Journal of Machine Learning Research 23 (2022) 1-42 Submitted 10/19; Revised 6/21; Published 2/22

Bayesian Multinomial Logistic Normal Models through Marginally Latent Matrix-T Processes

Justin D. Silverman JUSTINSILVERMAN@PSU.EDU
College of Information Science and Technology, Department of Statistics, and Institute for Computational and Data Science
Penn State University
University Park, PA, 16802, USA

Kimberly Roche KIMBERLY.ROCHE@DUKE.EDU
Program in Computational Biology and Bioinformatics
Duke University
Durham, NC, 27708, USA

Zachary C. Holmes ZACHARY.HOLMES@DUKE.EDU
Department of Molecular Genetics and Microbiology
Duke University
Durham, NC, 27708, USA

Lawrence A. David LAWRENCE.DAVID@DUKE.EDU
Department of Molecular Genetics and Microbiology and Center for Genomic and Computational Biology
Duke University
Durham, NC, 27708, USA

Sayan Mukherjee SAYAN@STAT.DUKE.EDU
Departments of Statistical Science, Mathematics, Computer Science, Biostatistics & Bioinformatics
Duke University
Durham, NC, 27708, USA

Editor: Erik Sudderth

Abstract

Bayesian multinomial logistic-normal (MLN) models are popular for the analysis of sequence count data (*e.g.*, microbiome or gene expression data) due to their ability to model multivariate count data with complex covariance structure. However, existing implementations of MLN models are limited to small datasets due to the non-conjugacy of the multinomial and logistic-normal distributions. Motivated by the need to develop efficient inference for Bayesian MLN models, we develop two key ideas. First, we develop the class of Marginally Latent Matrix-T Process (Marginally LTP) models. We demonstrate that many popular MLN models, including those with latent linear, non-linear, and dynamic linear structure are special cases of this class. Second, we develop an efficient inference scheme for Marginally LTP models with specific accelerations for the MLN subclass. Through application to MLN models, we demonstrate that our inference scheme are both highly accurate and often 4-5 orders of magnitude faster than MCMC.

Keywords: Bayesian Statistics, Multivariate Analysis, Count Data, Microbiome, Gene Expression

1. Introduction

Motivated by the growing need for efficient inference for a wide class of multinomial logistic-normal (MLN) models, in this article we develop two key ideas. First, we introduce the class of Marginally Latent Matrix-T Process (Marginally LTP) models. As the name suggests, Marginally LTP models are defined by a shared canonical marginal form which is a multivariate generalization of Student-t processes (Shah et al., 2014) and allow for non-Gaussian likelihoods. We show that this class is extremely flexible, encompassing many useful models including generalized linear models, generalized Gaussian process models, and generalized dynamic linear models. Second, we develop a general inference scheme for Marginally LTP models (which we term the collapse-uncollapse sampler) with specific accelerations (namely a marginal Laplace approximation) for the subclass of MLN models. Through both simulations and analyses of real datasets using MLN models, we show that our inference schemes are both highly accurate and often 4-5 orders of magnitude faster than MCMC.

MLN models are used for the analysis of compositions measured through multivariate counting. In contrast to multinomial Dirichlet models, MLN models permit both positive and negative covariation between multinomial categories (Aitchison and Shen, 1980). While multinomial logistic-normal topic models have been used in natural language processing for some time (Blei and Lafferty, 2006; Glynn et al., 2019), more recently these models have been adopted for regression and time-series modeling of microbiome data (Grantham et al., 2017; Silverman et al., 2018a; Äijö et al., 2017).

Yet, inference in MLN models is challenging due to lack of conjugacy between the multinomial and the logistic normal. Early work with MLN models used Metropolis within Gibbs samplers (Cargnoni et al., 1997; Billheimer et al., 2001) and could scale to just a small number multinomial categories (*i.e.*, less than 5). Recently, Pólya–Gamma data augmentation was proposed as a means of inference in MLN regression by augmenting Pólya–Gamma random variables between the multinomial and logistic normal components of a model. Yet for MLN models, the number of Gibbs sampling steps scales linearly with the number of multinomial categories (Polson et al., 2013). Numerous authors have found this approach too computationally intensive to scale to large multinomial models and have instead developed augmentation methods based on a stick-breaking representation of the multinomial (Linderman et al., 2015; Zhang and Zhou, 2017). However, this stick breaking representation does not maintain the logistic-normal form of the model and is sensitive to the labeling of multinomial categories (Linderman et al., 2015). Most recently, several authors (Silverman et al., 2018a; Äijö et al., 2017; Grantham et al., 2017) have shown that Hamiltonian Monte Carlo (HMC) provides for a more efficient and scalable approach to inference in MLN models. In particular, Grantham et al. (2017) used a HMC within a Gibbs sampler whereas both Silverman et al. (2018a), and Äijö et al. (2017) found that the No-U-Turn-Sampling algorithm provided by the Stan Modeling language (Gelman et al., 2015), provided more scalable inference. However, both these approaches are still limited in the number of categories or samples that they can handle. Silverman et al. (2018a) analyzed approximately 800 samples each with only 10 multinomial categories; Äijö et al. (2017) analyzed 36 multinomial categories but had to run their model over the dataset using a sliding window of 60 samples at a time; and Grantham et al. (2017) analyzed 166 samples and 2662 categories but had to impose low rank structure on the logistic normal model for computational tractability. In this work we show that our inference methods scale to hundreds to thousands of categories and samples and permit inference for a wide variety of models including non-linear regression models (as in Äijö et al. (2017)), dynamic linear models (as in Silverman et al. (2018a)), and linear regression models (as in Grantham et al. (2017)).

The layout of this article is as follows. In Section 2 we introduce a common motivation for the use of MLN models. In Section 3 we introduce the class of Marginally LTP models which encompasses many useful MLN models. In Section 4 we develop inference methods for Marginally LTP Models as well as developing specific acceleration for MLN models. In Section 5 we demonstrate our approaches through extensive simulation studies of MLN models. Finally, in Section 6 and 7 we demonstrate the utility of our approaches by developing both linear and non-linear regression models for microbiome sequence count data. Finally, we close with a discussion in Section 8.

2. Multinomial Logistic-Normal (MLN) Models

Our primary motivation in this work was to develop efficient inference for a class of models we refer to as multinomial logistic-normal (MLN) models. Consider a dataset Y consisting of N observations of D-dimensional count vectors; where the counting process for each observation is modeled as multinomial. For example, in the analysis of microbiome data we may consider $Y_{\cdot j}$ to be a count vector with a total of $n_j = \sum_i Y_{ij}$ counts, representing the number of DNA molecules observed for each of D different bacterial taxa in sample j. Yet, in many such datasets, multinomial count variation is just one source important variation. Consider the task of modeling a hypothetical dataset of N political polls each collected in a different year and each counting the number of polled individuals who identify with one of D different political parties. In such a setting we may wish to develop a model of the form:

$$\begin{aligned} Y_{\cdot t} &\sim \text{Multinomial}(n_t, \pi_{\cdot t}) \\ \pi &\sim f(\theta) \end{aligned}$$

where $f(\theta)$ represents a time-varying stochastic process with parameters θ. Often, a logistic normal model represents an appealing form for f as it, in contrast to Dirichlet models, allows for both positive and negative covariation between the political parties (Aitchison and Shen, 1980). Furthermore, if ϕ represents a log-ratio transform such as the ALR_D transform, with inverse given by:

$$\text{ALR}_D^{-1}(\eta_{\cdot j}) = \left(\frac{e^{\eta_{1j}}}{1+\sum_{i=1}^{D-1} e^{\eta_{ij}}}, \cdots, \frac{e^{\eta_{(d-1)j}}}{1+\sum_{i=1}^{D-1} e^{\eta_{ij}}}, \frac{1}{1+\sum_{i=1}^{D-1} e^{\eta_{ij}}} \right), \tag{1}$$

then we can write a multinomial logistic normal (MLN) model as a multinomial transformed-multivariate normal model:

$$\begin{aligned} Y_{\cdot t} &\sim \text{Multinomial}(n_t, \pi_{\cdot t}) \\ \pi_{\cdot t} &= \phi^{-1}(\eta_{\cdot t}) \\ \text{vec}(\eta) &\sim N(\mu, \Sigma). \end{aligned}$$

This relationship between the logistic-normal and the multivariate normal demonstrates another appealing property of logistic-normal models: They can often be easily formulated as a transformation of existing multivariate normal models.

In what follows, we develop efficient inference methods for a class of models we term Marginally Latent Matrix-T (Marginally LTP) models. We show that the class of Marginally LTP models encompasses many useful MLN models such as linear regression models, non-linear regression models, and time-series models.

3. Modeling Overview

In this section we will introduce Marginally Latent Matrix-T Process (Marginally LTP) models as a flexible class of models capable of describing a wide variety of linear regression, non-linear regression, and time-series models.

3.1 Matrix-Normal and Matrix-T, Distributions and Processes

To build the class of Marginally LTP models we first review matrix-normal distributions and processes as well as matrix-t distributions and processes, highlighting properties we will make use of in this article.

Matrix-Normal Distribution The matrix-normal distribution is a generalization of the multivariate normal distribution to random matrices. We describe a random $m \times n$ matrix X as being distributed matrix-normal $Y \sim N(M, U, V)$ if $\text{vec}(Y) \sim N(\text{vec}(M), V \otimes U)$ where $\otimes$ denotes the Kronecker product, U is a $m \times m$ covariance matrix and V is a $n \times n$ covariance matrix.

Matrix-Normal Process We define a stochastic process Y as a matrix-normal process on the set $\mathcal{X} = \mathcal{X}^{(1)} \times \mathcal{X}^{(2)}$ and denoted $\mathsf{Y} \sim \mathsf{GP}(\mathsf{M}, \mathsf{K}, \mathsf{A})$ if Y evaluated on any two finite subsets $\mathbf{x}^{(1)} = (x_1^{(1)}, \dots, x_P^{(1)}) \in \mathcal{X}^{(1)}$ and $\mathbf{x}^{(2)} = (x_1^{(2)}, \dots, x_N^{(2)}) \in \mathcal{X}^{(2)}$ is distributed as $Y \sim N(M, K, A)$ where $M_{ij} = \mathsf{M}(x_i^{(1)}, x_j^{(2)})$, $K_{ij} = \mathsf{K}(x_i^{(1)}, x_j^{(1)})$, $A_{ij} = \mathsf{A}(x_i^{(2)}, x_j^{(2)})$ for matrix function M and kernel functions K and A. The requirement that K and A be kernel functions implies that the matrices K and A are covariance matrices (*i.e.*, they are symmetric positive definite).

Matrix-t Distribution The matrix-t distribution is a generalization of the multivariate-t distribution to random matrices. Like the multivariate-t, the matrix-t can be defined constructively through its relationship to the matrix-normal and inverse Wishart distributions. Let Σ denote a random covariance matrix such that $\Sigma \sim IW(\Xi, \upsilon)$ where Ξ represents a positive semi-definite scale matrix and $\upsilon > 0$. Also suppose that $X \sim N(0, I, V)$. If $CC^T = \Sigma$ then the distribution of $Y = CX$ is denoted as matrix-t such that $Y \sim T(\upsilon, 0, \Xi, V)$. For a random matrix $\eta \sim T(\upsilon, B, K, A)$ the log density of η may be written

$$\begin{aligned}\log T_{P\times N}(\eta \mid \upsilon, B, K, A) &= \log \Gamma_P\left(\frac{\upsilon + N + P - 1}{2}\right) - \log \Gamma_P\left(\frac{\upsilon + P - 1}{2}\right) - \frac{NP}{2}\log \pi \\ &- \frac{N}{2}\log|K| - \frac{p}{2}\log|A| - \frac{\upsilon + N + P - 1}{2}\log\left|I_p + K^{-1}[\eta - B]A^{-1}[\eta - B]^T\right| \quad (2)\end{aligned}$$

where $\Gamma_a(b)$ refers to the multivariate gamma function. These results follows directly from Gupta and Nagar (2018, p. 134).

Matrix-t Process Through analogy to our definition of matrix normal processes, we define a matrix-t process through its relationship to the matrix-t distribution. We define a stochastic process $\mathsf{Y} \sim \mathsf{TP}(\upsilon, \mathsf{B}, \mathsf{K}, \mathsf{A})$ defined on the set $\mathcal{X} = \mathcal{X}^{(1)} \times \mathcal{X}^{(2)}$ as a matrix-t process if Y evaluated on any two finite subsets $\mathbf{x}^{(1)} = (x_1^{(1)}, \ldots, x_P(1)) \in \mathcal{X}^{(1)}$ and $\mathbf{x}^{(2)} = (x_1^{(2)}, \ldots, x_N^{(2)}) \in \mathcal{X}^{(2)}$ is distributed as $Y \sim T(\upsilon, B, K, A)$ where υ is a scalar strictly greater than zero, $B_{ij} = \mathsf{B}(x_i^{(1)}, x_j^{(2)})$, $K_{ij} = \mathsf{K}(x_i^{(1)}, x_j^{(1)})$, and $A_{ij} = \mathsf{A}(x_i^{(2)}, x_j^{(2)})$ for matrix function B, and kernel functions K and A. Matrix-t processes can be alternatively seen as a multivariate generalization of Student-t processes which have found widespread use in statistical analysis (Shah et al., 2014).

3.2 Latent Matrix-t Processes (LTPs)

To generalize matrix-t processes to a more flexible set of data types, *e.g.*, count data, we now define LTPs as a generalization of a matrix-t processes. We accomplish this by defining a stochastic process Y as a hierarchical process formed by a process F having parameters that, with appropriate transformation ϕ, follow a matrix-t process. Additionally, we now explicitly denote dependence on model hyper-parameters which we collectively refer to as δ.

Definition 1 ***Latent Matrix-t Process*** *We define a stochastic process* Y *as a latent matrix-t process* $\mathsf{Y} \sim \mathsf{LTP}(\mathsf{F}, \phi, \upsilon, \mathsf{B}, \mathsf{K}, \mathsf{A}, \delta)$ *on the set* $\mathcal{X} = \mathcal{X}^{(1)} \times \mathcal{X}^{(2)}$ *if* Y *evaluated on any* P *dimensional finite subset* $\boldsymbol{x}^{(1)} \in \mathcal{X}^{(1)}$ *and any* N *dimensional finite subset* $\boldsymbol{x}^{(2)} \in \mathcal{X}^{(2)}$ *is distributed*

$$Y \sim f(\pi, \delta) \tag{3}$$

$$\pi = \phi^{-1}(\eta) \tag{4}$$

$$\eta \sim T(\upsilon, B(\delta), K(\delta), A(\delta)). \tag{5}$$

where η *denotes a* $P \times N$ *real valued matrix,* $B(\delta)$ *a* $P \times N$ *dimensional real valued matrix function of parameters* δ *defined by* $[B(\delta)]_{ij} = \mathsf{B}(x_i^{(1)}, x_j^{(2)}, \delta)$*,* $K(\delta)$ *is a* $P \times P$ *covariance matrix defined as* $[K(\delta)]_{ij} = \mathsf{K}(x_i^{(1)}, x_j^{(1)}, \delta)$*,* $A(\delta)$ *is an* $N \times N$ *covariance matrix defined as* $[A(\delta)]_{ij} = \mathsf{A}(x_i^{(2)}, x_j^{(2)}, \delta)$*,* υ *is a scalar subject to* $\upsilon > 0$*,* π *is an element of a space* Π *defined via the one-to-one mapping* $\phi^{-1} : \mathcal{R}^{P \times N} \to \Pi$*, and* f *denotes a probabilistic model for the observed data (a likelihood model), with parameters* π *and* δ*, which is itself an evaluation of the process* F *evaluated on a finite subset of the set* Π*.*

3.3 Marginally LTP Models

To allow us to represent latent processes beyond LTPs, we next introduce a generalization of LTPs to a larger class which we term Marginally LTP models. This definition is straightforward, we define Marginally LTP models as those models which have a marginal that is an LTP.

Definition 2 ***Marginally LTP models*** *If a model described by the joint distribution $p(\eta, \Psi, Y)$ may be written as $p(\Psi \mid \eta, Y)\, p(\eta, Y)$ where $p(\eta, Y)$ is an LTP, we refer to $p(\eta, \Psi, Y)$ as a Marginally LTP model and $p(\eta, Y)$ as the model's collapsed representation.*

In the next three subsections we demonstrate that Marginally LTP models provide a rich class of models. We give three examples of Marginally LTP models: (1) a class of multivariate generalized linear models; (2) a flexible class of models for inference in multivariate non-Gaussian time-series; and (3) a flexible class of multivariate generalized non-linear regression models.

3.3.1 Generalized Multivariate Conjugate Linear (GMCL) Models

First we develop generalization of Bayesian multivariate linear regression with conjugate priors which permits non-Gaussian observations (Rossi et al., 2012, p. 32). As in Section 2, let us consider Y to represent N independent D-variate measurements and consider X to represent N sets of Q-dimensional covariates. We define generalized multivariate conjugate linear (GMCL) models as

$$\begin{aligned} Y_{\cdot j} &\sim f(\pi_{\cdot j}) && (6)\\ \pi_{\cdot j} &= \phi^{-1}(\eta_{\cdot j}) && (7)\\ \eta_{\cdot j} &\sim N(\Lambda X_{\cdot j}, \Sigma) && (8)\\ \Lambda &\sim N(\Theta, \Sigma, \Gamma) && (9)\\ \Sigma &\sim IW(\Xi, \upsilon). && (10) \end{aligned}$$

We may describe the joint density of this model as $p(\Lambda, \Sigma, \eta, Y)$ which can be factored as $p(\Lambda, \Sigma \mid \eta, Y)\, p(\eta, Y)$. Therefore, to parallel to the definition of Marginally LTP models we may equate $\Psi = \{\Lambda, \Sigma\}$. In Appendix A we prove that $p(\eta, Y)$ is an LTP with parameters

$$\begin{aligned} B &= \Theta X \\ K &= \Xi \\ A &= I_N + X^T \Gamma X \end{aligned}$$

and with $\{\Theta, \Gamma, \Xi\} \in \delta$. This result demonstrates that all GMCL models are Marginally LTP models. Further, by letting f denote the multinomial distribution and ϕ^{-1} denote the inverse ALR transform, we can build multinomial logistic-normal linear models as a special case of GMCL models.

3.3.2 Generalized Multivariate Dynamic Linear Models (GMDLMs)

We develop a flexible class of multivariate time-series models for non-Gaussian observations. We term this class of models generalized multivariate dynamic linear models (GMDLMs). GMDLMs represent an extension of the multivariate dynamic linear models introduced in Quintana and West (1987) and developed further in West and Harrison (1997, Ch. 16) to non-Gaussian observations. Using notation from West and Harrison (1997, Ch. 16), let η_t^T denote a row-vector (*i.e.*, the transpose of the t-th column of η). We define the GMDLM as

$$Y_{\cdot j} \sim f(\pi_{\cdot j}) \qquad (11)$$

$$\begin{aligned}
\pi_{\cdot j} &= \phi^{-1}(\eta_{\cdot j}) && (12)\\
\eta_t^T &= F_t^T \Theta_t + \nu_t^T, \quad \nu_t \sim N(0, \gamma_t \Sigma) && (13)\\
\Theta_t &= G_t \Theta_{t-1} + \Omega_t, \quad \Omega_t \sim N(0, W_t, \Sigma) && (14)\\
\Theta_0 &\sim N(M_0, C_0, \Sigma) && (15)\\
\Sigma &\sim IW(\Xi, \upsilon) && (16)
\end{aligned}$$

where Θ_t represents a $Q \times P$ matrix describing the state of the time-series at time t, G_t denotes the $Q \times Q$ state transition matrix at time t, F_t denotes a $Q \times 1$ vector describing a linear model relating the latent space to the parameters η_t, Σ is a $P \times P$ covariance matrix specifying the covariation between the P dimensions of the time-series, W_t is a $Q \times Q$ covariance matrix describing the covariation of the perturbations affecting latent states, and γ_t is a scalar allowing an analyst to weight the importance of select observations (γ_t is typically equal to 1).

The joint model for the GMDLM can be written $p(\Theta, \Sigma, \eta, Y)$ which can be factored as $p(\Theta, \Sigma \mid \eta, Y)\, p(\eta, Y)$. To parallel the definition of Marginally LTP models, here we have $\Psi = \{\Theta, \Sigma\}$. In Appendix B we prove that $p(\eta, Y)$ is an LTP with parameters

$$\begin{aligned}
B &= \begin{bmatrix} | & & | & & | \\ \alpha_1 & \cdots & \alpha_t & \cdots & \alpha_T \\ | & & | & & | \end{bmatrix}\\
\alpha_t &= (F_t^T \mathcal{G}_{t:1} M_0)^T\\
K &= \Xi\\
A_{t,t-k} &= \begin{cases} \gamma_t + F_t^T \left[W_t + \sum_{\ell=t}^{2} \mathcal{G}_{t:\ell} W_{\ell-1} \mathcal{G}_{\ell:t}^T + \mathcal{G}_{t:1} C_0 \mathcal{G}_{1:t}^T \right] F_t \text{ if } k = 0 \\ F_t^T \left[\mathcal{G}_{t:t-k+1} W_{t-k} + \sum_{\ell=t-k}^{2} \mathcal{G}_{t:\ell} W_{\ell-1} G_{\ell:t-k}^T + \mathcal{G}_{t:1} C_0 G_{1:t-k}^T \right] F_{t-k} \text{ if } k > 0 \end{cases}
\end{aligned}$$

where we have introduced $\mathcal{G}_{t:\ell}$ as a short hand notation for the product $G_t \cdots G_\ell$ and where we have hyper-parameters $\{\Xi, M_0, C_0, W_1, \dots, W_T, \gamma_1, \dots, \gamma_T, G_1, \dots, G_T, F_1, \dots, F_T\} \in \delta$. This result demonstrates that GMDLMs are a special case of Marginally LTP models.

3.4 Generalized Multivariate Gaussian Process (GMGP) Models

Finally, we develop a flexible class of generalized multivariate non-linear models based on the matrix normal processes discussed in Section 3.1. These models utilize a separable kernel structure to allow modeling of vector valued data as seen, for example, in coregionalization models Álvarez et al. (2012). As a motivating example, suppose that we wish to model a microbiome time-series. In particular, suppose we wish to predict the relative abundance of an unobserved taxa at an unobserved time-point. Let us consider X to encompass available temporal metadata for observed samples, e.g., time-indices as well as other relevant covariates influencing composition at each observed time-point. Further, let us consider Z to encompass available metadata regarding each observed bacterial taxa, e.g., 16S sequence as well as whether the bacteria is aerobic or anaerobic. In this section we describe a flexible class of models which we term Generalized Multivariate Gaussian Process (GMGP) models which are capable of performing this, as well as many other, analysis tasks.

To enable GMGP models to make predictions regarding unobserved multinomial categories (e.g., unobserved taxa) we must first define Inverse Wishart Processes. These processes can be defined constructively in a similar manner to the matrix normal and matrix-t processes we defined in Section 3.1. Given a set Z with P-dimentional finite subset $Z = [Z_{\cdot 1}, \dots, Z_{\cdot P}]$, a scalar $\nu > 0$, and a kernel function Ξ such that $\Xi_{ij} = \Xi(Z_{\cdot i}, Z_{\cdot j})$, we define a stochastic process $\mathsf{\Sigma} \sim \mathsf{IWP}(\Xi, \nu)$ as an Inverse Wishart Process on the set Z if $\mathsf{\Sigma}$ evaluated on any subset Z is distributed as $\Sigma \sim IW(\Xi, \nu + p)$. In words, an Inverse Wishart Process is a probability distribution over kernel functions.

Using the above construction of Inverse Wishart Processes, we can now define the GMGP model form:

$$Y_{\cdot j} \sim f(\pi_{\cdot j}) \tag{17}$$

$$\pi_{\cdot j} = \phi^{-1}(\eta_{\cdot j}) \tag{18}$$

$$\eta_{\cdot j} \sim N(\mathsf{\Lambda}(X_{\cdot j}), \mathsf{\Sigma}(Z)) \tag{19}$$

$$\mathsf{\Lambda} \sim \mathsf{GP}(\mathsf{\Theta}, \mathsf{\Sigma}, \mathsf{\Gamma}) \tag{20}$$

$$\mathsf{\Sigma} \sim \mathsf{IWP}(\Xi, \nu) \tag{21}$$

where $\mathsf{\Theta}$ is a mean function and $\mathsf{\Gamma}$ as well as Ξ are kernel functions.

We may describe the joint density of the above model as $p(\mathsf{\Lambda}, \mathsf{\Sigma}, \eta, Y, X)$ which can be factored as $p(\mathsf{\Lambda}, \mathsf{\Sigma} | \eta, Y, X) p(\eta | Y, X)$. In Appendix C, we prove that $p(\eta | Y, X)$ is an LTP with parameters $\mathsf{B} = \mathsf{\Theta}$, $\mathsf{K} = \Xi$, and $\mathsf{A} = \mathsf{I} + \mathsf{\Gamma}$ where I represents the identity kernel defined by:

$$\mathsf{I}(x_i, x_j) \begin{cases} 1 \text{ if } x_i = x_j \\ 0 \text{ otherwise} \end{cases} .$$

It should be noted that the LTP form of GMGP models is very similar to that of GMCL models; the major difference between GMGP and GMCL models being the use of mean and kernel functions in place of mean and covariance matrices. Still, we discuss these models separately as they will often be used in very different ways – GMCL models for inferring linear effects of covariates, GMGP models for non-linear smoothing and prediction. We demonstrate examples of both of these models using real data in Sections 6 and 7.

4. Inference in Marginally LTP Models

Our overarching goal was to develop efficient and accurate posterior inference for MLN models, many of which are a special case of Marginally LTP models. In this section, we demonstrate how the canonical LTP form of Marginally LTP Models can be exploited for efficient inference of this larger model class. types of parameters, η which are distributed matrix-t and of a model to produce a LTP form. The sampling η on η and observed data $(p(\Psi \mid \eta, Y))$. In Section 4.1 we introduce a sampling scheme for Marginally LTP models which we refer to as the collapse-uncollapse (CU) sampler which exploits the hierarchical structure of Marginally LTP models to improve computational efficiency for various types of inference. In Section 4.2 we further build on the CU sampler by introducing a Laplace approximation as a means of accelerating a bottleneck step in the CU sampler. In Sections

4.3 we discuss the CU sampler in the context of the GMCL, GMDLM and GMGP models introduced in the last section. In Section 4.4, we discuss error bounds for the Laplace approximation. In Section 4.5, we discuss inference of hyperparameters. Finally, in Section 4.6 we discuss the *fido* software package which implements a number of MLN models using the CU sampler with Laplace approximation based on these models Marginally LTP form.

4.1 The Collapse-Uncollapse (CU) Sampler

Consider the task of sampling from the posterior distribution of a Marginally LTP model with joint density $p(\Psi, \eta, Y)$. The corresponding posterior density can be decomposed as

$$p(\eta, \Psi \mid Y) = p(\Psi \mid \eta, Y)\frac{p(\eta, Y)}{p(Y)}.$$

This decomposition implies that, given a Marginally LTP model with joint probability $p(\eta, \Psi, Y)$, we may sample from the posterior by first sampling from the posterior of the collapsed (LTP) model $p(\eta, Y)$ and then given that sample of η and the observed Y we may then sample Ψ from the conditional $p(\Psi \mid \eta, Y)$. Together the sample of η and Ψ then represents a single sample from the posterior of the Marginally LTP model, $p(\Psi, \eta \mid Y)$ (Algorithm 1).

Algorithm 1: The Collapse-Uncollapse (CU) Sampler for Marginally LTP Models

```
Data: $Y, v, B, K, A$
Result: $S$ samples of the form $\{\Psi^{(s)}, \eta^{(s)}\}$
Sample $\{\eta^{(1)}, \dots, \eta^{(S)}\} \sim p(\eta \mid Y)$ where $p(\eta \mid Y)$ is an LTP;
for $s$ in $\{1, \dots, S\}$ do in parallel
    Sample $\Psi^{(s)} \sim p(\Psi \mid \eta^{(s)}, Y)$;
```

Our rationale for focusing on the CU sampler for inference in Marginally LTP models is as follows. We expect that many Marginally LTP models (such as those introduced in Section 3) have partial conjugacy. Exploiting this partial conjugacy is central to many popular methods such as Metropolis-within-Gibbs (Cargnoni et al., 1997). Yet, by embedding MCMC steps within a Gibbs sampler techniques such as adaptation (Gelman et al., 2015) or approximate methods such as Laplace approximations may not make sense as they would have to be recomputed at each step. In contrast, the CU sampler allows the non-conjugate sampling to occur up front (in the sampling of $p(\eta \mid Y)$) so that such techniques can be used. Moreover, after multiple samples of η have been produced, uncollapsing the model can be done in parallel for each sample of η. Therefore, the CU sampler may be advantageous as it permits the use of adaptive or approximate methods for sampling the non-conjugate model components and permits a degree of parallelism not allowed by Metropolis-within-Gibbs.

The CU Sampler for Marginally LTP Models therefore requires two features for efficient inference. First, we require an efficient means of producing samples from the collapsed (LTP) form $p(\eta \mid Y)$. As we will show in Section 5, sampling $p(\eta \mid Y)$ can be more efficient than sampling $p(\Psi, \eta \mid Y)$ just by virtue of the fact that the former has fewer dimensions. Therefore the CU sampler alone can be more efficient than sampling the full (uncollapsed) model. Still, in Section 4.2 we develop a Laplace approximation for $p(\eta \mid Y)$ which can further improve efficiency. Second, we require an efficient means of sampling from the posterior conditional $p(\Psi \mid \eta, Y)$. In Section 4.3 we discuss efficient means of sampling $p(\Psi \mid \eta, Y)$ for the GMCL, GMDLM, and GMGP model classes.

4.2 Laplace Approximation for the Collapsed Form

Sampling $p(\eta \mid Y)$ is often the major computational bottleneck when inferring Marginally LTP models via the CU sampler. To accelerate this step, we developed a Laplace approximation for the density $p(\eta \mid Y)$. This approximation is defined as $q(\eta \mid Y) = N(\operatorname{vec}\hat{\eta}, H^{-1}(\operatorname{vec}\hat{\eta}))$ where $\hat{\eta}$ denotes the *maximum a posteriori* (MAP) estimate of $p(\eta \mid Y)$ and $H^{-1}(\operatorname{vec}\hat{\eta})$ denotes the inverse Hessian matrix of $\log p(\eta \mid Y)$ evaluated at the point $\operatorname{vec}\hat{\eta}$. That is, $\hat{\eta}$ is defined as the solution to the following optimization problem

$$\hat{\eta} = \underset{\eta \in \mathcal{R}^{P \times N}}{\operatorname{argmin}} \left[-\log p(\eta \mid Y)\right]. \tag{22}$$

The solution to this optimization problem is discussed in Appendix F.

While the accuracy of our Laplace approximation will depend on a number of factors including the choice of likelihood, prior, and link function, we hypothesized that such a Laplace approximation would provide an accurate approximation to an LTP posterior in a number of common settings. First, all exponential family likelihoods are log-convex with respect to their natural parameters (Jordan, 2010). Therefore, we expect the Laplace approximation to be particularly useful with any choice of likelihood f from the exponential family (e.g., the multinomial distribution) and with a corresponding choice of ϕ such that η represents the natural parameters of f (e.g., the ALR transform). Second, with regards to the matrix-t prior, the matrix-normal can provide a good approximation for the matrix-t for suitably large υ (Gupta and Nagar, 2018, p. 137) as it is both globally symmetric and log-convex about the MAP estimate. We hypothesized that even though the matrix-t is not globally log-convex except as $\upsilon \to \infty$, in practice the log-convexity about the MAP estimate coupled with its global symmetry would be enough to provide a useful approximation even for small values of υ. We note that both our simulation studies in Section 5 and analyses of real data in Section 6 suggest this hypothesis is reasonable. Finally, specifically for models parameterized by probabilities (such as the Multinomial logistic-normal), MacKay (1998) showed that the softmax parameterization can produce more accurate Laplace approximations than the more traditional simplex basis. Notably, the inverse ALR parameterization we choose is a linear transformation of the softmax transform (Pawlowsky-Glahn et al., 2015) and therefore has identical accuracy to a Laplace approximation using the softmax parameterization (MacKay, 1998). Together, these features led us to hypothesize that a Laplace approximation could provide a useful and accurate approximation for the the posterior of an LTP.

Developing an efficient Laplace approximation for LTP models required closed form solutions for the gradient and Hessian of LTPs. To develop these tools note that, by Bayes rule, we may write

$$-\log p(\eta \mid Y) \propto -\log f(Y \mid \phi^{-1}(\eta)) - p(\eta). \tag{23}$$

By linearity of the derivative operator we may write the gradient and Hessian of $-\log p(\eta \mid Y)$ as

$$-\frac{d \log p(\eta \mid Y)}{d\text{vec}(\eta)} = -\frac{d \log f(Y \mid \phi^{-1}(\eta))}{d\text{vec}(\eta)} - \frac{d \log p(\eta)}{d\text{vec}(\eta)} \tag{24}$$

$$-\frac{d^2 \log p(\eta|Y)}{d\text{vec}(\eta)d\text{vec}(\eta)^T} = -\frac{d^2 \log f(Y \mid \phi^{-1}(\eta))}{d\text{vec}(\eta)d\text{vec}(\eta)^T} - \frac{d^2 \log p(\eta)}{d\text{vec}(\eta)d\text{vec}(\eta)^T}. \tag{25}$$

Thus we find that calculating the gradient and Hessian of LTPs reduces to calculating the gradient and hessian of $\log f(Y \mid \phi^{-1}(\eta))$ and the matrix-t density $\log p(\eta \mid X)$ separately. The additive structure of the gradient and Hessian are central to generalizing this approach to a variety of different observation distributions f and transformations ϕ^{-1}. In Appendix D we provide the gradient and Hessian for the matrix-t density. With these results, to derive a flexible class of multinomial logistic-normal models, we only need to provide the gradient and Hessian for the logit-parameterized multinomial which we give in Appendix E. We describe the implementation of the Laplace Approximation for an LTP in Appendix F.

4.3 Efficient Sampling from Posterior Conditionals

The second step of the CU sampler involves sampling from the density $p(\Psi \mid \eta, Y)$. While the density of $p(\Psi \mid \eta, Y)$ is specific to the particular Marginally LTP model, we develop efficient means of sampling from this density for the GMCL, GMDLM, and GMGP models in Appendices A, B, and C respectively. In particular, for all three of these model classes we make use of the fact that Ψ is conditionally independent of Y given η, that is $p(\Psi \mid \eta, Y) = p(\Psi \mid \eta)$. This conditional independence also reduces sampling from the conditionals to computing the posterior distribution of standard Bayesian multivariate linear regression for GMCL and GMGP model and conjugate multivariate dynamic linear models for the GMDLM model. That is, for all three of these model classes, sampling the conditionals reduces to posterior inference for equivalent Bayesian Gaussian models that have been well described previously and have efficient closed form solutions.

4.4 Error Rate for a Laplace Approximation to the Collapsed Form

The inference scheme we propose above for Marginally LTP models has two parts: First, sample from the collapsed LTP representation of the model; Second, uncollapse those samples to produce samples from the full Marginally LTP model. If, as we discuss above, we use a Laplace approximation to sample from the collapsed LTP representation, then the only error induced by this inference scheme is due to the Laplace approximation. We wanted to develop intuition regarding the error rate of this approximation when the observation distribution is a logit-parameterized multinomial. In Appendix K, we prove that for large υ this error rate is $O_p((D-1)\sum_{j=1}^{N} n_j^{-1})$. That is, the error is stochastically bounded by the sum of the inverse of the average number of counts in each sample. This result follows from theory recently proposed by Ogden (2018) and provides a more general error bound than those used by Kass and Steffey (1989) or Rue et al. (2009). In particular, this bound accounts for not only the number of observed multinomial samples (N) but the number of counts in each multinomial sample (n_j) and the dimension over which those counts are spread (D).

This error bound is intuitive in a number of ways. First, a multinomial sample j with n_j counts can be thought of as n_j independent observations; it is therefore intuitive that our error bound is proportional to n_j^{-1}. Second, the number of dimensions in the Laplace approximation to a multinomial sample grows linearly with one minus the number of multinomial categories; intuitively, our error bound is proportional to $D-1$. Third, the number of dimensions in the Laplace approximation to the collapsed LTP form grows linearly with the number of observed samples; intuitively, our error bound grows (approximately) linearly with the number of observed samples. Finally, based on the observation that the multinomial parameterized by log-ratio coordinates is globally log-convex (Jordan, 2010) whereas the matrix-t distribution is only log-convex near the mean; it makes intuitive sense that a stronger likelihood (implied by larger values of n_j) would decrease the error of the Laplace Approximation.

This error bound also sheds light on when this Laplace approximation in the CU sampler will provide a useful, accurate inference method for MLN models. For example, this error bound suggests that an ideal dataset for this Laplace approximation is one that has many non-zero counts and lower data-sparsity. In contrast, it suggests that the Laplace approximation should not be used for high-dimensional classification problems, where there are many multinomial categories but only a single non-zero entry per sample. Still, as we demonstrate in the next section, the Laplace approximation can handle substantial data sparsity and many small counts with only minimal error.

4.5 Hyperparameter Inference

Until this point we have not considered the presence of unknown hyperparameters in the LTP form (*i.e.* we have considered δ or ν as given). Yet, for a number of Marginally LTP models, we expect estimation of such hyperparameters will be of interest. For example, within the GMDLM model we anticipate researchers may want to allow the terms W_t to be subject to their own stochastic model. This would in turn require that some portion of δ is unknown. Alternatively, for GMCL models, analysts may want to infer the degree-of-freedom parameter ν empirically rather than setting it based on subjective prior information. Overall, we leave inference of ν and δ as future work but note a few potential avenues for practitioners looking to infer these parameters. When the hyper-parameter set $\{\nu, \delta\}$ is small, these parameters may be efficiently selected by cross-validation (Rasmussen, 2003). In contrast, when the set is large (*i.e.*, when δ is high-dimensional), alternative approaches are likely needed. In particular, we note that Type-II MAP estimation can provide an efficient means of empirically setting hyper-parameters in a variety of hierarchical Bayesian models (Riihimäki et al., 2014).

4.6 Software for Marginally LTP models with Multinomial Observations

For inference of Marginally LTP models with multinomial observations and log-ratio link functions, we developed the R package *fido* (Silverman, 2019). *Fido* implements the CU sampler with Laplace approximation described above using optimized C++ code. Estimation of $\hat{\eta}$ is performed using the L-BFGS optimizer which we have found provides efficient and stable numerical results. Additionally all code required to reproduce the results of the next two sections, including the alternative implementations of multinomial logistic-normal linear models discussed in Section 5 is available as a GitHub repository at github.com/jsilve24/fido_paper_code.

5. Simulations

We performed a series of simulation studies to evaluate the CU sampler both with and without the Laplace Approximation in terms of accuracy and efficiency of posterior inference of multinomial logistic-normal models. The only portion of our inference algorithm that is approximate is the Laplace approximation to the LTP form. As this form is shared by all marginally LTP models, we focus our simulations only on multinomial logistic-normal linear models for simplicity (*e.g.*, Equations (6)-(10) where f is the multinomial distribution and ϕ is the ALR_D transform). To evaluate the utility of the CU sampler we compared Hamiltonian Monte Carlo (HMC) of the full model (HMC Uncollapsed) to the CU sampler where sampling of the collapsed (LTP) form was performed using HMC (HMC Collapsed). Both HMC implementations were inferred using the highly optimized No-U-Turn-Sampler provided in the Stan modeling language (Gelman et al., 2015) which has been frequently used for the analysis of MLN models (Äijö et al., 2017; Silverman et al., 2018a). To further evaluate the utility of the Laplace approximation to the collapsed form in the CU sampler (LA Collapsed), we used the function *pibble* from the *fido* software package described in Section 4.6. Finally, to compare LA Collapsed to an alternative scheme for approximate inference, we included two mean-field automatic-differentiation Variational Bayes (VB) implementations (Kucukelbir et al., 2015). The first was a VB approximation to the full form (VB Uncollapsed), the second was a VB approximation to just the collapsed form of the CU sampler (VB Collapsed). VB Uncollapsed was unstable in practice and often resulted in error during optimization (likely due to the increased number of parameters in the uncollapsed model). As a result, only the results form VB Collapsed could be shown below.

In order to compare these implementations, we created a series of simulations based on the corresponding likelihood model (Appendix H).We identified three key parameters, the sample size (N), the observation dimension (D), and the number of model covariates (Q) which we varied in order to test each implementation over a wide range of conditions. By varying these parameters in different simulations we were able to vary the error bound for the Laplace approximation introduced in Section 4.4 (Figure S1). We designed these simulations to span a wide range of sparsity (Figure 1, Column 1). We choose the tuple $\{N = 100, D = 30, Q = 5\}$ as our base condition and independently varied each simulation parameter from that base condition (N from 10 to 1000, D from 3 to 500, and Q from 2 to 500). Each panel in Figure 1 shows a different simulation metric (*e.g.*, percent of data matrix Y that were zero counts or the performance of a given inference method on each simulation) for a given tuple when a particular element of the tuple (N, D, Q) is varied from the base condition. For example, the top left panel shows the sparsity of each dataset for $N = 100$, $Q = 5$, and where D is varied (x-axis). Additionally, to account for the stochastic nature of the simulations, three simulations were performed for each tuple $\{N, D, Q\}$. For each simulation, each of the five implementations were fit and allowed a maximum of 48 hours to produce 2000 samples. Prior hyper-parameters were chosen to reflect common default choices, *e.g.*, mean parameters set to zero, and covariance parameters set to the identity matrix. The prior degrees-of-freedom parameter ν is defined on the range $\nu > D$, this parameter was chosen as $\nu = D + 10$. Further details of the simulation and model fitting procedure can be found in Appendix H.

To quantify the accuracy and efficiency of each implementation we defined the following performance metrics. As a measure of efficiency, we calculated the average number of seconds needed for the implementation to produce one independent sample from the target posterior (*i.e.*, Seconds per Effective Sample - SpES). Specifying independent samples is important as HMC samplers produce autocorrelated samples. In contrast, both LA Collapsed and VB Collapsed produce independent samples from the approximate posterior, as a result, for these two methods, SpES equals the number of samples per second. To quantify the accuracy of point estimates from each implementation (*i.e.*, either the posterior mean or MAP estimates) we used the root mean squared error of the point estimate for Λ from its true simulated value. Notably, given finite N we do not expect that any implementation will be able to perfectly reconstruct the true simulated value for Λ; rather, this metric provides a means of comparing the relative performance of each implementation. Finally, to quantify the accuracy of uncertainty quantification from each implementation we compared posterior intervals against those of the HMC Collapsed model which was taken as a gold standard. In particular, we define the root mean squared error of standard deviations as the average difference between the estimated posterior standard deviations, $sd(\Lambda_{ij})$, compared to the estimates produced by HMC Collapsed.

Beyond our error bound for the Laplace approximation, we hypothesized that the proportion of zero values in the dataset would impact the accuracy of both the Laplace and variational approximations. In particular, we hypothesized that datasets with higher than 30% zero values would see a substantial degradation in approximation accuracy. As hypothesized we found that the proportion of zero values (the sparsity) of the dataset closely resembled approximation accuracy (Figure 1). Yet, we found that in practice, LA collapsed performed far better than expected: LA Collapsed provided nearly identical estimates of posterior uncertainty to both HMC implementations up to over 90% data sparsity. Additionally, LA Collapsed provided nearly identical point estimates to both HMC implementation over the full spectrum of simulations. Finally, LA Collapsed was often up to 5 orders of magnitude faster than HMC and often 1-2 orders of magnitude faster than VB.

5.1 Computational Efficiency

Overall, the CU sampler with a Laplace approximation (LA Collapsed) provided the most efficient inference across all tested conditions. More specifically LA Collapsed displayed speed-ups of between 1 to 5 orders of magnitude in comparison to HMC Collapsed and HMC Uncollapsed and often between 1-2 order of magnitude compared to VB Collapsed. Notably, HMC Uncollapsed failed to complete sampling within 48 hours for $D > 100$.

Beyond the high efficiency of LA Collapsed, our results also demonstrate that the CU sampler can improve inference in HMC without the use of approximate inference methods. These results likely stem from the smaller number of dimensions in HMC for the collapsed versus uncollapsed implementations. Most noticeably, the collapsed representation com-

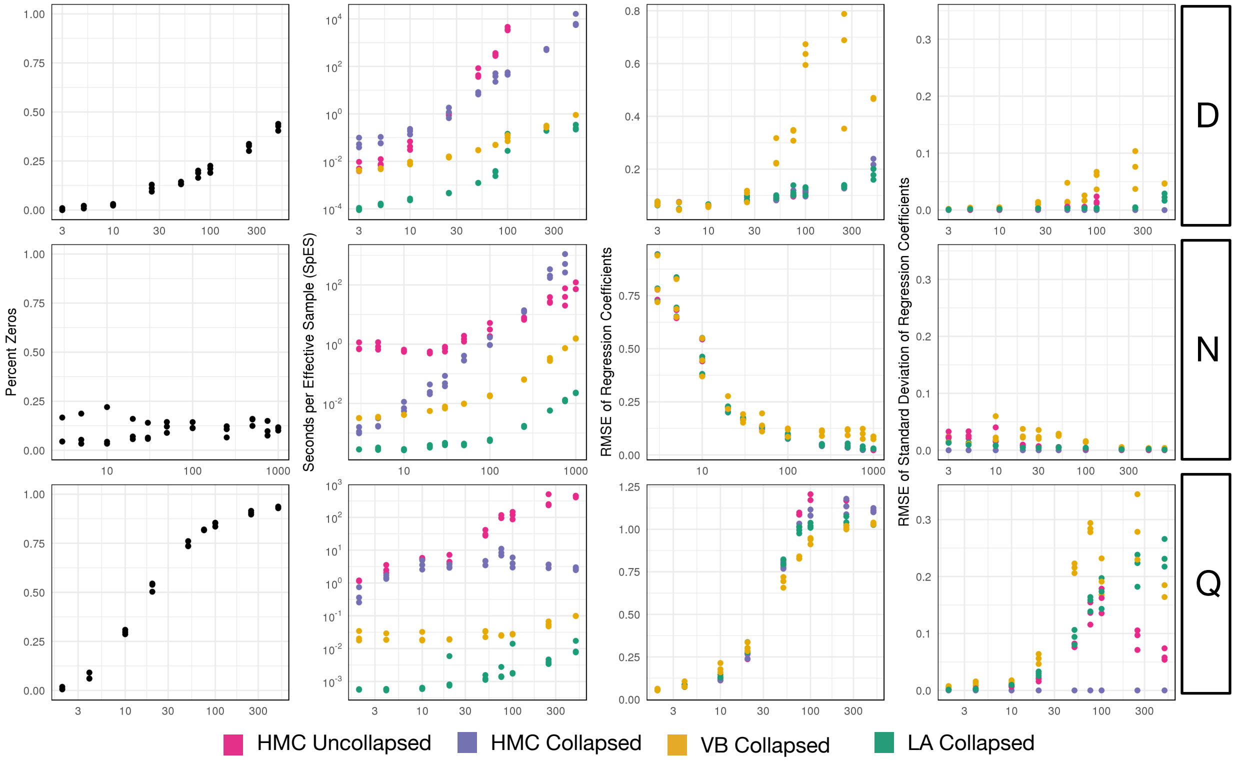

Figure 1: **Simulation study comparing multinomial logistic normal linear model implementations.** Each row of plots depicts simulation results for varying a different simulation parameter (D, the number of multinomial categories; N, the number of samples; and Q, the number of covariates). The percent of counts that were zero in each simulation is given in the first column. The error bound of the Laplace approximation, which was developed in Section 4.4, is shown in Figure S1. Implementations were compared in terms of efficiency (measured SpES), accuracy of point estimation (measured by RMSE of Regression Coefficients), and accuracy of uncertainty quantification (measured by RMSE of Standard Deviation of Regression Coefficients). For VB Collapsed and LA Collapsed, the number of effective samples is taken to be equal to the total number of samples as both methods produce independent samples from an approximation to the posterior.

pletely removes dependency on Q from HMC run-times as Λ is marginalized out of the collapsed representation. However, for large N the HMC Uncollapsed is more efficient than HMC Collapsed. This later result may reflect that the heavy tails of the matrix-t distribution produce a more challenging geometry for HMC than the expanded matrix normal and inverse Wishart forms. Such a finding has been well described previously for both univariate and multivariate-t distributions (Stan Development Team, 2018, Section 20).

5.2 Point Estimation

Overall point estimation using LA Collapsed (*i.e.*, MAP estimates) was nearly identical to point estimation using either HMC Collapsed or HMC Uncollapsed (*i.e.*, mean estimates). In contrast, point estimation using VB Collapsed produced substantially larger errors, especially for large values of D. Overall these results demonstrate that the CU sampler maintains accuracy in point estimation and that MAP estimation provides an excellent approximation to the mean in multinomial logistic normal models.

5.3 Uncertainty Quantification

Beyond accuracy of point estimates, we also wanted to study the accuracy of estimates of uncertainty from each implementation. We consider the HMC Collapsed implementation to be the gold standard on which we based our performance metric (*RMSE of standard deviations*). Except for values of Q greater or equal to 250 (where the proportion of zero values is >90%), the uncertainty estimates of LA Collapsed were nearly identical to those of both HMC implementations. Yet, at larger values of Q, when sparsity is >90%, we observed differences not only between LA Collapsed and HMC but between the two HMC implementations themselves. There are two possible explanations for this. First, that LA Collapsed had a slightly better point estimation accuracy in these same large Q simulations could point to the fact that LA Collapsed is correct and instead HMC estimates of uncertainty were incorrect due to the often small effective sample size for large Q. Alternatively, this could support our previous hypothesis that the Laplace approximation had higher error in uncertainty quantification with higher data sparsity. Given the ergodicity of HMC it seems more likely that the Laplace approximation is in error in these regions of high sparsity. Yet, that the approximation only began to show substantial error when sparsity is >90% is notable. Beyond LA Collapsed and the HMC implementations, VB Collapsed consistently demonstrated higher error in uncertainty quantification as compared to the other implementations.

Finally, to provide context regarding the size of the differences in uncertainty quantification, we provide direct visualizations of posterior intervals for all four implementations in Figure S3 and S4. These two simulations were chosen to highlight a case in which LA Collapsed was highly accurate (Figure S3) in terms of uncertainty quantification and a case in which it differed from HMC estimates (Figure S4). Notably, visualization of posterior intervals consistently demonstrated that the posterior mean was centered symmetrically in the 95% credible regions. This symmetry suggested that our metric *RMSE of standard deviations* captures much of the discrepancies in uncertainty quantification without higher order moments. Additionally, for context, we include a fifth implementation, PCLM (pseudo-count augmented linear model). The PCLM uses a pseudo-count based estimate of η which ignores the multinomial count variation. Such approximations are common in the analysis of microbiome sequence count data (Silverman et al., 2017; Gloor et al., 2016). Unsurprisingly, this PCLM implementation demonstrated substantially higher error rates than any of the other implementations (Figure S5).

6. Identifying Biomarkers of Crohn's Disease Using Microbiome Data

Crohn's Disease (CD) is a type of inflammatory bowel disease that has been linked to aberrant immune response to intestinal microbiota (Jostins et al., 2012; Khor and Hibberd, 2012; Gevers et al., 2014). To demonstrate that LA Collapsed (from the R package *fido*) provides an accurate and efficient means of modeling real microbiome data, we reanalyzed a previously published study comparing microbial composition in the terminal ileum of subjects with CD to healthy controls (Gevers et al., 2014). Only LA Collapsed could efficiently scale to this data size (49 taxa, 250 samples, 4 covariates). To allow us to compare to alternative implementations we randomly subset the data to contain 83 samples. On this subset HMC Uncollapsed and VB Collapsed repeatedly failed to run due to numerical instability. In addition, LA Collapsed produced posterior estimates nearly identical to HMC Collapsed but more than 1000 times faster.

Using the four model implementations introduced in Section 5, a Bayesian multinomial logistic normal linear model was fit to investigate the relationship between bacterial composition and CD. For both the full data-set and the subset, our regression model was defined for the j-th sample by the covariate vector

$$x_j = [1, x_{j(\text{CD})}, x_{j(\text{Inflamed})}, x_{j(\text{Age})}]^T$$

where $x_{j(\text{CD})}$ is a binary variable denoting whether the j-th sample was from a patient with CD or a healthy control, $x_{j(\text{Inflamed})}$ a binary variable denoting inflammation at time of sample collection, $x_{j(\text{Age})}$ denoting age of the subject, and the preceding 1 represents a constant intercept. To evaluate the impact of using small values for the degree-of-freedom parameter ν in model priors, we set $\nu = D + 3$. A full description of our prior assumptions is given in Appendix I and results of posterior predictive checks are shown in Figure S6.

Even though all four implementations were initialized identically, both the HMC Uncollapsed and VB Collapsed implementations repeatedly resulted in errors due to numerical instability. Thus only LA Collapsed and the HMC Collapsed implementations could fit this model for even the subset dataset. Whereas the HMC Collapsed model took approximately 30 minutes, LA Collapsed took only 3 seconds. Thus LA Collapsed is over 1000 times faster than HMC Collapsed on real data. Additionally, posterior estimates of Λ produced by both the HMC Collapsed and LA Collapsed implementations are nearly identical (Figure S7). These results demonstrate that in real data scenarios LA Collapsed can provide efficient and accurate posterior inference.

By modeling the full dataset we found the centered log-ratio (CLR) coordinates corresponding to 12 genera to be associated with CD status (95% credible interval not covering 0; Figure 2). These results are in general agreement with prior analyses (Gevers et al., 2014). As in prior analyses, we find a substantial increase in the abundance of proteobacteria in CD versus healthy controls. Similarly, we find that the families Pasteurellaceae and Enterobacteriacaeae, Gemellaceae, and Fusobacteriaceae are highly enriched and that the class

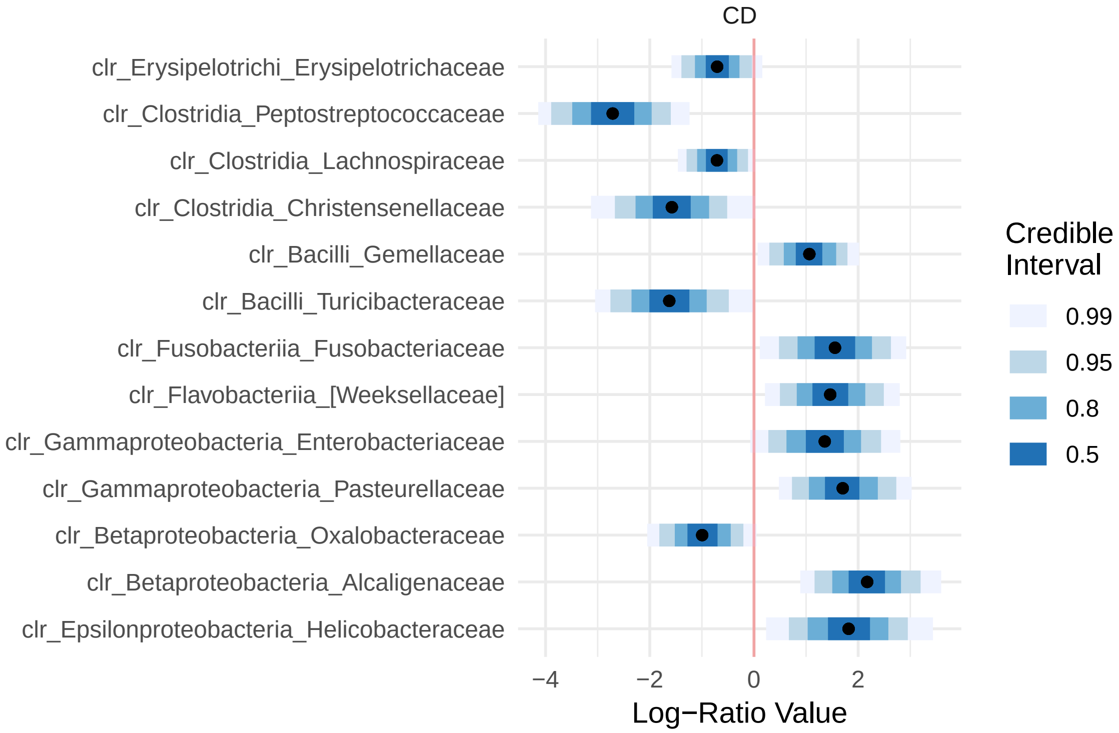

Figure 2: **Posterior mean and credible intervals for Λ of *fido::pibble* (LA Collapsed) applied to Crohn's disease data.** Only the 12 families found to be associated with Crohn's Disease (CD) (*i.e.*, Posterior 95% credible region not covering zero) are shown. Taxa are denoted as clr_[class]_[family]. Λ is represented in centered log-ratio (CLR) coordinates rather than additive log-ratio (ALR) so that each coordinate could be identified with a different bacterial taxa.

Clostridia are depleted in CD. Notably, Fusobacteria has been independently suggested as a marker of IBD (Strauss et al., 2011; Kostic et al., 2012). These findings serve to validate our results and build confidence in our methods.

In contrast, our results differ from prior analyses of this data in certain respects. We find that the family Peptostreptococcaceae is likely decreased in CD versus healthy controls and we find no association for Veillonellaceae. Three factors support our results. First, our analysis accounts for count variation and compositional constraints whereas prior analyses have not. It is well known that the handing of count variation and compositional constraints can have substantial impact on conclusions in the analysis of sequence count data (McMurdie and Holmes, 2014; Silverman et al., 2020; Gloor et al., 2017). Second, Peptostreptococcaceae has been found to be decreased in CD based on the analysis of independent data (Imhann et al., 2018). Third, in visualizing the count data for Peptostreptococcaceae and Veillonellaceae (Figure S8) we find no visual difference in Veillonellaceae but a notable difference in Peptostreptococcaceae. Therefore, we conclude that our approach has revealed novel associations in this data and excluded potentially spurious conclusions.

7. Inferring Microbial Trajectories in an Artificial Gut Model

Artificial gut models provide a powerful *in vitro* approach to studying microbial communities. To demonstrate the generality of our inference methods for Marginally LTP models, we reanalyzed a previously published high-resolution longitudinal study of 4 artificial gut models using a GMGP model (Silverman et al., 2018a). Each of the 4 artificial gut models represent a closed system that were maintained in nearly identical conditions and inoculated with an identical fecal slurry. Following Silverman et al. (2018a), We therefore chose to model each of the four vessels ($r \in 1, \dots, 4$) as independent but with a shared covariance structure using the following GMGP model:

$$\begin{aligned}
Y_{\cdot tr} &\sim \text{Multinomial}(n_{tr}, \pi_{\cdot tr}) \\
\pi_{\cdot tr} &= \text{ALR}_D^{-1}(\eta_{\cdot tr}) \\
\eta_{\cdot tr} &\sim N(\Lambda(X_{\cdot tr}), \Sigma(Z)) \\
\Lambda &\sim \mathsf{GP}(\Theta, \Sigma, \Gamma_{\text{vessel}} \circ \Gamma_{time}) \\
\Sigma &\sim \mathsf{IWP}(\Xi, \nu)
\end{aligned}$$

where Ξ is a kernel based on sequence similarity between bacterial taxa, Γ_{time} is a squared exponential kernel based on the time between samples, Γ_{vessel} is a block identity kernel, and $\circ$ denote the element-wise multiplication of kernel functions. To evaluate the impact of using small values for the degree-of-freedom parameter ν in model priors, we set $\nu = D + 2$. Details on these kernels as well as the matrix functions Θ are described further in Appendix J. The above GMGP model was inferred using the function *basset* from the R package *fido*. While there are differences between the generalized dynamic linear model used in Silverman et al. (2018a) and the above GMGP model, it is notable that the model used in Silverman et al. (2018a) took on the order of 5 hours while the GMGP model above, using CU sampler with Laplace approximation, ran in just 4 seconds.

Our results are in general agreement with those of Silverman et al. (2018a). Notably, we found a distinct decrease in the relative amount of the family Bacteroidaceae immediately after the introduction of a *B. ovatus* probiotic at hour 60. Still, our analyses revealed a number of features unappreciated in prior analyses. Most notably, our GMGP analyses suggests that the degree to which the community was undergoing sub-daily oscillations was far greater than was appreciated in Silverman et al. (2018a). Silverman et al. (2018a) noticed that the relative amount of Enterobacteriaceae displayed distinct, unsyncronized, sub-daily oscillations in all four artificial gut vessels. We too found evidence of unsyncronized sub-daily oscillations in all four vessels. However, we also found such oscillatory dynamics in numerous other dimensions including the CLR coordinates corresponding to the Lachnospiraceae, Desulfovibrionaceae, and Synergistaceae. We suspect that the flexibility provided by the non-linear GMGP model allowed these oscillatory patterns to be more easily revealed than the random walk dynamics originally modeled in Silverman et al. (2018a).

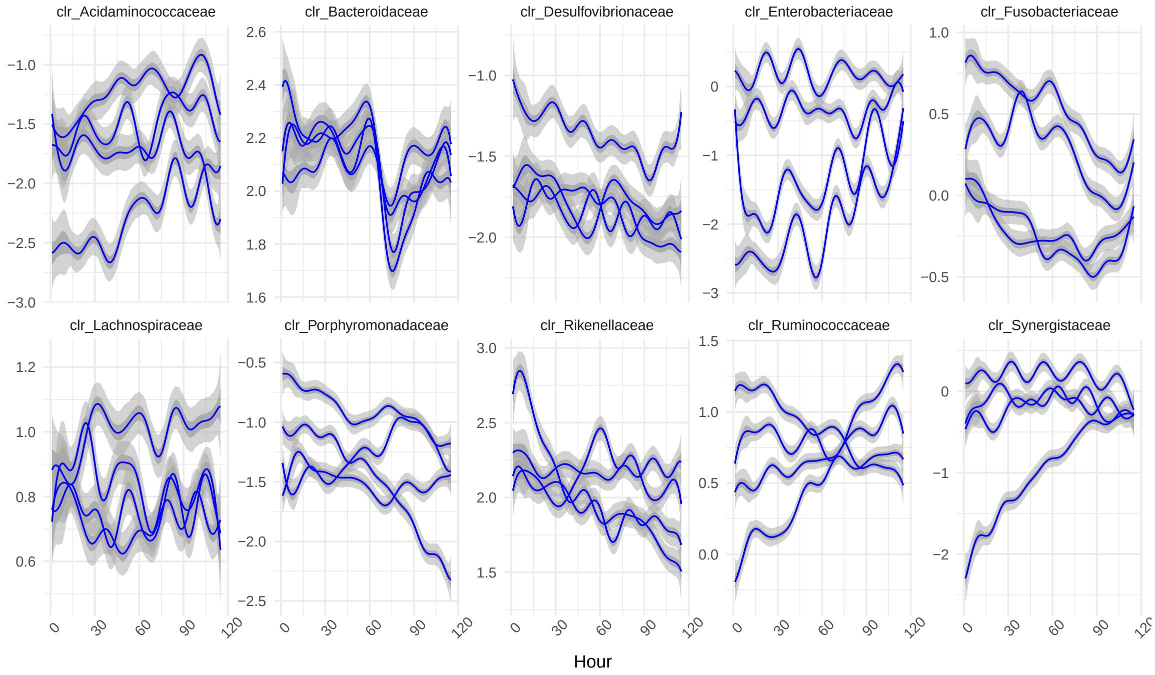

Figure 3: **Posterior mean and credible intervals for Λ of *fido::basset* applied to artificial gut data.** Following Silverman et al. (2018a) we analyze the data from 4 independent artificial guts at the bacterial family level. The posterior mean as well as 50% and 95% credible regions for Λ are depicted. The posterior is depicted with respect to centered log-ratio (CLR) coordinates so that each coordinate could be identified with a different bacterial family.

8. Conclusion

In this work we have developed efficient inference for the analysis of a large class of multinomial logistic-normal models through the use of a shared marginal representation. We demonstrated that, in comparison to HMC, the CU sampler with a marginal Laplace approximation improved sampler efficiency by up to 5 orders of magnitude while preserving accuracy of point estimation and uncertainty quantification. Yet, the performance of our Laplace approximation under observation distributions beyond the log-ratio parameterized multinomial is more uncertain. We hypothesize that our results could generalize to other exponential family distributions parameterized by natural parameters since such distributions are globally log-concave. Yet, we expect that there are other observation distributions where a Laplace approximation to the LTP form may be sub-optimal. Rather than resorting to using MCMC to infer the collapsed model form, we suggest that methods of particle refinement of the initial Laplace approximation (*e.g.*, parallel MCMC steps for each sample from the LTP form, or sequential importance resampling) may be more efficient. We believe such extensions are prime areas for future work.

Here we have compared the CU sampler with marginal Laplace approximation against HMC and VB for inference of MLN models, yet many other comparisons are possible. In particular, both the Integrated Nested Laplace Approximation (INLA) (Rue et al., 2009) and Pólya-gamma data-augmentation (Polson et al., 2013) are popular approaches for inferring Bayesian logistic models. Like INLA, our approach uses Laplace approximations to posterior marginals; yet INLA's requires that the number of hyper-parameters is small (e.g., ≤ 6) which proves limiting in inferring MLN models with potentially dense co variation between multinomial categories as we address here. In contrast to INLA, Pólya-gamma data augmentation uses a Gibbs sampling algorithm with augmented Pólya-gamma random variables. Yet numerous authors have found that Pólya-gamma data augmentation is too slow for scalable inference of MLN models due to two key limitations (Linderman et al., 2015; Glynn et al., 2019; Zhang and Zhou, 2017). First, MLN models do not permit block updates to Pólya-gamma random variables and as a result, the number of Gibbs steps required for each posterior sample scales linearly with the number of multinomial categories (Polson et al., 2013; Linderman et al., 2015; Zhang and Zhou, 2017). Second, when the number of multinomial categories is large, sampling Pólya-gamma random variables can become rate limiting (Glynn et al., 2019). Rather than INLA or Pólya-gamma data augmentation, we believe that the most fruitful comparisons involve alternative approximations for sampling the collapsed representation of Marginally LTP models. Notably, for inference of hierarchical Bayesian Gaussian processes, multiple authors have found expectation propagation to be more accurate, albeit an order of magnitude slower than, Laplace approximation (Jylänki et al., 2011; Nickisch and Rasmussen, 2008; Kuss et al., 2005). Overall, further comparisons will both help to clarify the use cases for the CU sampler with marginal Laplace approximation and point to potential future improvements.

One limitation of this work is that our derived error bound required the assumption that $\nu \to \infty$. This assumption was required so that the Matrix-t distribution was globally log-convex – a requirement of the theory introduced in (Ogden, 2018). In practice however, we expect practitioners to use finite values of ν and in these cases our error bound serves only as a tool for building intuition regarding the error rate of our Laplace approximation. Despite this limitation, our analyses of both simulated and real data suggest that the Laplace approximation provides accurate inference even when ν is small. Still, we expect some degradation of the accuracy of the Laplace approximation for smaller values of ν compared to larger values of ν.

Acknowledgments

JDS was supported in part by the Duke University Medical Scientist Training Program (GM007171). JDS and LAD were supported in part by the Global Probiotics Council, a Searle Scholars Award, the Hartwell Foundation, an Alfred P. Sloan Research Fellowship, the Translational Research Institute through Cooperative Agreement NNX16AO69A, the Damon Runyon Cancer Research Foundation, and NIH 1R01DK116187-01. SM and KR would like to acknowledge the support of grants NSF IIS-1546331, NSF DMS-1418261, NSF IIS-1320357, NSF DMS-1045153, NSF DMS1613261, and NSF DEB-1840223. The authors thank Rachel Silverman, Liat Shenhav, and Shaobo Han for their helpful comments.

Appendix A. Generalized Multivariate Conjugate Linear (GMCL) Models

Here we prove that the GMCL models defined in Equations (6)-(10) are Marginally LTP models, and in so doing, derive their collapsed (LTP) form. Additionally, we demonstrate that uncollapsing the LTP form can be done efficiently. Our proof relies on the following affine transformation property of the matrix normal distribution. Given matrices A, B, and C, as well as a random matrix $X \sim N(M, U, V)$; then for a random matrix $Z = A + BXC$ we have $Z \sim N(A + BMC, BUB^T, C^T VC)$ (Gupta and Nagar, 2018, p. 64).

Proposition 3 *The GMCL Models, as defined in Equations* (6)-(10)*, are a type of Marginally LTP models.*

Proof We prove this proposition by showing that the marginal $p(\eta, Y, X)$ of GMCL models is an LTP. By Definition 2, if $p(\eta, Y, X)$ is an LTP, then $p(\eta, \Lambda, \Sigma, Y, X)$ is a Marginally LTP model.

To begin, we note that equations (8)-(10) can alternatively be written as

$$\eta = \Lambda X + E^\eta \quad E^\eta \sim N(0, \Sigma, I_N) \tag{26}$$

$$\Lambda = \Theta + E^\Lambda \quad E^\Lambda \sim N(0, \Sigma, \Gamma) \tag{27}$$

$$\Sigma \sim IW(\Xi, \upsilon). \tag{28}$$

Using this form, in combination with the affine transformation property of the matrix normal distribution stated above, it is straightforward to marginalize over Λ producing the following form:

$$\begin{aligned} \eta &= \Theta X + E^\Lambda X + E^\eta \quad E^\eta \sim N(0, \Sigma, I_N) \quad E^\Lambda \sim N(0, \Sigma, \Gamma) \\ &= \Theta X + E^* \quad E^\star \sim N(0, \Sigma, I_N + X^T \Gamma X). \end{aligned} \tag{29}$$

Thus we may rewrite Equations (26)-(28) as

$$\eta = \Theta X + E^* \quad E^\star \sim N(0, \Sigma, I_N + X^T \Gamma X) \tag{30}$$

$$\Sigma \sim IW(\Xi, \upsilon). \tag{31}$$

By using the definition of the matrix-t given in Section 3.1 we can marginalize over Σ in Equations (30) and (31) to get

$$\eta \sim T(\upsilon, \Theta X, \Xi, I_N + X^T \Gamma X).$$

Finally, incorporating equations (6) and (7) allows us to write the marginalized form of GMCL models, $p(\eta, Y, X)$, as an LTP

$$\begin{aligned} Y &\sim f(\pi) \\ \pi &= \phi^{-1}(\eta) \\ \eta &\sim T(\upsilon, B, K, A) \end{aligned}$$

where $B = \Theta X$, $K = \Xi$, and $A = I_N + X^T \Gamma X$. ■

Next, we demonstrate that for GMCL models, the conditional posterior $p(\Lambda, \Sigma|\eta, Y, X)$ can be computed and sampled efficiently: That the collapsed model can be uncollapsed efficiently. As Λ and Σ are conditionally independent of Y given η in GMCL models, we may write

$$p(\Lambda, \Sigma \mid \eta, Y, X) = p(\Lambda, \Sigma \mid \eta, X) = p(\Lambda \mid \Sigma, \eta, X)\, p(\Sigma \mid \eta, X).$$

The right hand side of the above equation represents the posterior of a multivariate conjugate linear model that can be sampled efficiently using the following relations (Rossi et al., 2012, p. 32):

$$\begin{aligned}
\upsilon_N &= \upsilon + N \\
\Gamma_N &= (XX^T + \Gamma^{-1})^{-1} \\
\Lambda_N &= (\eta X^T + \Theta\Gamma^{-1})\Gamma_N \\
\Xi_N &= \Xi + (\eta - \Lambda_N X)(\eta - \Lambda_N X)^T + (\Lambda_N - \Theta)\Gamma^{-1}(\Lambda_N - \Theta)^T \\
p(\Sigma|\eta, X) &= IW(\Xi_N, \upsilon_N) \\
p(\Lambda|\Sigma, \eta, X) &= N(\Lambda_N, \Sigma, \Gamma_N).
\end{aligned}$$

Appendix B. Generalized Multivariate Dynamic Linear Model (GMDLM)

Here we prove that the GMDLMs defined in Equations (11)-(16) are Marginally LTP models. Additionally we provide a recursive procedure for uncollapsing an LTP to a GMDLM.

B.1 Derivation of Collapsed Form

Proposition 4 *The GMDLMs defined in Equations* (11)-(16)*, are a type of Marginally LTP models.*

Proof As in Proposition 3, we show that GMDLMs are Marginally LTPs by showing that a marginal of the GMDLMs, $p(\eta, Y)$, is an LTP.

We begin by deriving the marginal distribution $p(\eta, Y)$ in terms of the quantities F_t, G_t, W_t, Σ, M_0 and C_0. As all densities involved are multivariate or matrix-variate normal, the result must also be multivariate or matrix-variate normal and thus fully described by the mean and covariance of η. To derive the mean and covariance we first derive a useful alternative representation of η_t^T with respect to Θ_{t-k-1} for some positive integer $k < t$.

Substituting Equation (14) into Equation (13) allows η_t^T be expressed with respect to Θ_{t-1} as

$$\eta_t^T = F_t^T G_t \Theta_{t-1} + F_t^T \Omega_t + \nu_t^T. \tag{32}$$

Repeated substitution of Θ_{t-k} leads to the following form for η_t^T in terms of Θ_{t-k-1}

$$\eta_t^T = F_t^T \mathcal{G}_{t:t-k} \Theta_{t-k-1} + F_t^T \Omega_t + \sum_{\ell=t}^{t-k-1} F_t^T \mathcal{G}_{t:\ell} \Omega_{\ell-1} + \nu_t^T \tag{33}$$

where $\mathcal{G}_{t:t-k}$ is shorthand for $G_t G_{t-1} \cdots G_{t-k}$. Using the affine transformation property of the matrix normal given in Appendix A in combination with (15) we can marginalize over the random variables $\Omega_t, \ldots, \Omega_1, \nu_t$ in Equation (33) giving

$$\eta_t^T \sim N\left(F_t^T \mathcal{G}_{t:1} M_0, \gamma_t + F_t^T \left[W_t + \sum_{\ell=t}^{2} \mathcal{G}_{t:\ell} W_{\ell-1} \mathcal{G}_{\ell:t}^T + \mathcal{G}_{t:1} C_0 \mathcal{G}_{1:t}^T\right] F_t, \Sigma\right). \tag{34}$$

Next we calculate $Cov(\eta_t^T, \eta_{t-k}^T)$. In parallel to Equation (32) we may write η_{t-k}^T as

$$\eta_{t-k}^T = F_{t-k}^T G_{t-k} \Theta_{t-k-1} + F_{t-k}^T \Omega_{t-k} + \nu_{t-k}^T. \tag{35}$$

Using Equation (33) and (35) along with the fact $Cov(AX_1 + BX_2, Y) = A\,Cov(X_1, Y) + B\,Cov(X_2, Y)$ and that $Cov(\Theta_s, \nu_\ell) = Cov(\Theta_s, \Omega_\ell) = Cov(\Omega_\ell, \nu_s) = 0$ for all s and ℓ, we can write

$$Cov(\eta_t^T, \eta_{t-k}^T) = F_t^T \mathcal{G}_{t:t-k} Var(\theta_{t-k-1}) G_{t-k}^T F_{t-k} + F_t^T \mathcal{G}_{t:t-k+1} Var(\Omega_{t-k}) F_{t-k} \tag{36}$$

where $Var(\Theta_{t-k-1})$ can be written recursively as

$$Var(\Theta_{t-k-1}) = G_{t-k-1} Var(\Theta_{t-k-2}) G_{t-k-1}^T + Var(\Omega_{t-k-1})$$

and where $Var(\Omega_{t-k-1}) = \Sigma \otimes W_{t-k-1}$. Combining this recursive form with equation (36) gives

$$\begin{aligned} Cov(\eta_t^T, \eta_{t-k}^T) = F_t^T \mathcal{G}_{t:t-k+1} (\Sigma \otimes W_{t-k}) F_{t-k} + \sum_{\ell=t-k}^{2} F_t^T \mathcal{G}_{t:\ell} (\Sigma \otimes W_{\ell-1}) G_{\ell:t-k}^T F_{t-k} \\ + F_t^T \mathcal{G}_{t:1} (\Sigma \otimes C_0) G_{1:t-k}^T F_{t-k}. \end{aligned} \tag{37}$$

Together Equations (34) and (37) characterize the marginal distribution of η_t^T in terms of F_t, G_t, W_t, Σ, M_0 and C_0. Noting that if $X \sim N(M, U, V)$ then $X^T \sim N(M^T, V, U)$, it follows that

$$\begin{aligned} \eta &\sim N(B, \Sigma, A) \\ B &= \begin{bmatrix} | & & | & & | \\ \alpha_1 & \cdots & \alpha_t & \cdots & \alpha_T \\ | & & | & & | \end{bmatrix} \\ \alpha_t &= (F_t^T \mathcal{G}_{t:1} M_0)^T \\ A_{t,t-k} &= \begin{cases} \gamma_t + F_t^T \left[W_t + \sum_{\ell=t}^{2} \mathcal{G}_{t:\ell} W_{\ell-1} \mathcal{G}_{\ell:t}^T + \mathcal{G}_{t:1} C_0 \mathcal{G}_{1:t}^T\right] F_t \text{ if } k = 0 \\ F_t^T \left[\mathcal{G}_{t:t-k+1} W_{t-k} + \sum_{\ell=t-k}^{2} \mathcal{G}_{t:\ell} W_{\ell-1} G_{\ell:t-k}^T + \mathcal{G}_{t:1} C_0 G_{1:t-k}^T\right] F_{t-k} \text{ if } k > 0 \end{cases} \end{aligned}$$

Finally, using the marginalization property of the matrix normal and the inverse Wishart used in our definition of the matrix-t distribution and incorporating Equations (11), (12) and (16) it follows that

$$Y \sim f(\pi)$$

$$
\begin{aligned}
\pi &= \phi^{-1}(\eta) \\
\eta &\sim T(\upsilon, B, \Xi, A).
\end{aligned}
$$

■

B.2 Efficient Form for Uncollapsing

Here we provide an efficient means of sampling from the conditional density $p(\Theta, \Sigma \mid \eta, Y)$ for the GMDLM. First we recognize that Θ is conditionally independent of Y given η. Therefore, our task simplifies to sampling from $p(\Theta, \Sigma \mid \eta)$. The problem is identical to the standard filtering and simulation smoothing problem solved by West and Harrison (1997, p. 603-604). Again, the problem is defined by the following model (which we will refer to as the MDLM model)

$$
\begin{aligned}
\eta_t^T &= F_t^T \Theta_t + \nu_t^T, \quad \nu_t \sim N(0, \gamma_t \Sigma) && (38) \\
\Theta_t &= G_t \Theta_{t-1} + \Omega_t, \quad \Omega_t \sim N(0, W_t, \Sigma) && (39) \\
\Theta_0 &\sim N(M_0, C_0, \Sigma) && (40) \\
\Sigma &\sim IW(\Xi, \upsilon). && (41)
\end{aligned}
$$

Following West and Harrison (1997), below we restate the filtering and retrospective recursions needed to sample from $p(\Theta, \Sigma \mid \eta)$. Note that all densities in this subsection are conditional on the parameters F_t, G_t, W_t, Σ, M_0 and C_0 but that this dependence has been suppressed for notational simplicity. Let us introduce υ_t and Ξ_t as filtering parameters at step t. Further, we define $\upsilon_0 = \upsilon$ and $\Xi_0 = \Xi$. As a final piece of notation we introduce H_t^T as a shorthand for the set $\{\eta_t^T, \dots, \eta_1^T\}$

B.2.1 Filtering Recursions for MDLM Model

(1) Posterior at $t-1$:

$$
\begin{aligned}
p(\Sigma \mid H_{t-1}^T) &\sim IW(\Xi_{t-1}, \upsilon_{t-1}) \\
p(\Theta_{t-1} \mid \Sigma, H_{t-1}^T) &\sim N(M_{t-1}, C_{t-1}, \Sigma)
\end{aligned}
$$

(2) Prior at t:

$$
\begin{aligned}
A_t &= G_t M_{t-1} \\
R_t &= G_t C_{t-1} G_t^T + W_t \\
p(\Sigma \mid H_{t-1}^T) &\sim IW(\Xi_{t-1}, \upsilon_{t-1}) \\
p(\Theta_t \mid \Sigma, H_{t-1}^T) &\sim N(A_t, R_t, \Sigma)
\end{aligned}
$$

(3) One-step ahead forecast at t:

$$
\begin{aligned}
f_t^T &= F_t^T A_t \\
q_t &= \gamma_t + F_t^T R_t F_t
\end{aligned}
$$

$$\begin{aligned} p(\Sigma \mid H_{t-1}^T) &\sim IW(\Xi_{t-1}, \upsilon_{t-1}) \\ p(\eta_t \mid \Sigma, H_{t-1}^T) &\sim N(f_t, q_t\Sigma) \end{aligned}$$

(4) Posterior at t:

$$\begin{aligned} e_t^T &= \eta_t^T - f_t^T \\ S_t &= \frac{R_t F_t}{q_t} \\ M_t &= A_t + S_t e_t^T \\ C_t &= R_t - q_t S_t S_t^T \\ \upsilon_t &= \upsilon_{t-1} + 1 \\ \Xi_t &= \Xi_{t-1} + \frac{e_t e_t^T}{q_t} \\ p(\Sigma \mid H_{t-1}^T) &\sim IW(\Xi_t, \upsilon_t) \\ p(\Theta_t \mid \Sigma, H_t^T) &\sim N(m_t, C_t, \Sigma) \end{aligned} \tag{42}$$

Equation (42) differs slightly from the presentation in West and Harrison (1997) as the parameterization of the inverse-Wishart we employ throughout this paper differs from that source. Throughout this work we use the following parameterization for a random matrix $\Sigma \sim IW(\Xi, \upsilon)$:

$$p(\Sigma) \propto |\Sigma|^{-(P+\upsilon+1)/2} \exp\Big(-\frac{1}{2}\mathrm{tr}\big(\Xi\Sigma^{-1}\big)\Big).$$

B.2.2 Simulation Smoothing Recursion

The recursions provided here follow directly from Prado and West (2010, p. 268)
(1) Sample $\Sigma \sim IW(\Xi_T, \upsilon_T)$ and then $\Theta_T \sim N(M_t, C_t, \Sigma)$.
(2) For each time t from $T-1$ to 0, sample $p(\Theta_t|\Theta_{t+1}, H_T^T) \sim N(M_t^*, C_t^*, \Sigma)$ where

$$\begin{aligned} Z_t &= C_t G_{t+1}^T R_{t+1}^{-1} \\ M_t^* &= M_t + Z_t(\Theta_{t+1} - a_{t+1}) \\ C_t^* &= C_t - Z_t R_{t+1} Z_t^T. \end{aligned}$$

Appendix C. Generalized Multivariate Gaussian Process (GMGP) Models

Here we prove that the GMGP models defined in Equations (17)-(21) are marginally LTP models, and in so doing, derive their collapsed (LTP) form. Additionally, we demonstrate that uncollapsing the LTP form can be done efficiently. Finally, we provide a closed form algorithm for predicting and smoothing using GMGP models.

To facilitate this discussion we must first expand our notation to explicitly denote which quantities are inferred via smoothing versus prediction. In the context of Gaussian Processes, smoothing refers to inferring the value of the latent processes Λ and Σ over the finite set (X^o, Z^o) corresponding to observed data (Y). In contrast, prediction refers to inferring the value of the same latent processes over finite sets that do not correspond to observed data (X^u, Z^u). We therefore introduce the following expanded notation:

$$X = \begin{bmatrix} X^o & X^u \end{bmatrix} \tag{43}$$

$$Z = \begin{bmatrix} Z^o & Z^u \end{bmatrix} \tag{44}$$

$$\Sigma = \begin{bmatrix} \Sigma^{oo} & (\Sigma^{uo})^T \\ \Sigma^{uo} & \Sigma^{uu} \end{bmatrix} \tag{45}$$

$$\Lambda = \begin{bmatrix} \Lambda^o & \Lambda^u \end{bmatrix} \tag{46}$$

$$\Gamma = \begin{bmatrix} \Gamma^{oo} & (\Gamma^{uo})^T \\ \Gamma^{uo} & \Gamma^{uu} \end{bmatrix}. \tag{47}$$

Additionally, let $P = P_o + P_u$ and $N = N_o + N_u$ denote the number of dimensions of the observed and unobserved sets, *i.e.*, Σ^{uo} is a $P_u \times P_o$ matrix and Γ^{ou} is an $N_u \times N_o$ matrix.

Proposition 5 *The GMGP models defined by Equations* (17)-(21) *are Marginally LTP models.*

Proof The proof of this proposition follows directly from Proposition 3 noting that the finite evaluation of a GMGP model on any finite sets $X^o = (X_{\cdot 1}, \ldots, X_{\cdot N^o})$ and $Z^o = (Z_{\cdot 1}, \ldots, Z_{\cdot P_o})$ can be written as a GMCL model given the following identifications: $\upsilon^o = \nu + P_o$, $\Xi^{oo} = \Xi(Z^o)$, $\Sigma^{oo} = \mathsf{\Sigma}(Z^o)$ $\Gamma^{oo} = \mathsf{\Gamma}(X^o)$, $\Theta^o = \mathsf{\Theta}(X^o)$, and $\Lambda^o = \mathsf{\Lambda}(X^o)$. With these identifications the GMGP model reduces to the following GMCL model

$$\begin{aligned} Y_{\cdot j} &\sim f(\pi_{\cdot j}) \\ \pi_{\cdot j} &= \phi^{-1}(\eta_{\cdot j}) \\ \eta_{\cdot j} &\sim N(\Lambda^o I_N, \Sigma^{oo}) \\ \Lambda^o &\sim N(\Theta^o, \Sigma^{oo}, \Gamma^{oo}) \\ \Sigma^{oo} &\sim IW(\Xi^{oo}, \upsilon^o) \end{aligned}$$

which – by Proposition 3 – is a Marginally LTP model. ■

Based on the identifications provided in the above proposition, it is straightforward to develop an efficient means of sampling $p(\Lambda^o, \Sigma^{oo} | \eta, Y, X^o, Z^o)$ (uncollapsing the GMGP model):

$$\begin{aligned} \upsilon^o_{N_o} &= \nu + P^o + N_o \\ \Gamma^{oo}_{N_o} &= (I + (\Gamma^{oo})^{-1})^{-1} \\ \Lambda^o_{N_o} &= (\eta + \Theta^o\,(\Gamma^{oo})^{-1})\Gamma^{oo}_{N_o} \\ \Xi^{oo}_{N_o} &= \Xi^{oo} + (\eta - \Lambda^o_{N_o})(\eta - \Lambda^o_{N_o})^T + (\Lambda^o_{N_o} - \Theta^o)\,(\Gamma^{oo})^{-1}\,(\Lambda^o_{N_o} - \Theta^o)^T \\ p(\Sigma^{oo} | \eta, X) &= IW(\Xi^{oo}_{N_o}, \upsilon^o_{N_o}) \\ p(\Lambda^o | \Sigma^{oo}, \eta, X) &= N(\Lambda^o_{N_o}, \Sigma^{oo}, \Gamma^{oo}_{N_o}). \end{aligned}$$

So far we have described GMGP models for inferring the value of the latent stochastic processes $\mathsf{\Lambda}$ and $\mathsf{\Sigma}$ on the finite set corresponding to observed data (smoothing). Next we address the challenge of sampling from the posterior distribution of the latent stochastic process over a finite set corresponding to both observed and unobserved points (simultaneously smoothing and predicting).

As described above, the CU sampler can be used to produce samples of $p(\Lambda^o, \Sigma^{oo}|\eta, Y, X^o, Z^o)$. Conditioned on those samples we now describe a method of sampling $p(\Lambda, \Sigma|X^o, Z^o, Y, X^u, Z^u)$. Letting $\Sigma^{ou/oo} = (\Sigma^{oo})^{-1}(\Sigma^{uo})^T$ and $\Sigma^{uu\cdot oo} = \Sigma^{uu} - \Sigma^{uo}\Sigma^{ou/oo}$, and using the conditional properties of the inverse Wishart distribution (Gupta and Nagar, 2018, pg. 112), we can sample Σ conditioned on Σ^{oo}:

$$\begin{aligned}
\Sigma^{uu\cdot oo} &\sim IW(\Xi^{uu\cdot oo}, \nu + p_o + p_u)\\
\Sigma^{ou/oo}|\Sigma^{uu\cdot oo} &\sim N(\Xi^{ou/oo}, (\Xi^{oo})^{-1}, \Sigma^{uu\cdot oo})\\
\Sigma &= \begin{bmatrix} (\Sigma^{oo})^{-1} + \Sigma^{ou/oo}(\Sigma^{uu\cdot oo})^{-1}\left(\Sigma^{ou/oo}\right)^T & -\Sigma^{ou/oo}(\Sigma^{uu\cdot oo})^{-1}\\ -(\Sigma^{uu\cdot oo})^{-1}\left(\Sigma^{ou/oo}\right)^T & (\Sigma^{uu\cdot oo})^{-1}\end{bmatrix}.
\end{aligned}$$

Finally, we can sample Λ^u conditioned on Λ^o and Σ:

$$\begin{aligned}
\Gamma^{ou/oo} &= (\Gamma^{oo})^{-1}\Gamma^{ou}\\
\Gamma^{uu\cdot oo} &= \Gamma^{uu} - \Gamma^{uo}\Gamma^{ou/oo}\\
M &= \Theta^u + (\Lambda^o - \Theta^o)\Gamma^{ou/oo}\\
\Lambda^u|\Lambda^o, \Sigma &\sim N(M, \Sigma, \Gamma^{uu\cdot oo}).
\end{aligned}$$

Appendix D. Gradient and Hessian Calculations for the Matrix-T Distribution

Here we are concerned with calculating the gradient and Hessian of

$$\log p(\eta) \propto -\frac{\upsilon + N + P - 1}{2}\log\left|I_P + K^{-1}(\eta - B)A^{-1}(\eta - B)^T\right|.$$

Letting $S = I_P + K^{-1}(\eta - B)A^{-1}(\eta - B)^T$ we concern ourselves with calculating the quantities $\frac{d\log|S|}{d\text{vec}(\eta^T)}$ and $\frac{d\log|S|}{\text{vec}(d\eta)\text{vec}(d\eta)^T}$. We will use the identity $d\log|S| = Tr(S^{-1}dS)$ from matrix calculus (Minka, 2000, pg. 1):

$$\begin{aligned}
d\log|S| &= Tr(S^{-1}dS)\\
dS &= d(I_P + K^{-1}(\eta - B)A^{-1}(\eta - B)^T)\\
&= d(K^{-1}(\eta A^{-1}\eta^T - \eta A^{-1}B^T - BA^{-1}\eta^T))\\
&= K^{-1}(d\eta A^{-1}\eta^T + \eta A^{-1}d\eta^T - d\eta A^{-1}B^T - BA^{-1}d\eta^T)\\
&= K^{-1}(d\eta(A^{-1}\eta^T - A^{-1}B^T) + (\eta A^{-1} - BA^{-1})d\eta^T)\\
&= K^{-1}(d\eta C + C^T d\eta^T)
\end{aligned}$$

where in the last line we have defined the $N \times P$ matrix $C = A^{-1}(\eta^T - B^T)$. Further simplifying and using the identities $Tr(A) = Tr(A^T)$ and $Tr(AB) = Tr(BA)$ for matrices A and B we get

$$d\log|S| = Tr(S^{-1}K^{-1}(d\eta C + C^T d\eta^T))$$

$$\begin{aligned}
&= Tr(S^{-1}K^{-1}d\eta C) + Tr(S^{-1}K^{-1}C^T d\eta^T) \\
&= Tr(CS^{-1}K^{-1}d\eta) + Tr(CK^{-1}S^{-T}d\eta) \\
&= Tr(C(S^{-1}K^{-1} + K^{-1}S^{-T})d\eta) \\
&= \text{vec}([C(S^{-1}K^{-1} + K^{-1}S^{-T})]^T)^T \text{vec}(d\eta) \\
d\log|S| &= \text{vec}((S^{-1}K^{-1} + K^{-1}S^{-T})C^T)^T \text{vec}(d\eta) \qquad (48) \\
\frac{d\log|S|}{\text{vec}(d\eta)} &= \text{vec}((S^{-1}K^{-1} + K^{-1}S^{-T})C^T)^T
\end{aligned}$$

The Hessian $H = \frac{d^2 \log|S|}{\text{vec}(d\eta)\text{vec}(d\eta)^T}$ can then be calculated from equation (48) by taking the differential again and manipulating the result into the following canonical form $d^2 \log|S| = \text{vec}(d\eta)^T H \text{vec}(d\eta)$. In particular we make use of the following identities $\text{vec}(ABC) = (C^T \otimes A)\text{vec}(B)$ and $d(S^{-1}) = -S^{-1}dSS^{-1}$. We also make use of the vec-transposition matrix defined by $T_{m,n}\text{vec}(A) = \text{vec}(A^T)$ where A is an $m \times n$ matrix and $T_{m,n}$ is an $mn \times mn$ permutation matrix. The vec-transposition matrix also satisfies the following properties $T_{m,n} = T_{n,m}^T = T_{n,m}^{-1}$. Therefore we can write:

$$\begin{aligned}
d^2\log|S| &= \text{vec}((S^{-1}K^{-1} + K^{-1}S^{-T})dC^T + d(S^{-1})K^{-1}C^T + K^{-1}d(S^{-T})C^T)^T \text{vec}(d\eta) \\
&= \left[\text{vec}((S^{-1}K^{-1} + K^{-1}S^{-T})dC^T)^T + \text{vec}(d(S^{-1})K^{-1}C^T)^T + \text{vec}(K^{-1}d(S^{-T})C^T)^T\right] \text{vec}(d\eta) \\
&= [\#1 + \#2 + \#3]\,\text{vec}(d\eta) \\
\#1 &= \text{vec}((S^{-1}K^{-1} + K^{-1}S^{-T})d\eta A^{-1})^T \\
&= ((A^{-1} \otimes (S^{-1}K^{-1} + K^{-1}S^{-T}))\text{vec}(d\eta))^T \\
&= \text{vec}(d\eta)^T (A^{-1} \otimes (S^{-1}K^{-1} + K^{-1}S^{-T}))^T \\
\#2 &= -\text{vec}(S^{-1}dSS^{-1}K^{-1}C^T)^T \\
&= -((CK^{-1}S^{-T} \otimes S^{-1})\text{vec}(dS))^T \\
&= -\text{vec}(dS)^T (S^{-1}K^{-1}C^T \otimes S^{-T}) \\
\text{vec}(dS)^T &= \text{vec}(K^{-1}(d\eta C + C^T d\eta^T))^T \\
&= \text{vec}(K^{-1}d\eta C)^T + \text{vec}(K^{-1}C^T d\eta^T)^T \\
&= ((C^T \otimes K^{-1})\text{vec}(d\eta))^T + ((I_{D-1} \otimes K^{-1}C^T)\text{vec}(d\eta^T))^T \\
&= \text{vec}(d\eta)^T (C \otimes K^{-1}) + \text{vec}(d\eta^T)^T (I_P \otimes CK^{-1}) \\
\#2 &= [-\text{vec}(d\eta)^T (C \otimes K^{-1}) - \text{vec}(d\eta^T)^T (I_P \otimes CK^{-1})](S^{-1}K^{-1}C^T \otimes S^{-T}) \\
&= -\text{vec}(d\eta)^T (CS^{-1}K^{-1}C^T \otimes K^{-1}S^{-T}) - \text{vec}(d\eta^T)^T (S^{-1}K^{-1}C^T \otimes CK^{-1}S^{-T}) \\
&= -\text{vec}(d\eta)^T (CS^{-1}K^{-1}C^T \otimes K^{-1}S^{-T}) - \text{vec}(d\eta)^T T_{N,P} (S^{-1}K^{-1}C^T \otimes CK^{-1}S^{-T}) \\
\#3 &= \text{vec}(K^{-1}d(S^{-T})C^T)^T \\
&= -\text{vec}(K^{-1}S^{-T}dS^T S^{-T}C^T) \\
&= -((CS^{-1} \otimes K^{-1}S^{-T})\text{vec}(dS^T))^T \\
&= -\text{vec}(dS^T)^T (S^{-T}C^T \otimes S^{-1}K^{-1}) \\
\text{vec}(dS^T)^T &= \text{vec}((d\eta C + C^T d\eta^T)^T K^{-1})^T
\end{aligned}$$

$$\begin{aligned}
&= ((K^{-1}C^T \otimes I_P)\text{vec}(d\eta))^T + ((K^{-1} \otimes C^T)\text{vec}(d\eta^T))^T \\
\#3 &= [-\text{vec}(d\eta)^T(CK^{-1} \otimes I_P) - \text{vec}(d\eta^T)^T(K^{-1} \otimes C)](S^{-T}C^T \otimes S^{-1}K^{-1}) \\
&= -\text{vec}(d\eta)^T(CK^{-1}S^{-T}C^T \otimes S^{-1}K^{-1}) - \text{vec}(d\eta)^T T_{N,D-1}(K^{-1}S^{-T}C^T \otimes CS^{-1}K^{-1}) \\
d^2 \log|S| &= \text{vec}(d\eta)^T[(A^{-1} \otimes (S^{-1}K^{-1} + K^{-1}S^{-T}))^T - (CS^{-1}K^{-1}C^T \otimes K^{-1}S^{-T}) \\
&\quad - (CK^{-1}S^{-T}C^T) \otimes S^{-1}K^{-1}) \\
&\quad - T_{N,D-1}((S^{-1}K^{-1}C^T \otimes CK^{-1}S^{-T}) + (K^{-1}S^{-T}C^T \otimes CS^{-1}K^{-1}))]\text{vec}(d\eta)
\end{aligned}$$

Summarizing the above results we obtain

$$\begin{aligned}
S &= I_P + K^{-1}(\eta - B)A^{-1}(\eta - B)^T \\
C &= A^{-1}(\eta - B)^T \\
R &= S^{-1}K^{-1} \\
\frac{d \log|S|}{\text{vec}(d\eta)} &= \text{vec}((R + R^T)C^T)^T \\
L &= (CRC^T \otimes R^T) \\
\frac{d^2 \log|S|}{\text{vec}(d\eta)\text{vec}(d\eta)^T} &= (A^{-1} \otimes (R + R^T)) - (L + L^T) - T_{N,D-1}[(RC^T \otimes CR^T) + (R^TC^T \otimes CR)].
\end{aligned}$$

Finally, we note a computational trick which makes evaluation of this Hessian far more computationally efficient. We may quickly calculate $T_{m,n}X = X^*$ for an $m \times m$ matrix X having already computed X by noting that for $i \in 1 \dots m$ and $j \in 1 \dots n$ we can write $X^*_{(i-1)n+j,\cdot} = X_{(j-1)m+i,\cdot}$ where $X_{l,\cdot}$ denotes the l-th row of the matrix X.

Appendix E. Gradients and Hessians for the Log-Ratio Parameterized Multinomial

Unfortunately we cannot provide a general form for the gradient and Hessian of all possible likelihoods $f(Y \mid \phi^{-1}(\eta))$. For the purposes of this article, here we derive the gradient and Hessian for the case where f is multinomial and ϕ^{-1} is the inverse ALR transform:

$$\sum_j \log \text{Multinomial}(Y_{\cdot j} \mid n_j, \text{ALR}_D^{-1}(\eta_{\cdot j}))$$

which for notational simplicity we will refer to as g. Thus our goal is to find efficient forms for calculating g, $\frac{dg}{d\text{vec}(\eta)}$ and $\frac{d^2 g}{d\text{vec}(\eta)d\text{vec}(\eta)^T}$. Using the fact that $\log \text{Multinomial}(Y_{\cdot j} \mid n_j, \pi_{\cdot j}) \propto Y_{1j} \log \pi_{1j} + \cdots + Y_{Dj} \log \pi_{Dj}$ and Equation (1) we can write

$$g = \sum_{j=1}^{N} \left(\sum_{i=1}^{D-1} \eta_{ij} Y_{ij} - n_j \log \left(1 + \sum_{i=1}^{D-1} e^{\eta_{ij}} \right) \right).$$

Differentiating with respect to η_{ij} gives

$$\frac{dg}{d\eta_{ij}} = Y_{ij} - n_j \frac{e^{\eta_{ij}}}{1 + \sum_i e^{\eta_{ij}}}.$$

Differentiating again with respect to $\eta_{k\ell}$ gives

$$\frac{d^2 g}{d\eta_{ij} d\eta_{k\ell}} = \begin{cases} -n_j \left(\frac{e^{\eta_{ij}}}{1+\sum_i e^{\eta_{ij}}} - \frac{e^{2\eta_{ij}}}{(1+\sum_i e^{\eta_{ij}})^2}\right) & \text{if } \ell = j, i = k \\ n_j \left(\frac{e^{\eta_{ij}} e^{\eta_{kj}}}{(1+\sum_i e^{\eta_{ij}})^2}\right) & \text{if } \ell = j, i \neq k \\ 0 & \text{if } \ell \neq j. \end{cases}$$

These results directly imply the following matrix forms.

$$\begin{aligned} O &= \exp \eta \\ m &= 1_N + O^T 1_{D-1} \\ \rho &= \operatorname{vec}(O) \oslash \operatorname{vec}(1_{D-1} m^T) \\ n &= 1_D^T Y \\ g &= -\operatorname{vec}(\eta)^T \operatorname{vec}(Y_{/D\cdot}) - n \odot \log(m) \\ \frac{dg}{d\operatorname{vec}(\eta)} &= (\operatorname{vec}(Y_{/D\cdot}) - \operatorname{vec}(1_D n) \odot \rho)^T \\ W^{(j)} &= n_j (\rho_{(j)} \rho_{(j)}^T - \operatorname{diag}(\rho_{(j)})) \\ \frac{d^2 g}{d\operatorname{vec}(\eta) d\operatorname{vec}(\eta)^T} &= \operatorname{diag}\left(W^{(1)}, \dots, W^{(N)}\right) \end{aligned}$$

where $\exp X$ and $\log X$ refers to the element-wise exponentiation and logarithm of a matrix X, $\odot$ and $\oslash$ refer to element-wise product and division respectively, $Y_{/D\cdot}$ refers to the first $D-1$ rows of the matrix Y, $\rho_{(j)}$ denotes elements $(j-1)(D-1)+1$ to $j(D-1)$ in the vector ρ, and $\operatorname{diag}(X_1, \dots, X_D)$ refers to a block diagonal matrix where the i-th block is X_i.

Appendix F. Implementing the Laplace Approximation to an LTP

Implementing this Laplace approximation for an LTP requires three steps: finding the MAP estimates for η using optimization; calculating the hessian at the MAP estimate, and then sampling from the approximating normal distribution.

F.1 Finding the MAP estimate

The MAP estimate for η (denoted $\hat{\eta}$) is defined as the solution to the following optimization problem:

$$\hat{\eta} = \underset{\eta \in \mathcal{R}^{P \times N}}{\operatorname{argmin}} \left[-\log p(\eta \mid Y)\right] \tag{49}$$

where $\log p(\eta|Y)$ is the sum of the log-matrix-t prior and log-likelihood densities as shown in Equation (23):

$$-\log p(\eta \mid Y) \propto -\log f(Y \mid \phi^{-1}(\eta)) - p(\eta).$$

The form of $p(\eta)$ is given in Appendix D. In contrast, the form of $\log f(Y \mid \phi^{-1}(\eta))$ depends on the choice of likelihood (f) and link function (ϕ). When f and ϕ are given respectively by the multinomial and ALR transformation, the resulting form of $\log f(Y \mid \phi^{-1}(\eta))$ can be found in Appendix E.

As we expect the dimension of η to be large in most applications, we recommend using gradient based optimization methods such as L-BFGS over methods that require repeated calculation or inversion of a hessian matrix such as Newton-Raphson (Sun et al., 2019). Beyond calculating $\log p(\eta \mid Y)$, gradient based optimization additionally requires calculating the gradient $-\frac{d \log p(\eta|Y)}{d\text{vec}(\eta)}$, which is given by Equation (24):

$$-\frac{d \log p(\eta \mid Y)}{d\text{vec}(\eta)} = -\frac{d \log f(Y \mid \phi^{-1}(\eta))}{d\text{vec}(\eta)} - \frac{d \log p(\eta)}{d\text{vec}(\eta)}.$$

The form of $\frac{d \log p(\eta)}{d\text{vec}(\eta)}$ is given in Appendix D. For added computational efficiency when $N < P$, we provide an alternative method of calculating the gradient $\frac{d \log p(\eta)}{d\text{vec}(\eta)}$ using Sylvester's determinant identity in Appendix G. In contrast, the form of $\frac{d \log f(Y|\phi^{-1}(\eta))}{d\text{vec}(\eta)}$ depends on the choice of likelihood (f) and link function (ϕ). When f and ϕ are given respectively by the multinomial and ALR transformation, the resulting form of $\frac{d \log f(Y|\phi^{-1}(\eta))}{d\text{vec}(\eta)}$ can be found in Appendix E.

F.2 Calculating the hessian at the MAP estimate

Once the MAP estimate $\hat{\eta}$ has been found, the hessian H at the MAP estimate (denoted $H(\text{vec}\,\hat{\eta})$ can be calculated using Equation (25):

$$H = \frac{d^2 \log f(Y \mid \phi^{-1}(\eta))}{d\text{vec}(\eta) d\text{vec}(\eta)^T} + \frac{d^2 \log p(\eta)}{d\text{vec}(\eta) d\text{vec}(\eta)^T}.$$

The form of $\frac{d^2 \log f(Y|\phi^{-1}(\eta))}{d\text{vec}(\eta) d\text{vec}(\eta)^T}$ is given in Appendix D. In contrast, the form of $\frac{d^2 \log f(Y|\phi^{-1}(\eta))}{d\text{vec}(\eta) d\text{vec}(\eta)^T}$ depends on the choice of likelihood (f) and link function (ϕ). When f and ϕ are given respectively by the multinomial and ALR transformation, the resulting form of $\frac{d^2 \log f(Y|\phi^{-1}(\eta))}{d\text{vec}(\eta) d\text{vec}(\eta)^T}$ can be found in Appendix E.

F.3 Sampling from the approximating normal distribution

The Laplace approximation to the density $p(\eta|Y)$ is defined as $q(\eta|Y) = N(\text{vec}\,\hat{\eta}, H^{-1}(\text{vec}\,\hat{\eta}))$. While there are numerous ways of sampling from $q(\eta|Y)$ explicit inversion of H^{-1} can be avoided using a Cholesky decomposition. Letting U denote the upper Cholesky factor of the matrix $H^{-1}(\text{vec}\,\hat{\eta})$ such that $H^{-1}(\text{vec}\,\hat{\eta}) = U^T U$ we may sample a random variable $\text{vec}\,\eta^{(s)} \sim q(\eta|Y)$ by first sampling a vector of standard normal variables z and then transforming that sample as:

$$\text{vec}\,\eta^{(s)} = \text{vec}\,\hat{\eta} + U^{-1} z.$$

In practice, it is often more computationally efficient to directly calculate $U^{-1}z$ by backsolving rather than directly computing the term U^{-1}.

Appendix G. Accelerated Matrix-T Gradients via Sylvester's Determinant Identity

Sylvester's determinant identity states that for matrices A and B of size $m \times n$ and $n \times m$ respectively, $|I_m + AB| = |I_n + BA|$. This relationship can be used to speed up calculation of the log-likelihood and gradient of the matrix-t distribution when $N < P$ as the the log determinant or inverse of the matrix S can dominate computational time. To take advantage of this speed up we note that we can replace the relations given in Appendix D with

$$\begin{aligned}
S &= I_N + A^{-1}(\eta - B)^T K^{-1}(\eta - B)\\
C &= K^{-1}(\eta - B)\\
R &= S^{-1}A^{-1}\\
\frac{d\log|S|}{d\text{vec}(\eta)} &= \text{vec}(C(R + R^T))^T.
\end{aligned}$$

While this result can greatly accelerate inference for matrix-t gradients when $P \gg N$, this result provides only minimal improvement for calculating the corresponding Hessian terms. Therefore we suggest that, for simplicity, the Hessian form provided in Appendix D be used even if $P \gg N$.

Appendix H. Simulations and Model Fitting

To compare the performance of the multiple multinomial logistic-normal linear model implementations described in Section 5 over a range of sample sizes (N), observation dimensions (D), and covariate dimensions (Q), we created the following simulation scheme. For each evaluated triple (N, D, Q), three simulated data-sets were created based on the multinomial logistic-normal linear model with the following specified likelihood:

$$\begin{aligned}
Y_{\cdot j} &= \text{Multinomial}(n_j, \pi_{\cdot j})\\
\pi_{\cdot j} &= \text{ALR}_D^{-1}(\eta_{\cdot j})\\
\eta_{\cdot j} &= N(\Lambda X_{\cdot j}, \Sigma).
\end{aligned}$$

Additionally X, Λ, and Σ were simulated as

$$\begin{aligned}
\Lambda &\sim N(0, I, I)\\
\Sigma &\sim IW(I, D + 10)\\
X &\sim N(0, I, I).
\end{aligned}$$

The percent of zero counts naturally increased with large D or large Q relative to other parameters. We took advantage of this behavior to study the performance of all implementations in sparse data regimes.

For all model fits, priors parameter values for υ, Ξ, and Θ and values for hyperparameter Γ were chosen as their simulated values. All implementations were compiled and run using gcc version 6.2.0, R version 3.4.2, and Intel(R) Math Kernel Library version 2019 where possible. All replicates of the simulated count data were supplied to the various implementations independently and the models were fit on identical hardware, allotted 64GB RAM, 4 cores, and restricted to a 48-hour upper limit on run-time.

Appendix I. Priors for Crohn's Disease Data

Sequence count data was obtained from the R package MicrobeDS (github.com/twbattaglia/MicrobeDS). Only samples from the terminal ileum from healthy donors and patients with Crohn's Disease, who had no recent history of steroids, antibiotics, or biologics were included in the analysis. Samples with a sequencing depth below 5000 counts were excluded from analyses. Only families seen with at least 3 counts in at least 10% of samples were retained for subsequent analyses.

The regression model required that 4 hyper-parameters Γ, Θ, Ξ, and υ be specified. We set Θ to a $D \times Q$ matrix of zeros representing our prior assumption that, on average, there was no association between each covarariate and microbial composition. We specified $\Gamma = I_Q$ to constrain associations between microbial composition and covariates to remain small. We specified $\upsilon = D + 3$ and $\Xi_{ij} = (\upsilon - D)$ if $i = j$ and $\Xi_{ij} = (\upsilon - D)/2$ if $i \neq j$ to reflect our weak prior assumption that the log absolute abundance of each taxa is uncorrelated (Aitchison, 1986, p. 208-214).

Appendix J. Priors for Artificial Gut Data

Sequence count data was obtained from the R package Fido (github.com/jsilve24/fido). Only samples from the high-resolution hourly sampling period were included in the analysis.

The the developed GMGP model required that 4 hyper-parameters be specified: Θ, $\mathsf{\Gamma}_{\text{time}}$,$\mathsf{\Gamma}_{\text{vessel}}$, Ξ, and ν. Per the default in *fido*, these hyper-parameters were specified with respect to ALR$_D$ coordinates. We specified $\mathsf{\Theta} = \mathbf{0}$ which centered our prior about the neutral element of the simplex. We specified $\mathsf{\Gamma}_{\text{time}}$ as a squared exponential kernel

$$\mathsf{\Gamma}_{\text{time}}(t_i, t_j) = \exp \frac{-|t_j - t_i|^2}{2\rho_t^2}$$

where ρ_t was set to the median temporal distance between samples. To induce independence between vessels, letting r_i denote the vessel sample i was taken from, we specified $\mathsf{\Gamma}_{\text{vessel}}$ as

$$\mathsf{\Gamma}_{\text{vessel}}(r_i, r_j) = \begin{cases} 1 \text{ if } r_i = r_j \\ 0 \text{ otherwise} \end{cases} .$$

Next, following prior reports, we assumed that more evolutionary similar bacterial taxa would behave more similarly (Silverman et al., 2017). We encoded this prior information in a phylogenetic kernel for Ξ as follows. Let h_{ij} denote the Hamming distance between the 16S sequence of taxa i (s_i) and j (s_j). We created a squared exponential kernel based on these distances:

$$\Xi^*(s_i, s_j) = \exp \frac{-h_{ij}^2}{2\rho_s^2}$$

where ρ_s was set as the median hamming distance between sequences. To project the kernel $\Xi^*(s_i, s_j)$ into ALR$_D$ coordinates, we created a kernel Ξ which was specified as the projection of the Gram matrix of Ξ^*. Letting G denote the contrast matrix for ALR$_D$ ($G = [I_{D-1}, -1]$) and letting Ξ^* represent the Gram matrix of the kernel Ξ^* and Ξ represent the Gram matrix of the kernel Ξ, we specified the kernel Ξ implicitly as

$$\Xi = \text{corr}(G\Xi^* G^T)$$

where corr represents the normalized correlation matrix corresponding to a symmetric positive definite matrix. Finally, we specified $\nu = D + 2$ to reflect the fact that our prior knowledge regarding Ξ was weak.

Appendix K. Laplace Approximation Error

In this section we provide an analysis of the error rate of Laplace approximation used in Section 4.2.

Given an integral of the form

$$L = \int_{R^d} e^{-g(u)} du \tag{50}$$

Ogden (2018) examined conditions and rates for order-k Laplace approximations of the above integral. There were two regularity and convexity conditions required in the analysis.

Condition 1 *$g(\cdot)$ is a smooth function with a unique minimum.*

Condition 2 *Denote the unique minimum of $g(\cdot)$ as $\hat{u}$ and H_{ij} as the ij-th element of the Hessian of g evaluated at $\hat{u}$. For a collection of normalizing terms $\alpha_1, ..., \alpha_d > 0$ the normalized derivatives are*

$$k_{ij} = \frac{h_{ij}}{\alpha_i^{1/2}\alpha_j^{1/2}}.$$

For a $d \times d$ matrix A we denote $A = O_p^*(1)$ if for each $i, j \in 1, ..., d$ $\sum_j |A_{ij}| = O_p(1)$ and $\sum_i |A_{ij}| = O_p(1)$.

The second condition is there exist normalizing terms $\alpha_1, ..., \alpha_d$ such that $k^{-1} = O_p^*(1)$.

The main result we will use is (Ogden, 2018, Theorem 1) which states if Condition 1 and 2 are met then the error rate of the order-1 Laplace approximation to L is given by $\epsilon = O_p(\sum_{j=1}^d \alpha_j^{-1})$ where α_j are determined by Condition 2.

To prove the error rate of our Laplace approximation we will need the following Lemmas.

Lemma 6 *Let $\lambda_{\min}(X)$ denote the minimum eigenvalue of a matrix X. Assuming A and B denote Hermitian matrices. Then $\lambda_{\min}(A + B) \geq \lambda_{\min}(A) + \lambda_{\min}(B)$.*

Proof H is Hermitian so all eigenvalues must be real. For a Hermitian matrix A and non-zero vector x we have $x^T Ax \geq \lambda_{\min}(A)x^T x$ such that $x^T Ax = \lambda_{\min}(A)x^T x$ if and only if $x \in \text{Span}(x_{\min}(A))$ where $x_{\min}(A)$ denotes the set of eigenvectors corresponding to the minimum eigenvalue of A. It therefore follows that

$$\begin{aligned} x^T(A + B)x &= x^T Ax + x^T Bx \\ &\geq (\lambda_{\min}(A) + \lambda_{\min}(B))x^T x \end{aligned}$$

and this minimum bound is achieved if and only if $x \in \text{Span}(x_{\min}(A)) \cap \text{Span}(x_{\min}(B))$. It follows that $\lambda_{\min}(A + B) \geq \lambda_{\min}(A) + \lambda_{\min}(B)$ and equality is achieved only if there exists an x such that $x \in \text{Span}(x_{\min}(A)) \cap \text{Span}(x_{\min}(B))$. ■

Lemma 7 *For Hermitian matrices A and B, $\lambda_{\min}(A \circ B) \geq \lambda_{min}(A)\lambda_{\min}(B)$ where $\circ$ denotes the Hadamard (element-wise) product.*

Proof The Hadamard product $A \circ B$ is a principle sub-matrix of the Kroneker product $A \otimes B$. We may define the principle sub-matrix using a selection matrix E such that we may write $A \circ B = E^T(A \otimes B)E$. Letting $A = UD_AU^T$ and $B = VD_BV^T$ denote the eigen-decompositions of A and B respectively, we can then write

$$\begin{aligned} A \circ B &= E^T(UD_AU^T \otimes VD_BV^T)E \\ &= E^T[(U \otimes V)(D_A \otimes D_B)(U \otimes V)^T]E \end{aligned}$$

Therefore the eigenvalues of $A \circ B$ represent of subset of the eigenvalues of $A \otimes B$. As the eigenvalues of $A \otimes B$ are given by every pairwise product between one eigenvalue of A and one eigenvalue of B it is therefore clear that the minimum eigenvalue of $A \otimes B$ is $\lambda_{\min}(A)\lambda_{\min}(B)$. Therefore $\lambda_{\min}(A \circ B) \geq \lambda_{\min}(A)\lambda_{\min}(B)$. ■

Lemma 8 *For a $d \times d$ symmetric positive definite matrix H, $H = O_p^*(1)$ if $\lambda_{max}(H) = O_p(1)$ where $\lambda_{max}(.)$ denotes the maximum eigenvalue operator.*

Proof If H is symmetric then $H = H^T$ and therefore $\sum_j |H_{ij}| = O_p(1)$ for all $j \in \{1, \dots, d\}$ if and only if $\sum_i |H_{ij}| = O_p(1)$ for all $i \in \{1, \dots, d\}$. For a vector x we have $x^T Hx \leq \lambda_{\max}(H)x^T x$ with equality only if $x \in \text{Span}(x_{\max}(H))$ where $x_{\max}(H)$ denotes the set of eigenvectors corresponding $\lambda_{\max}(H)$. Letting r denote a D-vector with elements defined by $r_j = \sum_j H_{ij}$ it is clear that $r = H1_D$. Therefore, $r^T r = 1_d^T HH1_d \leq d\lambda^2_{\max}(H)$. Thus $r^T r \leq d\lambda^2_{\max}(H)$. As d is a constant and given that $\lambda_{\max}(H) = O_p(1)$ it follows that $r^T r = O_p(1)$. Noting that $r^T r = \sum_i \sum_j |H_{ij}|^2$ we can conclude that $\sum_j |H_{ij}| < r^T r$ for all $j \in \{1, \dots, D\}$. Therefore since $r^T r = O_p(1)$ we must have that $\sum_j |H_{ij}| = O_p(1)$ for all j and therefore, by definition, that $H = O_p^*(1)$. ■

Lemma 9 *For a $d \times d$ symmetric positive definite matrix H, if $\lambda_{\min}(H) = \Omega_p(1)$ then $H^{-1} = O_p^*(1)$, where $\Omega_p(1)$ a stochastic lower-bound of order at least 1.*

Proof Denoting the eigen-decomposition of H as $H = VDV^T$ we can write $H^{-1} = VD^{-1}V^T$ where D^{-1} is a diagonal matrix with elements $D^{-1}_{ii} = 1/D_{ii}$. It follows then $\lambda_{\max}(H^{-1}) = \lambda_{\min}(H)^{-1}$. If H is symmetric positive definite then all eigenvalues of H are positive. Therefore if $\lambda_{\min}(H)$ is lower bounded such that $\lambda_{\min}(H) = \Omega_p(1)$ then we can conclude that $\lambda_{\max}(H^{-1})$ is upper bounded by $\lambda_{\max}(H^{-1}) = O_p(1)$. Using Lemma 8 it follows that $H^{-1} = O_p^*(1)$. ■

Lemma 10 *The function $g(\eta) = \sum_j \log Multinomial(Y_{\cdot j}|n_j, ALR_D^{-1}(\eta_{\cdot j}))$ is strictly concave.*

Proof Let g_j denote the j-th element in the sum such that

$$g_j(\eta_{\cdot j}) = \log \text{Multinomial}(Y_{\cdot j}|n_j, \text{ALR}_D^{-1}(\eta_{\cdot j})).$$

g_j can then be equivalently written as

$$g_j(\eta_{\cdot j}) = \sum_{i=1}^{D-1} \eta_{ij} Y_{ij} - n_j \log\left(1 + \sum_{i=1}^{D-1} e^{\eta_{ij}}\right). \tag{51}$$

As the sum of concave functions is itself concave, our proof relies on showing that each g_j is concave.

Denoting the natural exponential family as $\log p(x|\gamma) \propto \gamma \cdot T(x) - A(\gamma)$ we can see that g_j corresponds to a natural exponential family density with natural parameters $\gamma = \eta$, sufficient statistic $T(x) = Y_{\cdot j}$ and log-partition function $A(\gamma) = n_j \log\left(1 + \sum_{i=1}^{D-1} e^{\eta_{ij}}\right)$. The hessian $\log p(x|\gamma)$ is

$$\frac{d^2 \log p(x|\gamma)}{d\gamma_i d\gamma_j} = -\frac{d^2 A(\gamma)}{d\gamma_i d\gamma_j}.$$

Furthermore, for all natural exponential family densities, the log-partition function $A(\eta)$ is strictly convex Jordan (2010). Therefore the hessian of g_j with elements $d^2 g_j(\eta_{\cdot j})/d\eta_{ij} d\eta_{kj}$ is positive definite for all values of $\eta_{\cdot j}$ and we can therefore conclude that g_j is strictly concave. Since g is the sum of strictly concave function we can conclude that g is strictly concave. ■

Proposition 11 *Let $\mathcal{Y}$ denote the finite realization of a D-dimensional LTP evaluated on an $N \times (D-1)$ finite set such that $\mathcal{Y}$ has the following form:*

$$\begin{aligned} Y_{\cdot j} &\sim \textit{Multinomial}(n_j, \pi_j) \\ \pi_{\cdot j} &= ALR_D^{-1}(\eta_{\cdot j}) \\ \eta &\sim T(\upsilon, B(\delta), K(\delta), A(\delta)). \end{aligned} \tag{52}$$

Assuming that $A(\delta)$ and $K(\delta)$ are symmetric positive definite and do not vary with any n_j. In the limit as $\upsilon \to \infty$ the error for the order-1 Laplace approximation to $\int p_{\mathcal{Y}}(Y, \eta) d\eta$ is $\epsilon = O_p((D-1)\sum_{j=1}^{N} n_j^{-1})$.

Proof Without loss of generality we may redefine $K(\delta) \to \upsilon K(\delta)$ such that (52) can be written as $\eta \sim T(\upsilon, B(\delta), \upsilon K(\delta), A(\delta))$. Given such a form, Theorem 4.3.4 of Gupta and Nagar (2018) proves that as $\upsilon \to \infty$, η converges in distribution to $\eta \sim N(B(\delta), K(\delta), A(\delta))$ such that in the limit we may write $\mathcal{Y}$ as

$$\begin{aligned} Y_i &\sim \text{Multinomial}(n_i, \pi_i) \\ \pi_i &= ALR_D^{-1}(\eta_i) \\ \eta &\sim N(B(\delta), K(\delta), A(\delta)). \end{aligned}$$

In this limit, we may write

$$\begin{aligned}\int p_{\mathcal{Y}}(Y,\eta)d\eta &= \int p(\eta)p(Y|\eta)d\eta \\ &= \int N(\eta|B(\delta),K(\delta),A(\delta))\prod_j \text{Multinomial}(Y_{\cdot j}|n_j,\text{ALR}_D^{-1}(\eta_{\cdot j}))d\eta \\ &= \int \exp\left\{-\log N(\eta|B(\delta),K(\delta),A(\delta)) - \sum_j \log\text{Multinomial}(Y_{\cdot j}|n_j,\text{ALR}_D^{-1}(\eta_{\cdot j}))\right\}d\eta \\ &= \int_{R^{P(D-1)}} \exp\{-g(\eta)\}d\eta\end{aligned}$$

The above integral therefore has the form studied by Ogden (2018) and our goal is to prove that Conditions 1 and 2 hold for some choice of normalizing constants $\alpha_1,\dots,\alpha_{P(D-1)}$.

To show that Condition 1 holds, we must show that $g(\eta)$ is smooth with unique optima. $g(\eta)$ can be represented as a sum $g(\eta) = -(g_N(\eta)+g_M(\eta))$ where

$$\begin{aligned}g_N(\eta) &= \log N(\eta|B(\delta),K(\delta),A(\delta)) \\ g_M(\eta) &= \sum_j \log\text{Multinomial}(Y_{\cdot j}|n_j,\text{ALR}_D^{-1}(\eta_{\cdot j})).\end{aligned}$$

It is clear that $g(\eta)$ is smooth for all $\epsilon \in R^{P(D-1)}$. We prove that $g(\eta)$ has a unique optima by showing that $g(\eta)$ is strictly convex. As $K(\delta)$ and $A(\delta)$ are positive definite it follows from the properties of the matrix-normal that $g_N(\eta)$ is strictly concave. Furthermore, in Lemma 10 we proved that $g_M(\eta)$ is strictly concave. Therefore $g(\eta)$ is the sum of two strictly convex functions and is therefore strictly convex. As $g(\eta)$ is strictly convex it therefore has a single unique optima.

To show that Condition 2 holds we chose normalizing constants $\alpha_{i\times j}$ for $i \in \{1,\dots,P\}$ and $j \in \{1,\dots,D-1\}$ such that $\alpha_{i\times j} = n_j$. We do this by bounding the minimum eigenvalue of the Hessian $h = g''(\hat{\eta})$ and using Lemma 9. Based on the linearity of the derivative operator we can write $h = -(h_N + h_M)$ where h_N and h_M are defined as the Hessian of g_N and g_M respectively evaluated at the optima $\hat{\eta}$. If $A(\delta)$ and $K(\delta)$ do not depend on $n_1,\dots,n_j$ then g_N and therefore h_N has no dependence on $n_1,\dots,n_j$. Noting that $h_N = A(\delta)\otimes K(\delta)$ we can then write

$$k_N = (A(\delta)\otimes K(\delta)) \oslash (\alpha \otimes \alpha^T)$$

where $\oslash$ denotes Hadamard (element-wise) division and α denotes the vector of normalizing constants. Note that $\alpha\otimes\alpha^T$ is strictly positive rank-1 matrix and therefore $\lambda_{\min}(\alpha\otimes\alpha^T)\geq 0$. Similarly, since $A(\delta)$ and $K(\delta)$ are both symmetric positive definite we have that $\lambda_{\min}(A(\delta)\otimes K(\delta))\geq 0$. Noting that $A\oslash B = A\circ(1\oslash B)$ we can use Lemma 7 to conclude that $\lambda_{min}(k_N)\geq 0$. Moving onto h_M, we use results in Appendix E to represent h_M as a block diagonal matrix

$$h_M = \text{diag}\left(n_1C^{(1)},\dots,n_NC^{(N)}\right).$$

It follows that

$$k_M = \text{diag}\left(C^{(1)},\dots,C^{(N)}\right)$$

where the blocks $C^{(j)}$ are $(D-1) \times (D-1)$ symmetric positive definite matrices of full rank. Therefore the minimum eigenvalue of k_M is greater than zero and does not vary with $n_1, \dots, n_N$. We now have shown that $\lambda_{\min}(k_N) \geq 0$ and $\lambda_{\min}(k_H) > 0$, the latter we have also shown has no dependence on $n_1, \dots, n_N$. Combining these results with Lemma 6 we can conclude that $\lambda_{\min}(k) \geq c > 0$ where c is a constant defined by $c = \lambda_{\min}(k_M)$. It follows that $\lambda_{\min}(k) = \Omega_p(c) = \Omega_p(1)$ and therefore from Lemma 9 that $k^{-1} = O_p^*(1)$.

We now have shown that $\int p_{\mathcal{Y}}(Y, \eta) d\eta$ is of the form (50), that Condition 1 and Condition 2 hold with normalizing constants $\alpha_{i \times j}$ for $i \in \{1, \dots, P\}$ and $j \in \{1, \dots, D-1\}$ such that $\alpha_{i \times j} = n_j$. Therefore we have $\epsilon = O_p((D-1)\sum_{j=1}^{N} n_j^{-1})$. ■

References

Tarmo Äijö, Christian L Müller, and Richard Bonneau. Temporal probabilistic modeling of bacterial compositions derived from 16s rRNA sequencing. *Bioinformatics*, 34(3):372–380, 2017.

J. Aitchison. *The statistical analysis of compositional data*. Monographs on statistics and applied probability. Chapman and Hall, London ; New York, 1986. ISBN 0412280604 (U.S.).

J. Aitchison and S. M. Shen. Logistic-normal distributions - some properties and uses. *Biometrika*, 67(2):261–272, 1980. ISSN 0006-3444. doi: Doi10.2307/2335470.

Dean Billheimer, Peter Guttorp, and William F Fagan. Statistical interpretation of species composition. *Journal of the American Statistical Association*, 96(456):1205–1214, 2001.

David M Blei and John D Lafferty. Dynamic topic models. In *Proceedings of the 23rd international conference on Machine learning*, pages 113–120. ACM, 2006.

Claudia Cargnoni, Peter Müller, and Mike West. Bayesian forecasting of multinomial time series through conditionally Gaussian dynamic models. *Journal of the American Statistical Association*, 92(438):640–647, 1997. ISSN 0162-1459.

Andrew Gelman, Daniel Lee, and Jiqiang Guo. Stan: A probabilistic programming language for Bayesian inference and optimization. *Journal of Educational and Behavioral Statistics*, 40(5):530–543, 2015. ISSN 1076-9986.

Dirk Gevers, Subra Kugathasan, Lee A Denson, Yoshiki Vázquez-Baeza, Will Van Treuren, Boyu Ren, Emma Schwager, Dan Knights, Se Jin Song, Moran Yassour, et al. The treatment-naive microbiome in new-onset Crohn's disease. *Cell host & microbe*, 15(3): 382–392, 2014.

Gregory B Gloor, Jean M Macklaim, Vera Pawlowsky-Glahn, and Juan J Egozcue. Microbiome datasets are compositional: and this is not optional. *Frontiers in Microbiology*, 8: 2224, 2017.

Gregory Brian Gloor, Jean M. Macklaim, Michael Vu, and Andrew D. Fernandes. Compositional uncertainty should not be ignored in high-throughput sequencing data analysis. *Austrian Journal of Statistics*, 45(4):73, 2016. doi: 10.17713/ajs.v45i4.122.

Chris Glynn, Surya T Tokdar, Brian Howard, David L Banks, et al. Bayesian analysis of dynamic linear topic models. *Bayesian Analysis*, 14(1):53–80, 2019.

Neal S. Grantham, Brian J. Reich, Elizabeth T. Borer, and Kevin Gross. Mimix: a Bayesian mixed-effects model for microbiome data from designed experiments. *ArXiv e-prints*, 1703:arXiv:1703.07747, 2017.

Arjun K Gupta and Daya K Nagar. *Matrix variate distributions*. Chapman and Hall/CRC, 2018.

Floris Imhann, Arnau Vich Vila, Marc Jan Bonder, Jingyuan Fu, Dirk Gevers, Marijn C Visschedijk, Lieke M Spekhorst, Rudi Alberts, Lude Franke, Hendrik M Van Dullemen, et al. Interplay of host genetics and gut microbiota underlying the onset and clinical presentation of inflammatory bowel disease. *Gut*, 67(1):108–119, 2018.

Michael Jordan. The exponential family: Basics, 2010. URL `https://people.eecs.berkeley.edu/~jordan/courses/260-spring10/other-readings/chapter8.pdf`.

Luke Jostins, Stephan Ripke, Rinse K Weersma, Richard H Duerr, Dermot P McGovern, Ken Y Hui, James C Lee, L Philip Schumm, Yashoda Sharma, Carl A Anderson, et al. Host–microbe interactions have shaped the genetic architecture of inflammatory bowel disease. *Nature*, 491(7422):119, 2012.

Pasi Jylänki, Jarno Vanhatalo, and Aki Vehtari. Robust Gaussian process regression with a Student-t likelihood. *Journal of Machine Learning Research*, 12(Nov):3227–3257, 2011.

Robert E Kass and Duane Steffey. Approximate Bayesian inference in conditionally independent hierarchical models (parametric empirical Bayes models). *Journal of the American Statistical Association*, 84(407):717–726, 1989.

Chiea Chuen Khor and Martin L Hibberd. Host–pathogen interactions revealed by human genome-wide surveys. *Trends in Genetics*, 28(5):233–243, 2012.

Aleksandar D Kostic, Dirk Gevers, Chandra Sekhar Pedamallu, Monia Michaud, Fujiko Duke, Ashlee M Earl, Akinyemi I Ojesina, Joonil Jung, Adam J Bass, Josep Tabernero, et al. Genomic analysis identifies association of Fusobacterium with colorectal carcinoma. *Genome research*, 22(2):292–298, 2012.

Alp Kucukelbir, Rajesh Ranganath, Andrew Gelman, and David Blei. Automatic variational inference in Stan. In *Advances in Neural Information Processing Systems*, pages 568–576, 2015.

Malte Kuss, Carl Edward Rasmussen, and Ralf Herbrich. Assessing approximate inference for binary Gaussian process classification. *Journal of machine learning research*, 6(10), 2005.

Scott Linderman, Matthew Johnson, and Ryan P Adams. Dependent multinomial models made easy: Stick-breaking with the pólya-gamma augmentation. In *Advances in Neural Information Processing Systems*, pages 3456–3464, 2015.

David JC MacKay. Choice of basis for Laplace approximation. *Machine learning*, 33(1): 77–86, 1998.

P. J. McMurdie and S. Holmes. Waste not, want not: why rarefying microbiome data is inadmissible. *PLoS Computational Biology*, 10(4):e1003531, 2014. ISSN 1553-7358 (Electronic) 1553-734X (Linking). doi: 10.1371/journal.pcbi.1003531.

Thomas P Minka. Old and new matrix algebra useful for statistics, 2000.
URL `www.stat.cmu.edu/minka/papers/matrix.html`.

Hannes Nickisch and Carl Edward Rasmussen. Approximations for binary Gaussian process classification. *Journal of Machine Learning Research*, 9(Oct):2035–2078, 2008.

Helen Ogden. On the error in Laplace approximations of high-dimensional integrals. *arXiv:1808.06341*, 2018.

Vera Pawlowsky-Glahn, Juan José Egozcue, and Raimon Tolosana-Delgado. *Modeling and analysis of compositional data*. John Wiley & Sons, 2015.

Nicholas G Polson, James G Scott, and Jesse Windle. Bayesian inference for logistic models using pólya–gamma latent variables. *Journal of the American statistical Association*, 108 (504):1339–1349, 2013.

Raquel Prado and Mike West. *Time series : modeling, computation, and inference*. Chapman & Hall/CRC texts in statistical science series. CRC Press, Boca Raton, 2010. ISBN 9781420093360.

Jose M Quintana and Mike West. An analysis of international exchange rates using multivariate DLMs. *The Statistician*, pages 275–281, 1987.

Carl Edward Rasmussen. Gaussian processes in machine learning. In *Summer School on Machine Learning*, pages 63–71. Springer, 2003.

Jaakko Riihimäki, Aki Vehtari, et al. Laplace approximation for logistic Gaussian process density estimation and regression. *Bayesian analysis*, 9(2):425–448, 2014.

P.E. Rossi, G.M. Allenby, and R. McCulloch. *Bayesian Statistics and Marketing*. Wiley Series in Probability and Statistics. Wiley, 2012. ISBN 9780470863688.

Håvard Rue, Sara Martino, and Nicolas Chopin. Approximate Bayesian inference for latent Gaussian models by using integrated nested Laplace approximations. *Journal of the royal statistical society: Series b (statistical methodology)*, 71(2):319–392, 2009.

Amar Shah, Andrew Wilson, and Zoubin Ghahramani. Student-t processes as alternatives to Gaussian processes. In *Artificial Intelligence and Statistics*, pages 877–885, 2014.

J. D. Silverman, A. D. Washburne, S. Mukherjee, and L. A. David. A phylogenetic transform enhances analysis of compositional microbiota data. *eLife*, 6, 2017. ISSN 2050-084X (Electronic) 2050-084X (Linking). doi: 10.7554/eLife.21887.

J. D. Silverman, H. K. Durand, R. J. Bloom, S. Mukherjee, and L. A. David. Dynamic linear models guide design and analysis of microbiota studies within artificial human guts. *Microbiome*, 6(1):202, 2018a. ISSN 2049-2618 (Electronic) 2049-2618 (Linking). doi: 10.1186/s40168-018-0584-3.

Justin D Silverman. fido: Multinomial logistic normal models, 2019. URL `https://github.com/jsilve24/fido`.

Justin D Silverman, Kimberly Roche, Sayan Mukherjee, and Lawrence A David. Naught all zeros in sequence count data are the same. *Computational and Structural Biotechnology Journal*, 18:2789, 2020.

Stan Development Team. Stan user's guide, 2018. URL `https://mc-stan.org/docs/2_18/stan-users-guide/index.html`.

Jaclyn Strauss, Gilaad G Kaplan, Paul L Beck, Kevin Rioux, Remo Panaccione, Rebekah DeVinney, Tarah Lynch, and Emma Allen-Vercoe. Invasive potential of gut mucosa-derived Fusobacterium nucleatum positively correlates with IBD status of the host. *Inflammatory Bowel Diseases*, 17(9):1971–1978, 2011.

Shiliang Sun, Zehui Cao, Han Zhu, and Jing Zhao. A survey of optimization methods from a machine learning perspective. *IEEE Transactions on Cybernetics*, 2019.

Mike West and Jeff Harrison. *Bayesian forecasting and dynamic models*. Springer series in statistics. Springer, New York, 2nd edition, 1997. ISBN 0387947256.

Quan Zhang and Mingyuan Zhou. Permuted and augmented stick-breaking Bayesian multinomial regression. *The Journal of Machine Learning Research*, 18(1):7479–7511, 2017.

Mauricio A. Álvarez, Lorenzo Rosasco, and Neil D. Lawrence. Kernels for vector-valued functions: A review. *Foundations and Trends in Machine Learning*, 4(3):195, 00 2012. doi: 10.1561/2200000036.