BAYESIAN ANALYSIS OF HIGH-THROUGHPUT SEQUENCING DATA

A Dissertation

by

SIAMAK ZAMANI DADANEH

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Chair of Committee,	Xiaoning Qian
Committee Members,	P. R. Kumar
	Bani K. Mallick
	Byung-Jun Yoon
Head of Department,	Miroslav M. Begovic

December 2019

Major Subject: Electrical Engineering

ABSTRACT

We develop a Bayesian framework for the analysis of high-throughput sequencing count data under a variety of settings, removing sophisticated *ad-hoc* pre-processing steps commonly required in existing algorithms. Specifically, we start by exploiting Bayesian nonparametric priors, including the gamma-Poisson, gamma-negative binomial, and beta-negative binomial processes, to model RNA sequencing (RNA-seq) count matrices. We then develop a novel Bayesian negative binomial regression (BNB-R) method for the analysis of RNA-seq count data. In particular, the natural model parameterization removes the needs for the normalization step, while the method is capable of tackling complex experimental design involving multivariate dependence structures.

In addition to studying genes individually, investigating coordinated expression variations of genes may help reveal the underlying cellular mechanisms to derive better understanding and more effective prognosis and intervention strategies. In chapter 4, We develop a fully Bayesian covariate-dependent negative binomial factor analysis method—dNBFA—for RNA-seq count data, to capture coordinated gene expression changes, while considering effects from covariates reflecting different influencing factors.

Finally, in the last chapter, we propose a fully generative hierarchical gamma-negative binomial (hGNB) model of single-cell RNA-seq (scRNA-seq) data, obviating the need for explicitly modeling zero inflation. hGNB can naturally account for covariate effects at both the gene and cell levels to identify complex latent representations of scRNA-seq data, without the need for commonly adopted pre-processing steps such as normalization.

DEDICATION

To my family, whose constant support led me in this path.

ACKNOWLEDGMENTS

First and foremost, I would like to thank my love, Megi, for all of her sacrifices during my PhD. Her patience, love, and encouraging attitude have been always driving me to seek for the best quality of work.

I am grateful for my adviser Professor Xiaoning Qian for his continuous support and guidance throughout my PhD, and for providing me the freedom to work on a variety of problems. I also would like to express my deepest gratitude to my PhD committee member, Professor Mingyuan Zhou, who instilled a deeper understanding and intellectual probity into my work.

Finally, I would like to thank the Texas A&M University High Performance Research Computing for providing computational resources to perform experiments in this dissertation.

CONTRIBUTORS AND FUNDING SOURCES

Contributors

This work was supported by a dissertation committee consisting of Professor Xiaoning Qian [advisor] and Professors P.R. Kumar and Byung-Jun Yoon of the Department of Electrical and Computer Engineering, Professor Bani Mallick of the Department of Statistics, and Professor Mingyuan Zhou of the Department of Statistics and Data Sciences at the University of Texas at Austin.

All the work conducted for the dissertation was completed by the student independently.

Funding Sources

Graduate study was supported by the National Science Foundation (NSF) Grants 1553281 and 1812641.

NOMENCLATURE

RNA-seq	RNA Sequencing
scRNA-seq	Single-Cell RNA Sequencing
GO	Gene Ontology
NB	Negative Binomial
NBP	Negative Binomial Process
GNBP	Gamma-Negative Binomial Process
BNBP	Beta-Negative Binomial Process
CRT	Chinese Restaurant Process
PG	Polya-Gamma
BNB-R	Bayesian Negative Binomial Regression
dNBFA	Covariate-Dependent Negative Binomial Factor Analysis

TABLE OF CONTENTS

Page

LIST OF FIGURES

FIGURE Page

LIST OF TABLES

1. INTRODUCTION

1.1 Background

Measuring gene expression is vital in studying many aspects of life systems including tissue differentiation, and genomic landscape of diseases, drugs, or other perturbations. There exists several molecular biology techniques to measure gene expression. In particular, advent of next generation sequencing (NGS) technologies such as RNA sequencing (RNA-seq) has revolutionized the field of molecular biology (2).

In a typical RNA-seq experiment (Figure 1.1), first long RNAs are fragmented into a library of cDNA. Subsequently, sequencing adaptors are added to each cDNA fragment and using the high-throughput sequencing technology a short sequence is obtained from each cDNA (1). The reads are then aligned to the reference genome corresponding to the species of interest. Aligned reads are assigned to genes in order to determine gene expression. Methods such as HTSeq (3) and featureCounts (4) aggregate the reads aligning to genomic intervals defined by an annotation, to produce gene counts.

1.2 Problem Statement: RNA Sequencing Data Analysis

There has been significant recent interest in analyzing RNA sequencing (RNA-seq) count data for studying life systems (1; 5). It is challenging to model RNA-seq data, not only because it is typically a large-p-small-n problem (6) where the data dimension is high while the sample size is small, but also because the sequencing counts are non-negative, skewed, having large dynamical ranges, and highly over-dispersed (7; 8). A key task in RNA-seq analysis is to identify the genes that are differentially expressed between different groups of samples ($e.g.$, samples measured under different medical conditions) (9; 10; 7; 11; 12; 13). The expression level of each RNA locus, here the gene, is determined by the number of sequenced reads to the transcript (14). Unlike a gene probe based method such as microarrays (15), the abundance of genes in RNA-seq is restricted by the sequencing depth and there often exist dependencies between the expressions of different

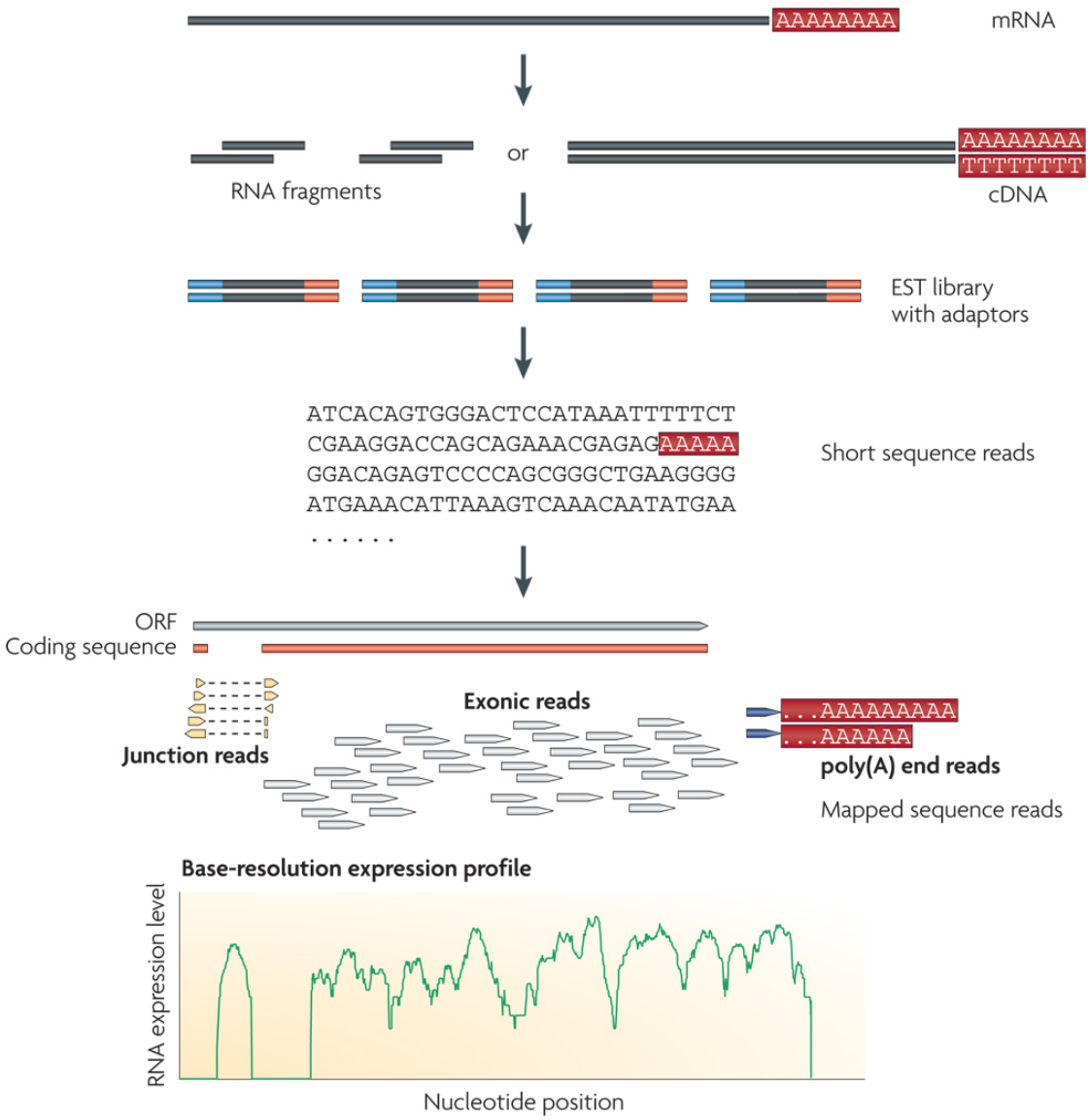


Figure 1.1: A typical RNA-seq experiment. Figure reprinted with permission from (1).

transcripts (16).

Modeling the sequencing counts using an over-dispersed count distribution, such as the negative binomial (NB) distribution (17; 18), is one of the most popular approaches for differential expression analysis (19; 7). In the null hypothesis that a gene is not differently expressed, it is common to assume that the expectations of the counts of that gene are the same across different groups, after making adjustments to account for both technical and biological variations. In par-

ticular, almost all existing comparative analysis algorithms, before downstream analyses, require normalizing the sequencing counts to compensate the variations of sequencing depths across samples (20; 21; 22). For instance, edgeR and DESeq, two widely used differential expression analysis R software packages adopt different *ad-hoc* normalization procedures: edgeR either calculates a trimmed mean of M-values (10) between each pair of samples or uses an upper quantile of samples (23) for normalization (19), while DESeq takes the median of the ratios of observed sample's counts to the geometric mean across samples as a scaling factor for that specific sample (7; 24).

Normalizing the sequencing counts, however, inevitably destroys the discrete nature of the raw data and makes the performance clearly depend on whether the introduced normalization is suitable for the structure of the RNA-seq data under study (20; 21; 22). If the normalization procedure extracts normalization constants from the data under study to parameterize the distributions of the gene counts, the discrete nature of the raw data is preserved, but the model can no longer be considered as a generative model. In addition, almost all existing normalization procedures assume that most of the genes are not differentially expressed, and the differentially expressed genes are equally likely to be up- and down-regulated (25; 26; 27; 28). The violation of the assumption may potentially be addressed by using external RNA control consortium (ERCC) spike-in sequences for controls; however, it is shown in (27; 28) that the read counts for ERCC spike-ins alone are usually not stable enough to be used for normalization. Moreover, despite that a wide array of methods have been proposed to adjust the counts to account for technical and biological variations, there is not a single one that clearly outperforms the others under various scenarios (20; 29; 21; 27; 22; 30).

1.2.1 Differential expression analysis

For J RNA-seq samples organized into the same group, let us denote n_{jk} as the number of reads in sequencing sample $j \in \{1, \ldots, J\}$ that are assigned to gene $k \in \{1, \ldots, K\}$, where K is the number of genes in the genome. Since the counts of a gene across samples are often over-dispersed, it is natural to model them using a NB distribution, where its variance σ^2 is related to its mean μ as $\sigma^2 = \mu + \phi\mu^2$, where ϕ is the dispersion parameter. As it is also common to refer to $r = \phi^{-1}$ as the dispersion parameter, to avoid ambiguity, we will refer to $r = \phi^{-1}$ as the NB shape

parameter.

Methods such as edgeR and DESeq propose different ways to estimate ϕ. EdgeR models the gene count n_{jk} as a NB distribution with mean $n_j\lambda_{jk}$ and dispersion ϕ_k, where n_j is the observed total count (or the sum of adjusted counts) for sample j, λ_{jk} represents the abundance of gene k in sample j, and ϕ_k is considered as the coefficient of biological variation that is estimated by conditional maximum likelihood (31). Furthermore, an empirical Bayes procedure is applied to shrink the dispersion parameters ϕ_k towards a common value (32).

DESeq also models the gene counts with the NB distribution. It considers two terms to estimate the variance σ^2_{jk} for gene k in sample j, where the first term (shot noise) is associated with the mean expression of the gene, and the second one (raw variance) takes into account the biological variations between replicates. More specifically, it lets $\sigma^2_{jk} = \mu_{jk} + n_j^2 v_{k,\rho(j)}$. Here, $\rho(j)$ is the group to which sample j belongs, and $v_{k,\rho(j)}$ is the per-gene raw variance, which is a smooth function of λ and ρ, an assumption that allows pooling data from different genes to estimate their variances.

Another widely used tool, baySeq (33), takes an empirical Bayesian approach to estimate the posterior probabilities of a set of models that define different patterns of differential expression for each gene. For instance, in the simplest case of a pairwise comparison between conditions A and B, with two biological replicates for each condition, the model for no differential expression is defined by the set of samples {A1, A2, B1, B2}, while differential expression between conditions A and B is defined by the sets {A1, A2} and {B1, B2}. The method then assumes that the counts follow the NB distribution and derives an empirically determined prior distribution from the data.

The final component of these methods is the test for gene differential expression. Both edgeR and DESeq use variations of Fisher's exact test, adjusted for the NB distribution, to compute exact p-values for the null hypothesis that the mean expressions of the genes are equal in both conditions under comparison. EdgeR also considers the generalized linear model approach to identify differentially expressed genes in its later versions; nevertheless, it has been shown to have similar performance to the method based on Fisher's exact test (34). Different from edgeR and DESeq,

baySeq ranks the genes based on the inferred posterior probabilities of differential expression.

1.2.2 Module Identification

Another class of approaches for the analysis of RNA-seq data, aim to detect genes with similar expression patterns as potential functional modules. The majority of these methods are inspired by tools developed for microarray technology, which construct networks from pairwise similarities between gene expressions and after normalizing the count data usually include (35; 36) the following steps:

- Build an *adjacency matrix* based on the correlation between gene expressions.
- Perform a *network clustering* to identify modules of genes with similar expression patterns.
- Apply *dimensionality reduction* techniques to extract *eigengenes* (35), which are representative of modules.
- *Regress* external covariates such as clinical factors on the eigengenes expressions.

This pipeline, however, requires heuristic tuning and careful choice of methods at each step. For instance, different choices of correlation measure, clustering methods and dimensionality reduction techniques can change the derived gene modules substantially.

There remains a lack of tools for gene module detection specifically designed for RNA-seq count data. Furthermore, the existing methods for RNA-seq often require prior knowledge from either manual annotations or other module identification methods. Specifically, they need to be supplied with prepared lists of genes as candidate functional modules. For example, (37) have proposed a network module-based generalized linear model for identifying differentially expressed pre-defined gene sets.

1.2.3 Single-Cell RNA-seq Analysis

Single-cell RNA-seq (scRNA-seq) is a technique that was developed in the last decade and has been quickly growing in popularity. In contrast with bulk RNA-seq technology, scRNA-seq

allows one to identify the messenger RNA corresponding to individual cells, making it the method of choice to assay gene expression of heterogeneous biological systems, such as cancer, developmental biology and differentiation, and complex tissues.

Many of Statistical tools developed for scRNA-seq data analysis include a dimensionality reduction step. This leads to the reduction of noise in the data, while retaining the often intrinsically low-dimensional signal of interest. Dimensionality reduction of scRNA-seq data is challenging. In addition to high gene expression variability due to cell heterogeneity, the excessive amount of zeros in scRNA-seq hinders the application of classical dimensionality reduction techniques such as principal component analysis (PCA).

Several existing computational tools adopt explicit zero-inflation modeling to infer the latent representation of scRNA-seq data. Despite its popularity, using an explicit zero-inflation term may place unnecessary emphasis on the zero counts, leading to complication in discovering the latent representation of scRNA-seq data.

1.3 Our Contributions

Bayesian modeling is an ideal choice for high-dimensional, small-sample size data prevalent in high-throughput genomics measurements, as it provides a rigorous mechanism to incorporate prior scientific knowledge, and also is capable of quantifying the uncertainty about the statistical discoveries. In this dissertation, we propose a fully Bayesian framework to address the challenges encountered in the statistical analysis of RNA-seq data. Core properties of our proposed methods include obviating the need for ad-hoc preprocessing of the RNA-seq data due to fully probabilistic nature of our models, and efficient inference of model parameters by taking advantage of novel data augmentation techniques. In what follows, we describe in more details how the aforementioned challenges in the analysis of RNA-seq count data are addressed under our proposed models.

1.3.1 DE Analysis

First, in chapter 2, we exploit Bayesian nonparametric priors, including the gamma-Poisson, gamma-negative binomial, and beta-negative binomial processes, to model RNA sequencing count

matrices. With different sequencing depths captured by sample-specific model parameters, the posterior distributions of certain gene-specific model parameters are used to detect the genes that are differentially expressed between different conditions. With the model parameters inferred by borrowing statistical strength across both the genes and samples, there is no need to adjust the raw counts using heuristics before downstream analyses, an important preprocessing step that is often required in previously proposed algorithms. Example results on both synthetic and real-world RNA-Seq data demonstrate the stateof-the-art performance of both the gamma- and beta-negative binomial processes based differential expression analysis algorithms. Given the success of the proposed randomprocess-based algorithms in differential expression analysis, it is of interest to investigate Bayesian nonparametric algorithms for many other real-world applications in biomedicine that require analyzing next-generation sequencing data.

Moving to more complicated experimental settings, in chapter 3, We propose a Bayesian NB regression (BNB-R) method for DE analysis of sequencing count data. On one hand, BNB-R is capable of handling complex experiments involving multiple factors. On the other hand, it does not require an ad-hoc normalization preprocessing step. By taking advantage of novel data augmentation techniques, BNB-R possesses efficient closed-form Gibbs sampling update equations and ranks differentially expressed genes based on a symmetric KL-divergence measure, exploiting the full posterior distributions of the model parameters. Experimental results on both synthetic and real-world RNA-seq data demonstrate the state-of-the-art performance of BNB-R in DE analysis of RNA-seq data.

1.3.2 Covariate-Dependent Module Identification

In chapter 4 of this dissertation, we develop a novel covariate-dependent NB factorization model for identifying gene modules in RNA-seq experiments. The proposed method, directly applied to gene counts from RNA-seq, obviates the need for multiple *ad-hoc* steps as required in above co-expression network based analyses. Additionally, by employing a flexible regression model in a fully Bayesian framework, our model is capable of tackling RNA-seq experiments with complex confounding factors, and quantifies the impact of these factors on the identified modules.

Finally, this new approach does not require an *ad-hoc* normalization step, as the model accounts for the sequencing-depth heterogeneity of different samples automatically.

1.3.3 scRNA-seq Data Analysis

In chapter 5 of this dissertation, we propose a hierarchical gamma-negative binomial (hGNB) model to both perform dimensionality reduction and adjust for the effects of the gene- and cell-level confounding factors simultaneously. Exploiting the hierarchical structure, the proposed hGNB model is capable of capturing the high over-dispersion present in the scRNA-seq data. More precisely, we factorize the logit of the negative-binomial (NB) distribution probability parameter to identify latent representation of the data. In addition to factorization, linear regression terms are also included in that logit function to adjust for the impact of covariates.

2. BAYESIAN NONPARAMETRIC DIFFERENTIAL EXPRESSION ANALYSIS [1]

In this chapter, we introduce a generative model to analyze differential expression directly on the raw sequencing counts, without the need to preprocess the data by normalization. Instead of using parametric count distributions to describe the counts, we use a stochastic process to model the observed sample-gene random count matrix in each group, whose model parameters are estimated by sharing statistical strength across both the genes and samples. The stochastic process can be used to explain not only the counts and the total number of expressed genes in the observed random count matrix, but also the number of newly expressed genes and the counts on both existing and newly expressed genes to be brought by a new sample. Such flexible random-process-based models lift the need of *ad-hoc* data normalization and strict parametric assumptions, allowing heterogeneity across samples and gene expression variations across different conditions to be well captured.

More specifically, moving beyond existing algorithms that model over-dispersed counts with the NB distribution, our Bayesian nonparametric (BNP) algorithms model the gene counts using the gamma-negative binomial process (GNBP) (38), which mixes the NB shape parameter for each gene with the distribution of the weight of an atom of a gamma process (39), or beta-negative binomial process (BNBP) (40; 41; 42), which mixes the NB probability parameter of each gene with the distribution of the weight of an atom of a beta process (43). In addition to the GNBP and BNBP, for comparison, we have extended the negative binomial process (NBP) of (38) by multiplying the gene-specific Poisson rates with gamma distributed sample-specific scaling parameters, and refer to it as the scaled NBP. While the NBP of (38) is not expected to work well since it does not explicitly model the variation of a sample's total count, the scaled NBP, even with a scaling parameter for each sample to capture that variation, is found to provide poor performance, indicating a clear limitation of the Poisson distribution assumption. We will show that while the variations of

[1] Reprinted with permission from S. Zamani Dadaneh, X. Qian, and M. Zhou, "BNP-Seq: Bayesian nonparametric differential expression analysis of sequencing count data," *Journal of the American Statistical Association*, vol. 113, no. 521, pp. 81–94, 2018. Copyright 2018 Taylor & Francis.

the gene counts across samples are well captured by neither the Poisson rates of the scaled NBP nor the normalized Poisson rates of the NBP, they are well modeled by both the GNBP and BNBP, using the NB shape and probability parameters, respectively.

Unlike previous algorithms for differential expression analysis, the proposed BNP algorithms require no normalization pre-processing steps and they infer the posterior distributions, instead of point estimates, of their model parameters, using Gibbs sampling with closed-form update equations, achieving state-of-the-art performance in detecting truly differentially expressed genes for both synthetic and real data.

2.1 Bayesian Nonparametric Differential Expression Analysis for RNA-seq

We consider a family of NB processes, each of which can be used to describe the row-by-row sequential construction of a sample-gene sequencing count matrix, where the addition of a new sample (row) brings counts at not only previously expressed genes (columns), but also previously unexpressed ones. We also describe the equivalent construction that draws a Poisson random number of independent, and identically distributed (i.i.d.) columns simultaneously, where each column corresponds to the counts of a gene that is expressed at least once across all the observed samples of a group. Showing these two equivalent constructions helps clearly understand the underlying statistical assumption made on the RNA-seq data by a BNP prior, and how the statistical strength is shared across both the genes and samples to estimate both the sample-specific model parameters, which account for the variations in sequencing depths, and the gene-specific model parameters, whose posterior distributions are used to detect differentially expressed genes.

Below we show how a stochastic process can be used to model the counts in each group, where the group index is omitted for brevity. We represent the counts of all expressed genes in a group as a random count matrix $\mathbf{N}_J \in \mathbb{Z}^{J\times K_J}$, where $\mathbb{Z} = \{0, 1, \ldots\}$ represents the set of nonnegative integers, K_J denotes the random number of genes that are expressed at least once in the J samples of the group, and the element n_{jk} represents the number of reads in sequencing sample $j \in \{1, \ldots, J\}$ that are assigned to gene $k \in \{1, \ldots, K_J\}$. Note that K_J, the number of expressed genes among the J samples, is smaller or equal to K, the total number of genes in the

genome, and K_J can potentially increase without bound as J increases.

2.1.1 NBP-Seq: Negative binomial process for RNA-seq

Let us denote G_0 as a finite and continuous base measure over a complete and separable metric space Ω, $c \in \mathbb{R}_+$ as a scale parameter, and $q_j \in \mathbb{R}_+$ as sample-specific scaling parameters, where $\mathbb{R}_+ := \{x : x > 0\}$. We define the scaled negative binomial process (NBP) that has sample-specific scaling parameters as

$$(X_1, \ldots, X_J) \,|\, c, G_0, \{q_j\}_{1,J} \sim \text{NBP}(G_0, c_0, q_1, \ldots, q_J),$$

which is obtained by marginalizing out a gamma process (39) $G \sim \Gamma\text{P}(G_0, 1/c)$ from J conditionally independent Poisson processes (44) $X_j \,|\, q_j, G \sim \text{PP}(q_j G)$, where for disjoint Borel sets $A_j \subset \Omega$, the gamma process G is defined such that $G(A_i) \sim \text{Gamma}[G_0(A_i), 1/c]$ are independent gamma random variables, and the Poisson process X_j is defined such that $X_j(A_i) \sim \text{Pois}[q_j G(A_i)]$ are independent Poisson random variables. With a draw from the gamma process expressed as $G = \sum_{k=1}^{\infty} r_k \delta_{\omega_k}$, where ω_k and r_k are the atoms and their weights, respectively, a draw from X_j can be expressed as

$$X_j = \sum_{k=1}^{\infty} n_{jk} \delta_{\omega_k}, \; n_{jk} \sim \text{Pois}(q_j r_k). \tag{2.1}$$

Note that if we fix $q_j = 1$ for all j, then the proposed NBP with sample-specific scaling parameters reduces to the NBP in (41) and (38).

The conditional likelihood of the observed J samples of a group can be written as

$$p(\{X_j\}_{j=1}^J \,|\, G) = e^{-q_\cdot G(\Omega \setminus \mathcal{D}_J)} \left[\prod_{k=1}^{K_J} \frac{r_k^{n_{\cdot k}} e^{-q_\cdot r_k}}{\prod_{j=1}^J n_{jk}!} \right] \left[\prod_{j=1}^J q_j^{n_j} \right], \tag{2.2}$$

where $\mathcal{D}_J = \{\omega_k\}_{k:n_{\cdot k}>0}$ is the set of points of discontinuity, $K_J = |\mathcal{D}_J| = \sum_k \delta(n_{\cdot k} > 0)$ is the number of genes that are expressed at least once, $q_\cdot = \sum_{j=1}^J q_j$, and $n_{\cdot k} = \sum_{j=1}^J n_{jk}$. We map the counts associated with the elements of $\mathcal{D}_J$ to the random count matrix $\mathbf{N}_J$. While the labelings of the atoms in $\mathcal{D}_J$ are arbitrary, they are mapped in one of the $K_J!$ possible ways to the columns of

$\mathbf{N}_J$. Similar to the derivation in (38), using a marginalization procedure shown in (45), one may marginalize out the gamma process G, leading to the distribution of the random count matrix as

$$f(\mathbf{N}_J \mid \gamma_0, c, q_1, \ldots, q_J) = \frac{p(\{X_j\}_{1,J} \mid \gamma_0, c, q_1, \ldots, q_J)}{K_J!} = \frac{\gamma_0^{K_J} \exp\left[-\gamma_0 \ln(\frac{q_\cdot + c}{c})\right]}{K_J!} \left[\prod_{k=1}^{K_J} \frac{\frac{\Gamma(n_{\cdot k})}{(q_\cdot + c)^{n_{\cdot k}}}}{\prod_{j=1}^{J} n_{jk}!}\right] \left[\prod_{j=1}^{J} q_j^{n_j}\right] . \tag{2.3}$$

One may verify by straightforward calculation that a scaled NBP random count matrix with the probability mass function (PMF) shown in (2.3) can be generated column by column as i.i.d. count vectors:

$$\begin{aligned} &\boldsymbol{n}_{\cdot k} \sim \text{Multinomial}(n_{\cdot k}, q_1/q_\cdot, \ldots, q_J/q_\cdot), \\ &n_{\cdot k} \sim \text{Logarithmic}[q_\cdot/(c + q_\cdot)], \\ &K_J \sim \text{Pois}\left\{\gamma_0 \left[\ln(c + q_\cdot) - \ln(c)\right]\right\} . \end{aligned} \tag{2.4}$$

It is clear from (2.4) that the columns of $\mathbf{N}_J$ are i.i.d. multivariate count vectors, which all follow the same logarithmic-multinomial (mixture) distribution. Thus the scaled NBP random count matrix $\mathbf{N}_J$ is column exchangeable. It is also row exchangeable if and only if the q_j are the same for all $j \in \{1, \ldots, J\}$.

Now consider the row-wise sequential construction of the scaled NBP random matrix. With the prior on $\mathbf{N}_J \in \mathbb{Z}^{J \times K_J}$ well defined, straightforward calculations using (2.4) yield the following form for this prediction rule, expressed in terms of familiar PMFs:

$$\begin{aligned} \frac{f(\mathbf{N}_{J+1} \,|\, \boldsymbol{\theta})}{f(\mathbf{N}_J \,|\, \boldsymbol{\theta})} = \frac{K_J! K_{J+1}^+!}{K_{J+1}!} &\prod_{k=1}^{K_J} \text{NB}\left(n_{(J+1)k}; n_{\cdot k}, \frac{q_{J+1}}{c + q_\cdot + q_{J+1}}\right) \\ &\times \prod_{k=K_J+1}^{K_{J+1}} \text{Logarithmic}\left(n_{(J+1)k}; \frac{q_{J+1}}{c + q_\cdot + q_{J+1}}\right) \\ &\times \text{Pois}\left\{K_{J+1}^+; \gamma_0 \left[\ln(c + q_\cdot + q_{J+1}) - \ln(c + q_\cdot)\right]\right\}, \end{aligned} \tag{2.5}$$

where $\boldsymbol{\theta} := \{\gamma_0, c, q_1, \ldots, q_J\}$. This formula indicates that, to add a new row to $\mathbf{N}_J \in \mathbb{Z}^{J\times K_J}$, we first draw count NB$[n_{\cdot k}, q_{J+1}/(c + q_\cdot + q_{J+1})]$ at each existing column. We then draw K^+_{J+1} new columns as $K^+_{J+1} \sim$ Pois$\{\gamma_0 [\ln(c + q_\cdot + q_{J+1}) - \ln(c + q_\cdot)]\}$. Finally, each entry in the new columns has a Logarithmic $\left[n_{(J+1)k}; q_{J+1}/(c + q_\cdot + q_{J+1})\right]$ distributed random count. It is clear in the sequential construction of the scaled NBP random count matrix, for a point of discontinuity $\omega_k \in \mathcal{D}_J$, the variance and mean are related as

$$\text{var}[n_{(J+1)k}] = \mathbb{E}[n_{(J+1)k}] + \frac{\mathbb{E}^2[n_{(J+1)k}]}{n_{\cdot k}}. \tag{2.6}$$

Since $n_{\cdot k}$, the total count of gene k of all the J samples of the group, is fixed, the above equation indicates a variance and mean relationship that does not change.

2.1.1.1 Inference for the scaled NBP

The parameters of the scaled NBP can be inferred using Gibbs sampling with closed-form update equations. Using likelihoods (2.2) and (2.3), with $\gamma_0 \sim$ Gamma$(e_0, 1/f_0)$, $c \sim$ Gamma$(c_0, 1/d_0)$, and $q_j \sim$ Gamma$(a_0, 1/b_0)$ in the prior, each Gibbs sampling iteration proceeds as

$$\begin{aligned}
&(\gamma_0 \,|\, -) \sim \text{Gamma}\Big(e_0 + K_J, \frac{1}{f_0 - \ln(\frac{c}{c+q_\cdot})}\Big),\\
&(r_k \,|\, -) \sim \text{Gamma}[n_{\cdot k}, 1/(c + q_\cdot)],\\
&[G(\Omega \setminus \mathcal{D}_J) \,|\, -] \sim \text{Gamma}[\gamma_0, 1/(c + q_\cdot)],\\
&(q_j \,|\, -) \sim \text{Gamma}\{a_0 + n_j, 1/[b_0 + G(\Omega)]\},\\
&(c \,|\, -) \sim \text{Gamma}\{c_0 + \gamma_0, 1/[d_0 + G(\Omega)]\},
\end{aligned} \tag{2.7}$$

where $G(\Omega) := G(\Omega\backslash\mathcal{D}_J) + \sum_{k=1}^{K_J} r_k$, given which the total gene count for sample j follows Poisson$[q_j G(\Omega)]$. Note that a gene that has at least one nonzero count among the J samples will be attached to a discrete atom (point of discontinuity) of the gamma process with weight r_k, while all the other countably infinite unexpressed genes are associated with the atoms in the absolute continuous space $\Omega\backslash\mathcal{D}_J$, whose total weight is $G(\Omega\backslash\mathcal{D}_J)$.

2.1.1.2 NBP-Seq differential expression analysis

To detect differentially expressed genes using the scaled NBP, we notice in the prior that

$$\mathbb{E}[n_{jk} \mid q_j, G] = \text{var}[n_{jk} \mid q_j, G] = q_j r_k$$

and in the conditional posterior shown in (2.7) that

$$\mathbb{E}[r_k \mid -] = n_{\cdot k}/(c + q_\cdot), \quad \mathbb{E}[q_j \mid -] = (a_0 + n_j)/[b_0 + G(\Omega)]. \tag{2.8}$$

Thus one may consider r_k as a gene-specific Poisson rate parameter that indicates the expression level of gene k, whose conditional posterior is related to both $n_{\cdot k}$, the total count of gene k across all the J samples of the group, and $q_\cdot$, the total sum of the sample-specific gamma distributed scaling parameters; one may consider q_j as a scaling factor to be inferred from the data, whose conditional posterior is determined not only by n_j, the total count of all genes in sample j that indicates the sequencing depth of sample j, but also by $G(\Omega)$, the total sum of all countably infinite gene-specific Poisson rate parameters; and the conditional posterior of γ_0 is clearly related to K_J, the total number of expressed genes in the group. Therefore, the scaled NBP borrows statistical strength across both the genes and samples to infer the conditional posterior of r_k.

To assess whether the difference between the expressions of the same gene at different sample groups is statistically significant, we collect posterior Markov chain Monte Carlo (MCMC) samples for each r_k in each group, and use these MCMC samples to measure the distance between the posterior distributions of the r_k of the same gene across different groups. Note that for a gene whose total count across all samples in a group is zero, the posterior values of its r_k would be fixed at 0.

Instead of using the scaled NBP that introduces q_j to model sample-specific sequencing depths, we also consider the original NBP of (38) with all q_j fixed at one. To compensate for the variations of sequencing depths between samples, for the original NBP, we normalize the inferred Poisson

rates r_k and use them to evaluate the significance of differential gene expressions.

2.1.2 GNBP-Seq: Gamma-negative binomial process for RNA-seq

To generate the random count matrix $\mathbf{N}_J$ in a group, we construct a gamma-negative binomial process (GNBP) (38) as

$$X_j \,|\, G \sim \text{NBP}(G, p_j), \quad G \sim \Gamma\text{P}(G_0, 1/c), \tag{2.9}$$

where $j \in \{1, .., J\}$ and $X_j \,|\, G \sim \text{NBP}(G, p_j)$ is defined as a NBP such that $X_j(A) \sim \text{NB}[G(A), p_j]$ for each Borel subset $A \subset \Omega$. Note that $X_j \,|\, G \sim \text{NBP}(G, p_j)$ can also be augmented as a gamma process mixed sum-logarithmic process (SumLogP) as

$$X_j \,|\, L_j \sim \text{SumLogP}(L_j, p_j),\ L_j \,|\, G \sim \text{PP}(q_j G), \tag{2.10}$$

where $q_j := -\ln(1 - p_j)$, *i.e.*, $p_j = 1 - e^{-q_j}$, and the SumLogP is defined in (38) such that $X_j(A) = \sum_{t=1}^{L_j(A)} u_t,\ u_t \sim \text{Logarithmic}(p_j)$ for each Borel subset $A \subset \Omega$. Thus the GNBP also can be expressed as a NBP mixed SumLogP as

$$X_j \,|\, L_j \sim \text{SumLogP}(L_j, p_j),\ (L_1, \ldots, L_J) \sim \text{NBP}(G_0, c, q_1, \ldots, q_J). \tag{2.11}$$

With a draw from the gamma process G expressed as $G = \sum_{k=1}^{\infty} r_k \delta_{\omega_k}$, a draw from X_j can be expressed as

$$X_j = \sum_{k=1}^{\infty} n_{jk} \delta_{\omega_k},\ n_{jk} \sim \text{NB}(r_k, p_j). \tag{2.12}$$

The GNBP employs sample-specific NB probability parameters p_j to model row heterogeneity. In the context of RNA-seq data, the variations of p_j can be used to account for those of sequencing depths.

Both the row-wise and column-wise constructions of the GNBP random count matrix mimic these of the NBP random count matrix. They are described in detail in (38) and hence omitted

here for brevity. We mention that the two key differences in their row-wise sequential constructions are that the GNBP uses the gamma-NB instead of NB distributions to model the counts at previously expressed genes brought by a new sample, and the GNBP uses the logarithmic mixed sum-logarithmic instead of logarithmic distributions to model the counts at newly expressed genes brought by a new sample.

As shown in (38), in the sequential construction of the GNBP random count matrix, for a point of discontinuity $\omega_k \in \mathcal{D}_J$, the variance and mean are related as

$$\text{var}[n_{(J+1)k}] = \frac{\mathbb{E}[n_{(J+1)k}]}{1 - p_{J+1}} + \frac{\mathbb{E}^2[n_{(J+1)k}]}{l_{\cdot k}}, \tag{2.13}$$

which depends on both p_{J+1} and $l_{\cdot k}$ that are random, where $l_{\cdot k} := \sum_{j=1}^{J} l_{jk}$, and $l_{jk} \sim \text{CRT}(n_{jk}, r_k)$ is the Chinese Restaurant Table (CRT) distribution. Comparing (2.6) and (2.13), it is clear that since $p_{J+1} < 1$ and $l_{\cdot k} \leq n_{\cdot k}$, the GNBP can model much more over-dispersed counts than the NBP.

2.1.2.1 Inference for the GNBP

Letting $\gamma_0 \sim \text{Gamma}(e_0, 1/f_0)$, $p_j \sim \text{Beta}(a_0, b_0)$, and $c \sim \text{Gamma}(c_0, 1/d_0)$ in the prior, as in (38), a Gibbs sampling iteration for the GNBP proceeds as

$$\begin{aligned}
&(\gamma_0 \,|\, -) \sim \text{Gamma}\bigg(e_0 + K_J, \frac{1}{f_0 - \ln(\frac{c}{c+q_\cdot})}\bigg),\\
&(l_{jk} \,|\, -) \sim \text{CRT}(n_{jk}, r_k),\ (r_k \,|\, -) \sim \text{Gamma}\big[l_{\cdot k}, 1/(c + q_\cdot)\big],\\
&\{G(\Omega \backslash \mathcal{D}_J) \,|\, -\} \sim \text{Gamma}\big[\gamma_0, 1/(c + q_\cdot)\big],\\
&(p_j \,|\, -) \sim \text{Beta}\big[a_0 + n_j, b_0 + G(\Omega)\big],\\
&(c \,|\, -) \sim \text{Gamma}\big\{c_0 + \gamma_0, 1/[d_0 + G(\Omega)]\big\}.
\end{aligned} \tag{2.14}$$

Note that given $G(\Omega)$, the total gene count for sample j follows $\text{NB}[G(\Omega), p_j]$.

2.1.2.2 GNBP-Seq differential expression analysis

In the GNBP, since in the prior we have

$$\mathbb{E}[n_{jk} \,|\, G, p_j] = r_k \frac{p_j}{1-p_j},$$

$$\text{var}[n_{jk} \,|\, G, p_j] = r_k \frac{p_j}{(1-p_j)^2} = \mathbb{E}[n_{jk} \,|\, G, p_j] + r_k^{-1} \mathbb{E}^2[n_{jk} \,|\, G, p_j],$$

and in the conditional posterior, if $b_0 + G(\Omega) > 1$, we have

$$\mathbb{E}[r_k \,|\, -] = l_{\cdot k}/(c + q_\cdot), \quad \mathbb{E}[p_j/(1-p_j) \,|\, -] = (a_0 + n_j)/[b_0 + G(\Omega) - 1]. \tag{2.15}$$

Thus one may interpret $p_j/(1-p_j)$ as a term that accounts for the sequencing depth of sample j, and may compare the posterior distributions of the NB shape parameter r_k of the same gene at different groups to assess differential expression of that gene. The conditional posterior of the scaling factor $p_j/(1-p_j)$ is determined by not only n_j, the total counts of genes in sample j, but also $G(\Omega)$, the total sum of all countably infinite gene-specific NB shape parameters; and the conditional expectation of r_k is related to both $l_{\cdot}$ and $q_\cdot$, which aggregate their corresponding sample-specific values across all the J samples. Therefore, the GNBP borrows statistical strength across both the genes and samples to infer the conditional posterior of r_k. For an unexpressed gene, whose total count across all samples in a group is 0, the posterior values of its r_k would be fixed at 0.

Comparing (2.8) and (2.15) shows that both the GNBP and scaled NBP have similar sample-specific scaling parameters, but, as in (2.14), since $\mathbb{E}[l_{jk} \,|\, -] = \sum_{t=1}^{n_{jk}} r_k/(r_k + t - 1)$ and hence $\mathbb{E}[l_{jk} \,|\, -] \approx r_k \ln(n_{jk} + r_k)$ for large n_{jk}, the posteriors of the gene-specific parameters r_k in the GNBP would be impacted much less by some genes whose expressions n_{jk} are significantly larger than their mean expression levels, which are commonly observed in genomic studies.

2.1.3 BNBP-Seq: Beta-negative binomial process for RNA-seq

Similar to the GNBP, the BNBP can be used to model RNA-seq samples. The BNBP can be constructed by sharing the NB probability parameters across the J sequencing samples of the same group as

$$X_j \,|\, r_j, B \sim \text{NBP}(r_j, B), \quad B \sim \text{BP}(c, B_0), \tag{2.16}$$

where $j \in \{1, \ldots, J\}$ and $B \sim \text{BP}(c, B_0)$ is a beta process with a finite and continuous base measure B_0 over Ω and a concentration parameter c, with Lévy measure

$$\nu(dpd\omega) = p^{-1}(1-p)^{c-1}dpB_0(d\omega). \tag{2.17}$$

With a draw from the beta process B expressed as $B = \sum_{k=1}^{\infty} p_k \delta_{\omega_k}$, where ω_k and p_k are atoms and their associated probability weights, respectively, a draw from X_j given B can be expressed as

$$X_j = \sum_{k=1}^{\infty} n_{jk}\delta_{\omega_k},\ n_{jk} \sim \text{NB}(r_j, p_k). \tag{2.18}$$

In the BNBP, different r_j's are used to model the sequencing depth variations.

Both the row-wise and column-wise constructions of the BNBP random count matrix, as described in detail in (38) and hence omitted here for brevity, mimic these of the scaled NBP random count matrix. We mention that the two key differences in their row-wise sequential constructions are that the BNBP uses the beta-NB instead of NB distributions to model the counts at previously expressed genes brought by a new sample, and the BNBP uses the digamma instead of logarithmic distributions to model the counts at newly expressed genes brought by a new sample.

As shown in (38), in the sequential construction of the BNBP random count matrix, for a point of discontinuity ω_k, the variance and mean are related as

$$\text{var}[n_{(J+1)k}] = \frac{\mathbb{E}[n_{(J+1)k}]}{\frac{c+r_\cdot-2}{n_{\cdot k}+c+r_\cdot-1}} + \frac{\mathbb{E}^2[n_{(J+1)k}]}{\frac{n_{\cdot k}(c+r_\cdot-2)}{n_{\cdot k}+c+r_\cdot-1}}, \tag{2.19}$$

which depends on both c and $r_{\cdot}$ that are random. Comparing (2.6) and (2.19), it is clear that since $\frac{c+r_{\cdot}-2}{n_{\cdot k}+c+r_{\cdot}-1} \leq 1$ and $\frac{n_{\cdot k}(c+r_{\cdot}-2)}{n_{\cdot k}+c+r_{\cdot}-1} < n_{\cdot k}$ for $c+r_{\cdot} > 2$, similar to the GNBP, the BNBP can also model much more over-dispersed counts than the scaled NBP.

The variance-mean relationships expressed by (2.6), (2.13), and (2.19) show that the GNBP and BNBP can model much more over-dispersed counts than the (scaled) NBP, and as shown in Figure 1 of (38), given the same expected total count, while the counts in NBP random count matrices usually have small dynamic ranges, the counts in both the GNBP and BNBP matrices can contain values that are significantly above the average. In RNA-seq, it is common to have large dynamical range for highly over-dispersed gene counts, which are likely to be better modeled by both the GNBP and BNBP than by the (scaled) NBP, as confirmed by our experiments in Section 2.2.

2.1.3.1 Inference for the BNBP

Letting $\gamma_0 \sim \text{Gamma}(e_0, 1/f_0)$, $p_j \sim \text{Beta}(a_0, b_0)$, and $c \sim \text{Gamma}(c_0, 1/d_0)$, as in (38), a Gibbs sampling iteration for the BNBP proceeds as

$$
\begin{aligned}
&(\gamma_0 \,|\, -) \sim \text{Gamma}\bigg(e_0 + K_J, \frac{1}{f_0 + \psi(c + r_{\cdot}) - \psi(c)}\bigg),\\
&(p_k \,|\, -) \sim \text{Beta}(n_{\cdot k}, c + r_{\cdot}),\ (p_* \,|\, -) \sim \text{logBeta}(\gamma_0, c + r_{\cdot}),\\
&(l_{jk}|-) \sim \text{CRT}(n_{jk}, r_j),\\
&(r_j \,|\, -) \sim \text{Gamma}\bigg(a_0 + l_{j\cdot}, \frac{1}{b_0 + p_* - \sum_{k=1}^{K_J} \ln(1 - p_k)}\bigg).
\end{aligned}
\tag{2.20}
$$

Inside each Gibbs sampling iteration, as in (38), an independence chain Metropolis-Hastings sampling step can be used to update the concentration parameter c.

2.1.3.2 BNBP-Seq differential expression analysis

In the BNBP, since in the prior we have

$$
\mathbb{E}[n_{jk} \,|\, r_j, B] = r_j \frac{p_k}{1 - p_k},\ \ \text{var}[n_{jk} \,|\, r_j, B] = r_j \frac{p_k}{(1 - p_k)^2} = (1 - p_k)^{-1} \mathbb{E}[n_{jk} \,|\, r_j, B], \tag{2.21}
$$

and in the conditional posterior, if $c + r_{\cdot} > 1$, we have

$$\mathbb{E}[p_k/(1-p_k) \,|\, -] = n_{\cdot k}/(c + r_{\cdot} - 1), \;\; \mathbb{E}[r_j \,|\, -] = \frac{a_0 + l_{j\cdot}}{b_0 + p_* - \sum_{k=1}^{K_J} \ln(1-p_k)}. \tag{2.22}$$

Thus one may consider that the NB sample-specific shape parameter r_j accounts for the sequencing depth of sample j, and may compare the posterior distributions of $p_k/(1-p_k)$ to evaluate differential expression of gene k between different groups. The posterior expectation of r_j in the BNBP is related to the NB probability parameters of all genes, which themselves are related to $r_{\cdot}$, the aggregation of the sample-specific scaling factors across all J samples. Thus the BNBP borrows statistical strength across all the genes and samples to infer the posterior distribution of $p_k/(1-p_k)$. Note that for an unexpressed gene, whose total count across all samples in a group is 0, the posterior values of its p_k would be fixed at 0.

Comparing (2.8) and (2.22) shows that the BNBP and scaled NBP have similar gene-specific parameters, but, as in (2.20), since $\mathbb{E}[l_{jk} \,|\, -] \approx r_j \ln(n_{jk} + r_j)$ for large n_{jk}, for some genes whose expressions n_{jk} are significantly larger than the mean expression levels, the posteriors of the sample-specific parameters r_j in the BNBP also would be impacted much less than these of the sample-specific parameters q_j in the scaled NBP.

2.1.4 Distance between posterior distributions

In order to compare the posterior distributions, we use the symmetric Kullback-Leibler (KL) divergence defined between two discrete distributions P and Q as

$$KL(P,Q) = \sum\nolimits_x \big[p(x) - q(x)\big] \log\big[p(x)/q(x)\big].$$

Supposing r is the parameter to be compared between two different groups, we estimate the symmetric KL-divergence between the posterior distributions of $r^{(1)}$ and $r^{(2)}$, the values of r of the first and second groups, respectively, using collected MCMC samples. We first find both the minimum and maximum values of the MCMC samples of r across both groups to define an interval

for r. After adjusting the lower- and upper-limits of the interval as $[\max(0, Q_1 - 1.5 * Q_\triangle),\ Q_3 + 1.5 * Q_\triangle]$, where Q_1 and Q_3 are 25% and 75% quantiles and $Q_\triangle = Q_3 - Q_1$, we equally divide the adjusted interval into $N = 100$ bins. For each group, we count the number of MCMC samples falling into each bin, and then normalize these bin counts to a 100 dimensional discrete probability vector, referred to as $\boldsymbol{\pi}^{(1)}$ and $\boldsymbol{\pi}^{(2)}$ for the first and second groups, respectively. Finally, with a small constant set as $\epsilon = 10^{-10}$, we calculate the symmetric KL-divergence as

$$KL(\boldsymbol{\pi}^{(1)}, \boldsymbol{\pi}^{(2)}) = \sum_{i=1}^{N} \left(\pi_i^{(1)} - \pi_i^{(2)}\right) \log\left(\frac{\pi_i^{(1)} + \epsilon}{\pi_i^{(2)} + \epsilon}\right). \tag{2.23}$$

2.2 Experimental Results

To evaluate the proposed BNP differential expression analysis algorithms, we compare their performance on both synthetic and real-world benchmark RNA-seq data with those of edgeR (19), DESeq (7), and baySeq (33), three widely used algorithms in biomedical studies. We also present a case study on *clear cell renal cell carcinoma* (ccRCC) (46), explaining the biomedical implications obtained by differential expression analysis using both our GNBP and BNBP methods. We first consider synthetic RNA-seq data generated under different models, and we show that the proposed GNBP and BNBP differential expression analysis algorithms consistently provide outstanding performance. We then consider the real-world benchmark RNA-seq data extracted from the SEquencing Quality Control (SEQC) project (47; 48) and the ccRCC case study extracted from The Cancer Genome Atlas (TCGA) (49). We consider the RNA-seq data from both Beijing Genomics Institute (BGI) and the Pennsylvania State University (PSU) provided in the SEQC project (47; 48), available in the R package SEQC on Bioconductor (50). Both the BGI and PSU datasets, which are the transcriptomic expression measurements of the RNA samples prepared at the same biological conditions but sequenced at different sequencing sites, contain the counts for approximately 26,000 genes. In our experiments, we employ sample groups A and B, which are derived from Agilent's Universal Human Reference RNA and Life Technologies' Human Brain Reference RNA cell lines, respectively. We collect the counts from the first flow cells of the sequencing

machines on five replicates for each group (condition).

On both synthetic and real-world RNA-seq count data, comparison of both the area under the receiver operating characteristic (ROC) curve (AUC-ROC) and area under the precision-recall (PR) curve (AUC-PR) shows that the proposed GNBP and BNBP algorithms clearly outperform the (scaled) NBP and previously proposed differential expression analysis algorithms, as described below in detail.

2.2.1 Synthetic data

We first generate synthetic RNA-seq data with the GNBP generative model, the BNBP generative model, or the NB distribution based procedure adopted in baySeq (33). For each setting, to make the synthetic data closely resemble real-world RNA-seq data, we first infer the parameters of the corresponding model on the BGI or PSU datasets from SEQC, and then generate synthetic sequencing counts using these inferred model parameters. To simulate samples from two different groups (conditions), each of which has 10,000 genes in five replicates, we randomly select 10% of the genes and set them to be differentially expressed between the two groups, with the fold change of differentially expressed genes chosen as an adjustable parameter. For quality control, we discard the bottom 10% of genes with low expressions across groups in data generation. In order to produce both up- and down-regulated differentially expressed genes, each differentially expressed gene is randomly set to be either up- or down-regulated. Below we denote $b > 1$ as the fold change to be set. We use the PSU dataset for the baySeq setting and the BGI dataset for both the GNBP and BNBP settings. Using different datasets to infer model parameters and different models to generate synthetic datasets allows us to assess the robustness of various methods in different practical settings.

In the GNBP setting, if gene k is up-regulated, then we generate its counts using $\mathrm{NB}(r_k, p_j)$ and $\mathrm{NB}(b\, r_k, p_j)$ for the samples in the first and second groups, respectively; whereas if gene k is down-regulated, then we generate its counts using $\mathrm{NB}(r_k, p_j)$ and $\mathrm{NB}(r_k/b, p_j)$ for the five samples in the first and second groups, respectively.

In the BNBP setting, if gene k is up-regulated, then we generate its counts using $\mathrm{NB}(r_j, p_k)$

and $\text{NB}(r_j, p'_k)$, where p'_k is selected to satisfy $bp_k/(1-p_k) = p'_k/(1-p'_k)$, for the samples in the first and second groups, respectively; whereas if gene k is down-regulated, then we generate its counts using $\text{NB}(r_j, \tilde{p}_k)$ and $\text{NB}(r_j, p_k)$, where $\tilde{p}_k$ is selected to satisfy $p_k/(1-p_k) = b\tilde{p}_k/(1-\tilde{p}_k)$, for the five samples in the first and second groups, respectively.

In the baySeq setting of (33) that generates a count from a NB distribution given its mean and dispersion parameters, if gene k is up-regulated, then we generate its counts using μ_k and $b\mu_k$ as the means for the first and second groups, respectively; whereas if gene k is down-regulated, then we generate its counts using μ_k and μ_k/b as the means for the first and second groups, respectively.

We infer the model parameters via Gibbs sampling for the proposed BNP differential expression analysis algorithms. For each algorithm, we collect 1,000 MCMC samples after 1,000 burn-in iterations. The example MCMC sample trace plots in Figure 2.1 suggest that the Markov chains for both the GNBP and BNBP methods converge fast and mix well, supporting the practice of performing downstream analysis with 2,000 MCMC iterations. For the analysis of the real-world dataset BGI on a single cluster node with Intel Xeon 2.5GHz E5-2670 v2 processor, it took around two hours for both the GNBP and BNBP methods with 2,000 MCMC iterations, about ten minutes for the other methods, including the NBP. Note that parallelization could further speed up the inference. We use the collected MCMC samples to calculate the symmetric KL divergence, as in (3.11), between two groups for each gene, and rank the genes according to these values. For edgeR and DESeq, we follow the standard analysis pipelines and rank the genes using the computed p-values; and for baySeq, we rank the genes using model likelihoods. We set the fold change b as 1.4, 1.6, 1.8, or 2 in simulating synthetic data to assess how sensitive the algorithms under study are to different levels of differential expression. For each fold change, we report the results of each algorithm based on ten independent random trials.

For these three different types of synthetic data, as shown in Figure 2.2, measured by both AUC-ROC and AUC-PR, baySeq has the worst overall performance even when the synthetic data are generated based on its model assumption, followed by the scaled NBP; the NBP, DESeq, and edgeR all have similar performance; and the GNBP and BNBP clearly outperform all the other

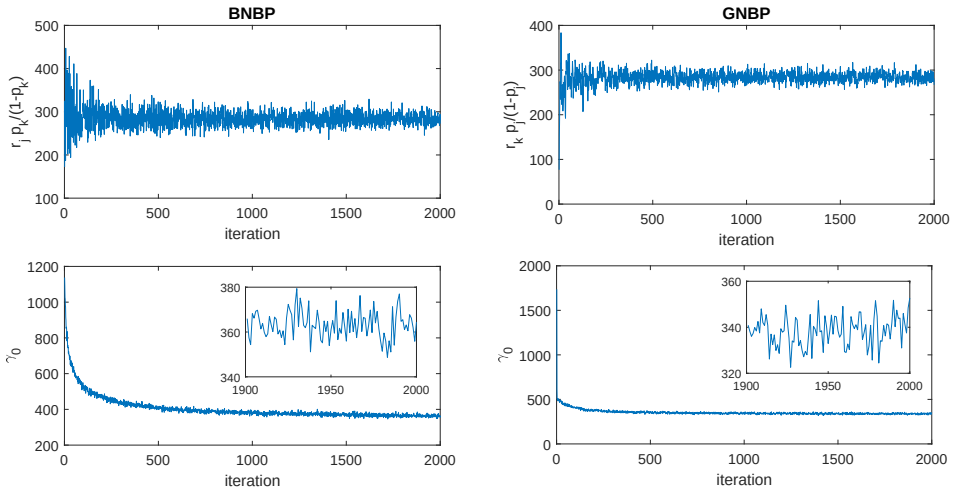


Figure 2.1: Trace plots of 2000 MCMC samples for example parameters of the BNBP (left column) and GNBP (right column) methods, applied to the BGI dataset.

differential expression analysis algorithms. To further compare the operating characteristics of different algorithms, we show in Figure 2.3 the full ROC and PR curves for the fold change of $b = 1.8$.

More carefully examining Figures 2.2 and 2.3, it is interesting to notice that for the synthetic data generated with either the GNBP or BNBP, the scaled NBP, which extends the original NBP with sample-specific scaling parameters q_j to model sample sequencing depth variations, in fact clearly underperforms the original NBP. Suggesting that explicitly modeling the sample sequencing depths, using the gamma-Poisson construction of the scaled NBP, is insufficient to model the over-dispersed gene counts generated using the gamma- or beta-NB constructions.

While the original NBP fixes $q_j = 1$ and hence does not explicitly model the sample sequencing depth variations, it performs as well as both DESeq and edgeR in all three settings, which may be explained by the fact that it normalizes the posterior Poisson rates before applying them to compare the gene expression levels between two groups, a post-processing step that plays a similar role as the pre-processing normalization steps used in both DESeq and edgeR to account for different sequencing depths.

It is also interesting to notice that while the GNBP consistently ranks the best or very close to

the best, in terms of both AUC-ROC and AUC-PR, the BNBP does so only in terms of AUC-ROC. For synthetic data generated using the GNBP and baySeq, the performance of the BNBP in terms of AUC-PR quickly deteriorates as the fold change reduces from 1.8 to 1.4, suggesting a large number of false positives among the top ranked genes of the BNBP when the fold change is not sufficiently large for the GNBP synthetic data. The disparity between the performance measured by AUC-ROC and that measured by AUC-PR, which only happens for the BNBP, indicates that the BNBP employs a distinct mechanism to detect differentially expressed genes, as carefully discussed below.

To compare the expression levels of the kth gene between two groups, the GNBP compares the posterior NB shape parameters r_k, whereas the BNBP compares the posterior NB probability parameters p_k. One may consider that the expression level of gene k is assumed to roughly follow a smooth function of the shape parameter r_k in the GNBP, and a smooth function of $p_k/(1-p_k)$ in the BNBP. The difference between the posterior NB shape parameters r_k explains the differences between both the means and dispersions, but does not explain that of the variance-to-mean ratios (VMR), of the counts of gene k at different groups, since if $n_{jk} \sim \text{NB}(r_k, p_j)$, then $\mathbb{E}[n_{jk}] = r_k p_j/(1-p_j)$, $\text{var}[n_{jk}] = \mathbb{E}[n_{jk}] + (\mathbb{E}[n_{jk}])^2/r_k$, and $\text{VMR}[n_{jk}] = 1 + \mathbb{E}[n_{jk}]/r_k$; whereas the difference between the posterior NB probability parameters p_k explains the differences between both the means and VMRs of the counts of gene k at different groups, since if $n_{jk} \sim \text{NB}(r_j, p_k)$, then $\mathbb{E}[n_{jk}] = r_j p_k/(1-p_k)$, $\text{var}[n_{jk}] = \mathbb{E}[n_{jk}] + (\mathbb{E}[n_{jk}])^2/r_j$, and $\text{VMR}[n_{jk}] = 1/(1-p_k)$. Therefore, for the counts of a gene generated with the GNBP, if the r_k is small, a small change in its value may lead to a significant change of $\text{VMR}[n_{jk}] = 1 + (\mathbb{E}[n_{jk}])/r_k$, which implies that a large difference in a gene's VMRs between two groups may not be taken by the GNBP as a strong evidence for differential expression. By contrast, since the gene-specific parameter p_k in the BNBP is explicitly responsible for the VMR, a large difference in a gene's VMRs between two groups may encourage the BNBP to rank that gene as strongly differentially expressed, which may be used to explain why the BNBP has good AUC-ROC but poor AUC-PR if the fold change is small for the GNBP synthetic data. In practice, however, it is often unclear whether it is the change

of the quadratic relationship between the variance and mean, as captured by the NB dispersion parameter, or the VMR, as captured by the NB probability parameter, that is responsible for the change of a gene's expression level. Thus it is often unclear whether the GNBP or BNBP would be a better choice for a real dataset, and it seems promising to combine the advantages of both for differential expression analysis, an attractive research topic beyond the scope of the paper that is to be investigated in our future study.

To more comprehensively evaluate the proposed methods, we consider several additional application scenarios. The performance comparisons with baySeq synthetic data under these different scenarios are shown in Figure 2.4. We first assess the sensitivity of different methods to the ratio of up- and down-regulated genes among the set of truly differentially expressed genes, which, the same as before, take 10% of the total number of genes. We assume a fold change of 2 for these truly differently expressed genes, and vary the percentage of up-regulated (down-regulated) genes among them from 20% (80%) to 40% (60%), 60% (40%), and 80% (20%). As shown in Figure 2.4(a), while the GNBP, BNBP, edgeR, and DESeq all show robustness to the change of that percentage, the performance of both the NBP based and baySeq methods significantly deteriorates as one increases the imbalance between the numbers of up- and down-regulated genes. We also note that both the GNBP and BNBP successfully preserve their out-performance margins under various ratios of up- to down-regulated genes.

To examine how the performance changes with the sample size, we consider increasing the number of samples for each group from 4, to 8, 12, and 16. This is a sensible choice, since in practice the number of samples per condition is often smaller than 16. In this experiment, 10% of genes are equally likely to be up- or down-regulated with a fold change of 2. Figure 2.4(b) illustrates the error bar plots for both the AUC of ROC curve and that of PR curve, under different sample sizes over 10 random trials. All methods show consistent improvements as the number of replicates in each group increases, which agrees with the expectation that more samples provide more information to assist parameter inference. In addition, we consider 100 genes with different sample sizes to investigate the performance of the proposed methods in the setting with a large

sample size but a small number of genes. Similar to previous simulations, 10% of the genes are assumed to be differentially expressed with a fold change of 2, and the number of replicates in each group is increased from 4 to 6, 8, 10, 20, 40, 60, 80, and 100. Figure 2.5 shows the error bar plots for both the AUC of ROC curve and that of PR curve, under different sample sizes over 10 random trials. As expected, adding more samples consistently enhances the recovery of true differential expression state of the genes for all methods, and when the number of samples reaches 100, almost all methods perform perfectly.

Last but not least, Figure 2.4(c) shows the box plots of the AUCs of ROC and PR curves when the true fold change of differentially expressed genes is uniformly distributed within the interval $[1.4, 2]$. The BNBP stands out as the best performing method followed by the GNBP, suggesting that the superior results of the proposed methods in previous simulations do not rely on setting the fold change to a fixed constant.

2.2.2 SEQC benchmark RNA-seq data and case study

In order to characterize various RNA-seq technologies and quantification pipelines in the SEQC project (47; 48), the same RNA samples for a comprehensive group of control genes are analyzed based on quantitative Reverse Transcription Polymerase Chain Reaction (qRT-PCR) using TaqMan assays (51), which is referred as the TaqMan benchmark data (52; 53). For sample groups A and B, the expression intensity values of 955 selected control genes have been derived in the TaqMan qPT-PCR analysis for sequencing benchmarking. Without knowing in practice which genes are truly differentially expressed between different conditions, we consider thresholding the qRT-PCR expression ratios between different conditions at a certain value to define the ground-truth set of differentially expressed genes. Based on these 955 genes in the TaqMan data, we evaluate the performance of different differential expression analysis pipelines. Note that although the replicates in SEQC are technical, they show notable amount of over-dispersion and have been used in the literature as a standard benchmark for assessing differential expression analysis tools (29).

While it is unknown which genes are truly differentially expressed for both the BGI and PSU RNA-seq data, we rely on the qRT-PCR expression intensity of the 955 genes in the TaqMan data

and set different cut-offs for the binary logarithm (log2) of the qRT-PCR expression ratio to define "truly" differentially expressed genes. We increase this log2 cut-off value gradually from 0.5 to 2, and calculate both AUC-ROC and AUC-PR. The symmetric KL divergence is used to assess differential expression. As shown in Figure 2.6 for both the BGI and PSU datasets, the GNBP and BNBP outperform all the other methods in both ROC and PR analyses with significant margins. Note that the performance gains of the GNBP and BNBP over the other methods become more significant as one increases the log2 cut-off for the qRT-PCR expression ratio, which reduces the number of genes that are considered as truly differentially expressed.

Comparing Figure 2.2 for synthetic data with Figure 2.6 for real-world data, one may notice that while both the AUC-ROC and AUC-PR curves in Figure 2.2 seem to monotonically increase as the fold change increases, the AUC-ROC and AUC-PR curves in Figure 2.6 do not necessarily share similar trends. It is not surprising to observe these seemingly distinct behaviors, since for the synthetic data in Figure 2.2, the set of truly differentially expressed genes are fixed and known exactly, remaining unchanged regardless of how one sets the fold change that is used to detect differentially expressed genes, whereas for the real-world data in Figure 2.6, the number of genes considered as truly expressed reduces as the cut-off value of the qRT-PCR expression ratio increases. In addition, we note that the results of edgeR, DESeq, and baySeq on both the BGI and PSU real datasets reported in this paper are similar to those reported in (29).

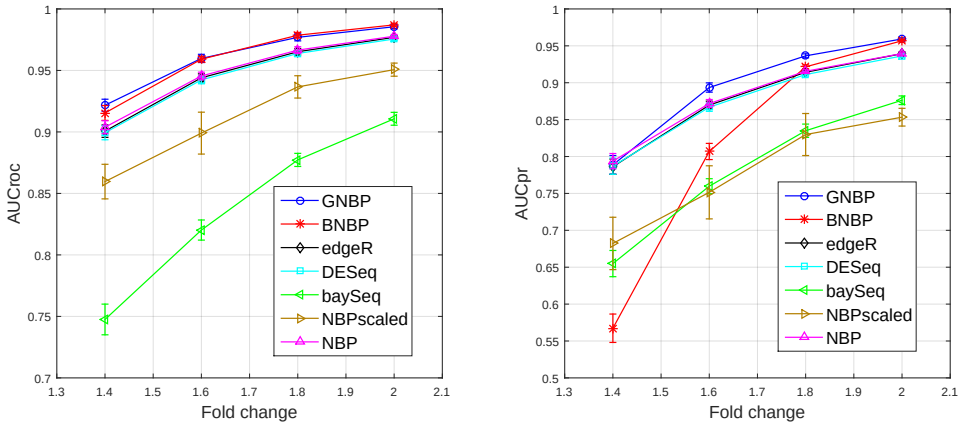
To investigate the experimental results more thoroughly, we fix the true positives and negatives at the log2 cut-off value of 2 and illustrate the ROC and PR curves for BGI dataset in Figure 2.7. In addition, we show in Table 2.1 the area under the ROC curve for the range with $FPR \leq 0.1$ and area under the PR curve for the range with Recall ≤ 0.1 for various algorithms. It is clear that both the GNBP and BNBP not only have higher AUC-ROC and AUC-PR, but also outperform all the other methods in almost all regions of the ROC and PR curves.

In addition to showing the ROC and PR curves, we also plot the number of false discoveries to highlight the performance on the top ranked genes. Since there are 400 truly differentially expressed genes based on the log2 cut-off value of 2, the top 400 genes detected by each approach

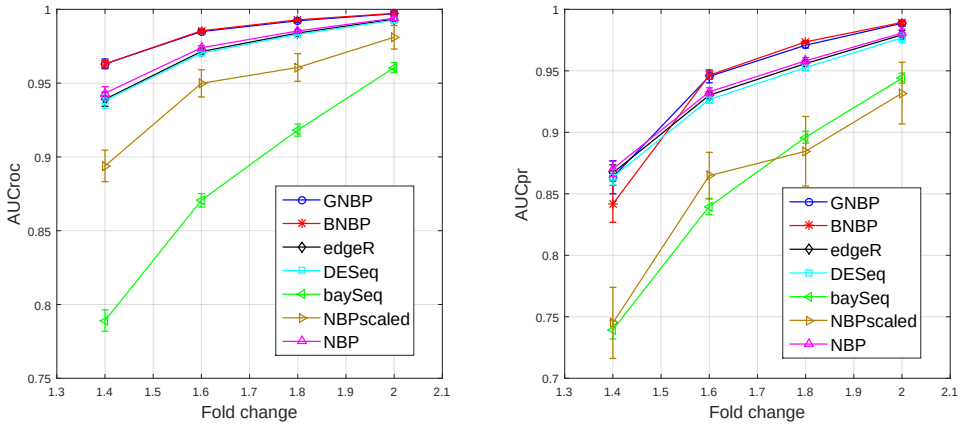
are selected and the number of false discoveries are plotted. It is clear from Figure 2.8 that both the GNBP and BNBP return much smaller number of false positives in comparison to all the other differential expression analysis algorithms.

Table 2.1: Area under the ROC curve for the range with $FPR \leq 0.1$ and area under the PR curve for the range with Recall ≤ 0.1 for both the PSU and BGI datasets, with the log2 cut-off value fixed at 2.

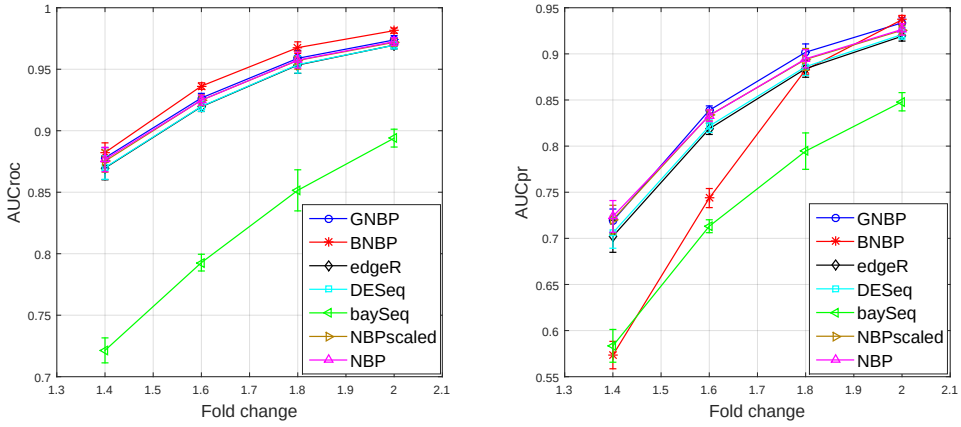
	PSU		BGI	
Method	AUCroc	AUCpr	AUCroc	AUCpr
GNBP	**0.0627**	**0.0980**	**0.0716**	**0.0995**
BNBP	**0.0628**	**0.0980**	0.0685	0.0986
edgeR	0.0587	**0.0980**	0.0527	**0.0995**
DESeq	0.0514	**0.0980**	0.0521	**0.0995**
baySeq	0.0533	**0.0980**	0.0258	0.0757
NBP	0.0356	0.0921	0.0390	0.0968
NBPscaled	0.0404	0.0911	0.0369	0.0957



(a) GNBP synthetic data

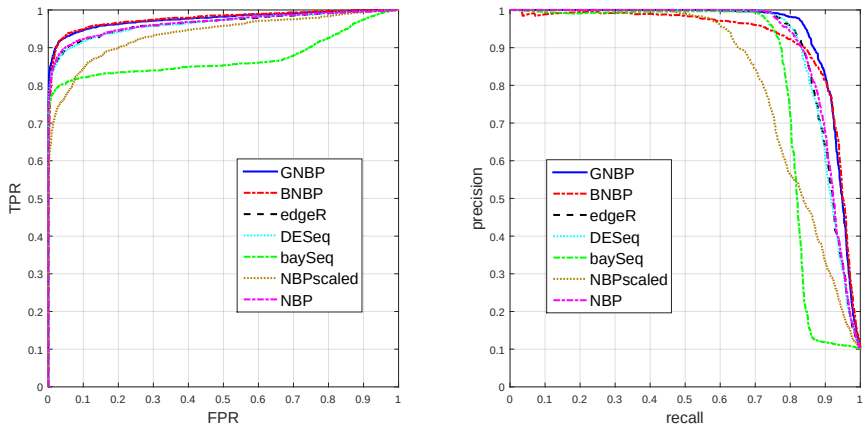


(b) BNBP synthetic data

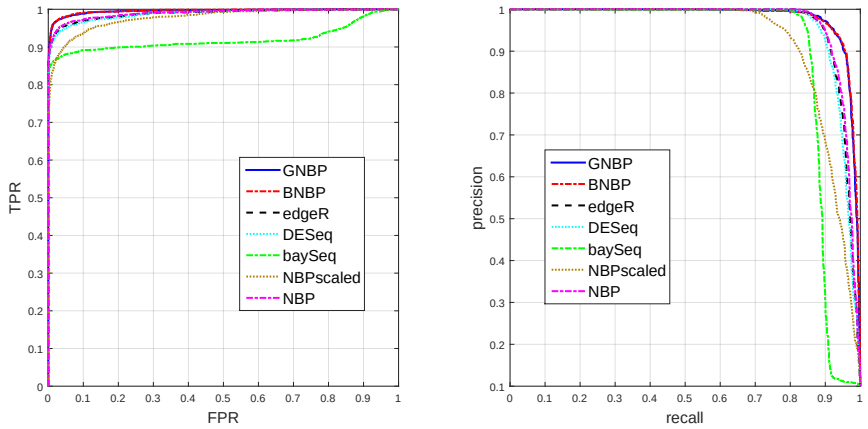


(c) baySeq synthetic data

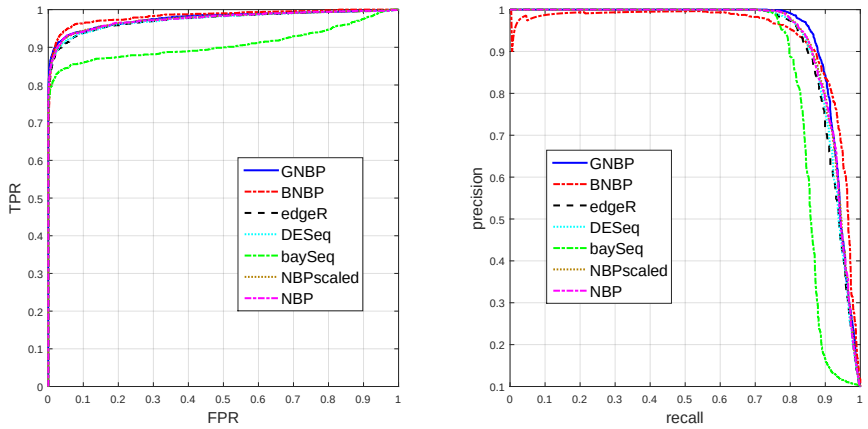
Figure 2.2: left column: AUC-ROC values, right column: AUC-PR values. Performance comparison of different methods in detecting differentially expressed genes under various fold changes, using synthetic data generated under three different negative binomial distribution based models.



(a) GNBP setup

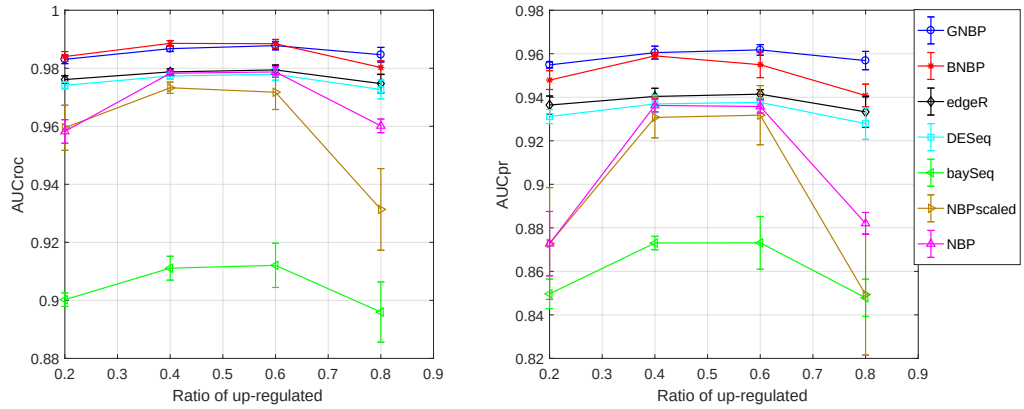


(b) BNBP setup

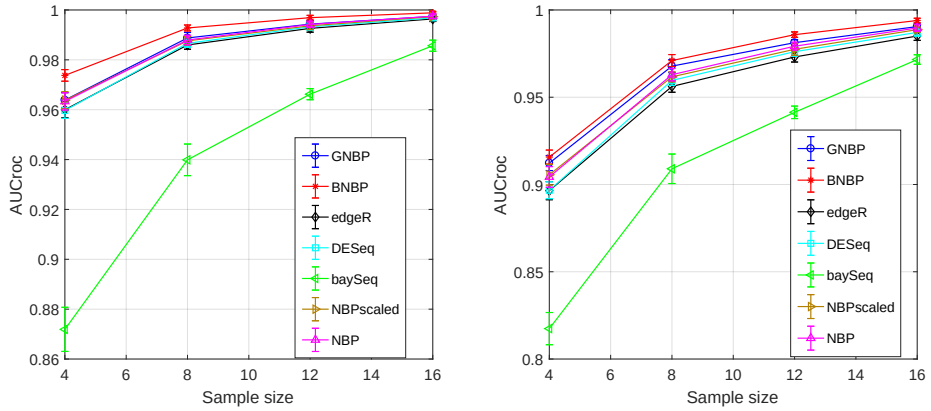


(c) baySeq setup

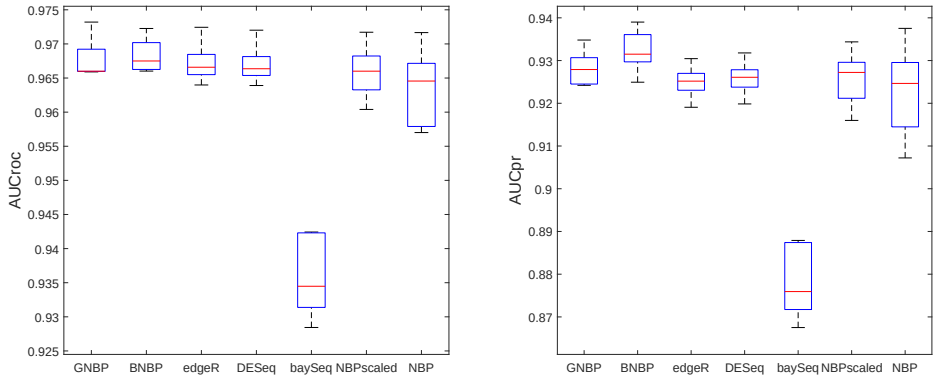
Figure 2.3: left column: ROC curve, right column: PR curve. Performance of different methods in detecting the differential expression of simulated data generated from different setups with a fold change of 1.8 for truly differentially expressed genes.



(a) Varying up/down-regulation proportions

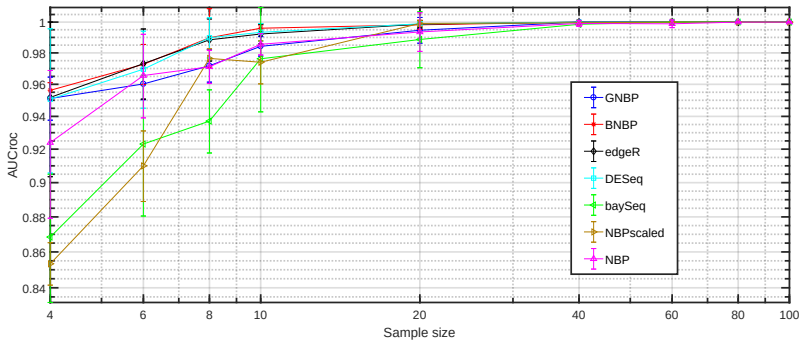


(b) Different sample sizes

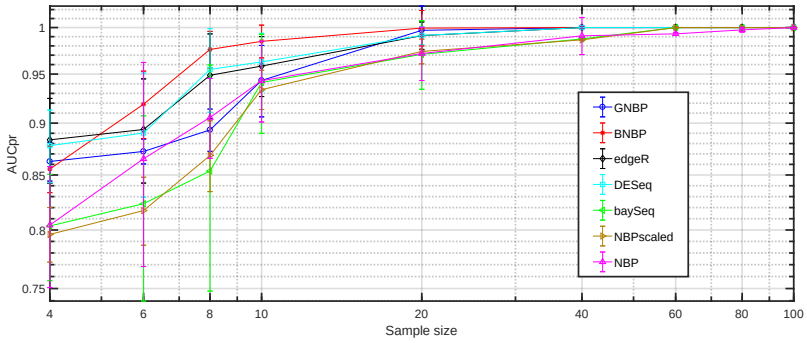


(c) Continuous true fold changes

Figure 2.4: left column: AUC-ROC values, right column: AUC-PR values. Performance comparison of different methods in detecting differentially expressed genes under various scenarios using synthetic data generated with baySeq. (a) The proportion of up-regulated genes in true differentially expressed genes increases from 20% to 80% with 20% increments. (b) The sample size in each group is increased from 4 to 16 with increments of size 4. (c) The true fold change of differentially expressed genes is sampled from a uniform distribution in the interval $[1.4, 2]$.

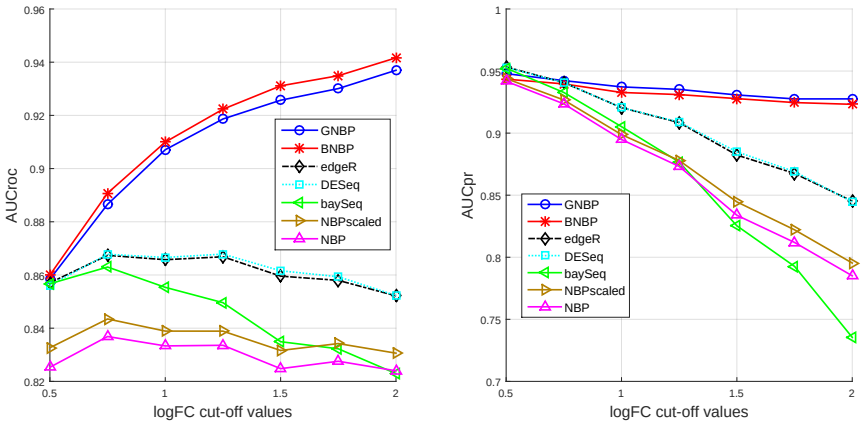


(a) AUC-ROC

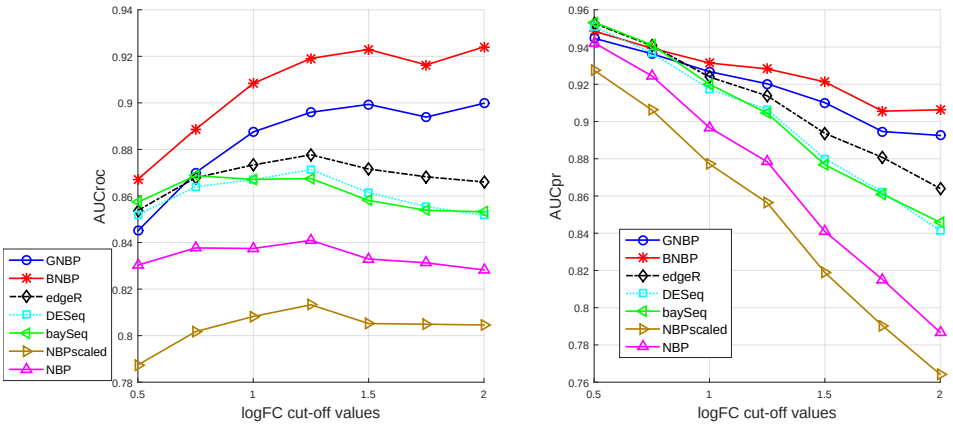


(b) AUC-PR

Figure 2.5: (a) AUC-ROC and (b) AUC-PR in the baySeq simulation setup with 100 genes and different sample sizes, where 10 genes are equally likely to be up- or down-regulated with a fold change of 2.



(a) BGI dataset



(b) PSU dataset

Figure 2.6: left column: AUC-ROC values, right column: AUC-PR values. Performance comparison of different methods in detecting differentially expressed genes on real-world benchmark RNA-seq data from the SEQC project.

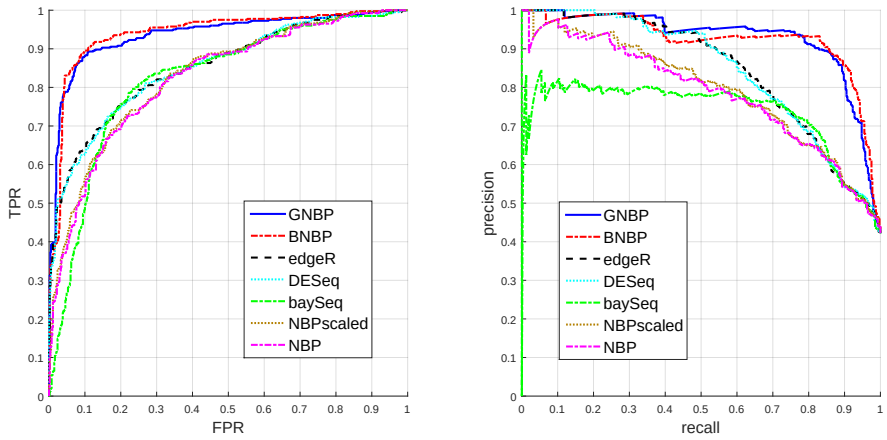


Figure 2.7: left: ROC curves, right: Precision-Recall (PR) curves. Performance comparison of different methods with the log2 cut-off value fixed at 2 for the BGI dataset from the SEQC project.

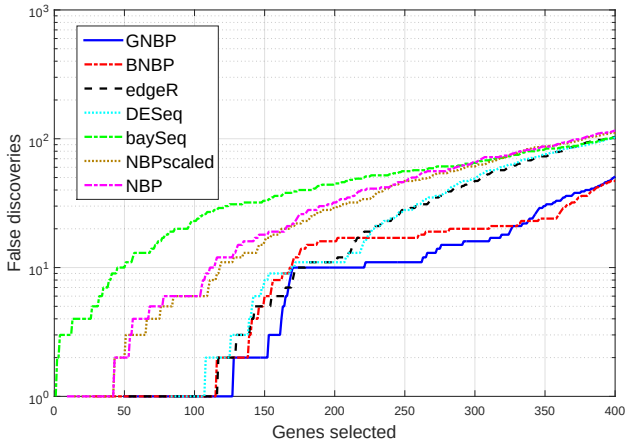


Figure 2.8: False discovery plots for different methods on the BGI dataset from the SEQC project, with the log2 cut-off value fixed at 2. The x-axis shows the number of genes selected, in order of their detected differential expression levels, while the y-axis shows the number of selected genes that are false positives.

3. BAYESIAN NEGATIVE BINOMIAL REGRESSION FOR DIFFERENTIAL EXPRESSION [1]

While the majority of differential expression analyses are conducted with respect to a main treatment factor, the presence of potential confounding factors in real-world experiments makes it desirable to take them into account in the developed tools to derive unbiased genotype-phenotype association results. There exists a rich set of methods on addressing this problem in microarray data analysis, such as the ones developed based on linear models (54; 55).

Several differential expression analysis methods have employed generalized linear models (GLMs) to adapt the NB distribution to experiments with complex design. For example, two widely used methods, edgeR (56) and DESeq2 (57), both use GLMs to model the mean of the NB distribution as a log-linear function of the covariates. The gene-wise dispersion parameters are then estimated using adjusted profile likelihood and GLM coefficients are estimated using Fisher scoring iterations.

We propose a fully Bayesian NB regression (BNB-R) method for differential expression analysis of RNA-seq data from experiments with complex multiple-factor design. Unlike all the existing differential expression methods based on the NB distribution, our method does not rely on *ad-hoc* approximations of various kinds, such as the fact that many statistical tests are only asymptotically valid (58). BNB-R quantifies the uncertainty of the estimations, and also allows for the incorporation of prior information. BNB-R directly model the influence from covariates of interest for differential expression analysis and therefore it does not need the surrogate variable analysis pre-processing step.

Moreover, this new approach does not require the *ad-hoc* normalization step either, as the model accounts for the sequencing-depth heterogeneity of different samples automatically, similar to the mechanisms employed in the BNP-Seq algorithms. By exploiting two novel data augmen-

[1] Reprinted with permission from S. Zamani Dadaneh, M. Zhou, and X. Qian, "Bayesian negative binomial regression for differential expression with confounding factors," *Bioinformatics*, vol. 34, no. 19, pp. 3349–3356, 2018. Copyright 2018 Oxford University Press.

tation techniques (59), closed-form posterior inference of BNB-R model parameters is derived in a Gibbs sampling procedure. Specifically, the dispersion parameter of NB distribution is inferred using the augmentation technique of (60), and regression coefficients are inferred in closed-forms by utilizing the Polya-Gamma distributed auxiliary variable technique of (61), removing the need for non-trivial Metropolis-Hastings correction steps (62).

3.1 BNB-R: NB regression differential expression analysis

We denote the number of sequencing reads mapped to gene $k \in \{1, \ldots, K\}$ in sequencing sample $j \in \{1, \ldots, J\}$ by n_{kj}, and model this count as a negative binomial random variable $n_{kj} \sim \text{NB}(r_j, p_{kj})$. The dispersion parameter r_j, which only depends on the sample index, can be considered as a parameter reflecting the heterogeneity of counts, due to the variation of the sequencing depths across different samples. This can be justified by the gene count expectation $\mathbb{E}[n_{kj}] = r_j \frac{p_{kj}}{1-p_{kj}}$, which is directly proportional to r_j. To establish the dependence between the gene expression and covariates (e.g., phenotypes, treatments, and other potential confounding factors) in different experimental setups, we impose a linear relationship between the logit function of the probability and covariates as $\text{logit}(p_{kj}) = \boldsymbol{x}_j^T \boldsymbol{\beta}_k$, where $\boldsymbol{x}_j = [1, x_{j1}, ..., x_{jV}]^T$ is the covariate vector for sample j and $\boldsymbol{\beta}_k = [\beta_{k0}, \beta_{k1}, ..., \beta_{kV}]^T$ is the regression coefficient vector for gene k. In our proposed model, the covariate variables can be numerical or categorical. Consequently, the expected gene expression can be expressed as $\mathbb{E}[n_{kj}] = r_j \exp(\boldsymbol{x}_j^T \boldsymbol{\beta}_k)$, which resembles the familiar form of negative binomial generalized linear model (63). Thereby, the effects of different experimental factors on gene expression are captured through the regression coefficients $\boldsymbol{\beta}_k$. In particular, by utilizing the Bayesian framework, the posterior distributions of different combinations of the regression coefficients can be estimated via a Markov chain Monte Carlo (MCMC) (64) inference procedure to assess how the covariates impact the expression changes.

To complete the hierarchal model, we place a gamma prior on each sequencing scaling parameter r_j and independent zero-mean normal priors on the regression coefficients $\boldsymbol{\beta}_k$. The full model

is expressed as:

$$
\begin{aligned}
n_{kj} \sim \text{NB}(r_j, p_{kj}), \quad \psi_{kj} &:= \text{logit}(p_{kj}) = \boldsymbol{x}_j^T \boldsymbol{\beta}_k \\
\boldsymbol{\beta}_k \sim \prod_{v=0}^{V} \text{N}(0, \alpha_v^{-1}), \quad \alpha_v &\sim \text{Gamma}(c_0, 1/d_0) \\
r_j \sim \text{Gamma}(a_0, 1/h), \quad h &\sim \text{Gamma}(b_0, 1/g_0).
\end{aligned} \tag{3.1}
$$

In addition to controlling the effects of multiple experimental factors via the regression coefficients $\boldsymbol{\beta}_k$, in BNB-R, the precision parameters of the normal distributions over these coefficients are shared between all genes to borrow signal strengths, a desirable property of the model that makes it robust especially in RNA-seq data analysis with a small sample size. In the following, we present our efficient MCMC inference of model parameters, which takes advantage of two novel data augmentation techniques, leading to closed-form parameter updates.

3.1.0.1 Parameter inference

We start by the inference of the dispersion parameter r_j, by using the data augmentation technique introduced in (60). In the first step of MCMC inference, we draw latent counts corresponding to gene expression as

$$
(\ell_{kj} \,|\, -) \sim \text{CRT}(n_{kj}, r_j). \tag{3.2}
$$

It can be shown that the ℓ_{kj} can be considered as the Poisson random count, expressed as $\ell_{kj} \sim \text{Pois}(-r_j \ln(1 - p_{kj}))$, used in the compound Poisson representation of the NB distribution $n_{jk} \sim \text{NB}(r_j, p_{kj})$. Hence, by taking advantage of the gamma-Poisson conjugacy, in each Gibbs sampling iteration, the parameter r_j can be updated as

$$
(r_j|-) \sim \text{Gamma}\left(\sum_k \ell_{kj} + a_0, \frac{1}{h - \sum_k \ln(1 - p_{kj})}\right). \tag{3.3}
$$

The second challenge is the inference of the regression coefficients, for which the lack of conditional conjugacy precludes immediate closed-form inference. Resorting to the methods such

as Metropolis-Hastings (62), however, requires a careful choice of the proposal distributions to avoid suffering from high rejection rates and subsequently slow convergence. To address these issues, we adopt an augmentation technique to infer the regression coefficients $\boldsymbol{\beta}_k$, relying on the Polya-Gamma (PG) data augmentation of (61). Denote ω_{kj} as a random variable drawn from the PG distribution as $\omega_{kj} \sim \text{PG}(n_{kj} + r_j, 0)$. We have $\mathbb{E}_{\omega_{kj}}[\exp(-\omega_{kj}\psi_{kj}^2/2)] = \cosh^{(n_{kj}+r_j)}(\psi_{kj}^2/2)$. Thus the likelihood of ψ_{kj} in (4.6) can be expressed as

$$\begin{aligned}\mathcal{L}(\psi_{kj}) &\propto \frac{(e^{\psi_{kj}})^{n_{kj}}}{(1+e^{\psi_{kj}})^{n_{kj}+r_j}} \\ &\propto \exp\Big(\frac{n_{kj}-r_j}{2}\psi_{kj}\Big)\mathbb{E}_{\omega_{kj}}[\exp(-\omega_{kj}\psi_{kj}^2/2)]. \end{aligned} \tag{3.4}$$

Exploiting the exponential tilting of the PG distribution in (61), we draw ω_{kj} as

$$(\omega_{kj}|-) \sim \text{PG}(n_{kj} + r_j, \psi_{kj}). \tag{3.5}$$

Given the values of the auxiliary variables ω_{kj} for $j = 1, ..., J$ and the prior in (4.6), the conditional posterior of $\boldsymbol{\beta}_k$ can be expressed as

$$p(\boldsymbol{\beta}_k|-) \propto \text{N}(\mathbf{0}, A^{-1}) \prod_{j=1}^{J} e^{-\frac{\omega_{kj}}{2}\left(\psi_{kj} - \frac{n_{kj}-r_j}{2\omega_{kj}}\right)^2}, \tag{3.6}$$

where $A = \text{diag}(\alpha_1, ..., \alpha_P)$. Thus in each Gibbs sampling iteration, we update the gene-wise regression coefficients $\boldsymbol{\beta}_k$ as

$$(\boldsymbol{\beta}_k|-) \sim \text{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k), \tag{3.7}$$

where the covariance and mean of this multivariate normal distribution are defined as $\boldsymbol{\Sigma}_k = \big(\sum_{j=1}^{J} \omega_{kj}\boldsymbol{x}_j\boldsymbol{x}_j^T + A\big)^{-1}$ and $\boldsymbol{\mu}_k = \boldsymbol{\Sigma}_k\big(\sum_{j=1}^{J}(\frac{n_{kj}-r_j}{2})\boldsymbol{x}_j\big)$, respectively.

Algorithm 1 Gibbs sampling inference for BNB-R

Inputs: gene expression counts, design matrix, N
Outputs: KL-divergence based ranking of DE genes

Initialize model parameters Please add the iteration loop
for $j = 1$ to N **do**
  Sample ℓ_{kj} using CRT
  Update r_j using the gamma-Poisson conjugacy
  Sample auxiliary variables ω_{kj}, using the PG distribution
end for
Update regression coefficients
Update α_p and h

Using the gamma-gamma conjugacy with respect to the gamma scale parameter, we have

$$(\alpha_v|-) \sim \text{Gamma}\big(K/2 + c_0, \frac{1}{d_0 + \sum_k \beta_{kv}^2/2}\big), \quad v = 0, ..., V.$$
$$(h|-) \sim \text{Gamma}\big(b_0 + Ja_0, \frac{1}{g_0 + \sum_j r_j}\big). \tag{3.8}$$

The Gibbs sampling steps in equations (3.2) to (3.8) are summarized in Algorithm 1.

3.1.0.2 Differential expression (DE) analysis

To detect differentially expressed genes using the inferred NB regression model, we notice in the prior that

$$\mathbb{E}[n_{kj}] = r_j \exp(\boldsymbol{x}_j^T \boldsymbol{\beta}_k) \tag{3.9}$$

and in the conditional posterior shown in (5.2)

$$\mathbb{E}[r_j|-] = \frac{\sum_k \ell_{kj} + a_0}{h + \sum_k \ln\big(1 + \exp(\boldsymbol{x}_j^T \boldsymbol{\beta}_k)\big)}. \tag{3.10}$$

Thus one may consider that the NB sample-specific dispersion parameter r_j, which depends on all the gene counts of sample j through latent counts ℓ_{kj}, accounts for the sequencing depth of sample j, and the quantity $\exp(\boldsymbol{x}_j^T \boldsymbol{\beta}_k)$ represents the expression of gene k in sample j after removing the sequencing-depth effect. To assess whether a certain experimental factor v causes significant ex-

pression differences across samples for gene k, we collect posterior MCMC samples for regression coefficients $\boldsymbol{\beta}_k$, and use these MCMC samples to measure the distance between the posterior distributions of $\exp(\beta_{k0})$ and $\exp(\beta_{k0}+\beta_{kv})$. More precisely, we use the symmetric Kullback-Leibler (KL) divergence defined between two discrete distributions P and Q as

$$\mathrm{KL}(P,Q) = \sum\nolimits_x \big[p(x) - q(x)\big] \log\big[p(x)/q(x)\big].$$

To calculate this distance, we follow the same steps as in (65), and construct a discrete probability vector for each group of collected MCMC samples, referred to as $\boldsymbol{\pi}^{(1)}$ and $\boldsymbol{\pi}^{(2)}$ for the first and second groups under comparison, respectively. Finally, with a small constant set as $\epsilon = 10^{-10}$, we calculate the symmetric KL-divergence as

$$\mathrm{KL}(\boldsymbol{\pi}^{(1)}, \boldsymbol{\pi}^{(2)}) = \sum_{i=1}^{N} \big(\pi_i^{(1)} - \pi_i^{(2)}\big) \log\Big(\frac{\pi_i^{(1)} + \epsilon}{\pi_i^{(2)} + \epsilon}\Big). \tag{3.11}$$

3.2 Results

To evaluate our Bayesian negative binomial regression differential expression analysis algorithm, referred to as BNB-R, we compare its performance on both synthetic and real-world benchmark data with those of edgeR (56), DESeq2 (57), and voom included in the package `limma` (58), three widely used methods capable of handling biomedical studies with complex experimental design. As it is common in practice, before applying these methods to real-world RNA-seq data, we first perform a surrogate variable analysis (SVA) to introduce surrogate variables as additional covariates to model potential unwanted batch effects (66), and then use them to adjust for these artifacts for unbiased differential expression analysis. We first consider synthetic RNA-seq data in simulated experiments with multiple factors, and we demonstrate that the proposed BNB-R consistently outperforms the other approaches. We then consider the real-world benchmark RNA-seq data extracted from the SEquencing Quality Control (SEQC) project (67). While this dataset does not possess explicit confounding factors, the results support the outstanding performance of

BNB-R for differential expression analysis method in general. On both synthetic and real-world RNA-seq count data, different methods are compared in terms of both the receiver operating characteristic (ROC) and precision-recall (PR) curves, and the area under these curves (AUC). Finally, we test BNB-R on a RNA-seq dataset of Th17 cell differentiation to study how incorporating the temporal information can lead to more meaningful biological discoveries.

3.2.1 Synthetic data

3.2.1.1 Incorporating covariates improves DE detection

We generate synthetic RNA-seq data with the NB regression generative model. To make the synthetic data closely resemble real-world RNA-seq data, the parameters of the NB regression model are first inferred from the SEQC dataset, and then synthetic sequencing counts are generated using these inferred model parameters. Throughout the simulations, we consider three experimental factors as *condition*, *gender*, and *dosage*, where condition and gender are categorical covariates with labels {*treated*, *untreated*} and {*male*, *female*}, respectively, and dosage is a numeric covariate in the interval $[0, 1]$, generated uniformly at random for each sample.

In the first simulation setting, the expression of gene k in sample j is simulated according to the distribution NB$(r_j, \frac{1}{1+\exp(-\boldsymbol{x}_j^T \boldsymbol{\beta}_k)})$, where for sample $j \in \{1, 2, \ldots, J\}$, the covariate vector is $\boldsymbol{x}_j = [x_{j0}, x_{j1}, x_{j2}, x_{j3}]$. The variable x_{jv} represents the value of covariate v for sample j. In the first simulation setup, $v = 0$ corresponds to the intercept term, and $v = 1, 2, 3$ correspond to *condition*, *gender*, and *dosage* covariates respectively. We use a binary scheme for coding the categorical covariates x_{j1} and x_{j2}. More precisely, $x_{j1} = 0$ if no treatment has been applied to sample j, and $x_{j1} = 1$ if this sample is under treatment. Also, $x_{j2} = 0$ if sample j belongs to a female individual and $x_{j2} = 1$ if it belongs to a male.

The effect of covariate v on the expression level of gene k is adjusted through the regression coefficient β_{kv}. We simulate this coefficient according to a zero-mean normal distribution with precision parameter α_v. For the *condition* covariate, we draw the precision parameter as $\alpha_1 \sim$ Gamma$(1.7e5, 1/1e4)$. Under this setting, the absolute value of β_{k1} is larger than 0.4 with

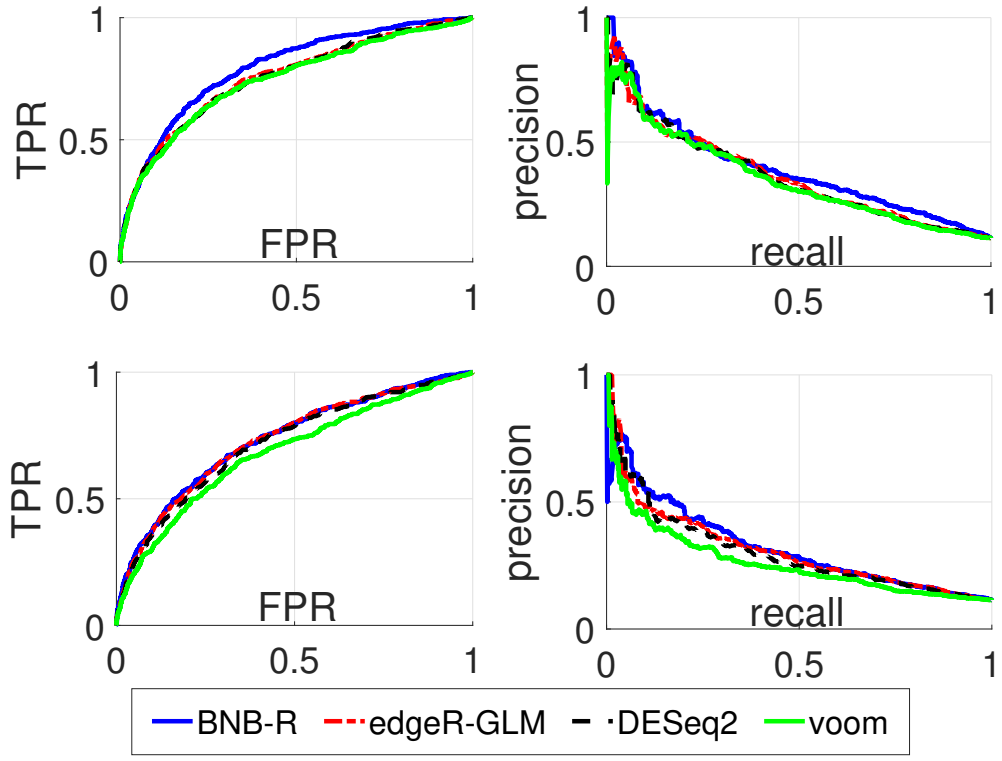


Figure 3.1: Left panel: ROC curve, Right panel: PR curve. Performance comparison of different methods in detecting differentially expressed genes generated under a negative binomial regression model with covariates: *condition*, *gender* and *dosage*. Panels in the top row correspond to the case that full covariate information is used in differential expression analysis. Panels in the bottom row correspond to the case that only condition covariate is used in differential expression analysis.

probability 10%. Thus on average, 10% of the genes exhibit an expression fold-change of at least $\exp(\beta_{k1}) = 1.5$ between two different conditions. In subsequent ROC and PR analyses, we consider gene k as true differentially expressed if $|\beta_{k1}| \geq 0.4$ and not differentially expressed otherwise. The other three precision parameters are simulated as follows:

$$\begin{aligned} \alpha_0 &\sim \text{Gamma}(2.7e6, 1/1e4) \\ \alpha_2 &\sim \text{Gamma}(3e3, 1/1e4) \\ \alpha_3 &\sim \text{Gamma}(3e5, 1/1e4), \end{aligned} \tag{3.12}$$

where α_0 determines the baseline gene expression independent of experimental factors, and α_2 and

α_3 adjust the heterogeneity of gene expressions due to the *gender* and *dosage* factors, respectively. Finally, to simulate the effect of different sequencing depths for different samples, the dispersion parameters r_j are independently drawn from Gamma$(50, 1/5)$, which is close to the posterior distribution of r_j inferred from the Beijing Genomics Institute (BGI) dataset of the SEQC benchmark.

In the first simulation setting, the gene-expression counts for a total of K =5,000 genes and J =12 samples, with three males and three females in each of the two conditions, are generated. We evaluate the performance of BNB-R based on this synthetic data, and compare it to edgeR, DESeq2, and voom. For BNB-R, model parameters are inferred via Gibbs sampling, where in each run of the algorithm, we collect 1,000 MCMC samples after 1,000 burn-in iterations, and then rank the genes using the symmetric KL-divergence measure developed in Section 3.1.0.2. For edgeR, DESeq2, and voom, we follow the standard analysis pipelines and rank the genes using the computed p-values.

Panels in the top row of Figure 3.1 illustrate the ROC and PR curves of BNB-R, edgeR, DESeq2, and voom under the first simulation setting, when all covariates are employed. The AUCs of these curves are presented in Table 3.1. The panels in the bottom row of Figure 3.1 represent the performance of BNB-R, edgeR, DESeq2, and voom on the synthetic data when using the *condition* covariate as the single experimental factor, while neglecting all the other covariates. Table 3.2 provides the AUCs of the curves in the latter scenario. Methods that exploit covariates' information clearly outperform the ones that only rely on the *condition* factor to identify differentially expressed genes, in terms of both the ROC and PR curves. This observation demonstrates the benefit of incorporating available experimental design information to better capture the heterogeneity of gene expression counts. In particular, BNB-R with covariates has the best performance with a significant margin over all the other algorithms. This may be explained by the hierarchical structure of BNB-R, where borrowing information from all genes to estimate precision parameters makes it robust in modeling overdispersed count data. In addition, we have also applied the BNBP and GNBP methods (65), which use only the *condition* factor to determine differential expression, to the synthetic data in this simulation. These two methods also perform closely to the algorithms

Table 3.1: AUC of ROC and PR curves presented in the panels, in the top row of Figure 3.1.

Method	AUC-ROC	AUC-PR
BNB-R	0.7952	0.3922
edgeR-GLM	0.7563	0.3622
DESeq2	0.7533	0.3587
voom	0.7450	0.3499

Table 3.2: AUC of ROC and PR curves presented in the panels, in the bottom row of Figure 3.1.

Method	AUC-ROC	AUC-PR
BNB-R	0.7343	0.3188
edgeR-GLM	0.7302	0.3087
DESeq2	0.7193	0.2999
voom	0.6832	0.2617

exploiting only the *condition* factor, confirming the observation that integrating additional covariates into a differential expression model can achieve more accurate and robust DE analysis for genotype-phenotype association.

3.2.1.2 Sensitivity to experimental design

To assess the sensitivity of BNB-R to the experimental design assumption employed in the differential expression analysis model, we consider a simulation setting with a more complex combination of experimental factors, including an interaction term between the *gender* and *condition* covariates. Similar to the previous simulation, the expression of gene k in sample j is drawn from NB$(r_j, \frac{1}{1+\exp(-\boldsymbol{x}_j^T \boldsymbol{\beta}_k)})$, where for sample $j = 1, 2, ..., J$, the covariate vector is $\boldsymbol{x}_j = [x_{j0}, x_{j1}, ..., x_{j4}]^T$. In this simulation setup, the elements x_{jv} in the covariate vector for $v = 0, 1, .., 4$ correspond to intercept, *gender*, *condition*, *dosage*, and the interaction between *gender* and *condition*, respectively. We employ the same binary coding scheme for the categorical covariates as those used in the previous simulation setting. Thus, for example, $x_{j4} = 1$ if sample j has been under treatment and belongs to a male individual, and $x_{j4} = 0$ otherwise. We also generate the *dosage* covariates x_{j3} from a uniform distribution in interval $[0, 1]$.

The presence of the interaction term in the regression model leads to the dependence of gene differential expression on both the *condition* and *gender* covariates. More precisely, in this simulation setting, the expected expression fold-change of gene k across two treatment conditions, for a female is $\exp(\beta_{k2})$, and for a male is $\exp(\beta_{k2} + \beta_{k4})$. Hence in ROC and PR analyses, gene k with $|\beta_{k2}| > 0.4$ is considered as truly differentially expressed across conditions for females, and when $|\beta_{k2} + \beta_{k4}| > 0.4$, it is considered as truly differentially expressed across conditions for males. We simulate the regression coefficient β_{kv} according to a zero-mean normal distribution with the precision parameter α_v, and we place the following Gamma distributions on the precision parameters:

$$\begin{aligned}
\alpha_0 &\sim \text{Gamma}(2.7e6, 1/1e4) \\
\alpha_1 &\sim \text{Gamma}(1e6, 1/1e4) \\
\alpha_2 &\sim \text{Gamma}(1.8e5, 1/1e4) \\
\alpha_3 &\sim \text{Gamma}(3e5, 1/1e4) \\
\alpha_4 &\sim \text{Gamma}(1.2e6, 1/1e4).
\end{aligned} \tag{3.13}$$

RNA-seq counts for a total of K =5,000 genes and J =12 samples, with three males and three females in each treatment condition, are generated. In this synthetic dataset, 516 genes are differentially expressed across treatment conditions for females and 653 genes are differentially expressed for males. First, we evaluate the performance of BNB-R, edgeR, DESeq2, and voom on this synthetic data, assuming that the true design matrix used for data generation is provided for all algorithms. Differentially expressed genes are identified using the same protocol as described in the previous subsection. The top and middle panels of Figure 3.2 illustrate the ROC and PR curves for the detection of differentially expressed genes across conditions in males and females, respectively. BNB-R clearly outperforms the other methods in terms of both ROC and PR, for gender-specific DE analyses.

Next, instead of assuming knowing the true underlying data generation mechanism, we exclude

the interaction term used for data generation for differential expression analysis with different methods, and use the covariate vector $\boldsymbol{x}_j = [x_{j0}, x_{j1}, x_{j2}, x_{j3}]$ for sample j, where the elements x_{jv} for $v = 0, 1, 2, 3$ represent the same covariates as in the data generation procedure. As a consequence of using this design, detected differentially expressed genes are not specific to a gender. Hence to evaluate the performance of BNB-R, edgeR, DESeq2, and voom when using this design matrix, we need to compare the detected genes to those that are truly differentially expressed across conditions independent of gender. In this simulation, there are 400 genes that are differentially expressed across the treatment conditions for both male and female groups. We consider these genes as truly differentially expressed independent of gender, and the rest of the genes as not differentially expressed. The ROC and PR curves plotted based on this setting are shown in the bottom row of Figure 3.2. In this case, BNB-R again exhibit the best performance in terms of the ROC and PR curves, confirming its superior performance even if the true mechanism of data generation is not fully known.

3.2.2 SEQC benchmark

In this section, we evaluate the performance of the proposed BNB-R method using SEQC benchmark (67). Specifically, we use the RNA-seq data from BGI provided in the R package SEQC on Bioconductor (68), containing the counts for about 26,000 genes. In our experiments, we employ sample groups A and B, which are derived from the Agilent's Universal Human Reference RNA and Life Technologies' Human Brain Reference RNA cell lines, respectively. We collect the counts from the first flow cells of the sequencing machines on five replicates for each group.

To evaluate the differential expression analysis methods, we note that in the SEQC project, the same RNA samples for a comprehensive group of control genes are analyzed based on quantitative Reverse Transcription Polymerase Chain Reaction (qRT-PCR) using TaqMan assays (51), which is referred as the TaqMan benchmark data (69; 67). More precisely, for sample groups A and B, the expression intensity values of 955 selected control genes have been derived in the TaqMan qPT-PCR analysis for sequencing benchmarking. In the absence of the knowledge on the genes that are truly differentially expressed across different conditions, we follow the approach in (70) to

threshold the qRT-PCR expression ratios across different conditions at a certain value to define the ground-truth set of differentially expressed genes. Based on these 955 genes in the TaqMan data, we evaluate the performance of different differential expression analysis pipelines.

Before applying edgeR, DESeq2, and voom to this dataset, we first perform a surrogate variable analysis (SVA) to adjust for un-modeled artifacts such as batch effects (66). More precisely, we use `svaseq` function of R package `sva` (71) with two introduced surrogate variables (SVs). In the downstream differential expression analysis, we use these two SVs as extra confounding factors for edgeR, DESeq2, and voom. Our experiment shows that incorporation of the SVs slightly improves the performance of these methods. Note that although for BNB-R no explicit experimental factor other than a sample's group is used in this experiment, our results suggest the performance of the proposed BNB-R differential expression analysis method is superior to those of stochastic processes inspired models in BNP-Seq, all of which achieve better ROC and PR curves than edgeR, DESeq2, and voom in conjunction with SVA, as described in detail below.

While truly differentially expressed genes are unknown for the SEQC RNA-seq data, we rely on the qRT-PCR expression intensity of the 955 genes in the TaqMan data and set different cut-offs for the binary logarithm (log2) of the qRT-PCR expression ratio to define "truly" differentially expressed genes. We increase this log2 cut-off value gradually from 0.5 to 2, and calculate both AUC-ROC and AUC-PR. For the analysis of the dataset BGI on a single cluster node with Intel Xeon 2.5GHz E5-2670 v2 processor, it took around two hours for BNB-R method with 2,000 MCMC iterations. The posterior distributions of the regression coefficients are used to assess differential expression. In addition to the methods used for synthetic data, we also include BNBP and GNBP (65), both of which are generative models designed specifically for a single factor setting. As shown in the bottom panels of Figure 3.3, the BNB-R method outperforms all the other methods in both ROC and PR analyses, followed very closely by BNBP and GNBP. Note that the performance gains of the three generative models over the other methods become more significant as one increases the log2 cut-off for the qRT-PCR expression ratio, which reduces the number of genes that are considered as truly differentially expressed.

To further investigate the experimental results, we fix the log2 cut-off value at 2 for the qRT-PCR expression intensity of the 955 genes in the TaqMan data, and illustrate the ROC and PR curves for the BGI dataset in the top panels of Figure 3.3. It is clear the BNB-R method along with GNBP and BNBP not only have higher AUC-ROC and AUC-PR, but also outperform edgeR, DESeq2, and voom used together with SVA in almost all regions of the ROC and PR curves.

3.2.3 Case study: Th17 cell differentiation

To further illustrate its potential biological significance when integrating other covariates in BNB-R for biomarker identification applications, we provide a case study with our BNB-R method on a RNA-seq dataset of early human T helper 17 (Th17) cell differentiation and T-cell activation (Th0). Th17 cells play an essential role in the pathogenesis of autoimmune and inflammatory diseases, and have been the focus of many recent research efforts (72). In particular, the knowledge of the early phase of Th17 differentiation helps to gain insight into the process of signal propagation through various pathways and gene regulatory networks (73). We use the RNA-seq dataset of (74) and (75), which contains gene expression profiling of Th0 and Th17 cells at the following five time points: 0, 12, 24, 48, and 72 hours after cell activation and stimulation, with three biological replicates at each time point. The data is obtained from *Gene Expression Omnibus*, with accession *GSE52260*.

The design matrix of the analysis is formed from an additive model formula as in our simulation studies, accounting for condition and time point factors. More precisely, for sample $j = 1, 2, ..., 15$ the covariate vector is $\boldsymbol{x}_j = [x_{j0}, x_{j1}, x_{j2}]^T$, where x_{j0} is the intercept, x_{j1} is the cell category (i.e., Th0 vs Th17), and x_{j2} is the sample time point. We apply BNB-R to identify differentially expressed genes, where after 1,000 burn-in iterations, 1,000 posterior samples are collected to calculate the symmetric KL-divergence between the posterior distributions of $\exp(\beta_{k0})$ and $\exp(\beta_{k0} + \beta_{k1})$ to rank the genes. The run-time of BNB-R with 2,000 MCMC sampling iterations for the Th17 dataset on the cluster node with configuration provided in Section 3.2.2 is around 6 hours.

We consider the top 100 genes ranked by the symmetric KL-divergence and perform *Gene On-*

tology (GO) analysis using LAGO (76) software[2], focusing on the ontology of biological processes. The top five significantly enriched GO terms discovered by LAGO, with their corresponding adjusted p-values shown in Table 3.3, illustrating the association between the differentially expressed genes and immune system activation and response to stimulus.

Table 3.3: Top five enriched GO terms associated with top 100 differentially expressed genes in TH17 dataset detected by BNB-R.

GO-ID	Term	P-value
GO:0002376	immune system process	4.74695e-13
GO:0046649	lymphocyte activation	3.33415e-11
GO:0006955	immune response	3.90728e-11
GO:0045321	leukocyte activation	1.6007e-10
GO:0050896	response to stimulus	1.89798e-10

In a closer look at the results, the top differentially expressed gene identified by BNB-R is gene COL6A3, an important organizer of the extracellular matrix proteins, contributing to adipose tissue inflammation (77). Also, the up-regulation of COL6A3 gene in Th17-polarizing cells is confirmed by microarray and RT-PCR assays in (72). The third ranked gene, Leukemia Inhibitory Factor (LIF), belongs to the IL-6 family of cytokines and resides within the core regulatory circuitry of T cells (78). The fourth gene, RORC, is a Th17 lineage-specific transcription factor (79), whose differential expression is also verified in the microarray study in (72). In addition, Western blotting results in (72) show that genes BATF, CTSL1, VDR, KDSR, ATP1B1, and BASP1 were highly expressed in Th17 cells compared with their expression in Th0 cells at various time points during the first three days of polarization. The rankings of these genes obtained by our BNB-R are 11, 13, 15, 20, 24, and 41, respectively, which confirms the significance of their expression changes. Moreover, microarray studies of (72) found out the up-regulation of CXCR5 and LMNA in $CD4^+$ T cells cultured under Th17-polarizing conditions compared with Th0 cells, and flow cytometric detection of CD52 at 48 and 72 hours showed down-regulation of this protein in $CD4^+$ T cells

[2]available at http://go.princeton.edu/cgi-bin/LAGO

cultured under Th17-polarizing conditions. These genes are ranked 14, 44, and 60, respectively, in our differential expression analysis, supporting their potential roles in Th17 cells differentiation process.

Next, to examine how incorporating the time course information changes the differential expression analysis results, we apply BNB-R on the Th17 dataset, considering only the condition factor but ignoring the temporal information of different samples. Although out of the top 100 differentially expressed genes, there are 84 genes common between these two differential analysis results, the GO analysis, when the time factor is neglected, results in a total of 36 significantly enriched terms with known annotations, which is less than 40 annotated enriched terms when including the time factor. Some of the GO terms missed include *cytokine-mediated signaling pathway*, *positive regulation of JAK-STAT*, *STAT cascades*, and *T cell activation involved in immune response*, which are all related to the immune system and can potentially lead to new hypotheses. In addition, BNB-R considering the time factor leads to smaller p-values overall in comparison to the analysis without time information, and hence more significantly enriched GO terms. For instance, the adjusted p-value obtained for *T cell differentiation* by the former analysis is $1.910e-4$, while the latter returns $2.554e-3$.

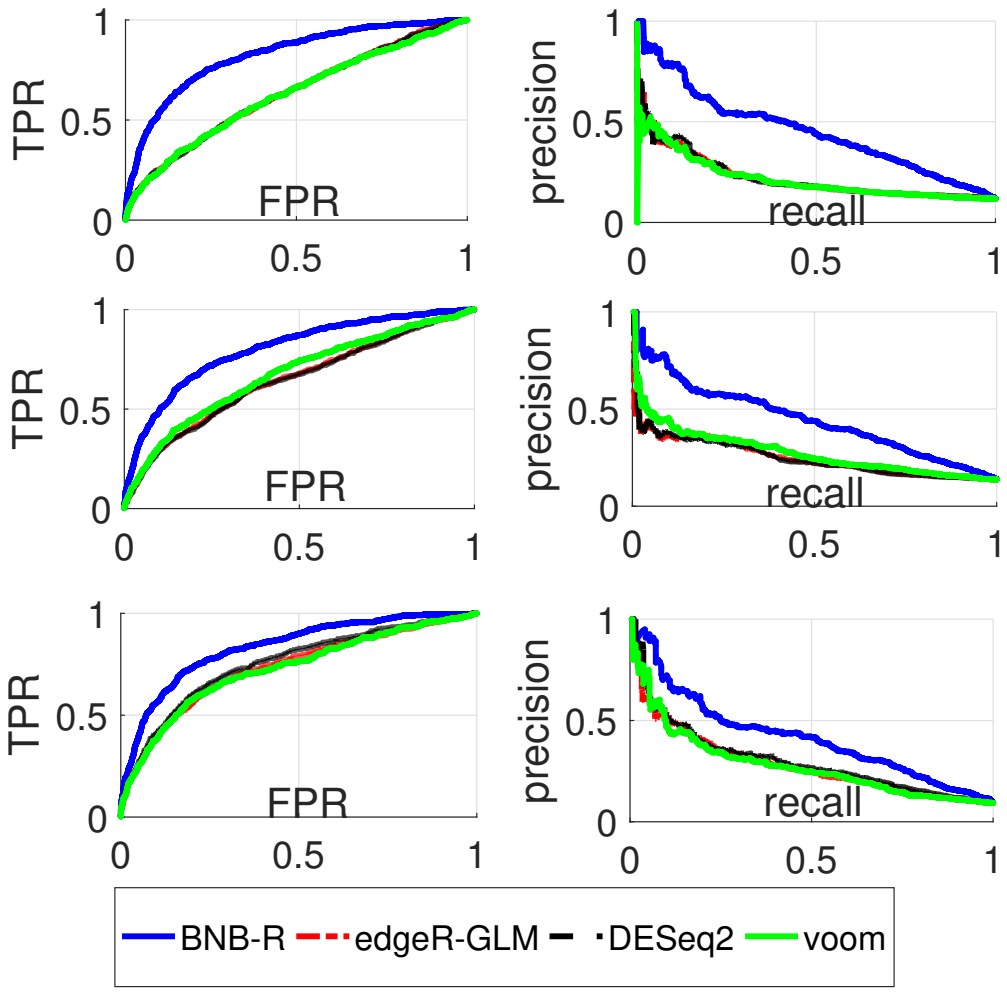


Figure 3.2: Left panels: ROC curve, Right panels: PR curve. Performance comparison of different methods in detecting differentially expressed genes generated under the negative binomial regression model with covariates: *condition*, *gender*, *dosage*, and interaction of *condition* and *gender*. The panels in the top and middle rows correspond to differentially expressed genes across conditions for males and females, respectively. The panels in the bottom row correspond to differentially expressed genes for the case that full covariate information is not employed, with the interaction term excluded from differential expression analyses by all the methods.

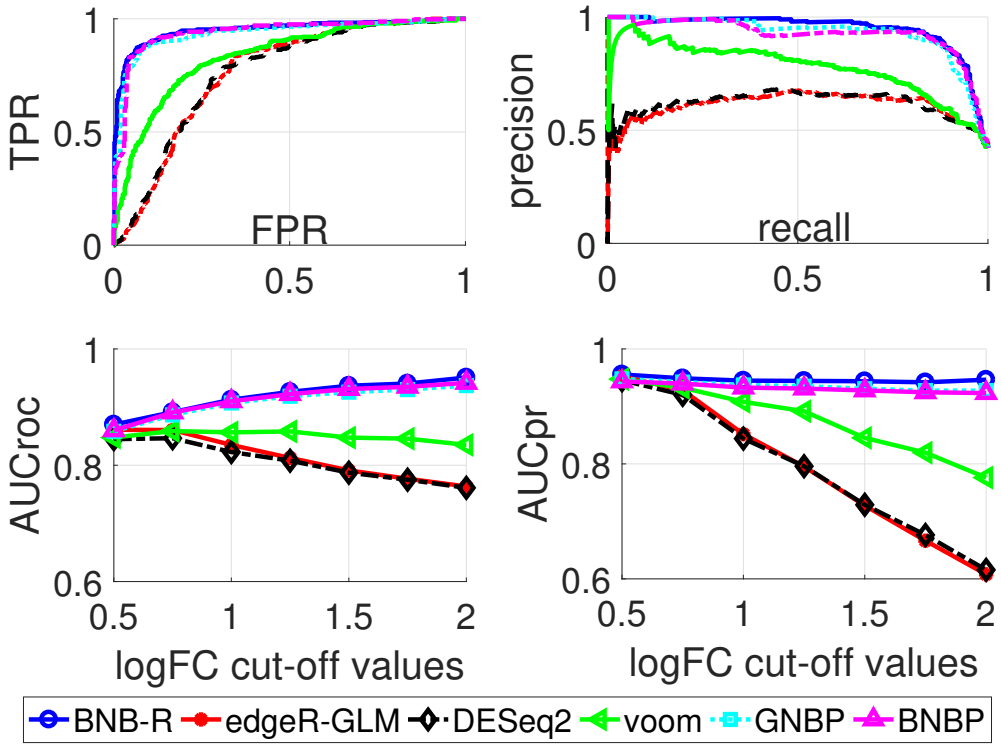


Figure 3.3: Top row: ROC and PR curves for a fixed cut-off, Bottom row: AUC of ROC and PR curves for different cut-off values. Performance comparison of different methods in detecting differentially expressed genes on real-world benchmark RNA-seq data from the SEQC project. edgeR, DESeq2, and voom are applied in conjunction with SVA with two surrogate variables.

4. COVARIATE-DEPENDENT NEGATIVE BINOMIAL FACTOR ANALYSIS OF RNA SEQUENCING DATA [1]

Living systems are complex and their behavior is coordinated by multiple components. Especially, when studying complex disease, phenotypic changes have been shown to be associated with coordinated regulation in functional modules of interacting genes (pathways or subnetworks) rather than statistically significant changes in individual genes (80). Therefore, a class of approaches have been developed to detect genes with similar expression patterns as potential functional modules. Weighted Gene Co-Expression Network Analysis (WGCNA) (35), a popular tool for gene co-expression network analysis, first constructs an adjacency matrix based on the pairwise co-expression measures, for example, based on the correlation between gene expressions across samples; then it assigns genes to different modules using the hierarchical clustering algorithm. DiffCoEx (36) builds on WGCNA, and by computing the matrix of adjacency differences between different experiment conditions, aims at identifying differentially co-expressed genes. Several *targeted* methods also have been proposed for studying co-expression changes across conditions, relying on pre-defined gene modules (81; 82; 83). For instance, (83) focuses on the analysis of modules based on known gene annotations, such as gene ontology categories.

All of the aforementioned methods were proposed for data generated from microarray based experiments; and thus there remains a lack of tools for gene module detection specifically designed for RNA-seq count data. Furthermore, the existing methods often require prior knowledge from either manual annotations or other module identification methods. They need to be supplied with prepared lists of genes as candidate functional modules. For example, (37) have proposed a network module-based generalized linear model for identifying differentially expressed pre-defined gene sets.

A suitable method for gene module identification based on RNA-seq data should explicitly

[1]Reprinted with permission from S. Zamani Dadaneh, M. Zhou, and X. Qian, "Covariate-dependent negative binomial factor analysis of RNA sequencing data," *Bioinformatics*, vol. 34, no. 13, pp. i61–i69, 2018. Copyright 2018 Oxford University Press.

model highly over-dispersed count data that are often skewed (8) to avoid potential bias introduced by inappropriate modeling. One of the most popular solutions to account for over-dispersion due to biological variability is using the negative binomial (NB) distribution, which possesses a quadratic variance-mean relationship. More importantly, the number of *ad-hoc* choices in modeling and data analytics should be minimized. Many existing methods, which often take two stages to first construct co-expression networks based on expression profile data and then identify co-expressed modules based on different clustering methods, may lead to uncertain results sensitive to different choices. Last but not least, when dealing with RNA-seq data, the variability of the sequencing depths across samples needs to be taken into account.

In this chapter, we propose a novel covariate-dependent NB factorization model for identifying gene modules in RNA-seq experiments. The proposed method, directly applied to gene counts from RNA-seq, obviates the need for multiple *ad-hoc* steps as required in co-expression network analyses of WGCNA (35) and DiffCoEx (36). In addition, by employing a flexible regression model for the scale parameter of the gamma distribution in our fully Bayesian NB factor analysis model, dNBFA is capable of tackling RNA-seq experiments with complex confounding factors, and quantifies the impact of these factors on the identified modules. Finally, similar to the mechanisms employed in (65), this new approach does not require an *ad-hoc* normalization step, as the model accounts for the sequencing-depth heterogeneity of different samples automatically.

4.1 Methods

4.1.1 NB factor analysis

In this section we first present the Negative Binomial Factor Analysis (NBFA) method for count data (84), and demonstrate how it can be applied in the context of RNA-seq data analysis for the identification of gene modules. Let n_{vj} denote the number of sequencing reads mapped to gene $v \in \{1, ..., V\}$ in sequencing sample $j \in \{1, ..., J\}$, and let the $V \times 1$ vector $\mathbf{n}_j$ contain all the gene counts for sample j. The negative binomial (NB) distribution is a popular choice to model RNA-seq count data, allowing one to account for over-dispersion due to technical and biological

variations (65; 85; 56). Under the NBFA model (84), the sample counts are factorized as

$$\mathbf{n}_j \sim \text{NB}(\boldsymbol{\Phi}\boldsymbol{\theta}_j, p_j), \tag{4.1}$$

where $n \sim \text{NB}(r, p)$ denotes the NB distribution with the probability mass function (PMF) $f_N(n) = \frac{\Gamma(n+r)}{n!\Gamma(r)}p^n(1-p)^r$, where $\Gamma(\cdot)$ is the gamma function and $n \in \{0, 1, 2, ...\}$. $\boldsymbol{\Phi} = (\boldsymbol{\phi}_1, ..., \boldsymbol{\phi}_K) \in \mathbb{R}_+^{V\times K}$ represents the factor loading matrix, $\boldsymbol{\Theta} = (\boldsymbol{\theta}_1, ..., \boldsymbol{\theta}_J) \in \mathbb{R}_+^{K\times J}$ represents the factor score matrix, and $\mathbb{R}_+ = \{x : x \leq 0\}$. For each latent factor, $\boldsymbol{\phi}_k = (\phi_{1k}, ..., \phi_{Vk})^T$ encodes the weights of the V genes associated with factor k and $\boldsymbol{\theta}_j$ encodes the popularity of K factors in sample j. The NBFA can be augmented as

$$n_{vj} = \sum_{k=1}^{K} n_{vjk}, \quad n_{vjk} \sim \text{NB}(\phi_{vk}\theta_{kj}, p_j). \tag{4.2}$$

From biological perspectives, K factors can correspond to the underlying biological processes or functional modules related to genotypic, phenotypic, or treatment condition changes. The corresponding sub-counts n_{vjk} can be viewed as the result of the contribution of underlying biological process k to the expression of gene v in sample j. The probability parameter p_j, which only depends on the sample index, can be considered as a parameter reflecting the potential heterogeneity of counts, due to the variation of the sequencing depths across different samples.

More precisely, using (4.2) and the formula for the mean of the NB distribution, the expected expression of gene v in sample j can be expressed as $\mathbb{E}[n_{vj}] = (\sum_{k=1}^{K}\phi_{vk}\theta_{kj})\frac{p_j}{1-p_j}$. The term $\frac{p_j}{1-p_j}$ can be interpreted as the effect of the sequencing-depth heterogeneity of sample j on the corresponding gene expression in this sample. This approach removes the need for an *ad-hoc* normalization step, as the model accounts for the sequencing-depth heterogeneity of different samples automatically, similar to the mechanisms employed in (65). The remaining term in this expectation, $\sum_{k=1}^{K}\phi_{vk}\theta_{kj}$, can represent the true abundance of gene v in sample j. Specifically, it comprises of contributions from all latent factors, where each contribution is encoded as the product of the gene association with latent factors as modules and the contribution of those modules to sample j.

NBFA proceeds by placing the Dirichlet and gamma prior distributions on ϕ_{vk} and θ_{kj}, respectively, and appropriate prior distributions on the other model parameters. A Gibbs sampling algorithm that exploits novel data augmentation techniques has been derived for inferring the model parameters (84).

4.1.2 Covariate-dependent NBFA

In real-world RNA-seq experiments, it is often desirable to identify the functional modules corresponding to critical biological processes specific to the behavior of interest by the design of experiments. Often, the presence of potential confounding factors also requires that the developed factor analysis method based on RNA-seq data can take them into account (when the corresponding conditions are given) to derive correct functional module results. The aforementioned NBFA model neglects such information about sequencing samples from designed experiments. In order to empower the NBFA model in tackling the setups with complex experiment design, we extend its framework to make it capable of incorporating the external covariate information (e.g., phenotypes, treatments, and other confounding factors) into the factor analysis model to derive the new covariate-dependent NBFA (dNBFA) model.

The graphical representation of dNBFA is illustrated as a hierarchical model in Figure 4.1. In the first layer of dNBFA, similar to NBFA, the gene counts are modeled using the same NB distribution as in (4.2). Then, in the next layer we place a gamma prior distribution on θ_{kj} as

$$\theta_{kj} \sim \text{Gamma}(r_k, e^{\boldsymbol{\beta}_k^T \mathbf{x}_j}), \tag{4.3}$$

where $\mathbf{x}_j$ is the $P \times 1$ vector of covariates for sample j, reflecting the corresponding experiment design. In this model, both numerical and categorical covariates can be used.

Employing the law of total expectation, and removing the sequencing depth effect by the related terms containing p_j, we have $\mathbb{E}[n_{vjk}] \propto \phi_{vk} r_k e^{\boldsymbol{\beta}_k^T \mathbf{x}_j}$. This new layer of model, splits the effect of the latent factor k on sample j into two parts; r_k, which can be considered as representing the baseline expression of the factor k across all samples, and the exponential term $e^{\boldsymbol{\beta}_k^T \mathbf{x}_j}$, which

adjusts the effect of the latent factor on the sample according to its traits. We note that including an intercept in $\boldsymbol{\beta}_k^T \mathbf{x}_j$ may weaken the identifiably of r_k, as in the expectation of the count n_{vjk} a product term $r_k e^{\beta_{k0}}$ depending only on latent factor k appears. Thus in all subsequent experiments a separate intercept term is not used when considering covariate effects. The parameters of the dNBFA model with their interpretations in the context of RNA-seq experiments are presented in Table 5.1.

We place independent zero-mean normal distributions on the components of the regression coefficient parameters as

$$\boldsymbol{\beta}_k \sim \prod_{p=1}^{P} \mathrm{N}(0, \alpha_p^{-1}), \tag{4.4}$$

where α_p is the precision parameter of the normal distribution. By assuming identical precisions for components of the regression coefficients across all latent factors, dNBFA burrows statistical strengths to infer these precision parameters.

Similar to NBFA, a Dirichlet prior distribution with the smoothing parameter η is imposed on the gene-module association parameters ϕ_{vk}:

$$(\phi_{1k}, ..., \phi_{Vk}) \sim \mathrm{Dir}(\eta, ..., \eta). \tag{4.5}$$

The Dirichlet smoothing parameter η controls the sparsity of the inferred latent factors. Generally speaking, the smaller η is, the more sparse and specific the inferred factors are encouraged to be.

A challenge in NB factorization is how to determine the number of latent factors K. To address this issue, one can employ a reasonably large K, and then according to the inference step for r_k (refer to (4.12) below), the baseline expression inferred for non-important latent factors vanishes as the number of assigned gene sub-counts to it decreases.

We complete the model by placing conjugate priors on hyperparameters. Specifically, we exploit the gamma-Poisson conjugacy, beta-negative binomial conjugacy with respect to the probability parameter, and gamma-gamma conjugacy with respect to the scale parameter of the gamma

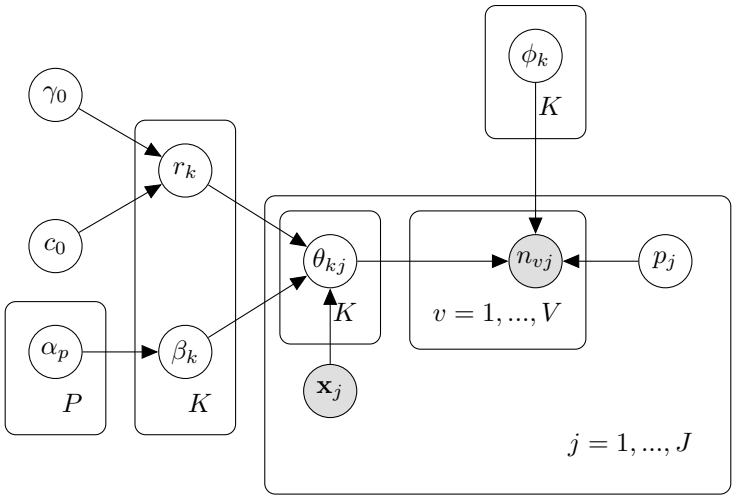


Figure 4.1: Graphical representation of covariate-dependent negative binomial factor analysis (dNBFA).

distribution. The complete dNBFA model is presented below:

$$
\begin{aligned}
n_{vj} = \sum_{k=1}^{K} n_{vjk}, \quad n_{vjk} \sim \text{NB}(\phi_{vk}\theta_{kj}, p_j), \\
\theta_{kj} \sim \text{Gamma}(r_k, e^{\boldsymbol{\beta}_k^T \mathbf{x}_j}), \quad r_k \sim \text{Gamma}(\gamma_0/K, 1/c_0), \\
(\phi_{1k}, ..., \phi_{Vk}) \sim \text{Dir}(\eta, ..., \eta), \quad \boldsymbol{\beta}_k \sim \prod_{p=1}^{P} \text{N}(0, \alpha_p^{-1}), \\
\gamma_0, c_0, \alpha, \eta \sim \text{Gamma}(e_0, 1/f_0), \quad p_j \sim \text{Beta}(a_0, b_0).
\end{aligned} \tag{4.6}
$$

Throughout the experiments in this paper, we set the hyperparameters as $e_0 = f_0 = 0.01$ and $a_0 = b_0 = 1$. In the following section, we provide an efficient inference algorithm that adopts novel data augmentation techniques tailored to our dNBFA model.

4.1.3 Inference via Gibbs sampling

By utilizing a few data augmentation techniques (86; 60; 87), we derive an efficient Gibbs sampling algorithm for inferring the model parameters in (4.6), as described below. Algorithm 1 summarizes all the steps in the Gibbs sampling algorithm.

Table 4.1: Parameters of covariate-dependent negative binomial factor analysis (dNBFA) and their interpretations in the context of RNA-seq data. The inputs of dNBFA are gene counts n_{vj} and vector of covariates $\mathbf{x}_j$.

Parameter	Constraint	Interpretation
r_k	$r_k > 0$	module baseline expression
p_j	$0 < p_j < 1$	sequencing depth
ϕ_{vk}	$\sum_{v=1}^{V} \phi_{vk} = 1, \phi_{vk} > 0$	gene-module association
θ_{kj}	$\theta_{kj} > 0$	popularity of factor k in sample j
β_{kp}	$\beta_{kp} \in \mathbb{R}$	impact of covariate p on expressionof factor k

Sample ϕ_{vk} and θ_{kj}. We start with the data augmentation technique developed for inferring the NB dispersion parameter (60). More precisely, the negative binomial random variable $n \sim \text{NB}(r, p)$ can be generated from a compound Poisson distribution as

$$n = \sum_{t=1}^{\ell} u_t, \quad u_t \sim \text{Log}(p), \quad \ell \sim \text{Pois}(-r \ln(1-p)),$$

where $u \sim \text{Log}(p)$ corresponds to the logarithmic random variable (88), with the PMF $f_U(u) = -\frac{p^u}{u \ln(1-p)}$, $u \in \{1, 2, ...\}$. As shown in (60), given n and r, the distribution of ℓ is a Chinese Restaurant Table (CRT) distribution, $(\ell | n, r) \sim \text{CRT}(n, r)$, which can be generated as $\ell = \sum_{t=1}^{n} b_t, b_t \sim \text{Bernoulli}(\frac{r}{r+t-1})$.

Utilizing the above data augmentation technique, for each observed count n_{vj}, a latent count is sampled as

$$(\ell_{vj} | -) \sim \text{CRT}(n_{vj}, \sum_{k=1}^{K} \phi_{vk} \theta_{kj}). \tag{4.7}$$

These counts are then further split into latent sub-counts (Proposition 3 of (84)) using a multinomial distribution:

$$(\ell_{vj1}, ..., \ell_{vjK} | -) \sim \text{Mult}\Big(\ell_{vj}; (\frac{\phi_{v1}\theta_{1j}}{\sum_{k=1}^{K} \phi_{vk}\theta_{kj}}, ..., \frac{\phi_{vK}\theta_{Kj}}{\sum_{k=1}^{K} \phi_{vk}\theta_{kj}})\Big). \tag{4.8}$$

These latent counts can be considered as being generated as $\ell_{vjk} \sim \text{Pois}(q_j \phi_{vk} \theta_{kj})$, where

$q_j := -\ln(1-p_j)$. Hence, using gamma-Poisson conjugacy, ϕ_{vk} and θ_{kj} are updated as

$$
\begin{aligned}
(\phi_{1k}, ..., \phi_{Vk}|-) &\sim \text{Dir}(\eta + \ell_{1\cdot k}, ..., \eta + \ell_{V\cdot k}) \\
(\theta_{kj}|-) &\sim \text{Gamma}\big(r_k + \ell_{\cdot kj}, \frac{1}{q_j + e^{-\boldsymbol{\beta}_k^T \mathbf{x}_j}}\big),
\end{aligned} \tag{4.9}
$$

where $\ell_{v\cdot k} = \sum_{j=1}^{J} \ell_{vjk}$ and $\ell_{\cdot kj} = \sum_{v=1}^{V} \ell_{vjk}$.

Sample r_k and γ_0. Let us denote $\psi_{kj} := \boldsymbol{\beta}_k^T \mathbf{x}_j + \ln q_j$. Starting with $\ell_{\cdot jk} \sim \text{Pois}(q_j \theta_{kj})$, marginalizing out θ_{kj} leads to

$$
\ell_{\cdot jk} \sim \text{NB}\Big(r_k, \frac{1}{1+e^{-\psi_{kj}}}\Big). \tag{4.10}
$$

Employing the CRT augmentation technique as

$$
(\tilde{\ell}_{jk}|-) \sim \text{CRT}(\ell_{\cdot jk}, r_k), \tag{4.11}
$$

the Gibbs sampling update for r_k can be written as

$$
(r_k|-) \sim \text{Gamma}\big(\gamma_0/K + \tilde{\ell}_{\cdot k}, \frac{1}{c_0 + \sum_j \ln(1+e^{\psi_{kj}})}\big). \tag{4.12}
$$

Following a similar procedure for γ_0, first we draw

$$
(\tilde{\tilde{\ell}}_k|-) \sim \text{CRT}(\tilde{\ell}_{\cdot k}, \gamma_0/K), \tag{4.13}
$$

and then we update the conditional posterior of γ_0 as

$$
(\gamma_0|-) \sim \text{Gamma}\big(e_0 + \sum_k \tilde{\tilde{\ell}}_k, \frac{1}{f_0 - \sum_k \ln(1-\tilde{p}_k)/K}\big), \tag{4.14}
$$

where $\tilde{p}_k := \frac{\sum_j \ln(1+e^{\psi_{kj}})}{c_0 + \sum_j \ln(1+e^{\psi_{kj}})}$.

Sample $\boldsymbol{\beta}_k$. For the regression coefficients modeling potential covariate effects, the lack of

conditional conjugacy precludes immediate closed-form inference. Therefore we adopt another data augmentation technique, specifically designed for dNBFA, to infer the regression coefficients $\boldsymbol{\beta}_k$, relying on the Polya-Gamma (PG) data augmentation of (86; 87). Denote ω_{kj} as a random variable drawn from the PG distribution as $\omega_{kj} \sim \text{PG}(\ell_{\cdot jk} + r_k, 0)$.

Since $\mathbb{E}_{\omega_{kj}}[\exp(-\omega_{kj}\psi_{kj}^2/2)] = \cosh^{(\ell_{\cdot jk}+r_k)}(\psi_{kj}^2/2)$, the likelihood of ψ_{kj} in (4.10) can be expressed as

$$\begin{aligned}\mathcal{L}(\psi_{kj}) &\propto \frac{(e^{\psi_{kj}})^{\ell_{\cdot jk}}}{(1+e^{\psi_{kj}})^{\ell_{\cdot jk}+r_k}} \\ &\propto \exp\Big(\frac{\ell_{\cdot jk}-r_k}{2}\psi_{kj}\Big)\mathbb{E}_{\omega_{kj}}[\exp(-\omega_{kj}\psi_{kj}^2/2)]. \end{aligned} \tag{4.15}$$

Exploiting the exponential tilting of the PG distribution in (87), we draw ω_{kj} as

$$(\omega_{kj}|-) \sim \text{PG}(\ell_{\cdot jk} + r_k, \psi_{kj}). \tag{4.16}$$

Given the values of the auxiliary variables ω_{kj} for $j = 1, ..., J$ and the prior in (4.6), the conditional posterior of $\boldsymbol{\beta}_k$ can be updated as

$$(\boldsymbol{\beta}_k|-) \sim \text{N}(\mu_k, \Sigma_k), \tag{4.17}$$

where $\Sigma_k = \Big(\text{diag}(\alpha_1, ..., \alpha_P) + \sum_j \omega_{kj}\mathbf{x}_j\mathbf{x}_j^T\Big)^{-1}$ and $\mu_k = \Sigma_k\big[\sum_j(\frac{\ell_{\cdot jk}-r_k}{2} - \omega_{kj}\ln(q_j))\mathbf{x}_j\big]$.

Sample η. To derive the update steps for Dirichlet hyperparameters, we note that the likelihood for $\{\boldsymbol{\phi}_k\}$ is

$$\mathcal{L}(\{\boldsymbol{\phi}_k\}) \propto \prod_{k=1} \text{Mult}(\ell_{1\cdot k}, ..., \ell_{V\cdot k}; \ell_{\cdot\cdot k}, \boldsymbol{\phi}_k). \tag{4.18}$$

Marginalizing out $\{\boldsymbol{\phi}_k\}$ from (4.18), the likelihood for η can be expressed as

$$\mathcal{L}(\eta) \propto \prod_{k=1} \text{DirMult}(\ell_{1\cdot k}, ..., \ell_{V\cdot k}; \ell_{\cdot\cdot k}, \eta, ..., \eta), \tag{4.19}$$

where DirMult denotes the Dirichlet-Multinomial distribution (84). Since the product of $\mathcal{L}(\eta)$ and $\prod_k \text{Beta}(q_k; \ell_{\cdot\cdot k}, \eta V)$ can be written as

$$\mathcal{L}(\eta) \prod_k \text{Beta}(q_k; \ell_{\cdot\cdot k}, \eta V) \propto \prod_k \prod_v \text{NB}(\ell_{v\cdot k}; \eta, q_k), \tag{4.20}$$

we can further apply the data augmentation technique for the NB distribution of (60) to derive closed-form update equations for η as

$$\begin{aligned} (q_k|-) &\sim \text{Beta}(\ell_{\cdot\cdot k}, \eta V), \;\; (u_{vk}|-) \sim \text{CRT}(\ell_{v\cdot k}, \eta) \\ (\eta|-) &\sim \text{Gamma}\big(e_0 + \sum_{v,k} u_{vk}, \frac{1}{f_0 - V \sum_k \ln(1-q_k)}\big). \end{aligned} \tag{4.21}$$

Sample α_p, p_j and c_0. Using appropriate conditional conjugacies, we can sample the remaining parameters as

$$\begin{aligned} (\alpha_p|-) &\sim \text{Gamma}(e_0 + K/2, \frac{1}{f_0 + \sum_k \beta_{kp}^2/2}) \\ (p_j|-) &\sim \text{Beta}(a_0 + \sum_{v=1}^{V} n_{vj}, b_0 + \sum_{k=1}^{K} \theta_{kj}) \\ (c_0|-) &\sim \text{Gamma}(e_0 + \gamma_0, \frac{1}{f_0 + \sum_k r_k}). \end{aligned} \tag{4.22}$$

The Gibbs sampling steps in equations (5.6) to (5.14) are summarized in Algorithm 2.

4.2 Results

We evaluate our dNBFA for covariate-dependent factor analysis based on two sets of real-world RNA-seq data studying complex diseases, and compare its performance with those of WGCNA (35) and DiffCoEx (36), two commonly adopted two-stage co-expression network based methods.

The first set of RNA-seq data was extracted from The Cancer Genome Atlas (TCGA) (89), including three datasets on breast invasive carcinoma (BRCA), lung squamous cell carcinoma (LUSC), and kidney renal clear cell carcinoma (KIRC). These data were retrieved using the TCGA2STAT

Algorithm 2 dNBFA model inference

Inputs: RNA-seq counts, design matrix of covariate effects, N
Outputs: gene module membership matrix

Initialize model parameters
# Do Gibbs sampling:
for $iter = 1$ to N **do**
    Sample ℓ_{vjk} using the CRT distribution (eq. (5.6))
    Update ϕ_{vk} and θ_{kj} using the gamma-Poisson conjugacy (eq. (4.9))
    Sample $\tilde{\ell}_{jk}$ using the CRT distribution (eq. (4.11))
    Update r_k and γ_0 using the gamma-Poisson conjugacy (eq. (4.12),(4.14))
    Sample auxiliary variables ω_{kj}, using the PG distribution (eq. (5.9))
    Update regression coefficients (eq. (5.10))
    Update η using auxiliary beta distributed random variables (eq. (4.21))
    Update α_p, p_j and c_0 (eq. (5.14))
end for

R package (90). Using TCGA data we expect to illustrate the higher differential expression significance of gene modules identified by dNBFA with respect to the disease factor compared to the results from WGCNA and DiffCoEx.

The second experiment was performed on a RNA-seq dataset of the Autism study in (91), where samples were obtained from three brain regions: the cerebral cortex Brodmann area (BA) 19, anterior prefrontal cortex (BA10), and a part of the frontal cortex (BA44). For this dataset we demonstrate how incorporating covariate information may enhance the chance of achieving meaningful biological discoveries.

For both TCGA and Autism experiments, dNBFA was run using 3,000 MCMC iterations, where after the first 1,000 burn-in iterations, the posterior samples with the highest likelihood were collected as the point estimates of model parameters. The total number of latent factors for both TCGA and Autism were initially set as $K = 250$, and after the parameter inference, only the top 100 factors with non-negligible baseline expressions were kept for further analyses. In addition, to determine module membership, for each latent factor k, only the top 20 genes with highest ϕ_{vk} were considered as members of module k. It should be noted that when an evaluation metric that can take advantage of the whole association matrix $\mathbf{\Phi}$ exists, this ad-hoc step of using

a cut-off for the gene-association parameter can be avoided.

For WGCNA, the adjacency matrix was built by first computing the pairwise Pearson correlation coefficients between gene expression profiles and then applying the soft threshold $\beta = 6, 9$ for TCGA and Autism data, respectively. The gene modules were identified by applying a hierarchical clustering algorithm to the derived topological overlap dissimilarity matrix (92). A similar procedure was followed for DiffCoEx, except that the topological overlap matrix was built upon the matrix of adjacency difference (36). Our experiments show that the discovered modules by WGCNA and DiffCoEx comprise of large lists of genes, where no further modeling capability is provided to narrow down the gene sets for more consequent exploratory analysis.

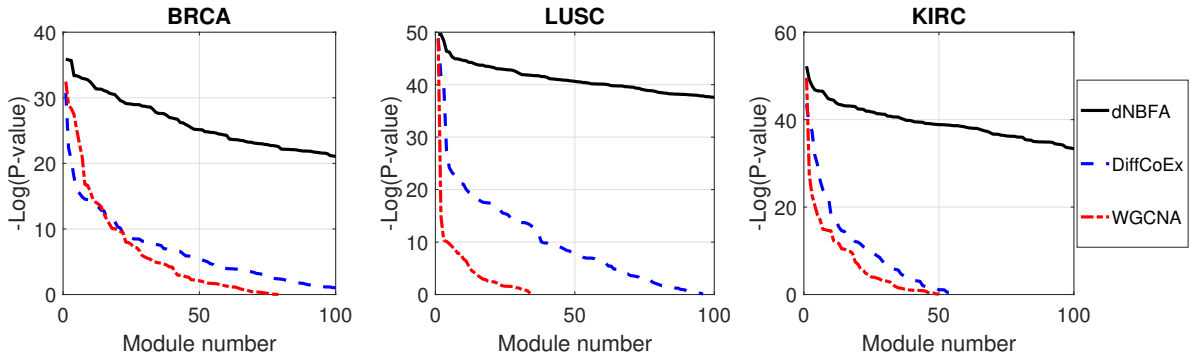


Figure 4.2: Significance of differential expression for eigengenes associated with gene modules identified by dNBFA, WGCNA, and DiffCoEx applied to three TCGA datasets. The panels show the sorted negative logarithm of P-values of the derived modules. P-values are calculated using the student's t-test on association between module eigengene expression and the samples' condition factor (cancerous vs. normal).

4.2.1 TCGA data

For all TCGA datasets, we have filtered out the genes whose total read counts across all samples are less than 50, resulting in roughly 20,000 genes in each dataset. The total numbers of samples for BRCA, LUSC, and KIRC datasets are, respectively, 40, 34, and 40, where in each case the number of primary tumor and normal samples are equal.

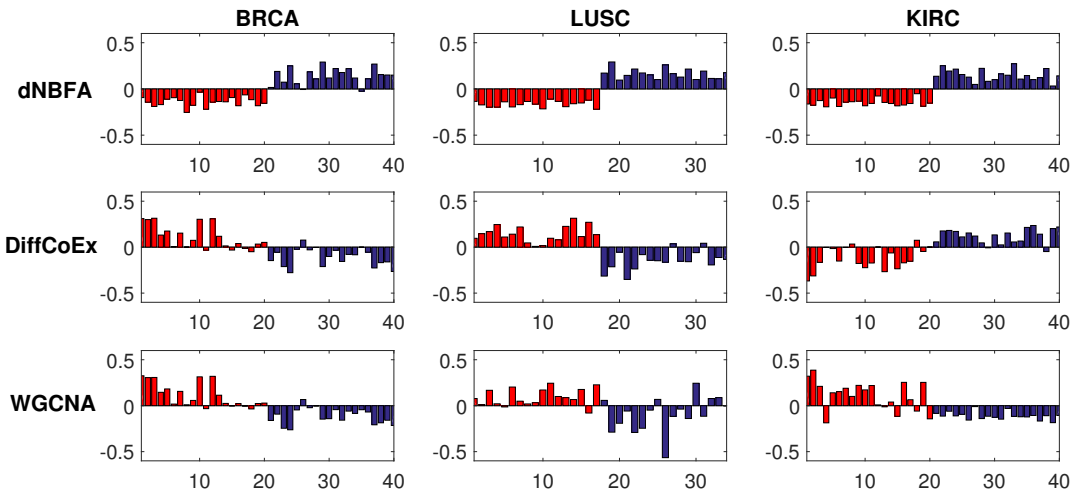


Figure 4.3: Per-sample eigengene expression of modules with the 10th lowest P-values discovered by dNBFA, WGCNA, and DiffCoEx, across cancerous and normal samples for the three TCGA datasets. In each figure the y-axis is the eigengene expression, and the x-axis is the sample number. Red and blue bars correspond to the normal and cancer groups respectively. Figures in top, middle, and bottom row are the results of dNBFA, DiffCoEx, and WGCNA, respectively. Figures in left, middle, and right columns correspond to BRCA, LUSC, and KIRC datasets, respectively.

Based on the resulting RNA-seq count data, dNBFA, WGCNA, and DiffCoEx have been applied to derive functional gene modules using the aforementioned settings. To assess the significance of differential expression of identified modules with respect to the disease status of samples, we follow the framework of (35). More precisely, for each detected module, first the *eigengene* (35) is computed via the first principal component of the expression matrix of the corresponding derived module. The module eigengene is used to summarize and represent the expression profiles of the module genes (93). Then, the association of the eigengene expression with the disease status is evaluated and finally the significance of the association is assessed based on the student's t-test.

We calculate the P-values for gene modules identified by dNBFA, WGCNA, and DiffCoEx, applied to the three TCGA datasets. The sorted P-values (based on -log(P-value)) are illustrated in Figure 4.2. The eigengenes of the modules detected by dNBFA are remarkably more differentially expressed than those detected by WGCNA and DiffCoEx in all three TCGA datasets. To further investigate the results, we present the per-sample eigengenes of the module ranked 10th for differ-

ential expression, which was identified by dNBFA, WGCNA, and DiffCoEx for the three TCGA datasets in Figure 4.3. The per-sample eigengenes of dNBFA modules are more consistently differentially expressed with respect to the disease status covariate for all three TCGA datasets, while per-sample eigengenes of WGCNA and DiffCoEx demonstrate higher variations within each group of samples with the same disease status. To ensure that the gene modules detected by dNBFA are not redundant, we also have examined the modules for significant overlap. Except a minor overlap between two modules, the rest of the modules identified by dNBFA are completely disjoint. These results suggest that dNBFA can be a powerful untargeted module identification tool, without pre-defined gene lists, for genomic experiments that study coordinated gene expression pattern changes across multiple groups.

To further verify the advantages of dNBFA that it avoids overfitting when the initial number of modules K is set high, we present in Figure 4.4 the learned r_k's, representing the baseline expression associated with the derived modules, for three TCGA datasets. Only the top 40 r_k's are included in this figure. For all datasets, only a fraction of modules have significantly large baseline expression; and thus in practice, a threshold can be used to extract the modules that contribute significantly to coordinated gene expression changes specific to the experiment design factors of interest.

For the analysis of the real-world TCGA dataset on a single cluster node with Intel Xeon 2.5GHz E5-2670 v2 processor, on average it took around eight hours for both the dNBFA and NBFA methods with 3,000 MCMC iterations, and about one hour for both WGCNA and DiffCoEx.

4.2.2 Autism data

Autism is a neuro-developmental disorder, in which the affected individuals are characterized by impairments in social and communicative developments (91). To apply dNBFA to the RNA-seq dataset of the Autism study in (91), we first discard the samples with low sequencing depths, resulting in a dataset with 36 samples from the control group and 23 Autism samples. The following analyses are performed using a subset of 12,010 genes that have a count of at least three per sample across 90% of the samples. In this experiment, site of sample collection, age, sex, and brain region

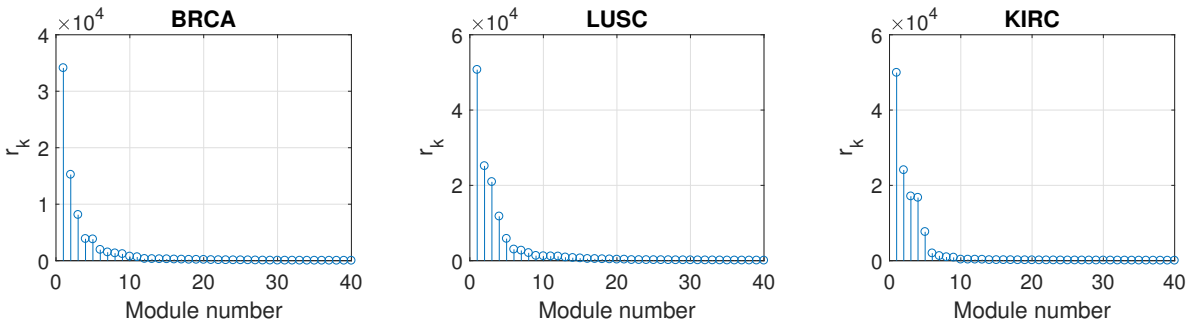


Figure 4.4: Inferred baseline expression r_k for modules detected by dNBFA in the three TCGA datasets. Only the top 40 r_k's are included in this figure.

are available as the covariate information in factor analysis. To examine how these information can alter the NB factor analysis results, in the first set of experiments we use the covariates to apply dNBFA; and in the second set of experiments, we neglect all covariate information and run the naive NBFA on the dataset.

We perform gene set enrichment analysis (GSEA) on the discovered modules by applying dNBFA and NBFA respectively to the Autism data, covering molecular function (MF), cellular component (CC), and biological process (BP) ontology domains. We calculate the significance of GO terms using Fisher's exact test and depict the sorted negative logarithm of P-values for both dNBFA and NBFA in Figure 4.5. The modules detected by dNBFA have, in general, lower P-values than those identified by NBFA without covariates, suggesting that incorporating covariate information may increase the chance of discovering biologically meaningful modules.

To investigate the gene ontology results more thoroughly, the top 10 GO terms with the lowest P-values are presented in Tables 4.2 and 4.3 for dNBFA and NBFA methods, respectively. In these tables, each row is the most significant GO term corresponding to one module identified by dNBFA or NBFA. The top modules discovered by dNBFA provide more explicit connections to neural system. Especially, the top module identified by dNBFA, which was not detected by NBFA, is associated with GO term '*type I interferon signaling pathway*', where type I Interferon responses in the brain are classically attributed to viral infections (94), which in turn are connected to Autism (95). Another important module detected only by dNBFA, the third module in Table 4.2, is related

to adaptive immune response which is closely correlated to the development of Autism spectrum disorders (96; 97). More precisely, this module includes the human leukocyte antigen (HLA) genes that play an instrumental role in many innate and adaptive immune responses (98). Many reports have provided the evidence on associations between Autism and HLA genes/haplotypes, suggesting an underlying dysregulation of the immune system mediated by HLA genes (98; 99; 100). A third important module identified only by dNBFA is associated with GO term '*neuron differentiation* (GO:0030182, P-value $= 1.4 \times 10^{-08}$). Specifically, this module includes calmodulin 1 (CALM1) gene. Significant defects in CALM1 interaction modules, which regulate voltage-independent calcium-activated action potentials at the neuronal synapse, are reported in autistic patients (101).

Table 4.2: Top enriched GO terms identified by dNBFA algorithm applied to Autism RNA-seq data.

GO-ID	Aspect	Term	P-value
GO:0060337	BP	type I interferon signaling pathway	4.377782e-15
GO:0043209	CC	myelin sheath	1.522628e-14
GO:0002460	BP	* see blow	9.487407e-13
GO:0061024	BP	membrane organization	2.250911e-11
GO:0044456	CC	synapse part	4.010908e-10
GO:0005575	CC	cellular component	4.950009e-10
GO:0033693	BP	neurofilament bundle assembly	3.982179e-09
GO:0031720	MF	haptoglobin binding	3.982179e-09
GO:0000982	MF	** see blow	6.267732e-09
GO:0001504	BP	neurotransmitter uptake	1.015843e-08

* adaptive immune response based on somatic recombination of immune receptors built from immunoglobulin superfamily domains

** transcription factor activity, RNA polymerase II core promoter proximal region sequence-specific binding

Other GO terms directly related to the nervous system associated with the top modules discovered by dNBFA include '*Myelin sheath*', '*synapse part*', '*neurofilament bundle assembl*', and '*neurotransmitter uptake*'. Specifically, The decreased thickness of myelin in the orbitofrontal cortex region is closely related to Autism disorders (102). In addition, the module detected by

Table 4.3: Top enriched GO terms identified by NBFA algorithm applied to Autism RNA-seq data.

GO-ID	Aspect	Term	P-value
GO:0022625	CC	cytosolic large ribosomal subunit	6.175728e-17
GO:0097458	CC	neuron part	1.134089e-10
GO:0006735	BP	NADH regeneration	1.153145e-10
GO:0005575	CC	cellular component	4.950009e-10
GO:0051050	BP	positive regulation of transport	7.042487e-08
GO:0007399	BP	nervous system development	1.841155e-07
GO:0065010	CC	extracellular membrane-bounded organelle	5.614232e-07
GO:0017111	MF	nucleoside-triphosphatase activity	1.144582e-06
GO:0048630	BP	skeletal muscle tissue growth	1.586369e-05
GO:0071208	MF	histone pre-mRNA DCP binding	1.586369e-05

dNBFA corresponding to GO term '*synapse part*' has the highest association with the gene SNAP-25, whose reduced expression level is responsible for the cognitive deficits in children affected by Autism spectrum disorders (103).

Examining the detected modules by both dNBFA and NBFA, we observe that multiple GO terms relevant to Autism, such as '*myelin sheath*', '*NADH regeneration*', and '*nervous system development*', are revealed by both algorithms. NADH is mainly involved in catabolic reactions (energy metabolism and mitochondrial function), whose decreased level has been reported in some children with Autism (104). On the other hand, defects in Autism appear closely tied to late developmental steps of nervous system that depend on synaptic activity and activity-dependent transcriptional changes (105). Hence the relevance of the discovered GO terms by both dNBFA and NBFA to Autism is confirmed.

Finally, by examining the trace plots of model parameters, such as c_0 and r_k, we find that the Markov chains for the dNBFA method converge fast and mix well, supporting the practice of performing downstream analysis with 3,000 MCMC iterations.

In summary, both NBFA and dNBFA methods emerge as useful module identification tools in RNA-seq data analysis, as in comparison to other available methods for gene module detection, they require minimum user adjustments. Specifically, the experimental results on the Autism dataset show that the incorporation of covariate information by dNBFA may lead to the discovery

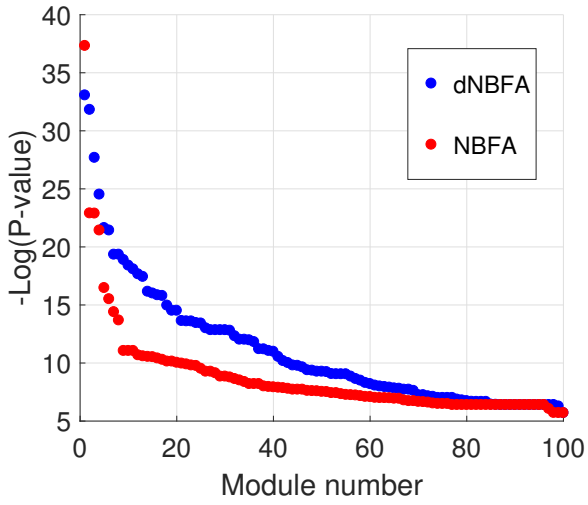


Figure 4.5: Negative logarithm of P-values for GO term enrichment analysis of modules detected by dNBFA and NBFA, applied to Autism RNA-seq data. For dNBFA, site of sample collection, age, sex and brain region are used as covariate information, while no such information is incorporated for NBFA.

of more significant Autism-relevant modules, which otherwise would be missed by NBFA.

5. BAYESIAN GAMMA-NEGATIVE BINOMIAL MODELING OF SINGLE-CELL RNA SEQUENCING DATA

Single-cell RNA sequencing (scRNA-seq) has emerged as a powerful tool for unbiased identification of previously uncharacterized molecular heterogeneity at the cellular level (106). This is in contrast to standard bulk RNA-seq techniques (2), which measures average gene expression levels within a cell population, and thus ignore tissue heterogeneity. Consideration of cell-level variability of gene expressions is essential for extracting signals from complex heterogeneous tissues (107), and also for understanding dynamic biological processes, such as embryo development (108) and cancer (109).

A large body of statistical tools developed for scRNA-seq data analysis include a dimensionality reduction step. This leads to more tractable data, from both statistical and computational point of views. Moreover, the noise in the data can be decreased, while retaining the often intrinsically low-dimensional signal of interest. Dimensionality reduction of scRNA-seq data is challenging. In addition to high gene expression variability due to cell heterogeneity, the excessive amount of zeros in scRNA-seq hinders the application of classical dimensionality reduction techniques such as principal component analysis (PCA). For instance, in real datasets, it has been reported that the first or second principal components often depend more on the proportion of detected genes per cell (i.e., genes with at least one read) than on the actual biological signal (110).

Several existing computational tools adopt explicit zero-inflation modeling to infer the latent representation of scRNA-seq data. Zero-inflated factor analysis (ZIFA) (111) extends the framework of probabilistic PCA (112) to the zero-inflated setting, by modeling the excessive zeros using Bernoulli distributed random variables which indicate the dropout event. Zero-inflated negative binomial-based wanted variation extraction (ZINB-WaVE) (113) directly models the scRNA-seq counts using a zero-inflated negative binomial distribution, while accounting for both gene- and cell-level covariates. It infers the model parameters using a penalized maximum likelihood procedure.

Despite its popularity, using an explicit zero-inflation term may place unnecessary emphasis on the zero counts, leading to complication in discovering the latent representation of scRNA-seq data. In this chapter, we propose a hierarchical gamma-negative binomial (hGNB) to both perform dimensionality reduction and adjust for the effects of the gene- and cell-level confounding factors simultaneously. Exploiting the hierarchical structure, the proposed hGNB model is capable of capturing the high over-dispersion present in the scRNA-seq data. More precisely, we factorize the logit of the negative-binomial (NB) distribution probability parameter to identify latent representation of the data. In addition to factorization, linear regression terms are also included in that logit function to adjust for the impact of covariates.

In hGNB, a gamma distribution with varying rate parameter is used to model the cell dependent dispersion parameter of the NB distribution. The cell-level dispersion serves as a means of representing the prevalence of the dropout events. For instance, cells that are sequenced deeply will naturally include less dropped-out genes with zero counts, and thus this will be reflected in the cell specific dispersion parameter of NB distribution.

5.1 hGNB Model

In this section we present the hierarchical gamma-negative binomial (hGNB) model for factor analysis of scRNA-seq data. The graphical representation of hGNB is shown in Figure 5.1. The parameters of the hGNB model with their interpretations in the context of scRNA-seq experiments are presented in Table 5.1. Let n_{vj} denote the number of sequencing reads mapped to gene $v \in \{1, ..., V\}$ in the cell $j \in \{1, ..., J\}$. Under the hGNB model, gene counts are distributed according to a negative binomial (NB) distribution:

$$n_{vj} \sim \text{NB}(r_j, p_{vj}), \tag{5.1}$$

where r_j and p_{vj} are dispersion and probability parameters of NB distribution, respectively. The probability mass function (PMF) of this distribution can be expressed as $f_N(n_{vj}) = \frac{\Gamma(n_{vj}+r_j)}{n_{vj}!\Gamma(r_j)} p_{vj}^{n_{vj}} (1-p_{vj})^{r_j}$, where $\Gamma(\cdot)$ is the gamma function.

Data from scRNA-seq experiments exhibit high variability between different cells, even for genes with medium or high levels of expression. To capture this variability, we impose a gamma prior on the cell-level dispersion parameters as

$$r_j \sim \text{Gamma}(e_0, 1/h), \tag{5.2}$$

where for simplification, the hyper-parameter e_0 is set to 0.01 in our experiments, and the rate h is learned during the Gibbs sampling inference, presented in the following section. This hierarchical prior on the dispersion parameter, enhances the flexibility of NB distribution to capture the high over-dispersion of scRNA-seq counts, without the need for explicit zero-inflation modeling.

To account for various technical and biological effects common in scRNA-seq technologies, we impose a regression model on the logit of NB probability parameter as

$$\psi_{vj} = \text{logit}(p_{vj}) = \boldsymbol{\beta}_v^T \boldsymbol{x}_j + \boldsymbol{\delta}_j^T \boldsymbol{z}_v + \boldsymbol{\phi}_v^T \boldsymbol{\theta}_j. \tag{5.3}$$

The three terms in the summation are described below.

In the first term, $\boldsymbol{x}_j$ is a known vector of P covariates for cell j and $\boldsymbol{\beta}_v$ is the regression-coefficient vector adjusting the effect of covariates on gene v. The covariate vector $\boldsymbol{x}_j$ can represent variations of interest, such as cell types, or unwanted variations, such as batch effects or quality control measures. An intercept term can also be included in these cell-level covariates to account for gene dependent baseline expressions.

In the second term, $\boldsymbol{z}_v$ is a vector of Q covariates for gene v, representing gene length or GC-content for example (114), and $\boldsymbol{\delta}_j$ is its associated regression-coefficient vector. We also include a fixed intercept element in $\boldsymbol{z}_v$ to account for cell-specific expressions, such as the size factors representing differences in sequencing depth.

In the third term, $\boldsymbol{\phi}_v^T \boldsymbol{\theta}_j$ corresponds to the latent factor representation of the count n_{vj}, after accounting for the effects of gene- and cell-level covariates. More precisely, the unknown $K \times 1$ vector $\boldsymbol{\phi}_v$ contains the factor loading parameters which determine the association between genes

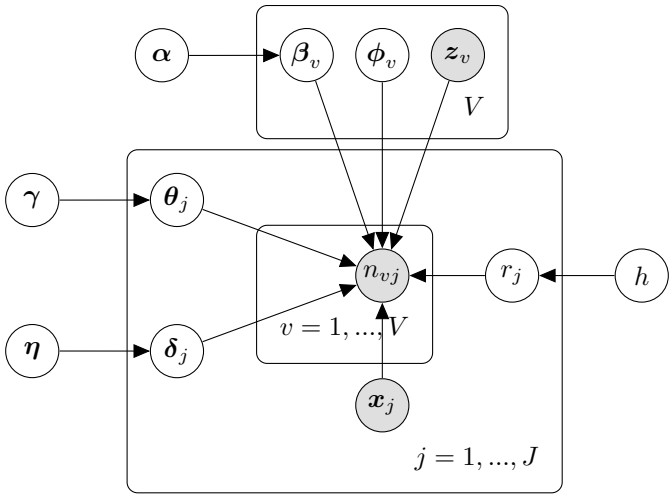


Figure 5.1: Graphical representation of the hierarchical gamma-negative binomial (hGNB) model.

and latent factors. Moreover, the unknown $K \times 1$ vector $\boldsymbol{\theta}_j$ encodes the popularity of the K factors in the expression of cell j.

We place independent zero-mean normal distributions on the components of the regression coefficient parameters $\boldsymbol{\beta}_v$ and $\boldsymbol{\delta}_j$ as

$$\begin{aligned} \boldsymbol{\beta}_v &\sim \prod_{p=1}^{P} \mathrm{N}(\beta_{vp}; 0, \alpha_p^{-1}), \\ \boldsymbol{\delta}_j &\sim \prod_{q=1}^{Q} \mathrm{N}(\delta_{jq}; 0, \eta_q^{-1}), \end{aligned} \tag{5.4}$$

where α_p and η_q are precision parameters of the normal distributions and gamma priors are imposed on them. These priors are known as automatic relevance determination (ARD), which are effective tools for pruning large numbers of irrelevant covariates (115; 116). In addition, by assuming identical precision for components of the regression coefficients across all genes or samples, hGNB borrows statistical strengths to infer these precision parameters.

We impose independent normal priors on latent factor loading and score parameters $\boldsymbol{\phi}_v$ and $\boldsymbol{\theta}_j$:

$$\begin{aligned} \boldsymbol{\phi}_v &\sim \mathrm{N}(\boldsymbol{\phi}_v; 0, I_K), \\ \boldsymbol{\theta}_j &\sim \prod_{k=1}^{K} \mathrm{N}(\theta_{jk}; 0, \gamma_k^{-1}). \end{aligned} \tag{5.5}$$

Note that the posterior for these terms is not generally independent or normal, but accounts for the statistical dependence as reflected in the data.

We complete the model by imposing a gamma prior on the precision parameters of normal distributions, and also the rate parameter of gamma distributions. Specifically, throughout the experiments, we set both the shape and rate of these gamma priors to 0.01.

Table 5.1: Parameters of the hierarchical gamma-negative binomial (hGNB) model and their interpretations in the context of scRNA-seq data. The inputs of hGNB are gene counts n_{vj} and vector of cell- and gene-level covariates $\mathbf{x}_j$ and $\mathbf{z}_v$.

Parameter	Constraint	Interpretation
r_j	$r_j > 0$	expression heterogeneity of genes in sample j
ϕ_{vk}	$\sum_{v=1}^{V} \phi_{vk} = 1, \phi_{vk} > 0$	gene-latent factor association
θ_{jk}	$\theta_{kj} > 0$	popularity of factor k in sample j
β_{vp}	$\beta_{vp} \in \mathbb{R}$	impact of cell covariate p on expression of gene v
δ_{jq}	$\beta_{vp} \in \mathbb{R}$	impact of gene covariate q on expression of cell j

5.1.1 Inference via Gibbs Sampling

In this section, we provide an efficient inference algorithm that adopts data augmentation techniques tailored to our hGNB model. Algorithm 3 summarizes all the steps in the Gibbs sampling algorithm.

Sample dispersion parameter. We start with the data augmentation technique developed for inferring the NB dispersion parameter (60). More precisely, the negative binomial random variable

$n \sim \text{NB}(r, p)$ can be generated from a compound Poisson distribution as

$$n = \sum_{t=1}^{\ell} u_t, \quad u_t \sim \text{Log}(p), \quad \ell \sim \text{Pois}(-r \ln(1-p)),$$

where $u \sim \text{Log}(p)$ corresponds to the logarithmic random variable (88), with the PMF $f_U(u) = -\frac{p^u}{u \ln(1-p)}$, $u \in \{1, 2, ...\}$. As shown in **(author?)**, given n and r, the distribution of ℓ is a Chinese Restaurant Table (CRT) distribution, $(\ell | n, r) \sim \text{CRT}(n, r)$, which can be generated as $\ell = \sum_{t=1}^{n} b_t, b_t \sim \text{Bernoulli}(\frac{r}{r+t-1})$.

Utilizing this augmentation technique, for each observed count n_{vj}, an auxiliary count is sampled as

$$(\ell_{vj} | -) \sim \text{CRT}(n_{vj}, r_j). \tag{5.6}$$

Using gamma-Poisson conjugacy, the cell-dependent dispersion parameters are updated as

$$(r_j | -) \sim \text{Gamma}\Big(e_0 + \sum_v \ell_{vj}, \frac{1}{h - \sum_v \ln(1 - p_{vj})}\Big). \tag{5.7}$$

Sample regression coefficients. For the regression coefficients modeling potential covariate effects, the lack of conditional conjugacy precludes immediate closed-form inference. Therefore we adopt another data augmentation technique, specifically designed for hGNB, to infer the regression coefficients $\boldsymbol{\beta}_v$ and $\boldsymbol{\delta}_j$, relying on the Polya-Gamma (PG) data augmentation (86; 87).

Denote ω_{vj} as a random variable drawn from the PG distribution as $\omega_{vj} \sim \text{PG}(n_{vj} + r_j, 0)$. Since $\mathbb{E}_{\omega_{vj}}[\exp(-\omega_{vj}\psi_{vj}^2/2)] = \cosh^{(n_{vj}+r_j)}(\psi_{vj}^2/2)$, the likelihood of ψ_{vj} in (5.3) can be expressed as

$$\begin{aligned} \mathcal{L}(\psi_{vj}) &\propto \frac{(e^{\psi_{vj}})^{n_{vj}}}{(1 + e^{\psi_{vj}})^{n_{vj}+r_j}} \\ &\propto \exp\Big(\frac{n_{vj} - r_j}{2}\psi_{vj}\Big)\mathbb{E}_{\omega_{vj}}[\exp(-\omega_{vj}\psi_{vj}^2/2)]. \end{aligned} \tag{5.8}$$

Exploiting the exponential tilting of the PG distribution in **(author?)**, we draw ω_{vj} as

$$(\omega_{vj}|-) \sim \text{PG}(n_{vj} + r_j, \psi_{vj}). \tag{5.9}$$

Given the values of the auxiliary variables ω_{vj} for $j = 1, ..., J$ and the prior in (5.4), the conditional posterior of $\boldsymbol{\beta}_v$ can be updated as

$$(\boldsymbol{\beta}_v|-) \sim \text{N}(\mu_v^{(\beta)}, \Sigma_v^{(\beta)}), \tag{5.10}$$

where $\Sigma_v^{(\beta)} = \Big(\text{diag}(\alpha_1, ..., \alpha_P) + \sum_j \omega_{vj} \boldsymbol{x}_j \boldsymbol{x}_j^T\Big)^{-1}$ and $\mu_v^{(\beta)} = \Sigma_v^{(\beta)} \Big[\sum_j \big(\frac{n_{vj} - r_j}{2} - \omega_{vj} (\boldsymbol{\delta}_j^T \boldsymbol{z}_v + \boldsymbol{\phi}_v^T \boldsymbol{\theta}_j) \big) \boldsymbol{x}_j \Big]$.

A similar procedure can be followed to derive the conditional updates for cell-level regression coefficients as

$$(\boldsymbol{\delta}_j|-) \sim \text{N}(\mu_j^{(\delta)}, \Sigma_j^{(\delta)}), \tag{5.11}$$

where $\Sigma_j^{(\delta)} = \Big(\text{diag}(\eta_1, ..., \eta_Q) + \sum_v \omega_{vj} \boldsymbol{z}_v \boldsymbol{z}_v^T\Big)^{-1}$ and $\mu_j^{(\delta)} = \Sigma_j^{(\delta)} \Big[\sum_v \big(\frac{n_{vj} - r_j}{2} - \omega_{vj} (\boldsymbol{\beta}_v^T \boldsymbol{x}_j + \boldsymbol{\phi}_v^T \boldsymbol{\theta}_j) \big) \boldsymbol{z}_v \Big]$.

Sample latent factor parameters. Using the likelihood function in (5.8) and the priors in (5.5), we can derive closed-form update steps for factor loading and score parameters. More specifically, the full conditional for factor loading $\boldsymbol{\phi}_v$ is a normal distribution:

$$(\boldsymbol{\phi}_v|-) \sim \text{N}(\mu_v^{(\phi)}, \Sigma_v^{(\phi)}), \tag{5.12}$$

where $\Sigma_v^{(\phi)} = \Big(I_K + \sum_j \omega_{vj} \boldsymbol{\theta}_j \boldsymbol{\theta}_j^T\Big)^{-1}$ and $\mu_v^{(\phi)} = \Sigma_v^{(\phi)} \Big[\sum_j \big(\frac{n_{vj} - r_j}{2} - \omega_{vj} (\boldsymbol{\beta}_v^T \boldsymbol{x}_j + \boldsymbol{\delta}_j^T \boldsymbol{z}_v) \big) \boldsymbol{\theta}_j \Big]$.

The full conditional for factor score $\boldsymbol{\theta}_j$ is also a normal distribution:

$$(\boldsymbol{\theta}_j|-) \sim \text{N}(\mu_j^{(\theta)}, \Sigma_j^{(\theta)}), \tag{5.13}$$

Algorithm 3 hGNB model inference

Inputs: scRNA-seq counts, design matrix of covariate effects, N
Outputs: gene module membership matrix

Initialize model parameters
# Do Gibbs sampling:
for $iter = 1$ to N **do**
  Sample ℓ_{vj} using the CRT distribution (eq. (5.6))
  Update r_j using the gamma-Poisson conjugacy (eq. (5.7))
  Sample auxiliary variables ω_{vj}, using the PG distribution (eq. (5.9))
  Update cell- and gene-level regression coefficients (eq. (5.11),(5.10))
  Update factor loadings and scores (eq. (5.12),(5.13))
  Update α_p, η_q and γ_k (eq. (5.14))
end for

where $\Sigma_j^{(\theta)} = \Big(\text{diag}(\gamma_1, ..., \gamma_K) + \sum_v \omega_{vj} \boldsymbol{\phi}_v \boldsymbol{\phi}_v^T\Big)^{-1}$ and $\mu_j^{(\theta)} = \Sigma_j^{(\theta)} \Big[\sum_v \big(\frac{n_{vj} - r_j}{2} - \omega_{vj} (\boldsymbol{\beta}_v^T \boldsymbol{x}_j + \boldsymbol{\delta}_j^T \boldsymbol{z}_v) \big) \boldsymbol{\phi}_v \Big]$.

Sample precision and rate. The precision parameters of normal distributions in (5.4) and (5.5) can be updated using the normal-gamma conjugacy:

$$\begin{aligned} \alpha_p &\sim \text{Gamma}\big(e_0 + V/2, \frac{1}{f_0 + \sum_{v=1}^V \beta_{vp}/2}\big), \\ \eta_q &\sim \text{Gamma}\big(e_0 + J/2, \frac{1}{f_0 + \sum_{v=1}^V \delta_{jq}/2}\big), \\ \gamma_k &\sim \text{Gamma}\big(e_0 + J/2, \frac{1}{f_0 + \sum_{v=1}^V \theta_{jk}/2}\big). \end{aligned} \tag{5.14}$$

Finally, the rate of gamma distribution in (5.2) can be updated using the gamma-gamma conjugacy with respect to the rate parameter:

$$h \sim \text{Gamma}\big(e_0(1 + J), \frac{1}{f_0 + \sum_{j=1}^J r_j}\big). \tag{5.15}$$

5.2 Results

We evaluate our hGNB model on four different sets of real-world scRNA-seq data from different platforms, and compare its performance to those of principal component analysis (PCA), ZIFA (111), and ZINB-WaVE (113). In the following, We briefly describe these scRNA-seq datasets. To pre-process these datasets when needed, we followed the same procedures as in **(author?)**.

V1 dataset. This dataset characterizes more than 1600 cells from the primary visual cortex (V1) in adult male mice, using a set of established Cre lines (117). A subset of three Cre lines, including Ntsr1-Cre, Rbp4-Cre, and Scnn1a-Tg3-Cre, that respectively label layer 4, layer 5, and layer 6 excitatory neurons were selected. We only retained 285 cells that passed the authors' quality control (QC) filters. The dimensionality reduction methods were only applied to the 1000 most variable genes.

S1/CA1 dataset. This dataset characterizes 3005 cells from the primary somatosensory cortex (S1) and the hippocampal CA1 region, using the Fluidigm C1 microfluidics cell capture platform followed by Illumina sequencing (118). Gene expression is quantified by UMI counts.

mESC dataset. This dataset includes the transcriptome measurement of 704 mouse embryonic stem cells (mESCs), across three culture conditions (serum, 2i, and a2i), using the Fluidigm C1 microfluidics cell capture platform followed by Illumina sequencing (119). We excluded the samples that did not pass the authors's QC filters, resulting in a total of 169 serum cells, 141 2i cells, and 159 a2i cells. The dimensionality reduction methods were only applied to the 1000 most variable genes.

OE dataset. This data characterizes 849 FACS-purified cells from the mouse OE, using the Fluidigm C1 microfluidics cell capture platform followed by Illumina sequencing (120). We followed the filtering procedure of (121), and filtered the cells that exhibited poor sample quality, retaining a total of 747 cells.

For all datasets, hGNB was run using 2000 MCMC iterations, where after the first 1000 burn-in iterations, the posterior samples with the highest likelihood were collected as the point estimates of model parameters corresponding to latent factors. In the dimensionality reduction analysis below,

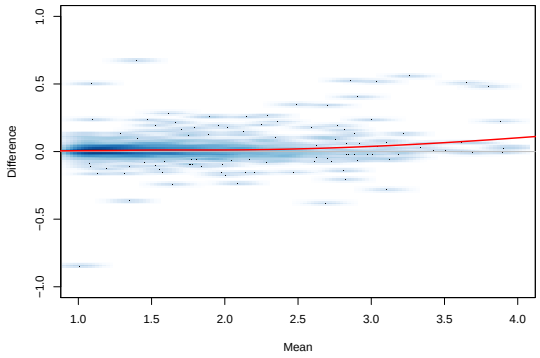


Figure 5.2: Mean-difference (MD) plot for S1/CA1 dataset. The solid red line represents the local regression fit to the data

following (113), for S1/CA1 dataset we set the number of latent factors $K = 3$, and for V1 and mESC we set $K = 2$.

5.2.1 Goodness-of-fit of hGNB Model

We have examined the goodness-of-fit of hGNB model on V1, S1/CA1 and mESC datasets, using the mean-difference (MD) plots. Figure 5.2 shows the MD plot for the S1/CA1 dataset, where the y-axis is the difference between observed counts and the expected counts under hGNB, and x-axis is the average of these two sets of counts. The solid red line in this figure, which represents the local regression fit (122) to the data, resides near zero for various average levels. This supports the good fit of hGNB model to the highly over-dispersed scRNA-seq data. Similar trends are observed for V1 and mESC datasets.

5.2.2 Capturing Zero-Inflation

Next we evaluate the performance of hGNB on simulated data based on the zero-inflated NB distribution of (113) to show that hGNB faithfully captures zero inflation without the need of explicit zero-inflation modeling. Specifically, the capability of hGNB to recover true clustering structure of cells under three zero-count prevalence levels with two different total numbers of cells. The parameters of the simulating zero-inflated model were learned based on the S1/CA1 dataset.

Genes that did not have at least five reads in at least five cells were filtered out and 1000 genes were then sampled at random for each dataset. The number of latent factors was set to $K = 2$. To simulate cell clustering, a K-variate Gaussian mixture distribution with three components was fitted to the inferred factor score parameters, and then for each simulated dataset, factor scores were generated from K-variate Gaussian distributions. By adjusting the value of regression coefficients in the zero-inflation term of ZINB-WaVE model, we generated synthetic datasets with three levels of zero-count percentages as 40%, 60% and 80% (for details refer to (113)). The number of cells were set to $J = 100$ and $J = 1000$. For each scenario, including cell numbers and zero-count prevalence (sparsity) levels, we simulated 10 datasets.

We evaluate the performance of our method for the clustering task based on the average silhouette width measure. The silhouette width s_j of sample j is defined as

$$s_j = \frac{b_j - a_j}{\max\{a_j, b_j\}},$$

where a_j is the average distance between sample j and all samples in the cluster that it belongs to, and b_j is the minimum average distance between sample j and samples in other clusters.

Figure 5.3 shows the clustering average silhouette width based on the above simulation setup, for different zero-count prevalence levels and cell numbers. In the setting with small sample size, for 40% and 60% zero fractions, hGNB has the best clustering silhouette width, and for the 80% zero fraction its performance is identical to that of ZINB-WaVE. In the setting with moderate sample size, hGNB has the best clustering silhouette width for 40% zero fraction, and for 60% and 80% zero fractions it closely follows the performance of ZINB-WaVE. This suggests that the hierarchical structure of hGNB equips it with the capacity to capture highly over-dispersed count data, even though an explicit zero-inflation term is not included in its model. Also, ZINB-WaVE requires large enough samples to have robust inference results due to the introduction of zero-inflation terms in its model. Finally, ZIFA and PCA have the worst performance, as they normalize the data before learning its latent representation.

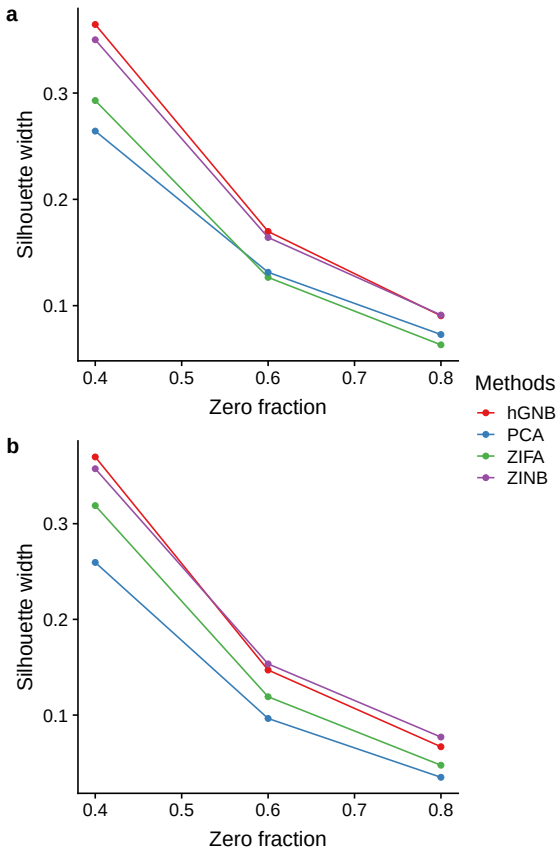


Figure 5.3: (a)$J = 100$, (b)$J = 1000$. Performance of different methods based on recovering the true cell clusters in synthetic data based on S1/CA1 dataset. Zero-inflated NB model of ZINB-WaVE is used to simulate scRNA-seq data.

5.2.3 Dimensionality Reduction

We applied hGNB to the three scRNA-seq datasets, V1, S1/CA1 and mESC, to assess its power to separate cell clusters in the low dimensional space, and compared it to PCA, ZIFA, and ZINB-WaVE methods. Figure 5.4 illustrates the projected scRNA-seq expression of profiled cells in the two-dimensional space for S1/CA1 dataset. The proposed hGNB model provides more biologically meaningful latent representations of scRNA-seq gene expressions for S1/CA1 cells, especially compared to PCA and ZIFA that do not model the counts directly. Furthermore, hGNB leads to more separated clusters of cells in the two-dimensional space, compared to ZINB-WaVE. Specifically, hGNB distinguishes `microglia` from `endothelialâĹŠmural` cells,

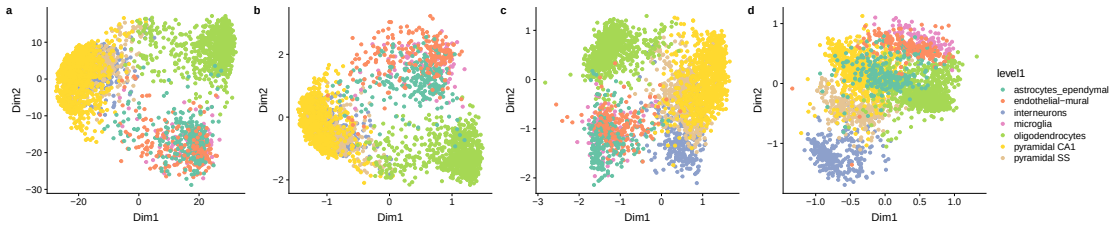


Figure 5.4: Low-dimensional representations of the S1/CA1 dataset. Panels correspond to (a) PCA (on total-count normalized data), (b) ZIFA (on total-count normalized data), (c) ZINB-WaVE, and (d) hGNB.

while ZINB-WaVE fails to accomplish this task.

To examine the dimensionality reduction results more carefully, we used the average silhouette width as a measure of goodness for clustering.

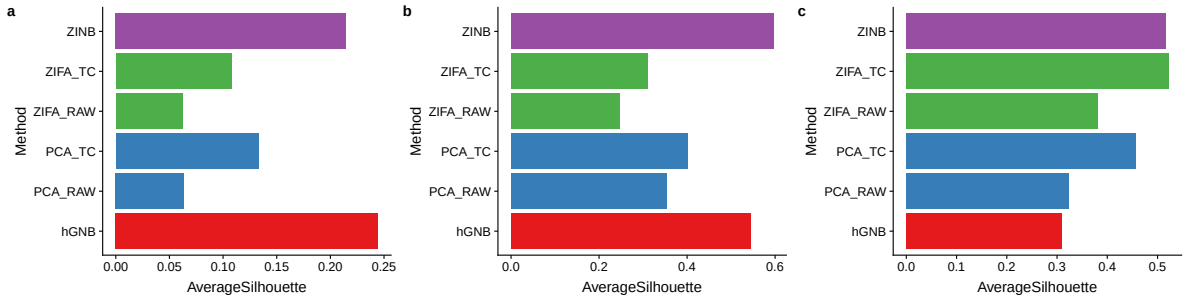


Figure 5.5: Average silhouette width in scRNA-seq datasets (a) S1/CA1, (b) mESC, and (c) V1. Silhouette widths were computed in the low-dimensional space, using the groupings provided by the authors of the original publications. PCA and ZIFA were applied with both unnormalized (RAW) data and after total count (TC) normalization.

Figure 5.5 shows the average silhouette width of different methods on V1, S1/CA1, and mESC datasets. For PCA and ZIFA, the results on both raw counts and normalized counts are included in this figure. For S1/CA1 dataset, which has the highest number of clusters, the proposed hGNB method outperforms all other methods in terms of clustering average silhouette. For mESC dataset, performance of hGNB is comparable to ZINB-WaVE, and it is significantly better than PCA and ZIFA. For V1 dataset, however, we observe that hGNB, besides PCA applied to raw counts, possess

Table 5.2: Correspondence between identified clusters and cell types in OE dataset.

Cell Type	Clusters
GBC	cl4,cl9
mSUS	cl2,cl3,cl5,cl11
mOSN	cl8,cl12,cl3
Immature Neurons	cl10
MV	cl14

the lowest average silhouette. By further examination of the latent representations of cells for this dataset, we observe that all methods split the Rbp4-Cre_KL100 cells into two clusters, one of them located near Scnn1a-Tg3-Cre cells, suggesting the presence of batch effects, which have led to confounding of latent representations (113).

5.2.4 Identification of Developmental Lineages

In addition to characterization of cell types, we further demonstrate the capability of hGNB to derive novel biological insights, by analyzing a set of cells from the mouse olfactory epithelium (OE). The samples were collected to identify the developmental trajectories that generate olfactory neurons (mOSN), sustentacular cells (mSUS), and microvillous cells (MV) (120).

We first performed dimensionality reduction on the OE dataset by applying hGNB with $K = 50$. Next, we clustered the cells using the low-dimensional factor score parameters θ_{kj}. More specifically, the resampling-based sequential ensemble clustering (RSEC) framework implemented in the `RSEC` function from the Bioconductor R package `clusterExperiment` (123) was applied to factor scores, leading to identification of 14 cell clusters. The correspondence between the detected clusters and the underlying biological cell types is presented in Table 5.2. In addition to these already known cell clusters in OE, hGNB is able to detect new clusters, potentially offering novel biological insights.

We further investigated the potential benefit of using the learned latent representation by our proposed hGNB model to infer branching cell lineages and order cells by developmental progression along each lineage. To infer the global lineage structure (i.e., the number of lineages and

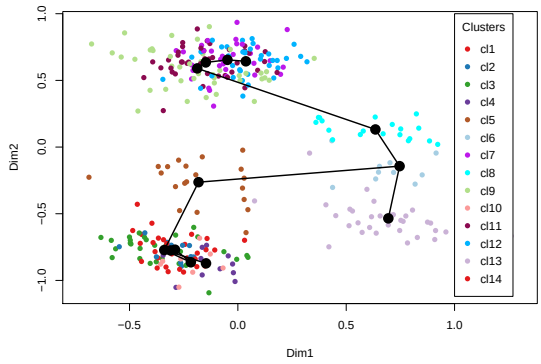


Figure 5.6: Lineage inference on the OE dataset. The low dimensional data representation derived by hGNB were used to cluster cells by `RSEC`. The minimum spanning tree (MST) of the derived clusters constructed by `slingshot` is also displayed.

where they branch), a minimum spanning tree (MST) was constructed on the clusters identified above by RSEC. We used the R package `slingshot` (124). Figure 5.6 illustrates the inferred lineages for the OE dataset, in a two-dimensional space obtained by applying multi-dimensional scaling (MDS) algorithm to the factor scores learned by hGNB. There are three branches in the inferred lineages, with endpoints located in microvillous (MV), mature olfactory sensory neurons (mOSN), and mature sustentacular (mSUS) cells.

6. CONCLUSION

In this thesis, we considered the problem of modeling count data from high-throughput RNA sequencing technologies in a fully Bayesian framework. In the begining, we exploited Bayesian nonparametric priors, including the gamma-Poisson, gamma-negative binomial, and beta-negative binomial processes, to model RNA sequencing count matrices. With different sequencing depths captured by sample-specific model parameters, the posterior distributions of certain gene-specific model parameters were used to detect the genes that are differentially expressed between different conditions. With the model parameters inferred by borrowing statistical strength across both the genes and samples, the need to adjust the raw counts using heuristics before downstream analyses, an important pre-processing step that is often required in previously proposed algorithms, was removed.

We then proposed a Bayesian negative binomial regression (BNB-R) method for differential expression analysis of sequencing count data. On one hand, BNB-R is capable of handling complex experiments involving multiple factors. On the other hand, it does not require an ad-hoc normalization preprocessing step. By taking advantage of novel data augmentation techniques, BNB-R possesses efficient closed-form Gibbs sampling update equations and ranks differentially expressed genes based on a symmetric KL-divergence measure, exploiting the full posterior distributions of the model parameters.

In the third section, we proposed a novel Bayesian covariate-dependent negative binomial factor analysis (dNBFA) method for analyzing RNA-seq count data. Our experimental results on real-world RNA-seq data demonstrate that dNBFA is capable of handling complex experiments involving multiple factors. What's more, dNBFA does not require any ad-hoc data normalization, data preprocessing, or co-expression network construction steps. By taking advantage of novel data augmentation techniques, dNBFA possesses efficient closed-form Gibbs sampling update equations. Experimental results on multiple RNA-seq data studying complex diseases, both cancer and Autism, demonstrate that our dNBFA can be directly applied to RNA-seq data to derive meaningful

functional modules and it has potential advantages over existing two-stage co-expression network based methods.

Finally, in section 5, we proposed a hierarchical Bayesian gamma-negative binomial (hGNB) model for extracting low dimensional representations from single-cell RNA sequencing (scRNA-seq) data. hGNB obviates the need for explicit modeling of the zero-inflation prevalent in scRNA-seq count data. Our hGNB can naturally account for covariate effects at both the gene and cell levels, and does not require the commonly adopted preprocessing steps such as normalization. By taking advantage of sophisticated data augmentation techniques, hGNB possesses efficient closed-form Gibbs sampling update equations. Our experimental results on real-world scRNA-seq data demonstrates that hGNB is capable of identifying insightful cell clusters, especially in complex settings.

REFERENCES

[1] Z. Wang, M. Gerstein, and M. Snyder, "RNA-Seq: a revolutionary tool for transcriptomics," *Nature Reviews Genetics*, vol. 10, no. 1, pp. 57–63, 2009.

[2] U. Nagalakshmi, Z. Wang, K. Waern, C. Shou, D. Raha, M. Gerstein, and M. Snyder, "The transcriptional landscape of the yeast genome defined by rna sequencing," *Science*, vol. 320, no. 5881, pp. 1344–1349, 2008.

[3] S. Anders, P. T. Pyl, and W. Huber, "Htseq–a python framework to work with high-throughput sequencing data," *Bioinformatics*, p. btu638, 2014.

[4] Y. Liao, G. K. Smyth, and W. Shi, "featurecounts: an efficient general purpose program for assigning sequence reads to genomic features," *Bioinformatics*, vol. 30, no. 7, pp. 923–930, 2013.

[5] M. L. Metzker, "Sequencing technologiesâĂŤthe next generation," *Nature Reviews Genetics*, vol. 11, no. 1, pp. 31–46, 2010.

[6] M. West, "Bayesian factor regression models in the "large p, small n" paradigm," in *Bayesian Statistics*, 2003.

[7] S. Anders and W. Huber, "Differential expression analysis for sequence count data," *Genome Biology*, vol. 11, no. 10, p. R106, 2010.

[8] S. Datta and D. Nettleton, *Statistical Analysis of Next Generation Sequencing Data*. Frontiers in Probability and the Statistical Sciences, Springer International Publishing, 2014.

[9] L. Wang, Z. Feng, X. Wang, X. Wang, and X. Zhang, "DEGseq: an R package for identifying differentially expressed genes from RNA-seq data," *Bioinformatics*, vol. 26, no. 1, pp. 136–138, 2010.

[10] M. D. Robinson and A. Oshlack, "A scaling normalization method for differential expression analysis of RNA-seq data," *Genome Biology*, vol. 11, no. 3, pp. 1–9, 2010.

[11] A. Oshlack, M. D. Robinson, and M. D. Young, "From RNA-seq reads to differential expression results," *Genome Biology*, vol. 11, no. 12, pp. 1–10, 2010.

[12] J. Li, D. M. Witten, I. M. Johnstone, and R. Tibshirani, "Normalization, testing, and false discovery rate estimation for RNA-sequencing data," *Biostatistics*, p. kxr031, 2011.

[13] J. Li and R. Tibshirani, "Finding consistent patterns: a nonparametric approach for identifying differential expression in RNA-Seq data," *Statistical Methods in Medical Research*, vol. 22, no. 5, pp. 519–536, 2013.

[14] A. Mortazavi, B. A. Williams, K. McCue, L. Schaeffer, and B. Wold, "Mapping and quantifying mammalian transcriptomes by RNA-Seq," *Nature methods*, vol. 5, no. 7, pp. 621–628, 2008.

[15] M. Schena, D. Shalon, R. W. Davis, and P. O. Brown, "Quantitative monitoring of gene expression patterns with a complementary DNA microarray," *Science*, vol. 270, no. 5235, p. 467, 1995.

[16] A. Roberts, C. Trapnell, J. Donaghey, J. L. Rinn, and L. Pachter, "Improving RNA-Seq expression estimates by correcting for fragment bias," *Genome biology*, vol. 12, no. 3, p. 1, 2011.

[17] M. Greenwood and G. U. Yule, "An inquiry into the nature of frequency distributions representative of multiple happenings with particular reference to the occurrence of multiple attacks of disease or of repeated accidents," *J. R. Stat. Soc.*, 1920.

[18] C. I. Bliss and R. A. Fisher, "Fitting the negative binomial distribution to biological data," *Biometrics*, vol. 9, no. 2, pp. 176–200, 1953.

[19] M. D. Robinson, D. J. McCarthy, and G. K. Smyth, "edger: a bioconductor package for differential expression analysis of digital gene expression data," *Bioinformatics*, vol. 26, no. 1, pp. 139–140, 2010.

[20] C. Soneson and M. Delorenzi, "A comparison of methods for differential expression analysis of RNA-seq data," *BMC Bioinformatics*, vol. 14, p. 91, 2013.

[21] M.-A. Dillies, A. Rau, J. Aubert, C. Hennequet-Antier, M. Jeanmougin, N. Servant, C. Keime, G. Marot, D. Castel, J. Estelle, *et al.*, "A comprehensive evaluation of normalization methods for illumina high-throughput RNA sequencing data analysis," *Briefings in*

Bioinformatics, vol. 14, no. 6, pp. 671–683, 2013.

[22] J. Zyprych-Walczak, A. Szabelska, L. Handschuh, K. Górczak, K. Klamecka, M. Figlerowicz, and I. Siatkowski, "The impact of normalization methods on RNA-Seq data analysis," *BioMed Research International*, vol. 2015, 2015.

[23] J. H. Bullard, E. Purdom, K. D. Hansen, and S. Dudoit, "Evaluation of statistical methods for normalization and differential expression in mRNA-Seq experiments," *BMC Bioinformatics*, vol. 11, no. 1, p. 94, 2010.

[24] M. I. Love, W. Huber, and S. Anders, "Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2," *Genome Biology*, vol. 15, no. 12, pp. 1–21, 2014.

[25] J. Lovén, D. A. Orlando, A. A. Sigova, C. Y. Lin, P. B. Rahl, C. B. Burge, D. L. Levens, T. I. Lee, and R. A. Young, "Revisiting global gene expression analysis," *Cell*, vol. 151, no. 3, pp. 476–482, 2012.

[26] D. J. Lorenz, R. S. Gill, R. Mitra, and S. Datta, "Using RNA-seq data to detect differentially expressed genes," in *Statistical Analysis of Next Generation Sequencing Data*, pp. 25–49, Springer, 2014.

[27] D. Risso, J. Ngai, T. P. Speed, and S. Dudoit, "Normalization of rna-seq data using factor analysis of control genes or samples," *Nature Biotechnology*, vol. 32, no. 9, pp. 896–902, 2014.

[28] D. Risso, J. Ngai, T. P. Speed, and S. Dudoit, "The role of spike-in standards in the normalization of RNA-seq," in *Statistical Analysis of Next Generation Sequencing Data*, pp. 169–190, Springer, 2014.

[29] F. Rapaport, R. Khanin, Y. Liang, M. Pirun, A. Krek, P. Zumbo, C. E. Mason, N. D. Socci, and D. Betel, "Comprehensive evaluation of differential gene expression analysis methods for RNA-seq data," *Genome Biology*, vol. 14, no. 9, p. R95, 2013.

[30] Z. H. Zhang, D. J. Jhaveri, V. M. Marshall, D. C. Bauer, J. Edson, R. K. Narayanan, G. J. Robinson, A. E. Lundberg, P. F. Bartlett, N. R. Wray, *et al.*, "A comparative study of techniques for differential expression analysis on RNA-Seq data," *PloS one*, vol. 9, no. 8,

p. e103207, 2014.

[31] G. Smyth and A. Verbyla, "A conditional likelihood approach to residual maximum likelihood estimation in generalized linear models," *J. R. Stat. Soc: Series B*, vol. 58, no. 3, pp. 565–572, 1996.

[32] M. D. Robinson and G. K. Smyth, "Moderated statistical tests for assessing differences in tag abundance," *Bioinformatics*, vol. 23, no. 21, pp. 2881–2887, 2007.

[33] T. J. Hardcastle and K. A. Kelly, "bayseq: empirical bayesian methods for identifying differential expression in sequence count data," *BMC Bioinformatics*, vol. 11, no. 1, p. 422, 2010.

[34] N. J. Schurch, P. Schofield, M. Gierliński, C. Cole, A. Sherstnev, V. Singh, N. Wrobel, K. Gharbi, G. G. Simpson, T. Owen-Hughes, *et al.*, "How many biological replicates are needed in an RNA-seq experiment and which differential expression tool should you use?," *RNA*, vol. 22, no. 6, pp. 839–851, 2016.

[35] P. Langfelder and S. Horvath, "WGCNA: an R package for weighted correlation network analysis," *BMC bioinformatics*, vol. 9, no. 1, p. 559, 2008.

[36] B. M. Tesson, R. Breitling, and R. C. Jansen, "DiffCoEx: a simple and sensitive method to find differentially coexpressed gene modules," *BMC bioinformatics*, vol. 11, no. 1, p. 497, 2010.

[37] M. Lei, J. Xu, L.-C. Huang, L. Wang, and J. Li, "Network module-based model in the differential expression analysis for RNA-seq," *Bioinformatics*, 2017.

[38] M. Zhou, O. H. M. Padilla, and J. G. Scott, "Priors for random count matrices derived from a family of negative binomial processes," *J. Amer. Statist. Assoc.*, vol. 111, no. 515, pp. 1144–1156, 2016.

[39] T. S. Ferguson, "A Bayesian analysis of some nonparametric problems," *Ann. Statist.*, vol. 1, no. 2, pp. 209–230, 1973.

[40] M. Zhou, L. Hannah, D. Dunson, and L. Carin, "Beta-negative binomial process and Poisson factor analysis," in *AISTATS*, pp. 1462–1471, 2012.

[41] M. Zhou and L. Carin, "Negative binomial process count and mixture modeling," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 2, pp. 307–320, 2015.

[42] T. Broderick, L. Mackey, J. Paisley, and M. I. Jordan, "Combinatorial clustering and the beta negative binomial process," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2015.

[43] N. L. Hjort, "Nonparametric Bayes estimators based on beta processes in models for life history data," *Ann. Statist.*, 1990.

[44] J. F. C. Kingman, *Poisson Processes*. Oxford University Press, 1993.

[45] F. Caron, Y. W. Teh, and B. T. Murphy, "Bayesian nonparametric Plackett-Luce models for the analysis of clustered ranked data," *Annal of Applied Statistics*, 2014.

[46] Cancer Genome Atlas Research Network *et al.*, "Comprehensive molecular characterization of clear cell renal cell carcinoma," *Nature*, vol. 499, no. 7456, pp. 43–49, 2012.

[47] J. Xu, Z. Su, H. Hong, J. Thierry-Mieg, D. Thierry-Mieg, D. P. Kreil, C. E. Mason, W. Tong, and L. Shi, "Cross-platform ultradeep transcriptomic profiling of human reference RNA samples by RNA-Seq," *Scientific Data*, vol. 1, pp. 140020–140020, 2013.

[48] SEQC/MAQC-III Consortium, "A comprehensive assessment of RNA-seq accuracy, reproducibility and information content by the Sequencing Quality Control Consortium," *Nature Biotechnology*, vol. 32, no. 9, pp. 903–914, 2014.

[49] R. McLendon, A. Friedman, D. Bigner, E. G. Van Meir, D. J. Brat, G. M. Mastrogianakis, J. J. Olson, T. Mikkelsen, N. Lehman, K. Aldape, *et al.*, "Comprehensive genomic characterization defines human glioblastoma genes and core pathways," *Nature*, vol. 455, no. 7216, pp. 1061–1068, 2008.

[50] R. C. Gentleman, V. J. Carey, D. M. Bates, B. Bolstad, M. Dettling, S. Dudoit, B. Ellis, L. Gautier, Y. Ge, J. Gentry, *et al.*, "Bioconductor: open software development for computational biology and bioinformatics," *Genome Biology*, vol. 5, no. 10, p. R80, 2004.

[51] C. Joyce, "Quantitative rt-pcr," *RT-PCR Protocols*, pp. 83–92, 2002.

[52] L. Shi, L. H. Reid, W. D. Jones, R. Shippy, J. A. Warrington, S. C. Baker, P. J. Collins, F. De Longueville, E. S. Kawasaki, K. Y. Lee, *et al.*, "The microarray quality control (maqc)

project shows inter-and intraplatform reproducibility of gene expression measurements," *Nature Biotechnology*, vol. 24, no. 9, pp. 1151–1161, 2006.

[53] MAQC Consortium, "The MicroArray Quality Control (MAQC)-II study of common practices for the development and validation of microarray-based predictive models," *Nature Biotechnology*, vol. 28, no. 8, pp. 827–838, 2010.

[54] G. K. Smyth, "Linear models and empirical bayes methods for assessing differential expression in microarray experiments," *Statistical applications in genetics and molecular biology*, vol. 3, no. 1, pp. 1–25, 2004.

[55] G. K. Smyth, "Limma: linear models for microarray data," in *Bioinformatics and computational biology solutions using R and Bioconductor*, pp. 397–420, Springer, 2005.

[56] M. D. Robinson, D. J. McCarthy, and G. K. Smyth, "edgeR: a Bioconductor package for differential expression analysis of digital gene expression data," *Bioinformatics*, vol. 26, no. 1, pp. 139–140, 2010.

[57] M. I. Love, W. Huber, and S. Anders, "Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2," *Genome biology*, vol. 15, no. 12, p. 550, 2014.

[58] C. W. Law, Y. Chen, W. Shi, and G. K. Smyth, "Voom: precision weights unlock linear model analysis tools for RNA-seq read counts," *Genome biology*, vol. 15, no. 2, p. R29, 2014.

[59] M. Zhou, L. Li, D. Dunson, and L. Carin, "Lognormal and gamma mixed negative binomial regression," in *ICML 2012*, 2012.

[60] M. Zhou and L. Carin, "Negative binomial process count and mixture modeling," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 2, pp. 307–320, 2015.

[61] N. G. Polson and J. G. Scott, "Default Bayesian analysis for multi-way tables: a data-augmentation approach," *arXiv preprint arXiv:1109.4180*, 2011.

[62] S. Chib and E. Greenberg, "Understanding the metropolis-hastings algorithm," *The american statistician*, vol. 49, no. 4, pp. 327–335, 1995.

[63] W. Gardner, E. P. Mulvey, and E. C. Shaw, "Regression analyses of counts and rates: Poisson, overdispersed Poisson, and negative binomial models.," *Psychological bulletin*, vol. 118, no. 3, p. 392, 1995.

[64] C. Andrieu, N. De Freitas, A. Doucet, and M. I. Jordan, "An introduction to mcmc for machine learning," *Machine learning*, vol. 50, no. 1-2, pp. 5–43, 2003.

[65] S. Z. Dadaneh, X. Qian, and M. Zhou, "BNP-Seq: Bayesian nonparametric differential expression analysis of sequencing count data," *Journal of the American Statistical Association*, no. in-press, doi:10.1080/01621459.2017.1328358, 2017.

[66] J. T. Leek, "Svaseq: removing batch effects and other unwanted noise from sequencing data," *Nucleic acids research*, vol. 42, no. 21, pp. e161–e161, 2014.

[67] SEQC/MAQC-III Consortium, "A comprehensive assessment of RNA-seq accuracy, reproducibility and information content by the Sequencing Quality Control Consortium," *Nature biotechnology*, vol. 32, no. 9, pp. 903–914, 2014.

[68] R. C. Gentleman, V. J. Carey, D. M. Bates, B. Bolstad, M. Dettling, S. Dudoit, B. Ellis, L. Gautier, Y. Ge, J. Gentry, *et al.*, "Bioconductor: open software development for computational biology and bioinformatics," *Genome biology*, vol. 5, no. 10, p. R80, 2004.

[69] Maqc Consortium and others, "The MicroArray Quality Control (MAQC) project shows inter-and intraplatform reproducibility of gene expression measurements," *Nature biotechnology*, vol. 24, no. 9, p. 1151, 2006.

[70] F. Rapaport, R. Khanin, Y. Liang, M. Pirun, A. Krek, P. Zumbo, C. E. Mason, N. D. Socci, and D. Betel, "Comprehensive evaluation of differential gene expression analysis methods for RNA-seq data," *Genome biology*, vol. 14, no. 9, p. 3158, 2013.

[71] J. T. Leek, W. E. Johnson, H. S. Parker, A. E. Jaffe, and J. D. Storey, "The sva package for removing batch effects and other unwanted variation in high-throughput experiments," *Bioinformatics*, vol. 28, no. 6, pp. 882–883, 2012.

[72] S. Tuomela, V. Salo, S. K. Tripathi, Z. Chen, K. Laurila, B. Gupta, T. Äijö, L. Oikari, B. Stockinger, H. Lähdesmäki, *et al.*, "Identification of early gene expression changes during

human Th17 cell differentiation," *Blood*, vol. 119, no. 23, pp. e151–e160, 2012.

[73] T. Äijö, V. Butty, Z. Chen, V. Salo, S. Tripathi, C. B. Burge, R. Lahesmaa, and H. Lähdesmäki, "Methods for time series analysis of RNA-seq data with application to human Th17 cell differentiation," *Bioinformatics*, vol. 30, no. 12, pp. i113–i120, 2014.

[74] S. Tuomela, S. Rautio, H. Ahlfors, V. Öling, V. Salo, U. Ullah, Z. Chen, S. Hämälistö, S. K. Tripathi, T. Äijö, *et al.*, "Comparative analysis of human and mouse transcriptomes of Th17 cell priming," *Oncotarget*, vol. 7, no. 12, p. 13416, 2016.

[75] Y. H. Chan, J. Intosalmi, S. Rautio, and H. Lähdesmäki, "A subpopulation model to analyze heterogeneous cell differentiation dynamics," *Bioinformatics*, vol. 32, no. 21, pp. 3306–3313, 2016.

[76] E. I. Boyle, S. Weng, J. Gollub, H. Jin, D. Botstein, J. M. Cherry, and G. Sherlock, "GO:: TermFinder—open source software for accessing Gene Ontology information and finding significantly enriched Gene Ontology terms associated with a list of genes," *Bioinformatics*, vol. 20, no. 18, pp. 3710–3715, 2004.

[77] M. Pasarica, B. Gowronska-Kozak, D. Burk, I. Remedios, D. Hymel, J. Gimble, E. Ravussin, G. A. Bray, and S. R. Smith, "Adipose tissue collagen VI in obesity," *The Journal of Clinical Endocrinology & Metabolism*, vol. 94, no. 12, pp. 5155–5162, 2009.

[78] S. Metcalfe, "LIF in the regulation of T-cell fate and as a potential therapeutic," *Genes and immunity*, vol. 12, no. 3, p. 157, 2011.

[79] C. Diveu, M. J. McGeachy, K. Boniface, J. S. Stumhofer, M. Sathe, B. Joyce-Shaikh, Y. Chen, C. M. Tato, T. K. McClanahan, R. de Waal Malefyt, *et al.*, "IL-27 blocks RORc expression to inhibit lineage commitment of Th17 cells," *The Journal of Immunology*, vol. 182, no. 9, pp. 5748–5756, 2009.

[80] D. Nam and S.-Y. Kim, "Gene-set approach for expression pattern analysis," *Briefings in bioinformatics*, vol. 9, no. 3, pp. 189–197, 2008.

[81] S. B. Cho, J. Kim, and J. H. Kim, "Identifying set-wise differential co-expression in gene expression microarray data," *BMC bioinformatics*, vol. 10, no. 1, p. 109, 2009.

[82] J. K. Choi, U. Yu, O. J. Yoo, and S. Kim, "Differential coexpression analysis using microarray data and its application to human cancer," *Bioinformatics*, vol. 21, no. 24, pp. 4348–4355, 2005.

[83] Y. Choi and C. Kendziorski, "Statistical methods for gene set co-expression analysis," *Bioinformatics*, vol. 25, no. 21, pp. 2780–2786, 2009.

[84] M. Zhou, "Nonparametric Bayesian negative binomial factor analysis," 2017. Bayesian Analysis, advance publication, November 2017, doi:10.1214/17-BA1070, https://doi.org/10.1214/17-BA1070.

[85] S. Anders and W. Huber, "Differential expression analysis for sequence count data," *Genome biology*, vol. 11, no. 10, p. R106, 2010.

[86] M. Zhou, L. Li, D. Dunson, and L. Carin, "Lognormal and gamma mixed negative binomial regression," in *ICML*, pp. 1343–1350, 2012.

[87] N. G. Polson, J. G. Scott, and J. Windle, "Bayesian inference for logistic models using Pólya–Gamma latent variables," *J. Amer. Statist. Assoc.*, vol. 108, no. 504, pp. 1339–1349, 2013.

[88] N. L. Johnson, A. W. Kemp, and S. Kotz, *Univariate discrete distributions*, vol. 444. John Wiley & Sons, 2005.

[89] Cancer Genome Atlas (TCGA) Research Network and others, "Comprehensive genomic characterization defines human glioblastoma genes and core pathways," *Nature*, vol. 455, no. 7216, p. 1061, 2008.

[90] Y.-W. Wan, G. I. Allen, and Z. Liu, "TCGA2STAT: simple TCGA data access for integrated statistical analysis in R," *Bioinformatics*, vol. 32, no. 6, pp. 952–954, 2015.

[91] S. Gupta, S. E. Ellis, F. N. Ashar, A. Moes, J. S. Bader, J. Zhan, A. B. West, and D. E. Arking, "Transcriptome analysis reveals dysregulation of innate immune response genes and neuronal activity-dependent genes in autism," *Nature communications*, vol. 5, p. 5748, 2014.

[92] E. Ravasz, A. L. Somera, D. A. Mongru, Z. N. Oltvai, and A.-L. Barabási, "Hierarchical

organization of modularity in metabolic networks," *science*, vol. 297, no. 5586, pp. 1551–1555, 2002.

[93] S. Horvath and J. Dong, "Geometric interpretation of gene coexpression network analysis," *PLoS computational biology*, vol. 4, no. 8, p. e1000117, 2008.

[94] S. Delhaye, S. Paul, G. Blakqori, M. Minet, F. Weber, P. Staeheli, and T. Michiels, "Neurons produce type I interferon during viral encephalitis," *Proceedings of the National Academy of Sciences*, vol. 103, no. 20, pp. 7835–7840, 2006.

[95] P. H. Patterson, "Maternal infection and immune involvement in autism," *Trends in molecular medicine*, vol. 17, no. 7, pp. 389–394, 2011.

[96] P. Ashwood, S. Wills, and J. Van de Water, "The immune response in autism: a new frontier for autism research," *Journal of leukocyte biology*, vol. 80, no. 1, pp. 1–15, 2006.

[97] L. Heuer, P. Ashwood, J. Schauer, P. Goines, P. Krakowiak, I. Hertz-Picciotto, R. Hansen, L. A. Croen, I. N. Pessah, and J. Van de Water, "Reduced levels of immunoglobulin in children with autism correlates with behavioral symptoms," *Autism Research*, vol. 1, no. 5, pp. 275–283, 2008.

[98] A. R. Torres, J. B. Westover, and A. J. Rosenspire, "HLA immune function genes in autism," *Autism research and treatment*, vol. 2012, 2012.

[99] R. P. Warren, J. D. Odell, W. L. Warren, R. A. Burger, A. Maciulis, W. W. Daniels, and A. R. Torres, "Strong association of the third hypervariable region of HLA-DRβ1 with autism," *Journal of Neuroimmunology*, vol. 67, no. 2, pp. 97–102, 1996.

[100] A. R. Torres, A. Maciulis, E. G. Stubbs, A. Cutler, and D. Odell, "The transmission disequilibrium test suggests that HLA-DR4 and DR13 are linked to autism spectrum disorder," *Human immunology*, vol. 63, no. 4, pp. 311–316, 2002.

[101] D. Hadley, Z.-l. Wu, C. Kao, A. Kini, A. Mohamed-Hadley, K. Thomas, L. Vazquez, H. Qiu, F. Mentch, R. Pellegrino, *et al.*, "The impact of the metabotropic glutamate receptor and other gene family interaction networks on autism," *Nature communications*, vol. 5, p. 4074, 2014.

[102] B. Zikopoulos and H. Barbas, "Changes in prefrontal axons may disrupt the network in autism," *Journal of Neuroscience*, vol. 30, no. 44, pp. 14595–14609, 2010.

[103] D. Braida, F. Guerini, L. Ponzoni, I. Corradini, S. De Astis, L. Pattini, E. Bolognesi, R. Benfante, D. Fornasari, M. Chiappedi, *et al.*, "Association between SNAP-25 gene polymorphisms and cognition in autism: functional consequences and potential therapeutic strategies," *Translational psychiatry*, vol. 5, no. 1, p. e500, 2016.

[104] J. B. Adams, T. Audhya, S. McDonough-Means, R. A. Rubin, D. Quig, E. Geis, E. Gehn, M. Loresto, J. Mitchell, S. Atwood, *et al.*, "Nutritional and metabolic status of children with autism vs. neurotypical children, and the association with autism severity," *Nutrition & metabolism*, vol. 8, no. 1, p. 34, 2011.

[105] C. A. Walsh, E. M. Morrow, and J. L. Rubenstein, "Autism and brain development," *Cell*, vol. 135, no. 3, pp. 396–400, 2008.

[106] E. Shapiro, T. Biezuner, and S. Linnarsson, "Single-cell sequencing-based technologies will revolutionize whole-organism science," *Nature Reviews Genetics*, vol. 14, no. 9, p. 618, 2013.

[107] E. Z. Macosko, A. Basu, R. Satija, J. Nemesh, K. Shekhar, M. Goldman, I. Tirosh, A. R. Bialas, N. Kamitaki, E. M. Martersteck, *et al.*, "Highly parallel genome-wide expression profiling of individual cells using nanoliter droplets," *Cell*, vol. 161, no. 5, pp. 1202–1214, 2015.

[108] Q. Deng, D. Ramsköld, B. Reinius, and R. Sandberg, "Single-cell rna-seq reveals dynamic, random monoallelic gene expression in mammalian cells," *Science*, vol. 343, no. 6167, pp. 193–196, 2014.

[109] A. P. Patel, I. Tirosh, J. J. Trombetta, A. K. Shalek, S. M. Gillespie, H. Wakimoto, D. P. Cahill, B. V. Nahed, W. T. Curry, R. L. Martuza, *et al.*, "Single-cell rna-seq highlights intratumoral heterogeneity in primary glioblastoma," *Science*, vol. 344, no. 6190, pp. 1396–1401, 2014.

[110] G. Finak, A. McDavid, M. Yajima, J. Deng, V. Gersuk, A. K. Shalek, C. K. Slichter, H. W.

Miller, M. J. McElrath, M. Prlic, *et al.*, "Mast: a flexible statistical framework for assessing transcriptional changes and characterizing heterogeneity in single-cell rna sequencing data," *Genome biology*, vol. 16, no. 1, p. 278, 2015.

[111] E. Pierson and C. Yau, "Zifa: Dimensionality reduction for zero-inflated single-cell gene expression analysis," *Genome biology*, vol. 16, no. 1, p. 241, 2015.

[112] M. E. Tipping and C. M. Bishop, "Probabilistic principal component analysis," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 61, no. 3, pp. 611–622, 1999.

[113] D. Risso, F. Perraudeau, S. Gribkova, S. Dudoit, and J.-P. Vert, "A general and flexible method for signal extraction from single-cell rna-seq data," *Nature communications*, vol. 9, no. 1, p. 284, 2018.

[114] D. Risso, K. Schwartz, G. Sherlock, and S. Dudoit, "Gc-content normalization for rna-seq data," *BMC bioinformatics*, vol. 12, no. 1, p. 480, 2011.

[115] D. P. Wipf and S. S. Nagarajan, "A new view of automatic relevance determination," in *Advances in neural information processing systems*, pp. 1625–1632, 2008.

[116] M. E. Tipping, "Sparse bayesian learning and the relevance vector machine," *Journal of machine learning research*, vol. 1, no. Jun, pp. 211–244, 2001.

[117] B. Tasic, V. Menon, T. N. Nguyen, T. K. Kim, T. Jarsky, Z. Yao, B. Levi, L. T. Gray, S. A. Sorensen, T. Dolbeare, *et al.*, "Adult mouse cortical cell taxonomy revealed by single cell transcriptomics," *Nature neuroscience*, vol. 19, no. 2, p. 335, 2016.

[118] A. Zeisel, A. B. Muñoz-Manchado, S. Codeluppi, P. Lönnerberg, G. La Manno, A. Juréus, S. Marques, H. Munguba, L. He, C. Betsholtz, *et al.*, "Cell types in the mouse cortex and hippocampus revealed by single-cell rna-seq," *Science*, vol. 347, no. 6226, pp. 1138–1142, 2015.

[119] A. A. Kolodziejczyk, J. K. Kim, J. C. Tsang, T. Ilicic, J. Henriksson, K. N. Natarajan, A. C. Tuck, X. Gao, M. Bühler, P. Liu, *et al.*, "Single cell rna-sequencing of pluripotent states unlocks modular transcriptional variation," *Cell stem cell*, vol. 17, no. 4, pp. 471–485, 2015.

[120] R. B. Fletcher, D. Das, L. Gadye, K. N. Street, A. Baudhuin, A. Wagner, M. B. Cole, Q. Flores, Y. G. Choi, N. Yosef, *et al.*, "Deconstructing olfactory stem cell trajectories at single-cell resolution," *Cell stem cell*, vol. 20, no. 6, pp. 817–830, 2017.

[121] F. Perraudeau, D. Risso, K. Street, E. Purdom, and S. Dudoit, "Bioconductor workflow for single-cell rna sequencing: Normalization, dimensionality reduction, clustering, and lineage inference," *F1000Research*, vol. 6, 2017.

[122] W. M. Shyu, E. Grosse, and W. S. Cleveland, "Local regression models," in *Statistical models in S*, pp. 309–376, Routledge, 2017.

[123] E. Purdom and D. Risso, "clusterexperiment: Compare clusterings for single-cell sequencing," *R package version*, vol. 1, no. 0, 2017.

[124] K. Street, D. Risso, R. B. Fletcher, D. Das, J. Ngai, N. Yosef, E. Purdom, and S. Dudoit, "Slingshot: Cell lineage and pseudotime inference for single-cell transcriptomics," *BMC genomics*, vol. 19, no. 1, p. 477, 2018.

APPENDIX A

ADDITIONAL TABLES

Table A.1: AUC-ROC in the GNBP simulation setup for different true fold changes.

	Fold change			
Method	1.4	1.6	1.8	2
GNBP	**0.9226** $\pm$ 0.006	**0.9625** $\pm$ 0.003	0.9777 $\pm$ 0.003	0.9864 $\pm$ 0.002
BNBP	0.9156 $\pm$ 0.005	0.9610 $\pm$ 0.003	**0.9783** $\pm$ 0.002	**0.9875** $\pm$ 0.002
edgeR	0.9004 $\pm$ 0.007	0.9463 $\pm$ 0.004	0.9653 $\pm$ 0.003	0.9778 $\pm$ 0.003
DESeq	0.8986 $\pm$ 0.008	0.9444 $\pm$ 0.004	0.9634 $\pm$ 0.003	0.9764 $\pm$ 0.003
baySeq	0.7542 $\pm$ 0.008	0.8247 $\pm$ 0.012	0.8752 $\pm$ 0.003	0.9114 $\pm$ 0.008
NBP	0.9035 $\pm$ 0.007	0.9476 $\pm$ 0.004	0.9665 $\pm$ 0.003	0.9786 $\pm$ 0.003
NBPscaled	0.8596 $\pm$ 0.014	0.8990 $\pm$ 0.017	0.9366 $\pm$ 0.009	0.9506 $\pm$ 0.0053

Table A.2: AUC-PR in the GNBP simulation setup for different true fold changes.

	Fold change			
Method	1.4	1.6	1.8	2
GNBP	0.7873 $\pm$ 0.011	**0.8998** $\pm$ 0.006	**0.9382** $\pm$ 0.003	**0.9607** $\pm$ 0.003
BNBP	0.5660 $\pm$ 0.011	0.8189 $\pm$ 0.008	0.9213 $\pm$ 0.005	0.9563 $\pm$ 0.002
edgeR	0.7857 $\pm$ 0.015	0.8742 $\pm$ 0.007	0.9136 $\pm$ 0.003	0.9403 $\pm$ 0.003
DESeq	0.7848 $\pm$ 0.014	0.8714 $\pm$ 0.007	0.9107 $\pm$ 0.002	0.9369 $\pm$ 0.004
baySeq	0.6517 $\pm$ 0.012	0.7655 $\pm$ 0.015	0.8329 $\pm$ 0.003	0.8756 $\pm$ 0.004
NBP	**0.7934** $\pm$ 0.014	0.8770 $\pm$ 0.007	0.9156 $\pm$ 0.003	0.9399 $\pm$ 0.003
NBPscaled	0.6822 $\pm$ 0.035	0.7515 $\pm$ 0.036	0.8298 $\pm$ 0.028	0.8533 $\pm$ 0.012

Table A.3: AUC-ROC in the BNBP simulation setup for different true fold changes.

	Fold change			
Method	1.4	1.6	1.8	2
GNBP	**0.9648** $\pm$ 0.001	**0.9847** $\pm$ 0.001	**0.9914** $\pm$ 0.0014	**0.9968** $\pm$ 0.001
BNBP	0.9635 $\pm$ 0.001	**0.9848** $\pm$ 0.002	**0.9922** $\pm$ 0.0009	**0.9971** $\pm$ 0.0009
edgeR	0.9399 $\pm$ 0.001	0.9706 $\pm$ 0.003	0.9829 $\pm$ 0.0017	0.9929 $\pm$ 0.00189
DESeq	0.9383 $\pm$ 0.002	0.9694 $\pm$ 0.003	0.9818 $\pm$ 0.0016	0.9920 $\pm$ 0.0018
baySeq	0.7919 $\pm$ 0.007	0.8699 $\pm$ 0.07	0.9167 $\pm$ 0.007	0.9590 $\pm$ 0.0041
NBP	0.9438 $\pm$ 0.001	0.9729 $\pm$ 0.003	0.9844 $\pm$ 0.002	0.9935 $\pm$ 0.0016
NBPscaled	0.8939 $\pm$ 0.0107	0.9499 $\pm$ 0.0092	0.9606 $\pm$ 0.0094	0.9811 $\pm$ 0.008

Table A.4: AUC-PR in the BNBP simulation setup for different true fold changes.

	Fold change			
Method	1.4	1.6	1.8	2
GNBP	0.8632 $\pm$ 0.011	**0.9431** $\pm$ 0.005	**0.9703** $\pm$ 0.003	**0.9881** $\pm$ 0.002
BNBP	0.8356 $\pm$ 0.012	**0.9432** $\pm$ 0.003	**0.9725** $\pm$ 0.002	**0.9889** $\pm$ 0.003
edgeR	0.8674 $\pm$ 0.006	0.9275 $\pm$ 0.005	0.9557 $\pm$ 0.003	0.9783 $\pm$ 0.004
DESeq	0.8634 $\pm$ 0.004	0.9240 $\pm$ 0.005	0.9523 $\pm$ 0.003	0.9759 $\pm$ 0.003
baySeq	0.7413 $\pm$ 0.015	0.8408 $\pm$ 0.01	0.8963 $\pm$ 0.007	0.9434 $\pm$ 0.003
NBP	**0.8708** $\pm$ 0.006	0.9302 $\pm$ 0.005	0.9577 $\pm$ 0.003	0.9798 $\pm$ 0.003
NBPscaled	0.7450 $\pm$ 0.03	0.8648 $\pm$ 0.019	0.8846 $\pm$ 0.028	0.9318 $\pm$ 0.025

Table A.5: AUC-ROC in the baySeq simulation setup for different true fold changes.

	Fold change			
Method	1.4	1.6	1.8	2
GNBP	0.8772 $\pm$ 0.009	0.9286 $\pm$ 0.005	0.9585 $\pm$ 0.004	0.9738 $\pm$ 0.001
BNBP	**0.8823** $\pm$ 0.005	**0.9382** $\pm$ 0.004	**0.9674** $\pm$ 0.003	**0.9812** $\pm$ 0.0015
edgeR	0.8702 $\pm$ 0.008	0.9216 $\pm$ 0.0042	0.9518 $\pm$ 0.004	0.9687 $\pm$ 0.003
DESeq	0.8705 $\pm$ 0.0083	0.9220 $\pm$ 0.004	0.9520 $\pm$ 0.0036	0.9688 $\pm$ 0.003
baySeq	0.7222 $\pm$ 0.0089	0.7887 $\pm$ 0.0067	0.8489 $\pm$ 0.012	0.8911 $\pm$ 0.0099
NBP	0.8769 $\pm$ 0.0075	0.9270 $\pm$ 0.0045	0.9567 $\pm$ 0.0031	0.9725 $\pm$ 0.0026
NBPscaled	0.8752 $\pm$ 0.009	0.9248 $\pm$ 0.0044	0.9571 $\pm$ 0.0071	0.9719 $\pm$ 0.0031

Table A.6: AUC-PR in the baySeq simulation setup for different true fold changes.

	Fold change			
Method	1.4	1.6	1.8	2
GNBP	**0.7194** $\pm$ 0.015	**0.8372** 0.0095 $\pm$	**0.8984** $\pm$ 0.0074	**0.9332** $\pm$ 0.0041
BNBP	0.5733 $\pm$ 0.012	0.7448 $\pm$ 0.014	0.8826 $\pm$ 0.0055	**0.9337** $\pm$ 0.0058
edgeR	0.7004 $\pm$ 0.013	0.8152 $\pm$ 0.008	0.8787 $\pm$ 0.0066	0.9173 $\pm$ 0.0054
DESeq	0.7042 $\pm$ 0.013	0.8180 $\pm$ 0.0082	0.8813 $\pm$ 0.0058	0.9194 $\pm$ 0.005
baySeq	0.5806 $\pm$ 0.0096	0.7034 $\pm$ 0.0057	0.7877 $\pm$ 0.0104	0.8482 $\pm$ 0.0106
NBP	**0.7223** $\pm$ 0.0129	0.8312 $\pm$ 0.0075	0.8913 $\pm$ 0.0058	0.9248 $\pm$ 0.0055
NBPscaled	0.7203 $\pm$ 0.0155	0.8333 $\pm$ 0.006	**0.8940** $\pm$ 0.012	0.9256 $\pm$ 0.0047

APPENDIX B

CHINESE RESTAURANT TABLE (CRT) DISTRIBUTION

The negative binomial distribution $m \sim \text{NB}(r, p)$ with the probability mass function

$$f_M(m) = \frac{\Gamma(m+r)}{m!\Gamma(r)}(1-p)^r p^m, \;\; m \in \{0, 1, \ldots\}$$

can be augmented as a gamma mixed Poisson distribution as

$$m \sim \text{Pois}(\lambda), \; \lambda \sim \text{Gamma}(r, p/(1-p)),$$

where the gamma distribution is parametrized by its shape r and scale $p/(1-p)$. It can be augmented under a compound Poisson representation as

$$m = \sum_{t=1}^{\ell} u_t, \; u_t \sim \text{Log}(p), \; \ell \sim \text{Pois}(-r \ln(1-p)),$$

where $u \sim \text{Log}(p)$ is the logarithmic distribution with probability generation function $C_U(z) = \ln(1-pz)/\ln(1-p)$, $|z| < p^{-1}$. As in (41), we denote the conditional posterior distribution of ℓ given m and r by $(\ell \,|\, m, r) \sim \text{CRT}(m, r)$ and sample it with the summation of independent Bernoulli random variables as $\ell = \sum_{n=1}^{m} b_n$, $b_n \sim \text{Bernoulli}[r/(n-1+r)]$.