Biometrics 000, 000–000 DOI: 000
000 0000

Scalable Estimation and Regularization for the Logistic Normal Multinomial Model

Jingru Zhang* and Wei Lin**

School of Mathematical Sciences and Center for Statistical Science, Peking University,

Beijing 100871, China

**email:* jingruzhang@pku.edu.cn

***email:* weilin@math.pku.edu.cn

Summary: Clustered multinomial data are prevalent in a variety of applications such as microbiome studies, where metagenomic sequencing data are summarized as multinomial counts for a large number of bacterial taxa per subject. Count normalization with ad hoc zero adjustment tends to result in poor estimates of abundances for taxa with zero or small counts. To account for heterogeneity and overdispersion in such data, we suggest using the logistic normal multinomial (LNM) model with an arbitrary correlation structure to simultaneously estimate the taxa compositions by borrowing information across subjects. We overcome the computational difficulties in high dimensions by developing a stochastic approximation EM algorithm with Hamiltonian Monte Carlo sampling for scalable parameter estimation in the LNM model. The ill-conditioning problem due to unstructured covariance is further mitigated by a covariance-regularized estimator with a condition number constraint. The advantages of the proposed methods are illustrated through simulations and an application to human gut microbiome data.

Key words: Compositional data; Condition number; Hamiltonian Monte Carlo; Logistic normal multinomial; Microbiome; Stochastic approximation EM.

This paper has been submitted for consideration for publication in *Biometrics*

1. Introduction

Clustered multinomial data are commonly encountered in a variety of modern applications such as longitudinal studies, social surveys, and text analysis. In particular, in microbiome studies that motivate this work, metagenomic sequencing experiments yield read counts of hundreds of bacterial taxa for each subject; the multinomial data are clustered in that reads from the same subject tend to be sampled from the same microbiome composition and hence more alike than those from different subjects (Agresti, 2013, Chapter 13). The quantities of interest are the underlying microbiome compositions for individual subjects, which can be linked to clinical responses or nutrient intakes using statistical methods recently developed for compositional data (Li, 2015). It is common practice to normalize the bacterial counts into proportions, which tends to give poor estimates of abundances for taxa with zero or small counts. Moreover, since the proportions usually contain a lot of zeros while many statistical methods for compositional data require strictly positive proportions, ad hoc zero adjustment procedures are often applied, whose influence on subsequent data analysis remains obscure and unexplored.

To account for heterogeneity and overdispersion as is typical of multinomial data in microbiome studies, a principled approach is to treat the taxa composition as unobserved random variables, which leads to several useful models. In particular, the Dirichlet-multinomial model (Mosimann, 1962; Chen and Li, 2013), which imposes a Dirichlet distribution on the taxa composition, is computationally simple but entails correlation structures among the taxa that are too restrictive to be satisfied in practice. By contrast, the logistic normal multinomial (LNM) model assumes logistic normal compositions and is preferable for its flexibility to allow for a general correlation structure among the taxa. The LNM model has been considered and applied to the analysis of ecological and metagenomic data by Billheimer, Guttorp, and Fagan (2001) and Xia et al. (2013). However, their works focused on associating taxa compositions

with environmental disturbances or nutrient intakes, and were confined to only a few taxa owing to the computationally expensive MCMC implementations of their methods.

In this article, we treat the zero counts in the multinomial data as sampling zeros. We suggest using the LNM model with an arbitrary correlation structure to simultaneously estimate the taxa compositions by borrowing information across subjects. To make this approach feasible, a computationally efficient algorithm for fitting the LNM model, which can scale up to hundreds of taxa, is required. Parameter estimation in the LNM model is notoriously difficult due to its intractable likelihood. Existing methods for fitting the LNM model are mainly adapted from those developed for generalized linear mixed models (GLMMs); see, for example, Hartzel, Agresti, and Caffo (2001) and Hedeker (2003). GLMM fitting techniques such as Gauss–Hermite quadrature or Monte Carlo EM (MCEM) algorithms are computationally intensive and scale up to only a few taxa. Methods based on analytic likelihood approximation such as the Laplace approximation are faster but generally biased. Variational methods have also been applied to closely related models (Blei and Lafferty, 2007; Braun and McAuliffe, 2010), trading off accuracy for speed.

Our work is complementary to those of Billheimer et al. (2001) and Xia et al. (2013), and extends their work to the high-dimensional setting where the number of taxa is large. Our contributions are twofold. First, we develop a fast, scalable algorithm for maximum likelihood estimation in the LNM model. In view of the computational bottleneck in the simulation step of the MCEM algorithm, we propose an accelerated algorithm by combining the stochastic approximation EM (SAEM) algorithm (Delyon, Lavielle, and Moulines, 1999) with Hamiltonian Monte Carlo (HMC) sampling (Duane et al., 1987; Neal, 2011). The SAEM algorithm makes a more efficient use of past samples and requires fewer simulations than MCEM to converge, while HMC leverages gradient information to provide faster mixing rates than standard MCMC methods such as the random walk Metropolis. The combined speedup

is remarkable and allows our algorithm to scale up to hundreds of taxa. Second, to mitigate the ill-conditioning problem due to a large number of covariance parameters, we introduce a regularization method for the LNM model by imposing a condition number constraint on the covariance. The resulting procedure is still computationally highly efficient, while preserving permutation invariance and relying on no sparsity assumptions.

The rest of the article is structured as follows. Section 2 introduces the LNM model. Section 3 presents the SAEM algorithm with HMC sampling. The regularized estimator and algorithm are described in Section 4. Simulation studies and an application to human gut microbiome data are presented in Sections 5 and 6, respectively. We conclude with some discussion in Section 7.

2. Logistic Normal Multinomial Model

Suppose we observe independent count vectors $\boldsymbol{x}_i = (x_{i1}, \ldots, x_{ip})^T$, $i = 1, \ldots, n$, for p taxa on n subjects. We assume that $\boldsymbol{x}_i$ are multinomial with total counts m_i and taxa compositions $\boldsymbol{\pi}_i = (\pi_{i1}, \ldots, \pi_{ip})^T$, where $m_i = \sum_{j=1}^p x_{ij}$, $\pi_{ij} > 0$, and $\sum_{j=1}^p \pi_{ij} = 1$. Throughout we condition on m_i, but treat $\boldsymbol{\pi}_i$ as random to account for extra-multinomial variability. Using the pth taxon as the reference taxon, define the additive log-ratio transformation ϕ that maps $\boldsymbol{\pi}_i$ to $\boldsymbol{y}_i$ (Aitchison, 2003, Section 6.2) by

$$y_{ij} = \log(\pi_{ij}/\pi_{ip}), \qquad j = 1, \ldots, p-1,$$

and $\boldsymbol{y}_i = (y_{i1}, \ldots, y_{i,p-1})^T$. The logistic normal multinomial (LNM) model (Billheimer et al., 2001; Xia et al., 2013) assumes that $\boldsymbol{y}_i$ are independent and identically distributed as $N_{p-1}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ with mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$, that is,

$$\boldsymbol{y}_i = \boldsymbol{\mu} + \boldsymbol{\alpha}_i, \tag{1}$$

where $\boldsymbol{\mu} \in \mathbb{R}^{p-1}$ and $\boldsymbol{\alpha}_i \sim N_{p-1}(\mathbf{0}, \boldsymbol{\Sigma})$. Note that this may be viewed as a random effects model, where $\boldsymbol{\alpha}_i$ are unobserved, subject-specific random effects. When additional covariates

are available, model (1) can easily be extended to incorporate fixed or random covariate effects, which we do not pursue in this article.

The inverse of ϕ, the additive logistic transformation ψ, maps $\boldsymbol{y}_i$ back to $\boldsymbol{\pi}_i$ by

$$\pi_{ij} = \frac{e^{y_{ij}}}{1 + \sum_{k=1}^{p-1} e^{y_{ik}}}, \qquad j = 1, \ldots, p-1, \tag{2}$$

and

$$\pi_{ip} = \frac{1}{1 + \sum_{k=1}^{p-1} e^{y_{ik}}}. \tag{3}$$

Denote $\boldsymbol{x} = (\boldsymbol{x}_1^T, \ldots, \boldsymbol{x}_n^T)^T$ and $\boldsymbol{y} = (\boldsymbol{y}_1^T, \ldots, \boldsymbol{y}_n^T)^T$. The joint density of the complete data $(\boldsymbol{x}, \boldsymbol{y})$ is given by

$$\begin{aligned} f(\boldsymbol{x}, \boldsymbol{y}; \boldsymbol{\theta}) = {} & (2\pi)^{-n(p-1)/2} \det(\boldsymbol{\Sigma})^{-n/2} \\ & \times \prod_{i=1}^{n} \binom{m_i}{\boldsymbol{x}_i} \frac{\exp\left(\sum_{j=1}^{p-1} x_{ij} y_{ij}\right)}{\left(1 + \sum_{j=1}^{p-1} e^{y_{ij}}\right)^{m_i}} \exp\left\{ -\frac{1}{2} (\boldsymbol{y}_i - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{y}_i - \boldsymbol{\mu}) \right\}, \end{aligned} \tag{4}$$

where $\boldsymbol{\theta} = (\boldsymbol{\mu}, \boldsymbol{\Sigma})$. Integrating out the unobserved data $\boldsymbol{y}_i$ and taking the logarithm gives the log-likelihood function for model (1),

$$\begin{aligned} \ell(\boldsymbol{\theta}) = {} & -\frac{n(p-1)}{2} \log(2\pi) - \frac{n}{2} \log\det(\boldsymbol{\Sigma}) \\ & + \sum_{i=1}^{n} \log \int \binom{m_i}{\boldsymbol{x}_i} \frac{\exp\left(\sum_{j=1}^{p-1} x_{ij} y_{ij}\right)}{\left(1 + \sum_{j=1}^{p-1} e^{y_{ij}}\right)^{m_i}} \exp\left\{ -\frac{1}{2} (\boldsymbol{y}_i - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{y}_i - \boldsymbol{\mu}) \right\} d\boldsymbol{y}_i. \end{aligned} \tag{5}$$

Note that inference procedures based on (5) are generally invariant under any permutation ρ of the p taxa. Denote by $\boldsymbol{P}_\rho$ the $p \times p$ permutation matrix and by $\boldsymbol{y}_{i\rho}$ the permuted log-ratio vector. Let $\boldsymbol{F} = (\boldsymbol{I}_{p-1}, -\boldsymbol{1}_{p-1})$ and $\boldsymbol{H} = \boldsymbol{I}_{p-1} + \boldsymbol{1}_{p-1}\boldsymbol{1}_{p-1}^T$, where $\boldsymbol{I}_k$ is the $k \times k$ identity matrix and $\boldsymbol{1}_k$ is the k-vector of ones. By the permutation properties of the log-ratio transformation (Aitchison, 2003, Property 5.2), we have $\boldsymbol{y}_{i\rho} = \boldsymbol{Q}_\rho \boldsymbol{y}_i$, where $\boldsymbol{Q}_\rho = \boldsymbol{F}\boldsymbol{P}_\rho\boldsymbol{F}^T\boldsymbol{H}^{-1}$. It follows that the parameters $\boldsymbol{\theta}_\rho = (\boldsymbol{\mu}_\rho, \boldsymbol{\Sigma}_\rho)$ under ρ are given by $\boldsymbol{\mu}_\rho = \boldsymbol{Q}_\rho \boldsymbol{\mu}$ and

$$\boldsymbol{\Sigma}_\rho = \boldsymbol{Q}_\rho \boldsymbol{\Sigma} \boldsymbol{Q}_\rho^T. \tag{6}$$

With a fully unspecified covariance $\boldsymbol{\Sigma}$, the LNM model (1) allows greater flexibility than the Dirichlet-multinomial model in accounting for the dependence among the taxa. The log-

likelihood (5), however, involves a high-dimensional integral that does not have an analytic form, posing major challenges to likelihood-based inferences.

3. SAEM Algorithm for the LNM Model

In this section we develop a stochastic approximation EM (SAEM) algorithm for maximum likelihood estimation in the LNM model, where the Hamiltonian Monte Carlo (HMC) method is used to speed up the simulation step in high dimensions. In the EM framework, $\boldsymbol{y}$ is treated as missing data. The posterior density of $\boldsymbol{y}$ given the observed data $\boldsymbol{x}$ is

$$g(\boldsymbol{y}|\boldsymbol{x};\boldsymbol{\theta}) \propto \prod_{i=1}^{n} \frac{\exp\left(\sum_{j=1}^{p-1} x_{ij} y_{ij}\right)}{\left(1+\sum_{j=1}^{p-1} e^{y_{ij}}\right)^{m_i}} \exp\left\{-\frac{1}{2}(\boldsymbol{y}_i-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\boldsymbol{y}_i-\boldsymbol{\mu})\right\}. \tag{7}$$

The E-step of the EM algorithm requires the calculation of the conditional expectation of the complete data log-likelihood, given the observed data $\boldsymbol{x}$ and the current parameter estimate $\boldsymbol{\theta}^{(k-1)}$,

$$Q(\boldsymbol{\theta};\boldsymbol{\theta}^{(k-1)}) = E_{\boldsymbol{\theta}^{(k-1)}}\{\log f(\boldsymbol{x},\boldsymbol{y};\boldsymbol{\theta})|\boldsymbol{x}\} = \int \log f(\boldsymbol{x},\boldsymbol{y};\boldsymbol{\theta}) g(\boldsymbol{y}|\boldsymbol{x};\boldsymbol{\theta}^{(k-1)})\, d\boldsymbol{y},$$

which is approximated by a Monte Carlo integration in the MCEM algorithm and, more efficiently, by a stochastic averaging procedure in the SAEM algorithm (Delyon et al., 1999).

Our SAEM algorithm for estimating $\boldsymbol{\theta}$ in model (1), at the kth iteration, consists of the following steps.

(1) *Simulation*: use the HMC method to sample $\boldsymbol{y}_j^{(k)}$, $j = 1,\ldots,N$, from the posterior density $g(\boldsymbol{y}|\boldsymbol{x};\boldsymbol{\theta}^{(k-1)})$.

(2) *Stochastic approximation*: update $Q^{(k)}(\boldsymbol{\theta})$ by

$$Q^{(k)}(\boldsymbol{\theta}) = Q^{(k-1)}(\boldsymbol{\theta}) + \gamma_k \left\{\frac{1}{N}\sum_{j=1}^{N} \log f(\boldsymbol{x},\boldsymbol{y}_j^{(k)};\boldsymbol{\theta}) - Q^{(k-1)}(\boldsymbol{\theta})\right\},$$

where $\{\gamma_k\}$ is a sequence of step sizes.

(3) *Maximization*: find $\boldsymbol{\theta}^{(k)} = (\boldsymbol{\mu}^{(k)}, \boldsymbol{\Sigma}^{(k)})$ that maximizes $Q^{(k)}(\boldsymbol{\theta})$.

In contrast to MCEM, the SAEM algorithm reuses simulated samples from the previous

iterations and, therefore, converges faster than MCEM for the same number of simulations. This improvement is critical for scaling up our procedure, since the simulation step tends to dominate the computational cost in high dimensions. The speedup within a simulation and the implementation of the maximization step will be discussed in the following subsections.

3.1 *HMC Sampling*

MCMC methods are known to be inefficient and converge slowly when the target distribution has isolated modes (Neal, 1996). In our problem, we show that the multimodality issue does not occur. The following proposition can be derived from preservation results for log-concave functions, but we give a direct proof in Web Appendix A.

PROPOSITION 1: The posterior density $g(\boldsymbol{y}|\boldsymbol{x};\boldsymbol{\theta})$ in (7) is log-concave.

The time complexity of standard random walk MCMC algorithms is $O(p^2)$ after the burn-in period (Belloni and Chernozhukov, 2009), which tends to be prohibitive when p is large. The HMC method explores high-dimensional state spaces more efficiently by using Hamiltonian dynamics to generate gradient-informed proposals of new states, which brings the complexity down to $O(p^{5/4})$ (Neal, 2011). One can develop intuition about Hamiltonian dynamics by imagining a satellite that orbits a planet under a gravitational field. To prevent the satellite from crashing into the planet or escaping its gravitational pull, one must endow the satellite with a certain momentum to counteract the gravitational attraction. As the satellite moves toward or away from the planet, its potential energy and kinetic energy change in opposite directions, while the total energy remains conserved. We refer the reader to Betancourt (2017) for an accessible introduction to HMC.

Since $\boldsymbol{y}_i$ can be independently generated, in what follows we assume $n = 1$ and omit the subscript i. The physical intuition is connected to our sampling problem by setting the potential energy $U(\boldsymbol{y}) = -\log g(\boldsymbol{y}|\boldsymbol{x};\boldsymbol{\theta})$. To generate trajectories that explore a region of

the state space neither too near nor too far away from the mode, we artificially introduce a momentum vector $\boldsymbol{p} \in \mathbb{R}^{p-1}$ and hence the kinetic energy $K(\boldsymbol{p}) = \boldsymbol{p}^T \boldsymbol{M}^{-1} \boldsymbol{p}/2$, where $\boldsymbol{M}$ is a mass matrix to be discussed later. The Hamiltonian is then defined as the total energy

$$H(\boldsymbol{y}, \boldsymbol{p}) = U(\boldsymbol{y}) + K(\boldsymbol{p}).$$

Since the total energy is conserved over time t, the partial derivatives lead to the Hamiltonian equations

$$\frac{d\boldsymbol{y}}{dt} = \frac{\partial H}{\partial \boldsymbol{p}} = \boldsymbol{M}^{-1}\boldsymbol{p},$$
$$\frac{d\boldsymbol{p}}{dt} = -\frac{\partial H}{\partial \boldsymbol{y}} = -\nabla U(\boldsymbol{y}).$$

These differential equations have no analytic solutions and, thus, have to be discretized and solved numerically. In particular, with a chosen step size ε, the leapfrog method updates the state and momentum vectors by

$$\boldsymbol{p} \leftarrow \boldsymbol{p} - \frac{\varepsilon}{2}\nabla U(\boldsymbol{y}), \tag{8}$$

$$\boldsymbol{y} \leftarrow \boldsymbol{y} + \varepsilon \boldsymbol{M}^{-1}\boldsymbol{p}, \tag{9}$$

$$\boldsymbol{p} \leftarrow \boldsymbol{p} - \frac{\varepsilon}{2}\nabla U(\boldsymbol{y}). \tag{10}$$

Note that, in practice, (10) and (8) from two consecutive iterations can be combined into a single update $\boldsymbol{p} \leftarrow \boldsymbol{p} - \varepsilon \nabla U(\boldsymbol{y})$. Finally, owing to the approximation error, an accept–reject step is required after T steps of the leapfrog updates to retain the target distribution as the stationary distribution.

Each iteration of the HMC algorithm for sampling from $g(\boldsymbol{y}|\boldsymbol{x};\boldsymbol{\theta})$ is then summarized as follows:

(1) Draw the momentum vector $\boldsymbol{p}$ from $N_{p-1}(\boldsymbol{0}, \boldsymbol{M})$.

(2) Simulate the Hamiltonian dynamics for T steps using the leapfrog updates (8)–(10).

(3) Accept the proposed state $(\boldsymbol{y}^*, \boldsymbol{p}^*)$ with probability $\min[1, \exp\{-H(\boldsymbol{y}^*, \boldsymbol{p}^*) + H(\boldsymbol{y}, \boldsymbol{p})\}]$.

To further improve the performance of HMC sampling, several considerations are in order.

First, if the covariance matrix $\boldsymbol{\Sigma}^*$ of $\boldsymbol{y}$ under the posterior were known, one could linearly transform $\boldsymbol{y}$ to have an identity covariance matrix, which is equivalent to choosing $\boldsymbol{M} = (\boldsymbol{\Sigma}^*)^{-1}$ in the above HMC algorithm (Neal, 2011). In practice, $\boldsymbol{\Sigma}^*$ is unknown and can be substituted by an estimate of the prior covariance matrix $\boldsymbol{\Sigma}$, so that $\boldsymbol{M} = (\boldsymbol{\Sigma}^{(k-1)})^{-1}$ at the kth iteration of the SAEM algorithm. Second, following Shahbaba et al. (2014), we may speed up the HMC algorithm by splitting the Hamiltonian and exploiting partial analytic solutions, as discussed in detail in Web Appendix B. Finally, we randomly choose the leapfrog step size ε and the number of steps T from some relatively small intervals, for example, $[0.055, 0.065]$ and $6, \ldots, 15$, to ensure the ergodicity of HMC (Neal, 2011).

3.2 *Implementation and Convergence of SAEM*

The maximization step of the SAEM algorithm can easily be implemented through a closed-form expression. In fact, since the joint density (4) belongs to an exponential family with natural parameter and sufficient statistic the same as those for $N_{p-1}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, the stochastic approximation step of the SAEM algorithm boils down to the updates

$$\boldsymbol{T}_1^{(k)} = (1-\gamma_k)\boldsymbol{T}_1^{(k-1)} + \gamma_k \frac{1}{N}\sum_{j=1}^{N} \boldsymbol{y}_j^{(k)}, \tag{11}$$

$$\boldsymbol{T}_2^{(k)} = (1-\gamma_k)\boldsymbol{T}_2^{(k-1)} + \gamma_k \frac{1}{N}\sum_{j=1}^{N} \boldsymbol{y}_j^{(k)}(\boldsymbol{y}_j^{(k)})^T. \tag{12}$$

The maximization of $Q^{(k)}(\boldsymbol{\theta})$ can then be carried out explicitly as

$$\begin{aligned} \boldsymbol{\mu}^{(k)} &= \boldsymbol{T}_1^{(k)}, \\ \boldsymbol{\Sigma}^{(k)} &= \boldsymbol{T}_2^{(k)} - \boldsymbol{T}_1^{(k)}(\boldsymbol{T}_1^{(k)})^T. \end{aligned}$$

With the explicit updates derived above, it is now easy to verify the assumptions for the convergence of the SAEM algorithm in Delyon et al. (1999). We summarize the convergence result in the following theorem.

THEOREM 1: *Assume that the step sizes* γ_k *satisfy* $0 \leqslant \gamma_k \leqslant 1$ *for all* k, $\sum_{k=1}^{\infty} \gamma_k = \infty$,

and $\sum_{k=1}^{\infty} \gamma_k^2 < \infty$*, and that* $\boldsymbol{T}_1^{(k)}$ *and* $\boldsymbol{T}_2^{(k)}$ *are almost surely uniformly bounded. Then any sequence* $\{\boldsymbol{\theta}^{(k)}\}$ *generated by the SAEM algorithm converges almost surely to a stationary point of* $\ell(\boldsymbol{\theta})$ *in* (5)*.*

The simulation sample size N and step sizes γ_k are tuning parameters that need to be carefully chosen to ensure fast convergence of the SAEM algorithm. As suggested by Delyon et al. (1999), since the maximization step is straightforward and computationally much cheaper than the simulation step, we set N to a small number, for example, $N = 5$. The choice of γ_k is more subtle. Although the convergence of the SAEM algorithm is guaranteed for a wide range of step sizes, it has been argued that the optimal choice of γ_k should depend on the rate of convergence of the underlying EM algorithm (Jank, 2006). In practice, one can assess the rate of convergence of EM by

$$r = 1 - \lim_{k \to \infty} \frac{\|\boldsymbol{\theta}^{(k+1)} - \boldsymbol{\theta}^{(k)}\|}{\|\boldsymbol{\theta}^{(k)} - \boldsymbol{\theta}^{(k-1)}\|}.$$

Let $\gamma_k = k^{-\alpha}$ for $1/2 < \alpha \leqslant 1$. A numerical relationship between the optimal choice of α and the rate of convergence of EM was given by Jank (2006). For simplicity, we fix $\alpha = 0.65$, which roughly corresponds to a typical rate of $r = 0.2$.

Owing to its stochastic approximation nature, the SAEM algorithm is not sensitive to the initial values, provided that $\boldsymbol{\Sigma}^{(0)}$ is well conditioned. In our implementation, we set $\boldsymbol{\theta}^{(0)} = (\boldsymbol{\mu}^{(0)}, \boldsymbol{\Sigma}^{(0)})$ as follows: replace all zeros by 0.05, obtain log-ratios $\boldsymbol{y}_i^{(0)}$, and set $\boldsymbol{\mu}^{(0)} = \widehat{\boldsymbol{\mu}}^{(0)}$ and $\boldsymbol{\Sigma}^{(0)} = \widehat{\boldsymbol{\Sigma}}^{(0)} + 5\boldsymbol{I}_{p-1}$, where $\widehat{\boldsymbol{\mu}}^{(0)}$ and $\widehat{\boldsymbol{\Sigma}}^{(0)}$ are the sample mean vector and covariance matrix of $\boldsymbol{y}_i^{(0)}$, respectively. Similarly, set $\boldsymbol{T}_1^{(0)} = \boldsymbol{\mu}^{(0)}$ and $\boldsymbol{T}_2^{(0)} = \boldsymbol{\Sigma}^{(0)} + \boldsymbol{\mu}^{(0)}(\boldsymbol{\mu}^{(0)})^T$.

4. Condition Number Regularization

Under the LNM model with parameter $\boldsymbol{\theta}$, one can estimate the taxa compositions by the posterior means

$$E_{\boldsymbol{\theta}}(\boldsymbol{\pi}_i|\boldsymbol{x}_i) = \int \psi(\boldsymbol{y}_i) g(\boldsymbol{y}_i|\boldsymbol{x}_i; \boldsymbol{\theta})\, d\boldsymbol{y}_i.$$

Substituting the maximum likelihood estimate $\widehat{\boldsymbol{\theta}}$ from the SAEM algorithm for $\boldsymbol{\theta}$ leads to the estimates

$$\widehat{\boldsymbol{\pi}}_i = E_{\widehat{\boldsymbol{\theta}}}(\boldsymbol{\pi}_i|\boldsymbol{x}_i) = \int \psi(\boldsymbol{y}_i) g(\boldsymbol{y}_i|\boldsymbol{x}_i;\widehat{\boldsymbol{\theta}})\, d\boldsymbol{y}_i.$$

To evaluate the above integral, a Monte Carlo approximation using HMC sampling as in the SAEM algorithm is required. Since the HMC algorithm involves the inversion of $\boldsymbol{\Sigma}$, an ill-conditioned estimate of $\boldsymbol{\Sigma}$ may result in inferior approximations of $\widehat{\boldsymbol{\pi}}_i$. Similarly, the ill-conditioning problem can cause numerical instability in the iterates of the SAEM algorithm, thus affecting the estimation of $\boldsymbol{\theta}$.

Inspired by Won et al. (2013), we propose to regularize the covariance structure by directly imposing a condition number constraint. For a symmetric and positive definite matrix $\boldsymbol{A}$, the L_2-norm condition number of $\boldsymbol{A}$ is defined by

$$\text{cond}(\boldsymbol{A}) = \|\boldsymbol{A}\|_2 \|\boldsymbol{A}^{-1}\|_2 = \lambda_{\max}(\boldsymbol{A})/\lambda_{\min}(\boldsymbol{A}),$$

where $\lambda_{\max}(\boldsymbol{A})$ and $\lambda_{\min}(\boldsymbol{A})$ are the largest and smallest eigenvalues of $\boldsymbol{A}$, respectively. Special care is needed to maintain the permutation invariance of the regularized estimation procedure. With a fully unspecified covariance $\boldsymbol{\Sigma}$, our estimation procedure is permutation invariant owing to the invariance property of $\det(\boldsymbol{\Sigma})$ and $(\boldsymbol{y}_i - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{y}_i - \boldsymbol{\mu})$ (Aitchison, 2003, Property 5.3). Regularizing the condition number of $\boldsymbol{\Sigma}$, however, does not preserve this property. Consider, for example, the case of $\boldsymbol{\Sigma} = \boldsymbol{I}_2$ with $\text{cond}(\boldsymbol{\Sigma}) = 1$, and the permutation $\rho(x_1, x_2, x_3) = (x_2, x_3, x_1)$. From (6) we have

$$\boldsymbol{\Sigma}_\rho = \boldsymbol{Q}_\rho \boldsymbol{I}_2 \boldsymbol{Q}_\rho^T = \begin{pmatrix} 2 & 1 \\ 1 & 1 \end{pmatrix}$$

with $\text{cond}(\boldsymbol{\Sigma}_\rho) = (7 + 3\sqrt{5})/2 \neq 1$. To find a permutation invariant condition number, we note the following fact. The proof is provided in Web Appendix C.

PROPOSITION 2: For any square root $\boldsymbol{K}$ of the matrix $\boldsymbol{H}$ and any permutation ρ of the

p taxa,

$$\mathrm{cond}(\boldsymbol{K}^{-1}\boldsymbol{\Sigma}_\rho\boldsymbol{K}^{-1}) = \mathrm{cond}(\boldsymbol{K}^{-1}\boldsymbol{\Sigma}\boldsymbol{K}^{-1}).$$

Besides its permutation invariance, the relevance of the condition number mentioned in Proposition 2 to our estimation problem can be appreciated by finding a reparametrization that involves the matrix $\boldsymbol{K}^{-1}\boldsymbol{\Sigma}\boldsymbol{K}^{-1}$. Denote by

$$\boldsymbol{K} = \boldsymbol{H}^{1/2} = \boldsymbol{I}_{p-1} + \frac{\sqrt{p}-1}{p-1}\mathbf{1}_{p-1}\mathbf{1}_{p-1}^T$$

the principal square root of $\boldsymbol{H}$. Write $\boldsymbol{\eta} = (\boldsymbol{\nu}, \boldsymbol{D})$ and $\boldsymbol{z} = (\boldsymbol{z}_1^T, \ldots, \boldsymbol{z}_n^T)^T$ with $\boldsymbol{\nu} = \boldsymbol{H}^{-1/2}\boldsymbol{\mu}$, $\boldsymbol{D} = \boldsymbol{H}^{-1/2}\boldsymbol{\Sigma}\boldsymbol{H}^{-1/2}$, and $\boldsymbol{z}_i = \boldsymbol{H}^{-1/2}\boldsymbol{y}_i$. Then the log-likelihood (5) becomes

$$\begin{aligned}\widetilde{\ell}(\boldsymbol{\eta}) = &-\frac{n(p-1)}{2}\log(2\pi) - \frac{n}{2}\log\det(\boldsymbol{D}) + \sum_{i=1}^n \log\int \binom{m_i}{\boldsymbol{x}_i} \\ &\times \frac{\exp\big(\sum_{j=1}^{p-1} x_{ij}z_{ij} + (\sqrt{p}-1)(m_i - x_{ip})\overline{\boldsymbol{z}}_i\big)}{\big(1+\sum_{j=1}^{p-1} e^{z_{ij}+(\sqrt{p}-1)\overline{\boldsymbol{z}}_i}\big)^{m_i}} \exp\left\{-\frac{1}{2}(\boldsymbol{z}_i-\boldsymbol{\nu})^T\boldsymbol{D}^{-1}(\boldsymbol{z}_i-\boldsymbol{\nu})\right\} d\boldsymbol{z}_i,\end{aligned}$$

where $\overline{\boldsymbol{z}}_i = \sum_{j=1}^{p-1} z_{ij}/(p-1)$. The reparametrized joint density $\widetilde{f}(\boldsymbol{x}, \boldsymbol{z}; \boldsymbol{\eta})$ and posterior density $\widetilde{g}(\boldsymbol{z}|\boldsymbol{x}; \boldsymbol{\eta})$ can similarly be derived. Therefore, regularizing the condition number of $\boldsymbol{D}$ is indeed relevant. We then consider the condition number regularized problem

$$\begin{aligned}\text{minimize} \quad & \widetilde{\ell}(\boldsymbol{\eta}) \\ \text{subject to} \quad & \mathrm{cond}(\boldsymbol{D}) \leqslant \kappa,\end{aligned} \tag{13}$$

where $\kappa > 0$ is a tuning parameter. We describe an efficient algorithm for solving problem (13) in the following subsection, and discuss a data-driven choice of κ by K-fold cross-validation in Web Appendix E.

4.1 *Regularized SAEM Algorithm*

The SAEM algorithm described in Section 3 can be extended to solve problem (13) by adding the condition number constraint to the maximization step. Let $\widetilde{\boldsymbol{T}}_1^{(k)}$ and $\widetilde{\boldsymbol{T}}_2^{(k)}$ be the sufficient statistics obtained from updates similar to (11) and (12) with $\boldsymbol{y}_j^{(k)}$ replaced by $\boldsymbol{z}_j^{(k)}$, and let

$$\widetilde{\boldsymbol{\Sigma}}^{(k)} = \widetilde{\boldsymbol{T}}_2^{(k)} - \widetilde{\boldsymbol{T}}_1^{(k)}(\widetilde{\boldsymbol{T}}_1^{(k)})^T.$$

In view of the discussion in Section 3.2, the regularized maximization step reduces to the optimization problem

$$\begin{aligned} \text{minimize} \quad & \log\det(\boldsymbol{D}) + \text{tr}(\boldsymbol{D}^{-1}\widetilde{\boldsymbol{\Sigma}}^{(k)}) \\ \text{subject to} \quad & \text{cond}(\boldsymbol{D}) \leqslant \kappa. \end{aligned} \tag{14}$$

Denote by $\lambda_1 \geqslant \cdots \geqslant \lambda_{p-1} \geqslant 0$ the eigenvalues of $\widetilde{\boldsymbol{\Sigma}}^{(k)}$, and $\widetilde{\boldsymbol{\Sigma}}^{(k)} = \boldsymbol{Q}\boldsymbol{\Lambda}\boldsymbol{Q}^T$ the spectral decomposition of $\widetilde{\boldsymbol{\Sigma}}^{(k)}$, where $\boldsymbol{\Lambda} = \text{diag}(\lambda_1, \ldots, \lambda_{p-1})$ and $\boldsymbol{Q}$ is an orthogonal matrix. As shown by Lemma 1 and Theorem 1 of Won et al. (2013), the solution to the optimization problem (14) is given by $\boldsymbol{D}^{(k)} = \boldsymbol{Q}\boldsymbol{S}\boldsymbol{Q}^T$ if $\kappa \leqslant \text{cond}(\widetilde{\boldsymbol{\Sigma}}^{(k)})$ and $\boldsymbol{D}^{(k)} = \widetilde{\boldsymbol{\Sigma}}^{(k)}$ otherwise, where $\boldsymbol{S} = \text{diag}(s_1, \ldots, s_{p-1})$ with $s_j = \min\{\max(\lambda_j, \tau^*), \kappa\tau^*\}$; τ^* is the unique solution to

$$\tau = \frac{\sum_{j=1}^{b(\tau)} \lambda_j/\kappa + \sum_{j=a(\tau)}^{p-1} \lambda_j}{b(\tau) + p - a(\tau)},$$

where $a(\tau) = \min\{1 \leqslant j \leqslant p-1 : \lambda_j < \tau\}$ and $b(\tau) = \max\{1 \leqslant j \leqslant p-1 : \lambda_j > \kappa\tau\}$. An algorithm of time complexity $O(p)$ for finding $1/(\kappa\tau^*)$ was outlined in Won et al. (2013); we present a slight variant of the algorithm for finding τ^* directly in Web Appendix D.

5. Simulation Studies

In this section we present simulation studies to examine the finite sample performance of the proposed SAEM and regularized SAEM algorithms for the LNM model, referred to as the LNM and LNM+ methods respectively. We compare our methods with the following commonly used procedures:

- Mult: the multinomial model with subject-specific parameters $\boldsymbol{\pi}_i$;
- DM: the Dirichlet-multinomial model, where $\widehat{\boldsymbol{\pi}}_i$ are obtained as the posterior means;
- LN1: the logistic normal model with $\widehat{\boldsymbol{\pi}}_i$ obtained from replacing all zeros by 0.5 and normalization;
- LN2: the logistic normal model with $\widehat{\boldsymbol{\pi}}_i$ obtained from adding a pseudocount of 1 to all counts and normalization;

- MCEM: the MCEM algorithm of Xia et al. (2013) for the LNM model.

The zero replacement procedure in LN1 is similar to those suggested by Aitchison (2003, Section 11.5) and has been adopted by, for example, Lin et al. (2014) and Cao, Lin, and Li (2018). The pseudocount approach in LN2, also known as add-one or Laplace smoothing, is in widespread use in text analysis and metagenomics; see, for example, Manning, Raghavan, and Schütze (2008) and Friedman and Alm (2012).

The simulated data were generated as follows. We first generated $\boldsymbol{w}_i = (w_{i1}, \ldots, w_{ip})^T$ from the multivariate normal distribution $N_p(\boldsymbol{\xi}, \boldsymbol{\Omega})$ and obtained the taxa compositions $\boldsymbol{\pi}_i$ through the transformation

$$\pi_{ij} = \frac{e^{w_{ij}}}{\sum_{k=1}^{p} e^{w_{ik}}}, \qquad j = 1, \ldots, p.$$

Consequently, $\boldsymbol{y}_i = \phi(\boldsymbol{\pi}_i)$ follows the multivariate normal distribution $N_{p-1}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, where $\boldsymbol{\mu} = \boldsymbol{F}\boldsymbol{\xi}$ and $\boldsymbol{\Sigma} = \boldsymbol{F}\boldsymbol{\Omega}\boldsymbol{F}^T$ (Aitchison, 2003, Property 6.1). Here $\boldsymbol{\xi}$ was sampled uniformly from the interval $[0, 10]$ and $\boldsymbol{\Omega} = (0.5^{|i-j|})$. We then generated the count data from the multinomial distribution $\text{Mult}(m_i, \boldsymbol{\pi}_i)$, where m_i were sampled uniformly from $20p, \ldots, 20p + 1000$. We considered the dimensions $p = 15$, 50, 100, and 200 and sample size $n = 100$. These settings were intended to mimic the high-dimensional and heterogeneous nature of microbiome data, and resulted in proportions of zeros 33.1%, 40.0%, 42.9%, and 43.9% for $p = 15$, 50, 100, and 200, respectively. Since the MCEM algorithm would take a prohibitive amount of time to converge even for moderately high-dimensional settings, it was included in our comparisons only for $p = 15$.

In the SAEM algorithm, we set the simulation sample size $N = 5$ and step sizes $\gamma_k = k^{-\alpha}$ with $\alpha = 0.65$. In HMC sampling, we randomly chose the leapfrog step size ε from $[0.055, 0.065]$ and the number of steps T from $6, \ldots, 15$. The regularization parameter κ was selected by fivefold cross-validation. We repeated the simulation 100 times for each setting. In addition to the estimates of $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$, we obtained the estimates of taxa compositions

$\widehat{\boldsymbol{\pi}} = (\widehat{\boldsymbol{\pi}}_1^T, \ldots, \widehat{\boldsymbol{\pi}}_n^T)^T$ based on a Monte Carlo sample of size 1000. In the MCEM algorithm, we used 1000 Metropolis–Hastings samples in the E-step after a burn-in period of 1000.

The simulation results are summarized in Table 1, which reports the relative estimation errors of $\widehat{\boldsymbol{\mu}}$, $\widehat{\boldsymbol{\Sigma}}$, and $\widehat{\boldsymbol{\pi}}$ under various norms. Here, for any vector $\boldsymbol{v}$ and matrix $\boldsymbol{A}$, $\|\boldsymbol{v}\|_1$ and $\|\boldsymbol{v}\|_2$ denote the L_1- and L_2-norms of $\boldsymbol{v}$, and $\|\boldsymbol{A}\|_2$ and $\|\boldsymbol{A}\|_F$ the spectral and Frobenius norms of $\boldsymbol{A}$, respectively. Despite using an ad hoc procedure to eliminate zeros, the LN1 and LN2 methods are closely related to the LNM method in that they aim to estimate the logistic normal parameters $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$. From Table 1 we see that the LNM method performs much better than the LN1 and LN2 methods in terms of estimating $\boldsymbol{\mu}$, while having a close performance to LN1 and LN2 in terms of estimating $\boldsymbol{\Sigma}$ due to the unconstrained condition number. Figure 1 indicates that the condition number in the LNM method grows exponentially as the dimensionality increases. Owing to the condition number constraint, the LNM+ method further improves on the performance of LNM for estimating $\boldsymbol{\mu}$, while reducing the estimation error for $\boldsymbol{\Sigma}$ substantially.

[Table 1 about here.]

[Figure 1 about here.]

Since the multinomial and DM methods do not model the logistic normal covariance structure, performance comparisons can only be based on the estimation accuracy of $\boldsymbol{\pi}$. From Table 1 we note first that the LN2 method has the worst performance among all methods except MCEM, because the Dirichlet prior imposed by add-one smoothing is incompatible with the logistic normal model. The DM method performs better than the two ad hoc procedures LN1 and LN2, and slightly improves on the multinomial method in terms of the L_1-norm. The LNM and LNM+ methods further improve on the DM estimation as a result of the more flexible correlation structure, and the LNM+ method achieves a greater performance boost in higher dimensions.

Although both MCEM and the proposed LNM method conduct maximum likelihood estimation, the numerical performance of LNM is substantially superior to that of MCEM even in the low-dimensional setting of $p = 15$. While MCEM still greatly outperforms the LN1 and LN2 methods in terms of estimating $\boldsymbol{\mu}$, it has the worst performance in estimating $\boldsymbol{\Sigma}$ and $\boldsymbol{\pi}$ among all methods. This is largely due to the inefficiency of the random walk Metropolis algorithm in exploring the state space. To examine the computational efficiency of MCEM and our algorithms, we report the average numbers of iterations and run times for each setting in Table 2. All timings were carried out on an Intel Xeon 2.6 GHz processor. We see that the LNM and LNM+ methods are about two orders of magnitude faster than MCEM. The condition number constraint incurs only a small extra cost per iteration, although it may require more iterations to converge in high dimensions. In the most challenging setting, the computation can be done in about one hour, indicating that our algorithms scale reasonably well to hundreds of taxa.

[Table 2 about here.]

We further investigate the performance of different methods with varying zero proportions. The simulation results are presented in Web Appendix F.

6. Application to Microbiome Data

We now illustrate the proposed methods by application to a human gut microbiome dataset from Wu et al. (2011). DNA from stool samples of 98 healthy subjects were analyzed by 454/Roche pyrosequencing of 16S rRNA gene segments. Demographic data on the subjects, including body mass index (BMI), were also collected. Taxonomic assignment gave rise to read counts for 3068 operational taxonomic units, which were combined into 87 genera that appeared in at least one sample. The total read counts vary widely across samples, ranging from 1242 to 14,616, resulting in highly sparse data with a zero proportion of 72.2%. Previous

studies normalized the data into proportions after zero counts were replaced by 0.5 (e.g., Lin et al., 2014; Cao et al., 2018). In view of the heterogeneity and sparsity of the data, we are interested in whether the composition estimates can be further improved by our methods.

We applied the proposed and competing methods to estimate the microbiome compositions at the genus level. Heatmaps of the estimated compositions are displayed in Figure 2. We observe clear patterns of light vertical stripes in the heatmaps obtained by the LN1, LN2, and DM methods. This is because, in both LN1 and LN2 methods, the estimated proportions corresponding to the zero counts depend only on the total read counts, and all unobserved genera on the same subject are treated equally. The DM method distinguishes the zero counts from each other only slightly better. By contrast, in the heatmaps generated by the LNM and LNM+ methods, the stripe patterns are not obvious and the zeros tend to shrink toward the population means. This shrinkage effect indicates that our methods are more able to detect fine differences among the rare genera.

[Figure 2 about here.]

Previous studies have revealed that obesity is associated with reduced bacterial diversity and composition changes in the gut microbiome (Turnbaugh et al., 2009). Intuitively, it is expected that more accurate recovery of information for the rare genera is likely to strengthen the contrast between lean and obese individuals. To verify this assumption, we divided the subjects into a lean group of 63 subjects with BMI < 25 and an obese group of 35 subjects with BMI $\geqslant 25$. For each group, we obtained the estimated compositions using different methods, and calculated the following diversity measures:

- Shannon's index $H(\boldsymbol{\pi}_i) = -\sum_{j=1}^{p} \pi_{ij} \log \pi_{ij}$, and
- Simpson's index $D(\boldsymbol{\pi}_i) = \sum_{j=1}^{p} \pi_{ij}^2$.

A larger Shannon's index or a smaller Simpson's index indicates a more diverse ecological community (Morris et al., 2014). Boxplots of the estimated diversity indices for the lean

and obese groups are shown in Figure 3. While the differences in bacterial diversity between groups are apparent for all methods, the LNM+ method helps to uncover a stronger contrast between groups and an increased skewness within each group. Also, we carried out formal statistical tests to show that a stronger evidence for composition changes can be gained; see Web Appendix G. These results are consistent with previous findings and suggest that our methods can yield more accurate estimation by borrowing strength across subjects.

[Figure 3 about here.]

7. Discussion

Adjusting for zero observations in multinomial data has been a long-standing problem in metagenomic data analysis. The approach presented here can be viewed as an empirical Bayes method using a logistic normal prior, which has been advocated as much more flexible than the Dirichlet prior in a wide range of applications. To address the computational challenge in high dimensions, we have developed a fast SAEM algorithm with HMC sampling, which scales up to at least hundreds of taxa, for the LNM model. This will allow the LNM model to be more widely adopted as a useful alternative to the DM model in order to take advantage of its general correlation structure. Moreover, condition number regularization helps to alleviate the impact of dimensionality and is particularly useful when no structural assumptions on the covariance matrix are desired.

It would be possible to extend our methods to multinomial data with higher dimensionality and increased sparsity as arising in shotgun metagenomics. Although increased sparsity does not seem to be a major issue, higher dimensionality places a greater demand on computing resources, and further speedup by GPU computing or data subsampling is recommended. Also, if the sample size is orders of magnitude smaller than the dimensionality, it would constitute a major statistical challenge. In this case, condition number regularization alone

may not be sufficient, and it would be desirable to exploit the sparsity structure in the covariance or precision matrix by incorporating an L_1 penalty (Guo and Zhang, 2017). We leave these important directions for future research.

8. Supplementary Materials

Web Appendices A and B referenced in Section 3, Web Appendices C–E referenced in Section 4, Web Appendix F referenced in Section 5, Web Appendix G referenced in Section 6, and R code implementing the proposed methods are available with this article at the *Biometrics* website on Wiley Online Library.

Acknowledgements

This work was supported in part by NSFC grants 11671018 and 71532001 and National Key R&D Program of China grant 2016YFC0207703.

References

Agresti, A. (2013). *Categorical Data Analysis*. Wiley, Hoboken, NJ, 3rd edition.

Aitchison, J. (2003). *The Statistical Analysis of Compositional Data*. Blackburn Press, Caldwell, NJ.

Belloni, A. and Chernozhukov, V. (2009). On the computational complexity of MCMC-based estimators in large samples. *The Annals of Statistics* **37**, 2011–2055.

Betancourt, M. (2017). A conceptual introduction to Hamiltonian Monte Carlo. https://arxiv.org/abs/1701.02434.

Billheimer, D., Guttorp, P., and Fagan, W. F. (2001). Statistical interpretation of species composition. *Journal of the American Statistical Association* **96**, 1205–1214.

Blei, D. M. and Lafferty, J. D. (2007). A correlated topic model of *Science*. *The Annals of Applied Statistics* **1**, 17–35.

Braun, M. and McAuliffe, J. (2010). Variational inference for large-scale models of discrete choice. *Journal of the American Statistical Association* **105**, 324–335.

Cao, Y., Lin, W., and Li, H. (2018). Two-sample tests of high-dimensional means for compositional data. *Biometrika* **105**, 115–132.

Chen, J. and Li, H. (2013). Variable selection for sparse Dirichlet-multinomial regression with an application to microbiome data analysis. *The Annals of Applied Statistics* **7**, 418–442.

Delyon, B., Lavielle, M., and Moulines, E. (1999). Convergence of a stochastic approximation version of the EM algorithm. *The Annals of Statistics* **27**, 94–128.

Duane, S., Kennedy, A. D., Pendleton, B. J., and Roweth, D. (1987). Hybrid Monte Carlo. *Physics Letters B* **195**, 216–222.

Friedman, J. and Alm, E. J. (2012). Inferring correlation networks from genomic survey data. *PLoS Computational Biology* **8**, e1002687.

Guo, X. and Zhang, C. (2017). The effect of l_1 penalization on condition number constrained estimation of precision matrix. *Statistica Sinica* **27**, 1299–1317.

Hartzel, J., Agresti, A., and Caffo, B. (2001). Multinomial logit random effects models. *Statistical Modelling* **1**, 81–102.

Hedeker, D. (2003). A mixed-effects multinomial logistic regression model. *Statistics in Medicine* **22**, 1433–1446.

Jank, W. (2006). Implementing and diagnosing the stochastic approximation EM algorithm. *Journal of Computational and Graphical Statistics* **15**, 803–829.

Li, H. (2015). Microbiome, metagenomics, and high-dimensional compositional data analysis. *Annual Review of Statistics and Its Application* **2**, 73–94.

Lin, W., Shi, P., Feng, R., and Li, H. (2014). Variable selection in regression with compositional covariates. *Biometrika* **101**, 785–797.

Manning, C. D., Raghavan, P., and Schütze, H. (2008). *Introduction to Information Retrieval*. Cambridge University Press, New York.

Morris, E. K., Caruso, T., Buscot, F., Fischer, M., Hancock, C., Maier, T. S., et al. (2014). Choosing and using diversity indices: Insights for ecological applications from the German Biodiversity Exploratories. *Ecology and Evolution* **4**, 3514–3524.

Mosimann, J. E. (1962). On the compound multinomial distribution, the multivariate β-distribution, and correlations among proportions. *Biometrika* **49**, 65–82.

Neal, R. M. (1996). Sampling from multimodal distributions using tempered transitions. *Statistics and Computing* **6**, 353–366.

Neal, R. M. (2011). MCMC using Hamiltonian dynamics. In *Handbook of Markov Chain Monte Carlo*, pages 113–162. CRC Press, Boca Raton, FL.

Shahbaba, B., Lan, S., Johnson, W. O., and Neal, R. M. (2014). Split Hamiltonian Monte Carlo. *Statistics and Computing* **24**, 339–349.

Turnbaugh, P. J., Hamady, M., Yatsunenko, T., Cantarel, B. L., Duncan, A., Ley, R. E., et al. (2009). A core gut microbiome in obese and lean twins. *Nature* **457**, 480–484.

Won, J.-H., Lim, J., Kim, S.-J., and Rajaratnam, B. (2013). Condition-number-regularized covariance estimation. *Journal of the Royal Statistical Society, Series B* **75**, 427–450.

Wu, G. D., Chen, J., Hoffmann, C., Bittinger, K., Chen, Y.-Y., Keilbaugh, S. A., et al. (2011). Linking long-term dietary patterns with gut microbial enterotypes. *Science* **334**, 105–108.

Xia, F., Chen, J., Fung, W. K., and Li, H. (2013). A logistic normal multinomial regression model for microbiome compositional data analysis. *Biometrics* **69**, 1053–1063.

Received: date / Accepted: date

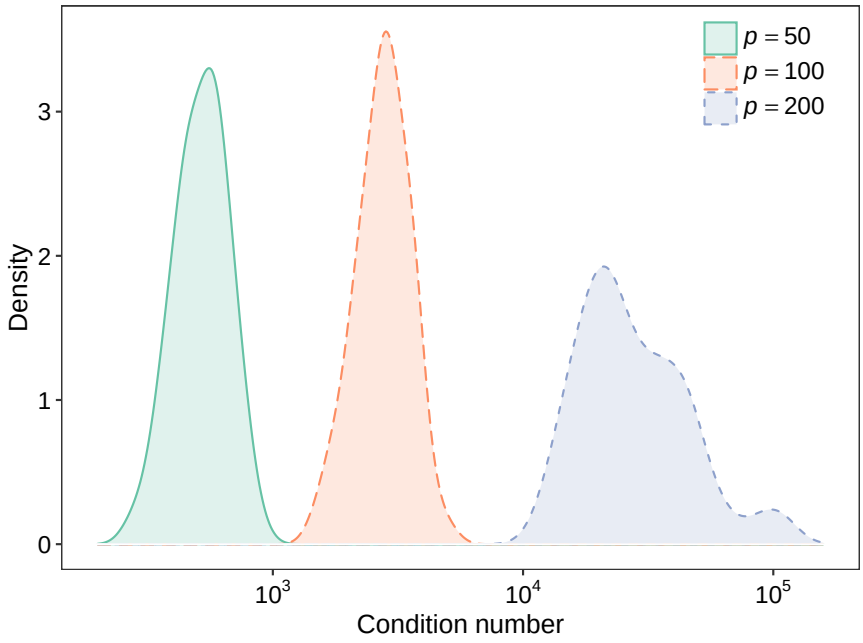

Figure 1. Density plots of condition numbers for the LNM method based on 100 simulations.

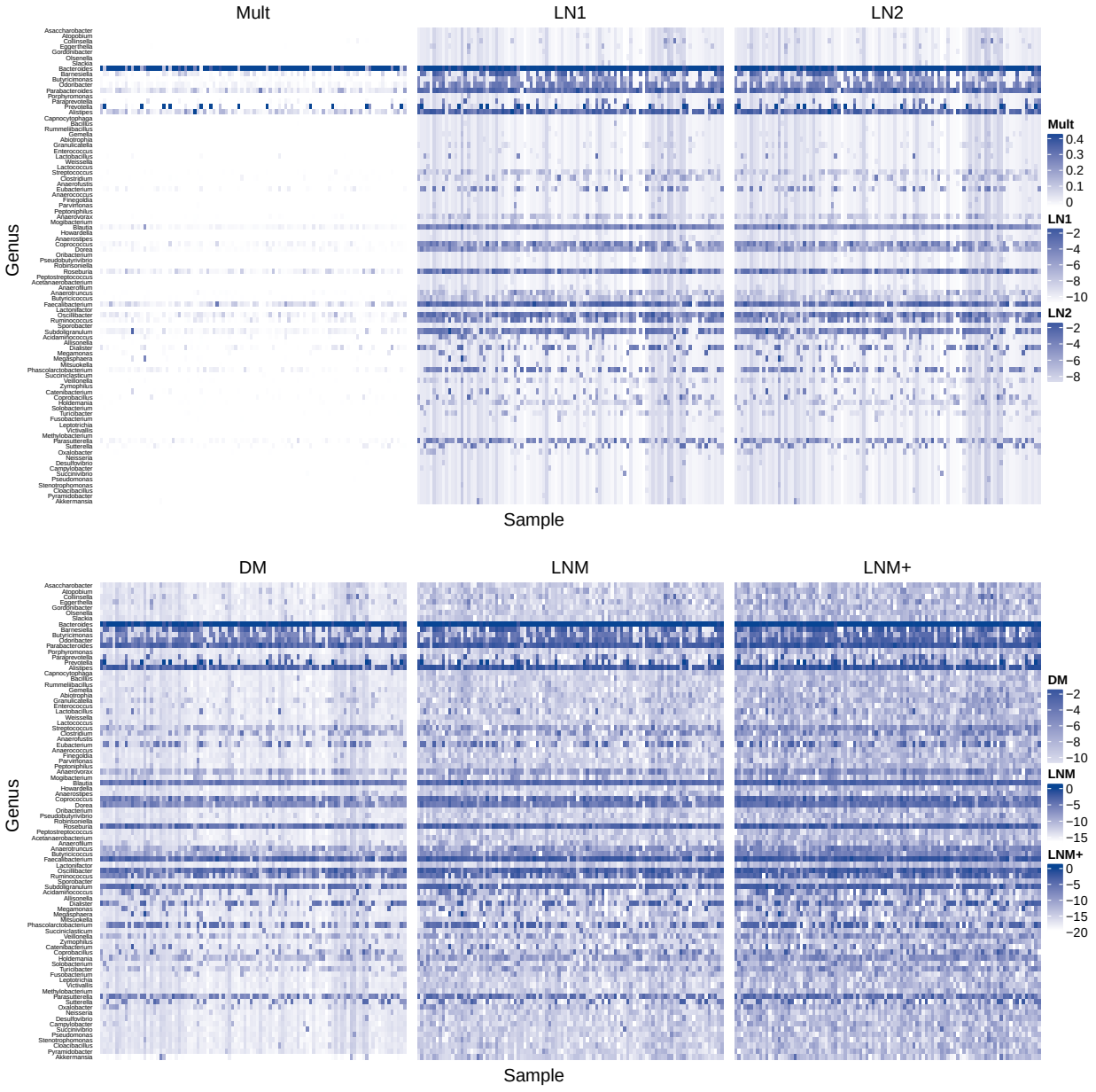

Figure 2. Heatmaps of estimated compositions for the gut microbiome data. Values for all methods except the multinomial method have been log-transformed.

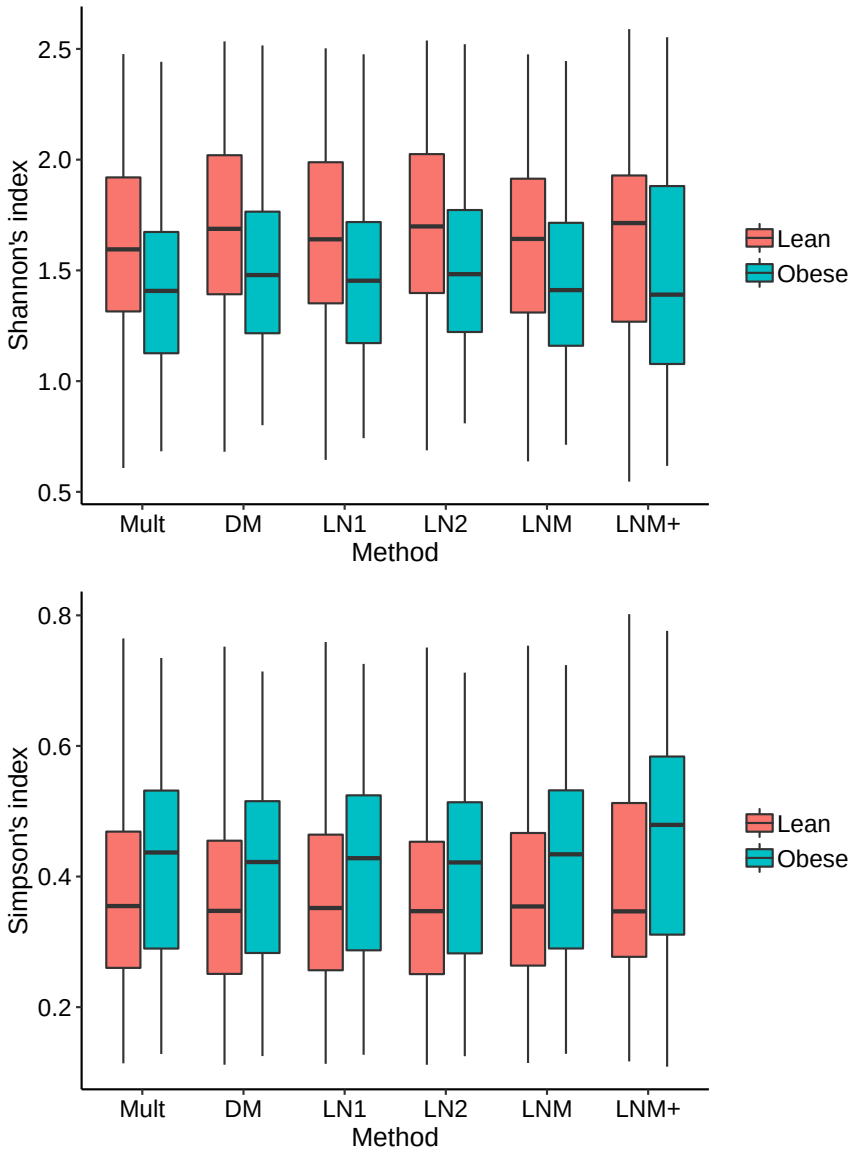

Figure 3. Boxplots of estimated diversity indices using different methods for the lean and obese groups in the gut microbiome data.

Table 1

Means and standard errors (in parentheses) of the relative errors of parameter estimates for various methods with $n = 100$ and varying dimensions based on 100 simulations. All values have been multiplied by 100.

p	Method	$\|\widehat{\boldsymbol{\mu}} - \boldsymbol{\mu}\|_1/\|\boldsymbol{\mu}\|_1$	$\|\widehat{\boldsymbol{\mu}} - \boldsymbol{\mu}\|_2/\|\boldsymbol{\mu}\|_2$	$\|\widehat{\boldsymbol{\Sigma}} - \boldsymbol{\Sigma}\|_2/\|\boldsymbol{\Sigma}\|_2$	$\|\widehat{\boldsymbol{\Sigma}} - \boldsymbol{\Sigma}\|_F/\|\boldsymbol{\Sigma}\|_F$	$\|\widehat{\boldsymbol{\pi}} - \boldsymbol{\pi}\|_1/\|\boldsymbol{\pi}\|_1$	$\|\widehat{\boldsymbol{\pi}} - \boldsymbol{\pi}\|_2/\|\boldsymbol{\pi}\|_2$
15	Mult	—	—	—	—	5.98 (0.09)	4.90 (0.09)
	DM	—	—	—	—	5.97 (0.08)	4.93 (0.09)
	LN1	24.6 (0.8)	30.6 (0.8)	35.7 (2.1)	39.3 (1.9)	6.08 (0.08)	4.90 (0.09)
	LN2	33.4 (0.8)	38.6 (0.8)	38.3 (2.4)	41.6 (2.1)	6.84 (0.06)	5.23 (0.08)
	MCEM	10.3 (0.7)	10.6 (0.6)	68.4 (6.1)	72.6 (5.5)	7.08 (0.10)	5.95 (0.10)
	LNM	5.8 (0.3)	6.4 (0.3)	26.2 (0.9)	33.7 (0.8)	5.74 (0.09)	4.84 (0.09)
	LNM+	5.2 (0.3)	5.7 (0.3)	22.8 (0.8)	26.4 (0.7)	5.72 (0.09)	4.83 (0.09)
50	Mult	—	—	—	—	8.07 (0.05)	5.79 (0.06)
	DM	—	—	—	—	8.04 (0.05)	5.88 (0.06)
	LN1	30.5 (0.8)	36.9 (0.7)	36.0 (2.5)	39.2 (2.3)	8.29 (0.05)	5.79 (0.06)
	LN2	39.6 (0.8)	45.3 (0.7)	33.7 (2.8)	37.2 (2.6)	9.56 (0.04)	6.44 (0.05)
	LNM	7.5 (0.2)	8.6 (0.2)	32.8 (2.3)	41.1 (2.1)	7.79 (0.06)	5.75 (0.06)
	LNM+	6.2 (0.2)	7.3 (0.2)	19.6 (0.6)	24.2 (0.5)	7.70 (0.05)	5.73 (0.06)
100	Mult	—	—	—	—	8.77 (0.04)	5.96 (0.05)
	DM	—	—	—	—	8.73 (0.04)	6.08 (0.05)
	LN1	36.6 (1.0)	41.3 (0.7)	45.6 (3.2)	47.7 (3.0)	9.07 (0.04)	5.98 (0.05)
	LN2	45.5 (0.9)	49.1 (0.7)	47.0 (3.5)	49.0 (3.3)	10.63 (0.03)	6.83 (0.04)
	LNM	11.3 (0.4)	12.8 (0.3)	41.7 (3.1)	48.6 (2.8)	8.55 (0.05)	5.95 (0.05)
	LNM+	8.6 (0.3)	10.1 (0.3)	22.4 (0.9)	25.0 (0.8)	8.36 (0.04)	5.91 (0.05)
200	Mult	—	—	—	—	9.24 (0.03)	6.03 (0.04)
	DM	—	—	—	—	9.21 (0.03)	6.20 (0.04)
	LN1	38.4 (1.1)	43.7 (0.9)	45.8 (3.2)	47.5 (3.1)	9.58 (0.03)	6.06 (0.04)
	LN2	47.0 (1.1)	51.3 (0.8)	46.0 (3.5)	47.5 (3.3)	11.34 (0.02)	7.08 (0.03)
	LNM	15.4 (0.7)	17.2 (0.6)	48.6 (3.5)	53.6 (3.2)	9.14 (0.03)	6.06 (0.04)
	LNM+	9.8 (0.5)	11.7 (0.4)	29.9 (1.1)	30.9 (1.1)	8.81 (0.03)	5.98 (0.04)

Table 2
Average numbers of iterations and run times (in seconds) for three methods with $n = 100$ and varying dimensions based on 100 simulations

p	Method	Iterations	Total time	Time per iter.
15	MCEM	104	2924.1	28.222
	LNM	1104	39.6	0.036
	LNM+	1017	53.1	0.052
50	LNM	1462	151.8	0.106
	LNM+	1778	249.3	0.141
100	LNM	1416	399.9	0.289
	LNM+	2298	795.3	0.349
200	LNM	1019	993.2	0.996
	LNM+	3188	4013.7	1.269