was calculated and added into the design matrix before model estimation.

For each simulation scenario, 100 datasets were generated and the DE genes were determined at a false discovery rate (FDR) of 0.05 for each method. SCDE uses the Holm (1979) procedure to adjust Z-scores, while all other methods utilize the Benjamini and Hochberg (1995) procedure to control for FDR. The measures of true positive rate (TPR), false positive rate (FPR), observed FDR, area under the receiver operating characteristic curve (AUC), and number of identified DE genes were used to compare methods.

In addition to the simulations discussed in the manuscript, another small sample size simulation ($N = 100$) was considered by changing the number of subpopulations per treatment group. In one scenario, a total of four subpopulations were simulated such that the cells within each treatment group were evenly divided into two subpopulations (twenty-five cells in each cluster). The $\gamma_{t,k}$'s within each treatment were randomly assigned either $-0.5\sigma_t$ or $0.5\sigma_t$ without replacement. In the other scenario, a total of sixteen subpopulations were simulated. We increased the number of cells per treatment group to be 56 in order to evenly divide the cells in each treatment into eight clusters of seven cells each, and the $\gamma_{t,k}$'s within each treatment were randomly assigned one of nine values without replacement: $-2.4\sigma_t$, $-1.8\sigma_t$, $-1.2\sigma_t$, $-0.6\sigma_t$, 0, $0.6\sigma_t$, $1.2\sigma_t$, $1.8\sigma_t$, and $2.4\sigma_t$. The results from these two variations are provided in Table A.2.

We also created variation in the small hurdle model simulations by increasing

the amount of information supplied by the subpopulations. In these simulations we considered equal cluster sizes and set $\sigma_*^2 = 0.3$ and $\sigma_0^2 = \sigma_1^2 = 0.7$. Here, the use of a smaller σ_*^2 value creates a larger within subpopulation correlation. Simulated datasets were generated with two, five, and eight subpopulations per treatment, and the results are displayed in Table A.3.

The overall results from these variations were generally similar to those already presented in Section 2.4.1 of the main manuscript. The "2 clusters per treatment" scenarios were the only small sample size, hurdle model simulations where the bulk method of DESeq2 selected more DE genes than CRE and IRE. However, this is due to the much higher FDR in DESeq2, which also leads to worse AUC performance than our proposal.

	σ_*^2	σ_0^2	σ_1^2
Equal Cluster Size Simulations	0.7	0.3	0.3
Unequal Cluster Size Simulations	0.8/0.7/0.6	0.3	0.3

Table A.1: Variance terms for $\gamma_{t,k}$ and ω_i^* in the hurdle model simulation studies.

Hurdle model: 2 equal clusters per treatment					
	TPR	**FPR**	**FDR**	**AUC**	**DE Genes**
CRE, SC3	0.701	0.009	0.044	0.963	1577
CRE, NN=3	0.700	0.009	0.044	0.963	1577
CRE, NN=7	0.700	0.009	0.044	0.963	1576
CRE, TRUE	0.700	0.009	0.044	0.963	1576
IRE	0.704	0.010	0.043	0.962	1591
NRE	0.694	0.014	0.062	0.958	1595
MAST	0.619	0.006	0.033	0.955	1378
SCDE	0.155	0.088	0.263	0.663	985
DESeq2	0.445	0.118	0.415	0.750	1834
edgeR	0.409	0.062	0.323	0.781	1336
Hurdle model: 8 equal clusters per treatment					
	TPR	**FPR**	**FDR**	**AUC**	**DE Genes**
CRE, SC3	0.722	0.010	0.046	0.965	1637
CRE, NN=3	0.723	0.010	0.045	0.965	1638
CRE, NN=7	0.722	0.010	0.045	0.965	1635
CRE, TRUE	0.722	0.010	0.045	0.965	1635
IRE	0.726	0.010	0.047	0.965	1646
NRE	0.711	0.024	0.098	0.952	1711
MAST	0.643	0.007	0.037	0.957	1443
SCDE	0.174	0.013	0.276	0.627	1340
DESeq2	0.436	0.088	0.394	0.765	1594
edgeR	0.436	0.071	0.342	0.780	1468

Table A.2: Additional results of performance measures from the hurdle model simulation structure with two and eight simulated subpopulations per treatment.

Hurdle model: 2 clusters per treatment, high correlation					
	TPR	**FPR**	**FDR**	**AUC**	**DE Genes**
CRE, SC3	0.713	0.009	0.042	0.965	1595
CRE, NN=3	0.713	0.009	0.041	0.965	1595
CRE, NN=7	0.713	0.009	0.041	0.965	1594
CRE, TRUE	0.714	0.009	0.042	0.965	1597
IRE	0.716	0.010	0.045	0.965	1609
NRE	0.713	0.010	0.048	0.964	1606
MAST	0.636	0.006	0.032	0.958	1409
SCDE	0.180	0.064	0.377	0.686	861
DESeq2	0.490	0.202	0.500	0.715	2545
edgeR	0.401	0.030	0.197	0.811	1078
Hurdle model: 5 clusters per treatment, high correlation					
	TPR	**FPR**	**FDR**	**AUC**	**DE Genes**
CRE, SC3	0.663	0.009	0.043	0.955	1506
CRE, NN=3	0.663	0.009	0.043	0.955	1506
CRE, NN=7	0.663	0.009	0.043	0.955	1504
CRE, TRUE	0.663	0.009	0.043	0.955	1506
IRE	0.667	0.010	0.048	0.954	1524
NRE	0.647	0.015	0.074	0.945	1522
MAST	0.568	0.006	0.037	0.944	1281
SCDE	0.031	0.008	0.066	0.677	126
DESeq2	0.380	0.048	0.298	0.778	1183
edgeR	0.360	0.053	0.335	0.768	1181
Hurdle model: 8 clusters per treatment, high correlation					
	TPR	**FPR**	**FDR**	**AUC**	**DE Genes**
CRE, SC3	0.701	0.010	0.047	0.959	1596
CRE, NN=3	0.701	0.010	0.048	0.959	1599
CRE, NN=7	0.701	0.010	0.047	0.959	1598
CRE, TRUE	0.702	0.010	0.047	0.959	1597
IRE	0.704	0.010	0.048	0.960	1606
NRE	0.686	0.021	0.092	0.948	1647
MAST	0.611	0.007	0.040	0.949	1381
SCDE	0.048	0.014	0.134	0.657	207
DESeq2	0.425	0.075	0.361	0.770	1480
edgeR	0.408	0.075	0.378	0.761	1447

Table A.3: Additional results of performance measures from the hurdle model simulation structure with high within subpopulation correlation for two, five, and eight subpopulations per treatment.

APPENDIX B

SPLAT SIMULATION DETAILS

The Splat scheme (Zappia et al., 2017) simulates count expression values from a gamma-Poisson distribution that is modified to vary the library sizes across cells and impose a mean-variance trend such that highly expressed genes are less variable than lowly expressed genes. This simulation structure also provides options for including abnormally high expression levels (outlier genes), high proportions of zeros (dropout), and additional technical variation (batch effects). For simulating differential expression between treatment groups of cells, Splat generates multiplicative factors from a lognormal distribution and applies them to the mean expression values of the respective genes in one of the treatment groups. Genes not simulated to be DE have multiplicative factors set to one.

A total of 100 small datasets (100 cells and 10,000 genes) and 100 larger datasets (1,000 cells and 10,000 genes) were generated with the *splatSimulateGroups* function from the Splatter R package (Zappia et al., 2017). The probability of a gene being DE was set to 0.1, and cells had equal probabilities of being assigned to one of two groups. We considered additional technical variation by setting the batch effect argument such that half of the cells belonged in one batch and the other half belonged in another batch. Dropout and outlier genes were also incorporated into the Splat simulation design.

The location parameters for the lognormal distributions of the batch effects factor and differential expression factor were both set to 2, while the scale parameters

for both factors set to their default values of 0.1 and 0.4, respectively. Again, in the larger datasets we observed very high TPRs and AUCs (nearly 1) for some of the methods when the expression factor was set to 2. Therefore, to help differentiate the methods in the larger Splat datasets, the differential expression factor was reduced to 0.5. Values for the remaining parameters of the *splatSimulateGroups* function not already mentioned above were estimated from the MEC data, and genes that were not expressed in at least 20% of the cells were removed from analysis.

In addition to using trimmed mean of M-values normalization and scaling the adjusted counts to counts per million for the clustering algorithms, as described in Section 2.4 of the main manuscript, we also applied the mutual nearest neighbors (MNN) correction (Haghverdi et al., 2018) to adjust for batch effects. The MNN method is available in the scran R package (Lun et al., 2016).

Since we focus on how the results from a clustering analysis can be incorporated into our model, not on how to perform a cluster analysis, we do assume that some consideration for potential bias has been taken when processing the data for clustering. Therefore, we included the MNN adjustment to remove any bias the simulated batch effects may impose during clustering. However, to keep the Splat simulation analyses consistent across all methods, we did not apply the MNN adjustment to the data analyzed with MAST since we did not adjust any of the other models for batch effects.

APPENDIX C

HMEC DROP-SEQ DETAILS

This dataset was generously provided to us by researchers at the University of Florida Health Cancer Center. To measure the cell-to-cell variability of gene expression in a mammalian cell line, Drop-seq was used to perform single cell expression analysis on MCF10A human mammary epithelial cells (HMEC) expressing either exogenous wild type or mutant histone H2B. Each sample included 10% mouse 3T3 cells to assist in determining the doublet rate and ambient RNA barcode noise. The Drop-seq apparatus was constructed in house according to published protocols (Macosko et al., 2015) with a microfluidic co-flow device purchased from Nanoshift (Emeryville, CA).

Briefly, 1,000 cells were encapsulated in droplets with companion bar coded primer-coated microparticles. After cell lysis, cDNA libraries were prepared by hybridization of mRNA to primer beads and reverse transcriptase treatment with Maxima H Minus RT (ThermoFisher) to produce single-cell transcriptomes attached to microparticles (STAMPS) and amplified with Kapa HiFi Hotstart ReadyMix. The cDNA library was tagmented with Nextara XT to produce pools high throughput sequencing on a NextSeq 500 (Illumina) with a 75 bp flow cell.

A cluster analysis was performed on this data to examine the variability within potential subpopulations of cells. In brief, the SC3 algorithm (Kiselev et al., 2017) was utilized to initially estimate an "optimal" number of clusters within each cell type. A visual inspection was then imposed to ensure that each cluster had a reasonable

number of cells (at least 1% of the total number of cells). Based on this analysis, the wild type cells were clustered into seven subpopulations and the mutant cells were clustered into nine subpopulations.

APPENDIX D

ADDITIONAL MEC ANALYSES

Sensitivity Analysis

To evaluate the choice of Inverse Gamma$(0.001, 0.001)$ as the prior on λ_2, we ran an additional analysis on the MEC data after changing this prior to Inverse Gamma$(1, 1)$ in our models. With the new prior, CRE detected 4,933 DE genes compared to the 4,927 DE genes in the original analysis. The overlap of genes was 4,926. For the IRE model, both versions identified exactly 4,947 DE genes, with 100% overlap. Based on these results, the prior choice for λ_2 seems to be relatively innocuous.

DAVID Functional Annotation Clustering

The Database for Annotation, Visualization, and Integrated Discovery, better known as DAVID (Huang et al., 2009a,b), was utilized to help determine the biological relevance of genes identified as being DE in the MEC analysis. Clusters of gene ontology (GO) categories for different subsets of genes were created with the DAVID functional annotation clustering tool. An Enrichment Score is calculated by DAVID for each cluster to help identify clusters that are involved in more enriched (important) biological roles. Each Enrichment Score is determined by the geometric mean of the modified Fisher Exact p-values for all annotation terms that belong to a given cluster. A negative log base 10 transformation is applied on each geometric mean to emphasize that this measurement is a relative score, rather than an exact value. Thus, a higher

score indicates that the genes annotated to the GO categories within the cluster are involved in more enriched roles.

It has been recommended that more attention should be given to groups with scores greater than or equal to 1.3 (Huang et al., 2009b). An Enrichment Score of 1.3 corresponds to clusters with a geometric mean of $10^{-1.3} = 0.05$. We used this threshold of 1.3 to classify clusters as enriched. Additionally, we made special note of the clusters with Enrichment Scores greater than or equal to 4 (i.e., the geometric mean of p-values is 10^{-4} or less) by classifying them as highly enriched.

We examined the 1,335 genes in the MEC analysis that were detected by CRE and undetected in the other methods to help evaluate the biological relevance of these new discoveries. For comparison, we also analyzed the genes uniquely identified by only DESeq2 and only edgeR. These methods were chosen as a comparison because they detected the second (293) and third (172) highest number of unique DE genes. The scRNA-seq methods of SCDE and MAST only detected 7 and 2 unique genes, respectively.

A total of five GO category clusters were considered to be highly enriched in the analysis of genes identified only CRE, and twenty additional clusters had scores greater than 1.3. Categories of RNA splicing, mRNA processing, and spliceosomal complex were clustered together and had the highest Enrichment Score of 10.62. The cluster containing GO categories of relating to cell-cell adhesion had the second highest Enrichment Score (7.13) and the cluster containing categories related to DNA binding and transcription both had the third highest Enrichment Score (7.10). GO categories of cell cycle, mitotic nuclear division, and cell division were clustered together had an Enrichment Score of 4.73. Lastly, the fifth most highly enriched cluster had an Enrichment Score of 4.51 and included categories relating to ligase activity and protein ubiquitination.

In comparison, the genes detected only by DESeq2 had one highly enriched GO

category cluster and five clusters with scores greater than 1.3. The highly enriched cluster had an Enrichment Score of 4.63 and contained categories related to RNA binding. The genes identified only by edgeR did not have any highly enriched GO clusters, but did have two clusters with a score greater than 1.3.

Naturally, we would expect the genes detected by our CRE model to form more clusters simply because there are more of them, but the fact that five of GO category clusters were highly enriched and twenty more had Enrichment Scores greater than 1.3 suggests that many of these genes have similarities in their biological roles. The subset of 989 genes in common among all methods except for MAST did form ten GO clusters with Enrichment Scores greater than 1.3, but none of them had a score greater than 4. Additionally, the 1,148 genes in common among CRE, DESeq2, and edgeR formed fourteen GO clusters with scores greater than 1.3 with two of those clusters being highly enriched. Therefore, based on the comparison to these larger subsets, our CRE method is not only able to identify a larger number of DE genes, but these genes are also annotated to similar and important biological functions and processes.

CURRICULUM VITA

NAME: Michael Sekula

ADDRESS: Department of Biostatistics and Bioinformatics
University of Louisville
Louisville, KY 40202

EDUCATION: Master of Science, University of Louisville, 2015

Bachelor of Science, Saginaw Valley State University, 2010

PUBLICATIONS: Sekula, M., Gaskins, J., and Datta, S. (2019). Detection of differentially expressed genes in discrete single-cell RNA sequencing data using a hurdle model with correlated random effects. *Biometrics*, 75(4), 1051-1062.

Sekula, M., Datta, S., and Datta, S. (2017). optCluster: An R Package for Determining the Optimal Clustering Algorithm. *Bioinformation*, 13(3), 101.

PRESENTATIONS: "A sparse Bayesian factor model for the construction of gene co-expression networks from discrete single-cell RNA sequencing data". Bioinformatics and Biostatistics Seminar Series. University of Louisville, Louisville, KY. September 6, 2019.

"A sparse Bayesian factor model for the construction of gene co-expression networks from discrete single-cell RNA sequencing data". Poster session presented at the Southern Regional Council on Statistics Summer Conference. General Butler State Resort Park, KY. June 3, 2019.

"A sparse Bayesian factor model for the construction of gene regulatory networks from discrete single-cell RNA sequencing data". Kentucky American Statistical Association Chapter Meeting. University of Louisville, Louisville, KY. April 4, 2019.

"A correlated random effects hurdle model for detecting differentially expressed genes in discrete single-cell RNA sequencing data". Contributed paper session at East North American Region (ENAR). Philadelphia, PA. March 26, 2019.

"A correlated random effects hurdle model for detecting differentially expressed genes in discrete single-cell RNA sequencing data". Contributed paper session at Joint Statistical Meetings (JSM). Vancouver, British Columbia, Canada. July 30, 2018.

"Detection of differentially expressed genes in discrete single-cell RNA sequencing data using a hurdle model with correlated random effects". Poster session presented at the Southern Regional Council on Statistics Summer Conference. Virginia Beach, VA. June 5, 2018.

"Advanced R programming". University of Louisville Biostatistics Club Seminar. University of Louisville, Louisville, KY. April 6, 2018.

"A correlated random effects hurdle model for differential gene expression analysis of discrete single-cell RNA sequencing data". Kentucky American Statistical Association Chapter Meeting. University of Louisville, Louisville, KY. March 2, 2018.

"Hurdle model with correlated random effects for differential expression of single-cell RNA sequencing data". Bioinformatics and Biostatistics Seminar Series. University of Louisville, Louisville, KY. October 27, 2017.

HONORS/AWARDS: University of Louisville Faculty Favorite Nominee, 2019

Best Student Presentation in Complex Modeling Techniques Session, Kentucky American Statistical Association Chapter, 2019

Boyd Harshbarger Travel Award, Southern Regional Council on Statistics, 2018, 2019

Best Student Presentation in Biostatistics Session, Kentucky American Statistical Association Chapter, 2018

President of University of Louisville Biostatistics Club, 2016-2017

University Fellowship, University of Louisville, 2015-2017

ETS Recognition of Excellence for Mathematics: Content Knowledge, 2010

Outstanding Senior in Mathematics Education, 2010