Variational Inference for Count Response Semiparametric Regression: A Convex Solution

Virginia Murru and Matt P. Wand

Università di Padova and University of Technology Sydney

14th October 2025

Abstract

We develop a version of variational inference for Bayesian count response regression-type models that possesses attractive attributes such as convexity and closed form updates. The convex solution aspect entails numerically stable fitting algorithms, whilst the closed form aspect makes the methodology fast and easy to implement. The essence of the approach is the use of Pólya-Gamma augmentation of a Negative Binomial likelihood, a finite-valued prior on the shape parameter and the structured mean field variational Bayes paradigm. The approach applies to general count response situations. For concreteness, we focus on generalized linear mixed models within the semiparametric regression class of models. Real-time fitting is also described.

Keywords: Generalized additive models; generalized additive mixed models; Negative Binomial regression; Pólya-Gamma augmentation; real-time semiparametric regression; structured mean field variational Bayes.

1 Introduction

Variational approximation is an alternative to Monte Carlo methods for Bayesian inference and can be useful in applications where speed and scalability are at a premium. For count response semiparametric regression, Luts & Wand (2015) provided variational inference algorithms using a fixed form, or semiparametric, mean field variational Bayes approach. Despite its good accuracy, the non-convexity of fixed form approaches can lead to numerical problems. For example, in the simulation study described in Section 3 of Wand & Yu (2022), the fixed form variational approach to Poisson nonparametric regression failed to converge properly in 13.6% of the replications. In this article we devise a variational inference approach for which all component optimization problems are convex. The Luts & Wand (2015) methodology also involved some numerical integration steps, whereas our new approach has totally closed form updates. These attributes lead to fast and stable algorithms that are easy to implement.

Our approach to variational inference for count response semiparametric regression has similarities with the binary response model approaches of Jaakola & Jordan (2000) and Durante & Rigon (2019). For a given model the coordinate ascent updates in these two articles are identical, but the latter makes use of Pólya-Gamma augmentation (e.g. Polson *et al.*, 2013) which has the advantage of couching the algorithm within the ordinary mean field variational Bayes framework. As explained in, for example, Pillow & Scott (2012) and Zhou *et al.* (2012) Pólya-Gamma augmentation can also aid the fitting of Bayesian regression-type models with Negative Binomial likelihoods. An additional difficulty compared with the binary response setting is approximate Bayesian inference for the Negative Binomial shape parameter, which we denote here by κ. We deal with this problem via a *structured* mean field variational Bayes approach (e.g. Saul & Jordan, 1996; Wand *et al.* 2011). This involves restriction of κ to a finite set and performing a variational version of Bayesian model averaging, where individual models correspond to the atoms of the prior distribution of κ. Since κ is more of a nuisance parameter, and any finite set can be specified, there is little cost to this discretisation of κ.

Apart from Luts & Wand (2015), we are aware of some other approaches to variational inference for count response regression-type models. In particular, Zhou *et al.* (2012) and Miao *et al.* (2020) each contain variational inference algorithms which are also based on Pólya-Gamma augmentation of Negative Binomial likelihoods. Their approaches make use of Logarithmic series or, equivalently, Chinese Restaurant Process, representations of Negative Binomial response models. However, when this representation is combined with Pólya-Gamma augmentation there is no single joint distribution to which minimum Kullback-Leibler divergence, the underpinning of mean field variational Bayes, is being applied. In addition, the square-root quantity that arises in the tilting parameter of the Pólya-Gamma $\mathfrak{q}$-density, such as the $\xi_i^{(t)}$ quantity in Algorithm 2 of Durante & Rigon (2019) and the $\boldsymbol{c}_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)}$ quantity in Algorithm 1 of this article, is absent from the Zhou *et al.* (2012) and Miao *et al.* (2020) algorithms. Our implementations of the Miao *et al.* (2020) approach resulted in low accuracy compared with the variational approximation strategy developed here.

Our new convex solution for count response semiparametric regression also benefits real-time fitting and inference. In Section 5 and Algorithm 2 of Luts & Wand (2015) we presented an online variational algorithm for real-time count response semiparametric regression. However, storage of the predictor data and spline basis design matrices was required. In Section 4 of this article we present a new real-time algorithm for the same class of models that is *purely* online, in that only sufficient statistics-type quantities need to be updated and stored. The streaming data can be discarded after they are processed.

Notation used throughout this article is given in Section 1.1. Section 2 describes the specific count response semiparametric regression models to which we gear our methodological development. The new variational inference scheme is described in Section 3. Real-time semiparametric using online adaptations of our variational inference approach is described in Section 4. Section 5 contains numerical results. We provide some conclusions in Section 6.

1.1 Notation

A real-valued function, which is defined and prominent in Jaakkola & Jordan (2000), and also important here is that given by

$$\lambda_{\rm JJ}(x) \equiv \frac{\tanh(x/2)}{4x}, \quad x \in \mathbb{R}. \tag{1}$$

Scalar functions applied to a vector are evaluated in an element-wise fashion. For example, $\cosh([3\ 11]^T) \equiv [\cosh(3)\ \cosh(11)]^T$. Similarly, if $\boldsymbol{v}$ is a column vector then $\boldsymbol{v}^2$ is the vector of element-wise squares and $\|\boldsymbol{v}\| \equiv \sqrt{\boldsymbol{v}^T\boldsymbol{v}}$ is the Euclidean norm of $\boldsymbol{v}$. The notation diag($\boldsymbol{v}$) is used for the diagonal matrix containing the entries of $\boldsymbol{v}$ along its diagonal. If $\boldsymbol{M}$ is a $d \times d$ square matrix then diagonal($\boldsymbol{M}$) is the $d \times 1$ vector containing the diagonal entries of $\boldsymbol{M}$. Also, $\mathbf{1}$ is a column vector of ones. The symbol $\stackrel{\text{ind.}}{\sim}$ is shorthand for "independently distributed as".

2 Model Description

Throughout this article we focus on the following count response Bayesian semiparametric regression model:

$$\begin{gathered}
y_i|\,\boldsymbol{\beta},\boldsymbol{u},\kappa \stackrel{\text{ind.}}{\sim} \text{Negative-Binomial}\big(\exp\{(\boldsymbol{X\beta}+\boldsymbol{Zu})_i\},\kappa\big), \quad 1 \le i \le n,\\
\alpha_i|\,y_i,\boldsymbol{\beta},\boldsymbol{u},\kappa \stackrel{\text{ind.}}{\sim} \text{Pólya-Gamma}\big(y_i+\kappa,(\boldsymbol{X\beta}+\boldsymbol{Zu})_i+\log(\kappa)\big),\\
\boldsymbol{u}|\,\sigma_1^2,\ldots,\sigma_r^2 \sim N(\mathbf{0},\text{blockdiag}(\sigma_1^2\,\boldsymbol{I}_{K_1},\ldots,\sigma_r^2\,\boldsymbol{I}_{K_r})), \quad \boldsymbol{\beta} \sim N(\mathbf{0},\sigma_\beta^2\boldsymbol{I}_p),\\
\sigma_j \stackrel{\text{ind.}}{\sim} \text{Half-Cauchy}(s_\sigma), \quad 1 \le j \le r,
\end{gathered} \tag{2}$$

and κ has a discrete prior with atoms $\mathcal{K}$ and probabilities $\mathfrak{p}(\kappa)$, $\kappa \in \mathcal{K}$.

In (2) $\sigma_\beta > 0$ and $s_\sigma > 0$ are user-specified hyperparameters. Distributional notation used in (2) is defined later in this section.

Model (2) is a variant of what Zhao *et al.* (2006) label *general design* generalized linear mixed models. As explained in their Section 2, this is a very versatile structure and special cases include nested random effects models for grouped (e.g. longitudinal) data, crossed random effects models for item response data, generalized additive models, generalized additive mixed models, varying-coefficients models and low-ranking kriging.

The distributional notation in (2) is such that $x \sim \text{Negative-Binomial}(\mu, \kappa)$ denotes that the random variable x has Negative Binomial distribution with mean $\mu > 0$ and shape parameter $\kappa > 0$ with probability mass function:

$$\mathfrak{p}(x) = \frac{\kappa^\kappa \Gamma(x+\kappa)\mu^x}{\Gamma(\kappa)\big(\kappa+\mu\big)^{x+\kappa}\Gamma(x+1)}, \quad x = 0, 1, 2, \ldots.$$

The Pólya-Gamma distributional notation matches that used in Polson *et al.* (2013) and is described in Section S.1.2 of the online supplement. Also, $x \sim \text{Half-Cauchy}(s)$, with scale parameter $s > 0$, means that the random variable x has density function

$$\mathfrak{p}(x) = 2/[\pi\{1 + (x/s)^2\}s], \quad x > 0.$$

Variational inferential tractability is aided by the replacement of $\sigma_j \sim \text{Half-Cauchy}(s_\sigma)$ by

$$\sigma_j^2 | a_j \sim \text{Inverse-Gamma}(\tfrac{1}{2}, 1/a_j), \quad a_j \sim \text{Inverse-Gamma}(\tfrac{1}{2}, 1/s_\sigma^2), \tag{3}$$

where $x \sim \text{Inverse-Gamma}(\xi, \lambda)$ means that x has density function

$$\mathfrak{p}(x) = \frac{\lambda^\xi}{\Gamma(\xi)} x^{-\xi-1} \exp\big(-\lambda/x\big), \quad x > 0.$$

If we let

$$\boldsymbol{y} \equiv (y_1, \ldots, y_n), \quad \boldsymbol{\alpha} \equiv (\alpha_1, \ldots, \alpha_n), \quad \boldsymbol{\sigma}^2 \equiv (\sigma_1^2, \ldots, \sigma_r^2) \quad \text{and} \quad \boldsymbol{a} \equiv (a_1, \ldots, a_r)$$

then model (2) has directed acyclic graph representation as shown in Figure 1.

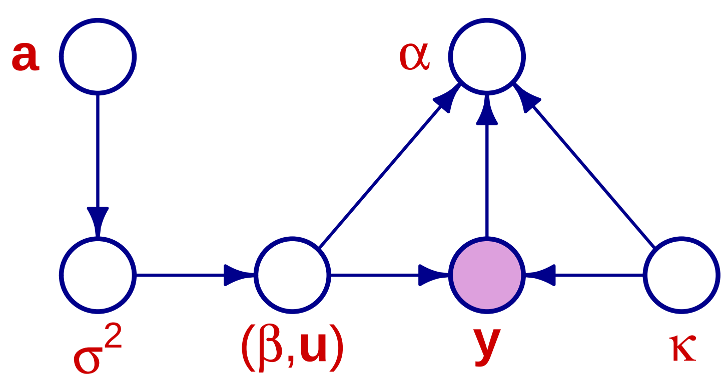

Figure 1: *Directed acyclic graph representation of model (2) with incorporation of the* $\boldsymbol{a} = (a_1, \ldots, a_r)$ *auxiliary variables as in (3). The* $\boldsymbol{y}$ *node is shaded to indicate that it contains observed data.*

Model (2) is similar to the class of Negative Binomial response models considered by Luts & Wand (2015). The differences are the presence of the Pólya-Gamma auxiliary variables and the imposition of a discrete prior on the shape parameter κ.

2.1 Extension to Covariance Matrix Parameters for Random Effects

To simplify the exposition, model (2) only contains scalar variance parameters. Extensions to covariance matrix parameters arise in the case of random intercept and slope models. A simple example of such a model having count responses is, for $1 \le i \le m$ and $1 \le j \le n_i$,

$$y_{ij}|\beta_0, \beta_1, u_{0i}, u_{1i} \stackrel{\text{ind.}}{\sim} \text{Negative-Binomial}\Big(\exp\left(\beta_0 + u_{0i} + (\beta_1 + u_{1i}\, x_{ij})\right), \kappa\Big)$$

where

$$\left[\begin{array}{c} u_{0i} \\ u_{1i} \end{array}\right] \Big| \boldsymbol{\Sigma} \stackrel{\text{ind.}}{\sim} N(\mathbf{0}, \boldsymbol{\Sigma})$$

and $\boldsymbol{\Sigma}$ is an unstructured 2×2 covariance matrix. Our new variational methodology has a straightforward extension to models containing covariance matrix parameters.

3 Variational Inference Scheme

Consider the Negative Binomial response additive model model given by (2) with the auxiliary variable replacement (3) and directed acyclic graph representation given in Figure 1. At its most general level, variational approximation of the full joint posterior density function of the parameters in (2) involves

$$\mathfrak{p}(\boldsymbol{\beta}, \boldsymbol{u}, \kappa, \boldsymbol{\alpha}, \boldsymbol{\sigma}^2, \boldsymbol{a}|\boldsymbol{y}) \approx \mathfrak{q}(\boldsymbol{\beta}, \boldsymbol{u}, \kappa, \boldsymbol{\alpha}, \boldsymbol{\sigma}^2, \boldsymbol{a}) \qquad (4)$$

where the $\mathfrak{q}$-density on the right-hand side of (4) is subject to specific restrictions. An initial restriction to consider is one involving the following product density form:

$$\mathfrak{q}(\boldsymbol{\beta}, \boldsymbol{u}, \kappa, \boldsymbol{\alpha}, \boldsymbol{\sigma}^2, \boldsymbol{a}) = \mathfrak{q}(\boldsymbol{\beta}, \boldsymbol{u}, \boldsymbol{a})\mathfrak{q}(\boldsymbol{\sigma}^2, \boldsymbol{\alpha})\mathfrak{q}(\kappa). \qquad (5)$$

Such a restriction is an example of ordinary mean field variational Bayes (e.g. Wainwright & Jordan, 2008) and, assuming tractability, the optimal $\mathfrak{q}$-densities can be found via coordinate ascent (e.g. Algorithm 1 of Ormerod & Wand, 2010) based on consistency conditions such as

$$\mathfrak{q}^*(\boldsymbol{\beta}, \boldsymbol{u}, \boldsymbol{a}) \propto E_{\mathfrak{q}(-(\boldsymbol{\beta}, \boldsymbol{u}, \boldsymbol{a}))}\big[\log\{\mathfrak{p}(\boldsymbol{\beta}, \boldsymbol{u}, \boldsymbol{a}|\text{rest})\}\big] \quad \text{and} \quad \mathfrak{q}^*(\kappa) \propto E_{\mathfrak{q}(-\kappa)}\big[\log\{\mathfrak{p}(\kappa|\text{rest})\}\big].$$

Here, for example, $E_{\mathfrak{q}(-(\boldsymbol{\beta}, \boldsymbol{u}, \boldsymbol{a}))}$ denotes expectation with respect to all $\mathfrak{q}$-densities other than $(\boldsymbol{\beta}, \boldsymbol{u}, \boldsymbol{a})$. Also, $\mathfrak{p}(\boldsymbol{\beta}, \boldsymbol{u}, \boldsymbol{a}|\text{rest})$ denotes the conditional density function of $(\boldsymbol{\beta}, \boldsymbol{u}, \boldsymbol{a})$ given the rest of the random variables in the model and called the full conditional density function of $(\boldsymbol{\beta}, \boldsymbol{u}, \boldsymbol{a})$. Attributes of (2) such as Pólya-Gamma augmentation lead to closed formed expressions for all but one of the *full conditional* density functions. The exception is $\mathfrak{p}(\kappa|\text{rest})$ which, as shown in Section S.2 of the online supplement, does not admit a closed form expression. This makes ordinary mean field variational Bayes impractical for model (2).

As a way of overcoming the difficulties with ordinary mean field variational Bayes we note that

$$\mathfrak{p}(\boldsymbol{\beta}, \boldsymbol{u}, \kappa, \boldsymbol{\alpha}, \boldsymbol{\sigma}^2, \boldsymbol{a}|\boldsymbol{y}) = \mathfrak{p}(\boldsymbol{\beta}, \boldsymbol{u}, \boldsymbol{\alpha}, \boldsymbol{\sigma}^2, \boldsymbol{a}|\kappa, \boldsymbol{y})\mathfrak{p}(\kappa|\boldsymbol{y}) \qquad (6)$$

and then consider approximation of the right-hand side of (6) using the product density form:

$$\mathfrak{q}(\boldsymbol{\beta}, \boldsymbol{u}, \kappa, \boldsymbol{\alpha}, \boldsymbol{\sigma}^2, \boldsymbol{a}) = \mathfrak{q}(\boldsymbol{\beta}, \boldsymbol{u}, \boldsymbol{a}|\kappa)\mathfrak{q}(\boldsymbol{\sigma}^2, \boldsymbol{\alpha}|\kappa)\mathfrak{q}(\kappa). \qquad (7)$$

which differs from (5) due to its conditioning on κ. Noting that κ is confined to a finite set, we apply a *structured* mean field variational Bayes approach to obtaining the optimal $\mathfrak{q}$-densities under restriction (7). In essence, this involves performing ordinary mean field variational Bayes for each $\kappa \in \mathcal{K}$ as if κ is fixed and then obtaining a weighted average of the resultant $\mathfrak{q}$-densities that depends on the κ prior distribution and the variational-approximate marginal log-likelihoods. A summary of this approach is given in Section 3.1 of Wand *et al.* (2011). From

Algorithm 1 *Structured mean field variational Bayes algorithm for achieving approximate Bayesian inference for model (2) according to product density restriction (7).*

Inputs: $\boldsymbol{y}\ (n\times 1)$, response vector having all entries non-negative integers,

$\boldsymbol{C}\ \left(n\times\left(p+\sum_{j=1}^{r}K_j\right)\right)$, combined design matrix,

$\sigma_\beta, s_\sigma>0$, hyperparameters for the $\boldsymbol{\beta}$ and σ_j prior distributions

$\mathcal{K}$, a finite set of positive numbers corresponding to the atoms of the prior distribution of κ.

Initialize: $\mu_{q(1/\sigma_j^2|\kappa)}>0$, $1\le j\le r$, and $\boldsymbol{c}_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)}$ $(n\times 1)$ all entries positive, for each $\kappa\in\mathcal{K}$.

For $\kappa\in\mathcal{K}$:

Cycle:

$$\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)}\longleftarrow 2(\boldsymbol{y}+\kappa\boldsymbol{1})\odot\lambda_{\rm JJ}(\boldsymbol{c}_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)})$$

$$\boldsymbol{M}_{q(1/\boldsymbol{\sigma}^2|\kappa)}\longleftarrow \text{blockdiag}(\sigma_\beta^{-2}\boldsymbol{I}_p,\mu_{q(1/\sigma_1^2|\kappa)}\boldsymbol{I}_{K_1},\ldots,\mu_{q(1/\sigma_r^2|\kappa)}\boldsymbol{I}_{K_r})$$

$$\boldsymbol{\Sigma}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}\longleftarrow\left\{\boldsymbol{C}^T\text{diag}\big(\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)}\big)\boldsymbol{C}+\boldsymbol{M}_{\mathfrak{q}(1/\boldsymbol{\sigma}^2|\kappa)}\right\}^{-1}$$

$$\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}\longleftarrow\boldsymbol{\Sigma}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}\left\{\tfrac{1}{2}(\boldsymbol{C}^T\boldsymbol{y}-\kappa\boldsymbol{C}^T\boldsymbol{1})+\log(\kappa)\boldsymbol{C}^T\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)}\right\}$$

$$\boldsymbol{c}_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)}\longleftarrow\sqrt{\text{diagonal}\left(\boldsymbol{C}\boldsymbol{\Sigma}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}\boldsymbol{C}^T\right)+\left(\boldsymbol{C}\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}-\log(\kappa)\boldsymbol{1}\right)^2}$$

For $j=1,\ldots,r$:

$$\lambda_{\mathfrak{q}(a_j|\kappa)}\longleftarrow\mu_{\mathfrak{q}(1/\sigma_j^2|\kappa)}+s_\sigma^{-2}\quad;\quad\mu_{\mathfrak{q}(1/a_j|\kappa)}\longleftarrow 1/\lambda_{\mathfrak{q}(a_j|\kappa)}$$

$$\lambda_{\mathfrak{q}(\sigma_j^2|\kappa)}\longleftarrow\mu_{\mathfrak{q}(1/a_j|\kappa)}+\tfrac{1}{2}\left\{\|\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{u}_j|\kappa)}\|^2+\text{tr}(\boldsymbol{\Sigma}_{\mathfrak{q}(\boldsymbol{u}_j|\kappa)})\right\}$$

$$\mu_{\mathfrak{q}(1/\sigma_j^2|\kappa)}\longleftarrow(K_j+1)/\big(2\lambda_{\mathfrak{q}(\sigma_j^2|\kappa)}\big)$$

$$\begin{aligned}\underline{\ell}_{\mathfrak{q}}(\kappa)\longleftarrow&\ \tfrac{1}{2}\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}^T\big(\boldsymbol{C}^T\boldsymbol{y}-\kappa\,\boldsymbol{C}^T\boldsymbol{1}\big)-(\boldsymbol{y}+\kappa\boldsymbol{1})^T\log\left\{\cosh\left(\tfrac{1}{2}c_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)}\right)\right\}\\&-\frac{\|\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{\beta}|\kappa)}\|^2+\text{tr}(\boldsymbol{\Sigma}_{\mathfrak{q}(\boldsymbol{\beta}|\kappa)})}{2\sigma_\beta^2}+\tfrac{1}{2}\log|\boldsymbol{\Sigma}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}|\end{aligned}$$

For $j=1,\ldots,r$:

$$\begin{aligned}\underline{\ell}_{\mathfrak{q}}(\kappa)\longleftarrow&\ \underline{\ell}_{\mathfrak{q}}(\kappa)+\mu_{\mathfrak{q}(1/\sigma_j^2|\kappa)}\left\{\lambda_{\mathfrak{q}(\sigma_j^2|\kappa)}-\mu_{\mathfrak{q}(1/a_j|\kappa)}-\tfrac{1}{2}\|\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{u}_j|\kappa)}\|^2-\tfrac{1}{2}\text{tr}\big(\boldsymbol{\Sigma}_{\mathfrak{q}(\boldsymbol{u}_j|\kappa)}\big)\right\}\\&+\mu_{\mathfrak{q}(1/a_j|\kappa)}\big(\lambda_{\mathfrak{q}(a_j|\kappa)}-s_\sigma^{-2}\big)-\tfrac{1}{2}(K_j+1)\log\left(\lambda_{\mathfrak{q}(\sigma_j^2|\kappa)}\right)-\log\left(\lambda_{\mathfrak{q}(a_j|\kappa)}\right)\end{aligned}$$

until the increase in $\underline{\ell}_{\mathfrak{q}}(\kappa)$ is negligible.

$$\underline{\ell}(\kappa)\longleftarrow\underline{\ell}_{\mathfrak{q}}(\kappa)+\boldsymbol{1}^T\log\{\Gamma(\boldsymbol{y}+\kappa\boldsymbol{1})\}+n\big[\tfrac{1}{2}\kappa\log(\kappa)-\log(2)\kappa-\log\{\Gamma(\kappa)\}\big]-\tfrac{1}{2}\log(\kappa)(\boldsymbol{y}^T\boldsymbol{1})$$

Outputs: $\left\{\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)},\boldsymbol{\Sigma}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)},\lambda_{\mathfrak{q}(\sigma_j^2|\kappa)},\underline{\ell}(\kappa):\kappa\in\mathcal{K},\ 1\le j\le r\right\}$

Section 10.1.1 of Bishop (2006), each individual (over $\kappa\in\mathcal{K}$) mean field optimization problem is convex. Also, induced factorization theory (e.g. Bishop, 2006; Section 10.2.5) implies that we have the additional product density forms $\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u},\boldsymbol{a}|\kappa)=\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)\mathfrak{q}(\boldsymbol{a}|\kappa)$ and $\mathfrak{q}(\boldsymbol{\sigma}^2,\boldsymbol{\alpha}|\kappa)=\mathfrak{q}(\boldsymbol{\sigma}^2|\kappa)\mathfrak{q}(\boldsymbol{\alpha}|\kappa)$ even though these are not imposed at the outset.

Algorithm 1 describes a suite of coordinate descent algorithms for obtaining, iteratively, the parameters of the optimal $\mathfrak{q}$-densities. It uses the following definitions:

$$\boldsymbol{C}\equiv[\boldsymbol{X}\ \boldsymbol{Z}]$$

and

$$\boldsymbol{u}_j \text{ is the } K_j\times 1 \text{ block of } \boldsymbol{u} \text{ according to the partition } \boldsymbol{u}=[\boldsymbol{u}_1^T\ \cdots\ \boldsymbol{u}_r^T]^T. \tag{8}$$

It also involves the $\lambda_{\rm JJ}$ function defined by (1).

Based on the output from Algorithm 1 and the structured mean field variational Bayes formulae given in Section 3.1 of Wand *et al.* (2011), the approximate posterior distributions of the model parameters are obtained as follows:

$$\mathfrak{q}^*(\kappa)=\frac{\mathfrak{p}(\kappa)\underline{\ell}(\kappa)}{\sum_{\kappa'\in\mathcal{K}}\mathfrak{p}(\kappa')\underline{\ell}(\kappa')},\quad \kappa\in\mathcal{K},$$

$$\begin{aligned}\mathfrak{q}^*(\boldsymbol{\beta},\boldsymbol{u})=&\sum_{\kappa\in\mathcal{K}}\mathfrak{q}^*(\kappa)\big|2\pi\boldsymbol{\Sigma}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}\big|^{-1/2}\\&\times\exp\left\{-\tfrac{1}{2}\left(\left[\begin{array}{c}\boldsymbol{\beta}\\\boldsymbol{u}\end{array}\right]-\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}\right)^T\boldsymbol{\Sigma}^{-1}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}\left(\left[\begin{array}{c}\boldsymbol{\beta}\\\boldsymbol{u}\end{array}\right]-\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}\right)\right\}\end{aligned}$$

and

$$\mathfrak{q}^*(\sigma_j^2)=\sum_{\kappa\in\mathcal{K}}\left\{\frac{\mathfrak{q}^*(\kappa)\lambda_{\mathfrak{q}(\sigma_j^2|\kappa)}^{(K_j+1)/2}}{\Gamma\big(\frac{1}{2}(K_j+1)\big)}\right\}(\sigma_j^2)^{-(K_j+1)/2-1}\exp\left(-\frac{\lambda_{\mathfrak{q}(\sigma_j^2|\kappa)}}{\sigma_j^2}\right),\ \ \sigma_j^2>0,\ \ 1\le j\le r.$$

3.1 Streamlined Variational Inference Alternatives

In grouped data situations, the sub-blocks of the $\boldsymbol{Z}$ matrix corresponding to random effects typically are quite sparse. Algorithm 1 is still valid for such $\boldsymbol{Z}$ matrices but, for large numbers of groups, tends to be inefficient when applied naïvely. Instead streamlined variational inference alternatives, which take advantage of sparse structure in $\boldsymbol{Z}$ matrices, are recommended. Lee & Wand (2016) provides streamlined variational methodology for binary response semiparametric regression models for grouped data. The same ideas apply to the count response setting treated here. The details will appear in the first author's upcoming doctoral dissertation.

4 Real-Time Count Response Semiparametric Regression

The structured mean field variational Bayes approach used in Algorithm 1 also lends itself to online fitting of streaming data. This allows real-time count response semiparametric regression. Moreover, unlike in Luts & Wand (2015), this new approach has the attractive aspects of only requiring low-dimensional sufficient statistics quantities to be stored and updated. There is no need to keep the full data in memory.

Our proposed algorithm for real-time count response semiparametric regression is similar to that used in Section 3 of Luts *et al.* (2014) for real-time binary response semiparametric regression. The main difference is the presence of the κ parameter and its finite set restriction. The essence of the approach is to express each of $\mathfrak{q}$-density parameters in terms of sufficient statistics quantities such as $\boldsymbol{C}^T\boldsymbol{y}$. If the current sample size is n then

$$\boldsymbol{C}^T\boldsymbol{y}=\sum_{i=1}^n\boldsymbol{c}_iy_i\quad\text{where}\quad\boldsymbol{c}_i\equiv i\text{th row of }\boldsymbol{C}.$$

When a new observation arrives with response value y_{new} and corresponding design matrix row $\boldsymbol{c}_{\text{new}}$ then the

$$\boldsymbol{C}^T\boldsymbol{y}\quad\text{sufficient statistic is incremented by}\quad\boldsymbol{c}_{\text{new}}y_{\text{new}}.$$

Similarly, for each $\kappa\in\mathcal{K}$, the

$$\boldsymbol{y}^T\log\{\cosh(\tfrac{1}{2}\boldsymbol{c}_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)})\}\quad\text{sufficient statistic is incremented by}\quad y_{\text{new}}\log\{\cosh(\tfrac{1}{2}c(\alpha|\kappa)_{\text{new}})\}$$

Algorithm 2 *Online structured mean field variational Bayes algorithm for achieving real-times approximate Bayesian inference for model (2) according to product density restriction (7).*

1. Perform batch-based tuning runs analogous to those described in Algorithm 2' of Luts, Broderick & Wand (2014) and determine a warm-up sample size n_{warm} for which convergence is validated.

2. Set $\boldsymbol{y}_{\text{warm}}$ and $\boldsymbol{C}_{\text{warm}}$ to be the response vector and design matrix and, for each $\kappa \in \mathcal{K}$, let $\boldsymbol{c}_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)}$ be the vector of Pólya-Gamma variational tilting parameters, based on the first n_{warm} observations. Then set $n \leftarrow n_{\text{warm}}$, $\boldsymbol{y}^T\mathbf{1} \leftarrow \boldsymbol{y}_{\text{warm}}^T\mathbf{1}$, $\boldsymbol{C}^T\mathbf{1} \leftarrow \boldsymbol{C}_{\text{warm}}^T\mathbf{1}$, $\boldsymbol{C}^T\boldsymbol{y} \leftarrow \boldsymbol{C}_{\text{warm}}^T\boldsymbol{y}_{\text{warm}}$. For each $\kappa \in \mathcal{K}$ set $\mathbf{1}^T\log\{\Gamma(\boldsymbol{y}+\kappa\mathbf{1})\} \longleftarrow \mathbf{1}^T\log\{\Gamma(\boldsymbol{y}_{\text{warm}}+\kappa\mathbf{1})\}$, $\boldsymbol{C}^T\lambda_{\text{JJ}}(\boldsymbol{c}_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)}) \longleftarrow \boldsymbol{C}_{\text{warm}}^T\lambda_{\text{JJ}}(\boldsymbol{c}_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)})$ and similar sufficient statistics quantities that appear in Step 3 below. Also, set $\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}$, $\boldsymbol{\Sigma}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}$, $\mu_{q(1/\sigma_{u1}^2|\kappa)},\ldots,\mu_{q(1/\sigma_{ur}^2|\kappa)}$ to be the values for these quantities obtained in the batch-based tuning run with sample size n_{warm}.

3. Let $\mathfrak{q}^*_{\text{warm}}(\kappa)$, $\kappa \in \mathcal{K}$, be the $\mathfrak{q}$-density of κ obtained from feeding $\boldsymbol{y}_{\text{warm}}$, $\boldsymbol{C}_{\text{warm}}$, σ_β and s_σ into Algorithm 1.

4. Cycle:

   read in y_{new} and $\boldsymbol{c}_{\text{new}}$; $n \longleftarrow n+1$

   $\boldsymbol{y}^T\mathbf{1} \longleftarrow \boldsymbol{y}^T\mathbf{1} + y_{\text{new}}$; $\boldsymbol{C}^T\mathbf{1} \longleftarrow \boldsymbol{C}^T\mathbf{1} + \boldsymbol{c}_{\text{new}}$; $\boldsymbol{C}^T\boldsymbol{y} \longleftarrow \boldsymbol{C}^T\boldsymbol{y} + \boldsymbol{c}_{\text{new}}y_{\text{new}}$

   Obtain the atom set $\mathcal{K}_n \subseteq \mathcal{K}$ for the current sample size such that the retained atoms coax $\mathfrak{q}^*(\kappa)$ to be more concentrated around the posterior mean of $\mathfrak{q}^*_{\text{warm}}(\kappa)$. (A practical recommendation for this step is described in Section 4.1.)

   For $\kappa \in \mathcal{K}_n$:

   $$\mathbf{1}^T\log\{\Gamma(\boldsymbol{y}+\kappa\mathbf{1})\} \longleftarrow \mathbf{1}^T\log\{\Gamma(\boldsymbol{y}+\kappa\mathbf{1})\} + \log\{\Gamma(y_{\text{new}}+\kappa)\}$$
   $$c(\alpha|\kappa)_{\text{new}} \longleftarrow \sqrt{\boldsymbol{c}_{\text{new}}^T\boldsymbol{\Sigma}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}\boldsymbol{c}_{\text{new}} + \{\boldsymbol{c}_{\text{new}}^T\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)} - \log(\kappa)\}^2}$$
   $$\boldsymbol{C}^T\lambda_{\text{JJ}}(\boldsymbol{c}_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)}) \longleftarrow \boldsymbol{C}^T\lambda_{\text{JJ}}(\boldsymbol{c}_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)}) + \lambda_{\text{JJ}}(c(\alpha|\kappa)_{\text{new}})\boldsymbol{c}_{\text{new}}$$
   $$\boldsymbol{C}^T\big(\boldsymbol{y}\odot\lambda_{\text{JJ}}(\boldsymbol{c}_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)})\big) \longleftarrow \boldsymbol{C}^T\big(\boldsymbol{y}\odot\lambda_{\text{JJ}}(\boldsymbol{c}_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)})\big) + y_{\text{new}}\lambda_{\text{JJ}}(c(\alpha|\kappa)_{\text{new}})\boldsymbol{c}_{\text{new}}$$
   $$\boldsymbol{C}^T\text{diag}\big(\lambda_{\text{JJ}}(\boldsymbol{c}_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)})\big)\boldsymbol{C} \longleftarrow \boldsymbol{C}^T\text{diag}\big(\lambda_{\text{JJ}}(\boldsymbol{c}_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)})\big)\boldsymbol{C} + \lambda_{\text{JJ}}(c(\alpha|\kappa)_{\text{new}})\boldsymbol{c}_{\text{new}}\boldsymbol{c}_{\text{new}}^T$$
   $$\boldsymbol{C}^T\text{diag}\big(\boldsymbol{y}\odot\lambda_{\text{JJ}}(\boldsymbol{c}_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)})\big)\boldsymbol{C} \longleftarrow \boldsymbol{C}^T\text{diag}\big(\boldsymbol{y}\odot\lambda_{\text{JJ}}(\boldsymbol{c}_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)})\big)\boldsymbol{C} + y_{\text{new}}\lambda_{\text{JJ}}(c(\alpha|\kappa)_{\text{new}})\boldsymbol{c}_{\text{new}}\boldsymbol{c}_{\text{new}}^T$$
   $$\mathbf{1}^T\log\{\cosh(\tfrac{1}{2}\boldsymbol{c}_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)})\} \longleftarrow \mathbf{1}^T\log\{\cosh(\tfrac{1}{2}\boldsymbol{c}_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)})\} + \log\{\cosh(\tfrac{1}{2}c(\alpha|\kappa)_{\text{new}})\}$$
   $$\boldsymbol{y}^T\log\{\cosh(\tfrac{1}{2}\boldsymbol{c}_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)})\} \longleftarrow \boldsymbol{y}^T\log\{\cosh(\tfrac{1}{2}\boldsymbol{c}_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)})\} + y_{\text{new}}\log\{\cosh(\tfrac{1}{2}c(\alpha|\kappa)_{\text{new}})\}$$
   $$\boldsymbol{M}_{q(1/\boldsymbol{\sigma}^2|\kappa)} \longleftarrow \text{blockdiag}(\sigma_\beta^{-2}\boldsymbol{I}_p, \mu_{q(1/\sigma_1^2|\kappa)}\boldsymbol{I}_{K_1},\ldots,\mu_{q(1/\sigma_r^2|\kappa)}\boldsymbol{I}_{K_r})$$
   $$\boldsymbol{\Sigma}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)} \longleftarrow \Big\{2\boldsymbol{C}^T\text{diag}\big(\boldsymbol{y}\odot\lambda_{\text{JJ}}(\boldsymbol{c}_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)})\big)\boldsymbol{C} + 2\kappa\boldsymbol{C}^T\text{diag}\big(\lambda_{\text{JJ}}(\boldsymbol{c}_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)})\big)\boldsymbol{C} + \boldsymbol{M}_{q(1/\boldsymbol{\sigma}^2|\kappa)}\Big\}^{-1}$$
   $$\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)} \longleftarrow \boldsymbol{\Sigma}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}\Big(\tfrac{1}{2}(\boldsymbol{C}^T\boldsymbol{y} - \kappa\boldsymbol{C}^T\mathbf{1}) + 2\log(\kappa)\big\{\boldsymbol{C}^T\big(\boldsymbol{y}\odot\lambda_{\text{JJ}}(\boldsymbol{c}_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)})\big) + \kappa\boldsymbol{C}^T\lambda_{\text{JJ}}(\boldsymbol{c}_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)})\big\}\Big)$$

   For $j = 1,\ldots,r$:

   $$\lambda_{\mathfrak{q}(a_j|\kappa)} \longleftarrow \mu_{\mathfrak{q}(1/\sigma_j^2|\kappa)} + s_\sigma^{-2} \quad ; \quad \mu_{\mathfrak{q}(1/a_j|\kappa)} \longleftarrow 1/\lambda_{\mathfrak{q}(a_j|\kappa)}$$
   $$\lambda_{\mathfrak{q}(\sigma_j^2|\kappa)} \longleftarrow \mu_{\mathfrak{q}(1/a_j|\kappa)} + \tfrac{1}{2}\big\{\|\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{u}_j|\kappa)}\|^2 + \text{tr}(\boldsymbol{\Sigma}_{\mathfrak{q}(\boldsymbol{u}_j|\kappa)})\big\}$$
   $$\mu_{\mathfrak{q}(1/\sigma_j^2|\kappa)} \longleftarrow (K_j+1)/\big(2\lambda_{\mathfrak{q}(\sigma_j^2|\kappa)}\big)$$

   continued on a subsequent page ...

where

$$c(\alpha|\kappa)_{\text{new}} = \sqrt{\boldsymbol{c}_{\text{new}}^T\boldsymbol{\Sigma}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}\boldsymbol{c}_{\text{new}} + \{\boldsymbol{c}_{\text{new}}^T\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)} - \log(\kappa)\}^2}.$$

and $\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}$ and $\boldsymbol{\Sigma}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}$ are the current $\mathfrak{q}$-density parameters of $(\boldsymbol{\beta},\boldsymbol{u})|\kappa$. Continuing in this fashion, we arrive at Algorithm 2 for real-time count response semiparametric regression, and only requiring low-dimensional sufficient statistics storage and updating.

Algorithm 2 continued. *This is a continuation of the description of this algorithm that commences on a preceding page.*

$$\underline{\ell}(\kappa) \longleftarrow \tfrac{1}{2}\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}^T\big(\boldsymbol{C}^T\boldsymbol{y} - \kappa\,\boldsymbol{C}^T\boldsymbol{1}\big) - \boldsymbol{y}^T\log\left\{\cosh\left(\tfrac{1}{2}c_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)}\right)\right\}$$
$$-\kappa\boldsymbol{1}^T\log\left\{\cosh\left(\tfrac{1}{2}c_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)}\right)\right\} - \frac{\|\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{\beta}|\kappa)}\|^2 + \mathrm{tr}(\boldsymbol{\Sigma}_{\mathfrak{q}(\boldsymbol{\beta}|\kappa)})}{2\sigma_{\beta}^2} + \tfrac{1}{2}\log|\boldsymbol{\Sigma}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}|$$
$$+\boldsymbol{1}^T\log\{\Gamma(\boldsymbol{y}+\kappa\boldsymbol{1})\} + n\big[\tfrac{1}{2}\kappa\log(\kappa) - \log(2)\kappa - \log\{\Gamma(\kappa)\}\big] - \tfrac{1}{2}\log(\kappa)(\boldsymbol{y}^T\boldsymbol{1})$$

For $j = 1,\ldots,r$:

$$\underline{\ell}(\kappa) \longleftarrow \underline{\ell}(\kappa) + \mu_{\mathfrak{q}(1/\sigma_j^2|\kappa)}\big\{\lambda_{\mathfrak{q}(\sigma_j^2|\kappa)} - \mu_{\mathfrak{q}(1/a_j|\kappa)} - \tfrac{1}{2}\|\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{u}_j|\kappa)}\|^2 - \tfrac{1}{2}\mathrm{tr}\big(\boldsymbol{\Sigma}_{\mathfrak{q}(\boldsymbol{u}_j|\kappa)}\big)\big\}$$
$$+\mu_{\mathfrak{q}(1/a_j|\kappa)}\big(\lambda_{\mathfrak{q}(a_j|\kappa)} - s_\sigma^{-2}\big) - \tfrac{1}{2}(K_j+1)\log\left(\lambda_{\mathfrak{q}(\sigma_j^2|\kappa)}\right) - \log\left(\lambda_{\mathfrak{q}(a_j|\kappa)}\right)$$

$$\mathfrak{q}^*(\kappa) \longleftarrow \mathfrak{p}(\kappa)\exp\{\underline{\ell}(\kappa)\}\Big/\sum_{\kappa'\in\mathcal{K}_n}\mathfrak{p}(\kappa')\exp\{\underline{\ell}(\kappa')\}$$

$$\mathfrak{q}^*(\boldsymbol{\beta},\boldsymbol{u}) \longleftarrow \sum_{\kappa\in\mathcal{K}_n}\Bigg[\mathfrak{q}^*(\kappa)|2\pi\boldsymbol{\Sigma}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}|^{-1/2}$$
$$\times\exp\left\{-\tfrac{1}{2}\left(\begin{bmatrix}\boldsymbol{\beta}\\ \boldsymbol{u}\end{bmatrix} - \boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}\right)^T\boldsymbol{\Sigma}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}^{-1}\left(\begin{bmatrix}\boldsymbol{\beta}\\ \boldsymbol{u}\end{bmatrix} - \boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}\right)\right\}\Bigg]$$

For $j = 1,\ldots,r$:

$$\mathfrak{q}^*(\sigma_j^2) \longleftarrow \sum_{\kappa\in\mathcal{K}_n}\left\{\frac{\mathfrak{q}^*(\kappa)\lambda_{\mathfrak{q}(\sigma_j^2|\kappa)}^{(K_j+1)/2}}{\Gamma\big(\tfrac{1}{2}(K_j+1)\big)}\right\}(\sigma_j^2)^{-(K_j+1)/2-1}\exp\left(-\frac{\lambda_{\mathfrak{q}(\sigma_j^2|\kappa)}}{\sigma_j^2}\right).\ \ \sigma_j^2 > 0,$$

Produce summaries based on $\mathfrak{q}^*(\boldsymbol{\beta},\boldsymbol{u})$, $\mathfrak{q}^*(\kappa)$ and $\mathfrak{q}^*(\sigma_j^2)$, $1 \le j \le r$.

until data no longer available or analysis terminated.

4.1 The Step of Algorithm 2 Involving the Reduced Atom Set $\mathcal{K}_n$

When developing and testing Algorithm 2 we first looked into using the original atom set $\mathcal{K}$ during the online updates. However, when $\mathcal{K}$ is kept fixed, there is a tendency for the κ probability mass to pile towards the left or right extremities of $\mathcal{K}$ which, in turn, adversely impacts the quality of the $\mathfrak{q}^*(\boldsymbol{\beta},\boldsymbol{u})$ and $\mathfrak{q}^*(\sigma_j^2)$ approximations. We do not have an explanation for the occurrence of this phenomenon. Some experimentation showed that the online $\mathfrak{q}$-densities of the main model parameters could achieve similar behaviour to their batch counterparts if the κ atoms were sequentially reduced and concentrated around the posterior mean of $\mathfrak{q}^*_{\text{warm}}(\kappa)$. Further research on this aspect seems warranted. However, with practicality in mind, we devised the following simple scheme for achieving such concentration. Given the skewed nature of posterior distributions of positive-valued parameters, we work with logarithm of κ. Let $\lambda \equiv \log(\kappa)$ and let $\mathfrak{q}^*_{\text{warm}}(\lambda)$ be the probability mass function of λ corresponding to $\mathfrak{q}^*_{\text{warm}}(\kappa)$. Next let $\mu^{\lambda}_{\text{warm}}$ and $\sigma^{\lambda}_{\text{warm}}$ denote the mean and standard deviation of $\mathfrak{q}^*_{\text{warm}}(\lambda)$. Asymptotic normality considerations dictate that most of the probability mass of $\mathfrak{q}^*_{\text{warm}}(\lambda)$ is in the interval

$$\big(\mu^{\lambda}_{\text{warm}} - \tau\sigma^{\lambda}_{\text{warm}}, \mu^{\lambda}_{\text{warm}} + \tau\sigma^{\lambda}_{\text{warm}}\big)\quad\text{where}\quad\tau\approx 3.$$

For $n \ge n_{\text{warm}}$ a reasonable away to coax $\mathfrak{q}^*(\kappa)$ to be more concentrated around $\mathfrak{q}^*_{\text{warm}}(\kappa)$ is to set

$$\mathcal{K}_n = \left\{\kappa\in\mathcal{K} : \mu^{\lambda}_{\text{warm}} - \tau\sigma^{\lambda}_{\text{warm}}\sqrt{n_{\text{warm}}/n} \le \log(\kappa) \le \mu^{\lambda}_{\text{warm}} + \tau\sigma^{\lambda}_{\text{warm}}\sqrt{n_{\text{warm}}/n}\right\}, \tag{9}$$

which is also based on standard asymptotic normality considerations. In our simulated data assessment of Algorithm 2, we adopted (9) with $\tau = 3.5$. Lastly, for very large n, we need to guard against $\mathcal{K}_n$ becoming null. We suggest to cease the atoms reduction when the number of atoms reaches a low value such as 5.

5 Numerical Results

We now evaluate and illustrate the performance of Algorithms 1 and 2 using both simulated and actual data. Section 5.1 uses the same simulated data setting as Luts & Wand (2015) to assess comparative accuracy and speed of Algorithm 1 against a Markov chain Monte Carlo benchmark. In Section 5.2 we conduct a simulation-based assessment of Algorith 2. In this study the real-time fits are compared with the more computationally expensive batch fits. As with any simulation study, the results of Sections 5.1 and Section 5.2 are necessarily limited in that they can only treat a few specific scenarios. In Section 5.3 we illustrate use of the methodology for some actual data concerning pollen counts.

5.1 Simulated Data Assessment of Algorithm 1

Our simulated data assessment of Algorithm 1 involved data generated according to the following Negative Binomial additive model:

$$y_i|x_{i1}, x_{i2} \stackrel{\text{ind.}}{\sim} \text{Negative-Binomial}\big(\exp\{\eta_{\text{true},1}(x_{1i}) + \eta_{\text{true},2}(x_{2i})\}\big), \kappa_{\text{true}}), \quad 1 \le i \le 500$$

where

$$\eta_{\text{true},1}(x) = \cos(4\pi x) + 2x, \quad \eta_{\text{true},2}(x) = 0.4\phi(x; 0.38, 0.08) - 1.02x + 0.018x^2 + 0.08\phi(x; 0.75, 0.03)$$

and $\phi(x; \mu, \sigma)$ denotes the density function of the Normal distribution with mean μ and standard deviation σ, evaluated at x. We set $\kappa_{\text{true}} = 3.8$, which corresponds to a relatively high amount of over-dispersion. The predictor data were generated according to

$$x_{i1}, x_{i2} \stackrel{\text{ind.}}{\sim} \text{Uniform}(0, 1), \quad 1 \le i \le 500.$$

We used the following penalized spline model for estimation of $\eta_{\text{true},1}(x_1) + \eta_{\text{true},2}(x_2)$:

$$\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \sum_{k=1}^{K_1} u_{1k} z_{1k}(x_1) + \sum_{k=1}^{K_2} u_{2k} z_{2k}(x_2), \quad u_{jk} \stackrel{\text{ind.}}{\sim} N(0, \sigma_j^2), \quad j = 1, 2. \tag{10}$$

Here z_{1k} and z_{2k} are canonical O'Sullivan spline bases as described in Section 4 of Wand & Ormerod (2008). The basis sizes were $K_1 = K_2 = 17$. This set-up is an $r = 2$ special case of model (2) with, for example,

$$\boldsymbol{X} = \big[1\ x_{1i}\ x_{2i}\big]_{1 \le i \le 500} \quad \text{and} \quad \boldsymbol{Z} = \big[z_{11}(x_{1i}) \cdots z_{1K_1}(x_{1i})\ z_{21}(x_{2i}) \cdots z_{2K_2}(x_{2i})\big]_{1 \le i \le 500}. \tag{11}$$

We imposed the prior distributions:

$$\beta_0, \beta_1, \beta_2 \stackrel{\text{ind.}}{\sim} N(0, 10^5), \quad \sigma_1, \sigma_2 \stackrel{\text{ind.}}{\sim} \text{Half-Cauchy}(10^5) \quad \text{and} \quad \mathfrak{p}(\kappa) \propto \exp(-\kappa/100), \quad \kappa \in \mathcal{K},$$

where the atom set $\mathcal{K}$ is a geometric sequence of size 50 between $\kappa^{\text{true}}/10$ and $10\kappa^{\text{true}}$. We fitted the special case of model (2), corresponding to (10) and (11), via Algorithm 1. The convergence was assessed by monitoring the relative change in $\log\{\underline{p}(\boldsymbol{y}; q|\kappa)\}$, with a stopping criterion set at 10^{-10}. One hundred simulation replications were performed.

To facilitate accuracy assessment we also obtained fits based on a Markov chain Monte Carlo approach. This was achieved using the JAGS Bayesian inference engine via the R package rjags (Plummer, 2025). For each replication, chains of length 10000 were obtained. The first 5000 values were discarded as burn-in Then, thinning by a factor of 5 was applied, leading to a retained samples of size of 1000.

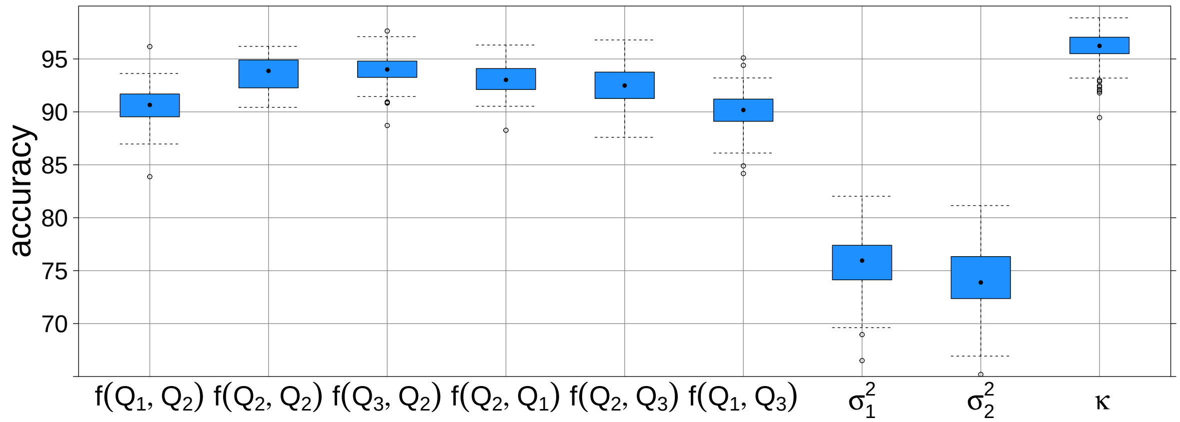

Figure 2: *Boxplots of accuracy scores, as defined by (12), for the Algorithm 1 simulation study.*

5.1.1 Assessment of Accuracy

The accuracy score for a structured mean field variational Bayes approximate posterior density function $\mathfrak{q}^*(\theta)$ of a generic continuous parameter θ is defined as

$$\text{accuracy}(\mathfrak{q}^*) = 100\left(1 - \frac{1}{2}\int_{-\infty}^{\infty}\left|\mathfrak{q}^*(\theta) - \mathfrak{p}(\theta|\boldsymbol{y})\right|d\theta\right)\%. \tag{12}$$

For the discrete parameter κ, an analogous definition applies with the integral replaced by the sum over $\mathcal{K}$. The density $\mathfrak{p}(\theta|\boldsymbol{y})$ was estimated from the Markov chain Monte Carlo samples for θ using the `bkde()` and `dpik()` kernel density estimation functions within the R package KernSmooth (Wand & Ripley, 2024).

Figure 2 displays the boxplots of the accuracy scores for estimation of the function $\exp(\eta_{\text{true},1} + \eta_{\text{true},2})$ evaluated at the sample quartiles of the x_{1i} and x_{2i} values. We use the notation Q_k, $k = 1, 2, 3$, to denote the quartiles. Accuracy scores for σ_1^2, σ_2^2 and κ were also obtained. The boxplots in Figure 2 indicate satisfactory accuracies for this simulation setting. When compared with Figure 4 of Luts & Wand (2015), improvements over that article's semiparametric mean field variational Bayes approach are observed, with the most notable gain being for κ.

Figure 3 shows the accuracy costs of the mean field approximation by comparing the $\mathfrak{q}$-density functions with those based on the Markov chain Monte Carlo for a randomly chosen replication. To aid visualisation we replaced the κ probability mass functions by polygons formed by joining each of the the atom/probability pairs. The polygons were then normalized to have areas under the curve equal to 1. It can be seen that variational approximate posterior densities are nearly as wide as the Markov chain Monte Carlo counterparts for the functions evaluated at the quartiles, and also centered about true values. However, some inaccuracy is apparent. The discrete densities of κ obtained via Algorithm 1 and Markov chain Monte Carlo and are very similar to each other and centered around κ^{true}.

5.1.2 Assessment of Speed

The execution time of Markov chain Monte Carlo represents a significant bottleneck in Bayesian analysis, which is one of the main motivations for exploring faster alternatives such as variational approximations. Algorithm 1 requires achieving convergence of $|\mathcal{K}|$ variational algorithms, where $|\mathcal{K}|$ is the cardinality of $\mathcal{K}$. We recommend the use of warm starts when marching, in order, through the κ atoms to accelerate convergence and thus reduce processing time. The overall computational burden is linear in $|\mathcal{K}|$, and appropriate choice of the κ prior can help prevent excessively long run times. That said, our method achieves very good speed. The average (standard deviation) elapsed time of the two methods across all 100 simulated datasets

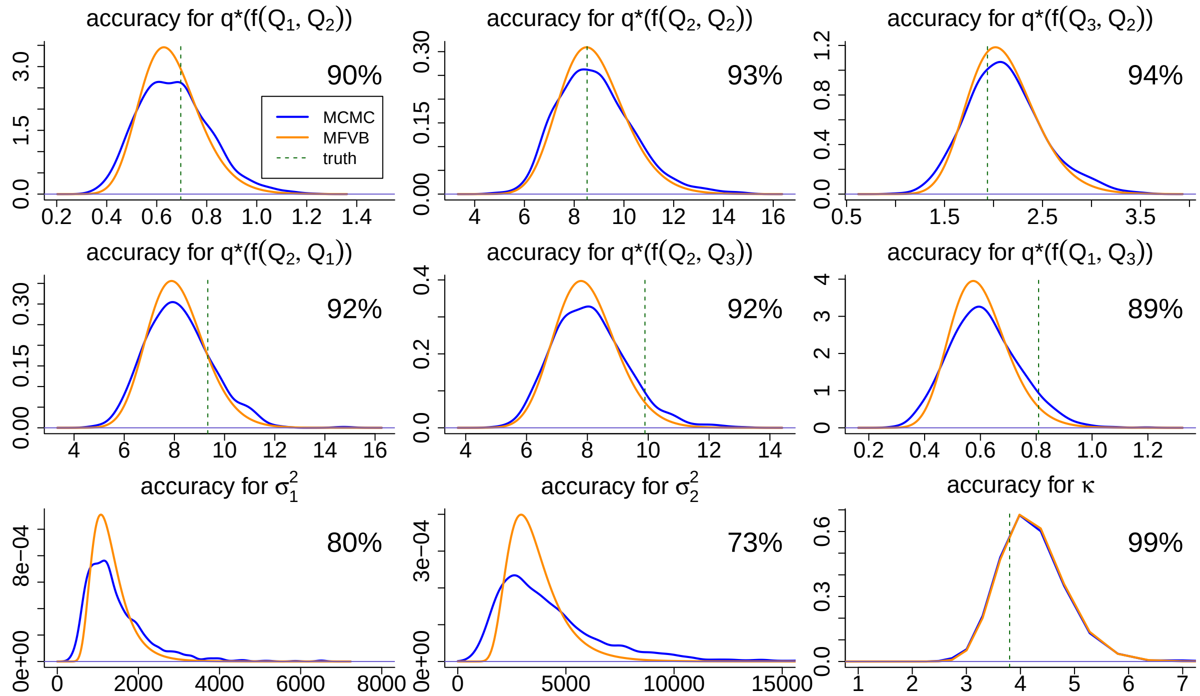

Figure 3: *Illustrations of the accuracy of the structured mean field variational Bayes (MFVB) posterior density functions and probability mass function approximations obtained from Algorithm 1. In each panel, the MFVB approximate density functions or probability mass function for a quantity of interest is compared with its Markov chain Monte Carlo (MCMC) counterpart. The percentage is the accuracy score according to (12). The vertical lines indicate true values according to the simulation set-up.*

is 117.8 (1.876) seconds for the Markov chain Monte Carlo approach, and 2.088 (0.1440) seconds for Algorithm 1. The simulations were run in R (R Core Team, 2025), version 4.4.3, on a machine with 12 cores and 24 gigabytes of random access memory.

5.2 Simulated Data Assessment of Algorithm 2

To assess the efficacy of Algorithm 2 we simulated data sets corresponding to the Negative binomial nonparametric regression model

$$y_i|x_i \overset{\text{ind.}}{\sim} \text{Negative-Binomial}\big(\eta_{\text{true}}(x_i), \kappa_{\text{true}}\big), \quad 1 \le i \le n_{\text{full}}, \tag{13}$$

where

$$\eta_{\text{true}}(x) \equiv 0.3\phi(x; 0.2, 0.08) - 0.3\phi(x; 0.65, 0.23) + 0.4\phi(x; 0.45, 0.08)$$

where $\phi(\cdot; \mu, \sigma)$ is as defined earlier in this section. In (13) n_{full} signifies the full real-time data sample size. In all of our examples we set $n_{\text{full}} = 1000$. The predictor data were generated according to $x_i \overset{\text{ind.}}{\sim} \text{Uniform}(0, 1)$.

Bayesian penalized splines of the form

$$\beta_0 + \beta_1\, x + \sum_{k=1}^{K} u_k z_k(x), \quad u_k|\sigma^2 \overset{\text{ind.}}{\sim} N(0, \sigma^2),$$

were used to model and estimate η_{true}. The z_k spline basis functions are analogous to those described in Section 5.1. In this real-time example we used 35 interior knots, which entails use of $K = 37$ basis functions. This set-up corresponds to the $r = 1$ special case of Algorithms 1 and 2. We also imposed the prior distributions:

$$\beta_0, \beta_1 \overset{\text{ind.}}{\sim} N(0, 10^5), \quad \sigma \sim \text{Half-Cauchy}(10^5) \quad \text{and} \quad \mathfrak{p}(\kappa) \propto \exp(-\kappa/100), \quad \kappa \in \mathcal{K}$$

where $\mathcal{K}$ is the geometric sequence of length 50 between $\kappa_{\text{true}}/10$ and $10\kappa_{\text{true}}$.

Real-time data scenarios based on (13) were simulated for each of

$$\kappa_{\text{true}} \in \{5, 10, 20, 40\},$$

corresponding to count responses with varying amounts of overdispersion. Three replications of (x_i, y_i), $1 \leq i \leq n_{\text{full}}$, data were generated within the R computing environment according to the command `set.seed(s)` with `s` set to each of 1, 2 and 3. These data were then fed into Algorithm 2 with samples of size $n = n_{\text{warm}}, n_{\text{warm}} + 1, \ldots, n_{\text{full}}$. In all but one case we used $n_{\text{warm}} = 100$ and achieved good convergence. The exception was $\kappa_{\text{true}} = 20$ with the third seed value, in which case the longer warm-up of $n_{\text{warm}} = 200$ was warranted. The same sequential data sets were fed into Algorithm 1 to allow comparison between the online fits and those obtained via ordinary batch processing.

The results of our simulation-based assessment of Algorithm 2 are presented as movies within the supplementary material. [1] Figure 4 shows some of the frames from the first $\kappa_{\text{true}} = 5$ movie.

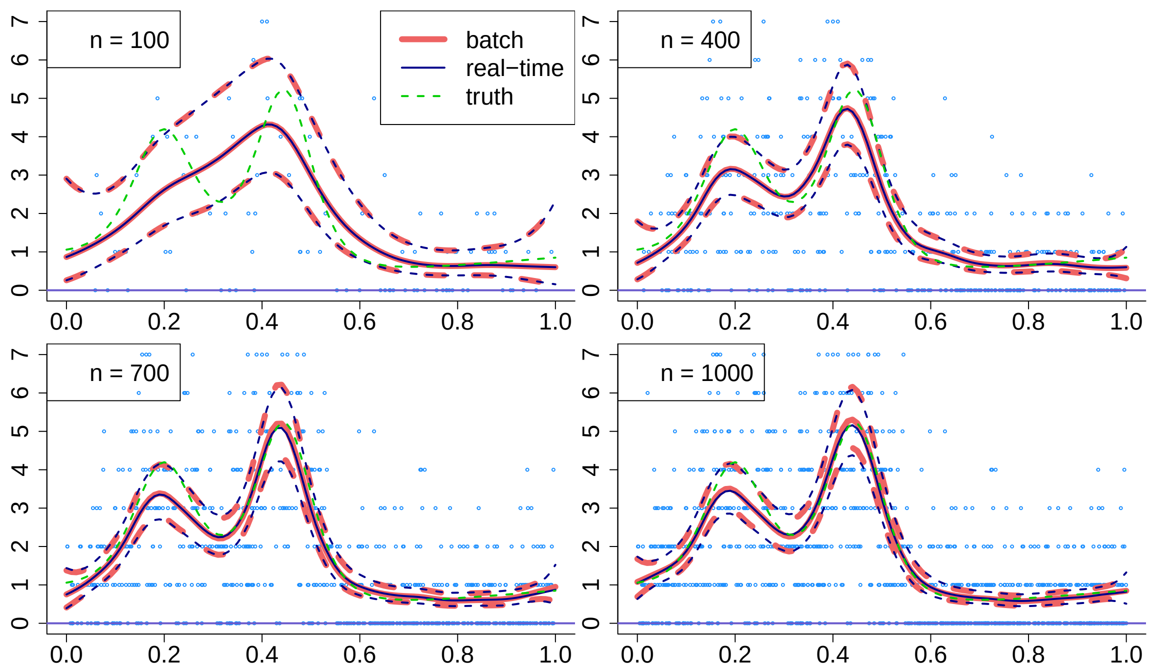

Figure 4: *Some illustrative comparisons for $\kappa_{true} = 5$ between the real-time Negative Binomial nonparametric regression estimates based on Algorithm 2 with the batch counterparts based on Algorithm 1. The solid curves correspond to posterior means. The dashed curves correspond to pointwise approximate 95% credible intervals. The scatterplots correspond to the current regression data.*

Figure 4 and, in particular, the movies show that Algorithm 2's real-time inference for the mean structure is quite similar to that obtained with successive batch fitting. Clearly, the former is preferable from a speed standpoint in streaming data applications. High quality real-time inference for the κ nuisance parameter appears to be out of reach. The strategy used in Figure 4 and the movies, and described in Section 4.1, aims to ensure that the running approximate posterior distributions of κ are reasonable enough to not adversely impact real-time estimation of the mean structure.

[1] For this pre-publication version of this article, the movies are on the following web-site: `https://matt-p-wand.net/M+Wmovies.html`

5.3 Application to Pollen Counts Data

This application involves data daily ragweed pollen counts in Kalamazoo, U.S.A., during the 1991–1994 ragweed seasons. The data correspond to the study described in Stark *et al.* (1997). The model is of the form

$$y_i \overset{\text{ind.}}{\sim} \text{Negative-Binomial}\big\{\exp\big(\beta_0 + \beta_1\,x_{1i} + \beta_2\,x_{2i} + \beta_3\,x_{3i} + \eta_{z_i}(x_{4i})\big), \kappa\big\}, \quad 1 \le i \le n, \tag{14}$$

where $n = 334$ corresponds to the total number of days when ragweed pollen was in season during 1991–1994. The variables in (14) are ragweed pollen count on the ith day (y_i), temperature residual on the ith day (x_{1i}), indicator of significant rain on the ith day (x_{2i}), wind speed in knots on the ith day (x_{3i}), day number of ragweed pollen season for the current year on which y_i was recorded (x_{4i}) and a categorical variable for the year in which y_i was recorded (one of 1991, 1992, 1993 or 1994) (z_i). Here temperature residuals are the residuals from fitting penalized splines, each having 5 effective degrees of freedom, to temperature (in degrees Fahrenheit) versus day number for each ragweed pollen season. Mixed model-based penalized splines were used for modelling the η_z, $z \in \{1991, 1992, 1993, 1994\}$. The variance parameters used in each of the four penalized spline components are σ_1^2, σ_2^2, σ_3^2 and σ_4^2. The full model is

$$\begin{aligned}
y_i|\boldsymbol{\beta}, \boldsymbol{u}, \kappa &\overset{\text{ind.}}{\sim} \text{Negative-Binomial}\big(\exp\{(\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{Z}\boldsymbol{u})_i\}, \kappa\big), \quad \boldsymbol{\beta} \sim N(\boldsymbol{0}, 10^{10}\boldsymbol{I})\\
\boldsymbol{u}|\sigma_1^2, \ldots, \sigma_4^2 &\sim N\big(\boldsymbol{0}, \text{blockdiag}(\sigma_1^2\,\boldsymbol{I}_{K_1}, \ldots, \sigma_4^2\,\boldsymbol{I}_{K_4})\big),\\
\sigma_1, \ldots, \sigma_4 &\overset{\text{ind.}}{\sim} \text{Half-Cauchy}(10^5), \quad \mathfrak{p}(\kappa) \propto \exp(-\kappa/100), \quad \kappa \in \mathcal{K}
\end{aligned} \tag{15}$$

where $\mathcal{K}$ is a geometric sequence of length 100 between 0.5 to 50. The design matrices in (15) are

$$\boldsymbol{X} = \begin{bmatrix} 1 & x_{11} & \cdots & x_{41} & I(z_1{=}1992) & x_{41}I(z_1{=}1992) & \cdots & I(z_1{=}1994) & x_{41}I(z_1{=}1994)\\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots\\ 1 & x_{1n} & \cdots & x_{4n} & I(z_n{=}1992) & x_{4n}I(z_n{=}1992) & \cdots & I(z_n{=}1994) & x_{4n}I(z_n{=}1994) \end{bmatrix}$$

and $\boldsymbol{Z} = [\boldsymbol{Z}_{1991}\,\boldsymbol{Z}_{1992}\,\boldsymbol{Z}_{1993}\,\boldsymbol{Z}_{1994}]$ where $\boldsymbol{Z}_{1991}$ is an $n \times K_j$ matrix with (i,k) entry equal to $I(z_i = 1991)z_k(x_{4i})$ and $\boldsymbol{Z}_{1992}, \ldots, \boldsymbol{Z}_{1994}$ are defined analogously. The z_k basis functions are of the same type used earlier in this section. The $\boldsymbol{\beta}$ and $\boldsymbol{u}$ vectors contain the coefficients to match the columns of $\boldsymbol{X}$ and $\boldsymbol{Z}$ respectively. Lastly, the spline basis sizes were $K_1 = K_2 = K_3 = 17$ and $K_4 = 16$. This difference in the spline basis sizes is due to the numbers days in the ragweed pollen season varying in length between the four years. They range from 78 for year 1994 to 92 to year 1991.

Model (15) is an $r = 4$ special case of (2). We used Algorithm 1 to perform approximate Bayesian inference.

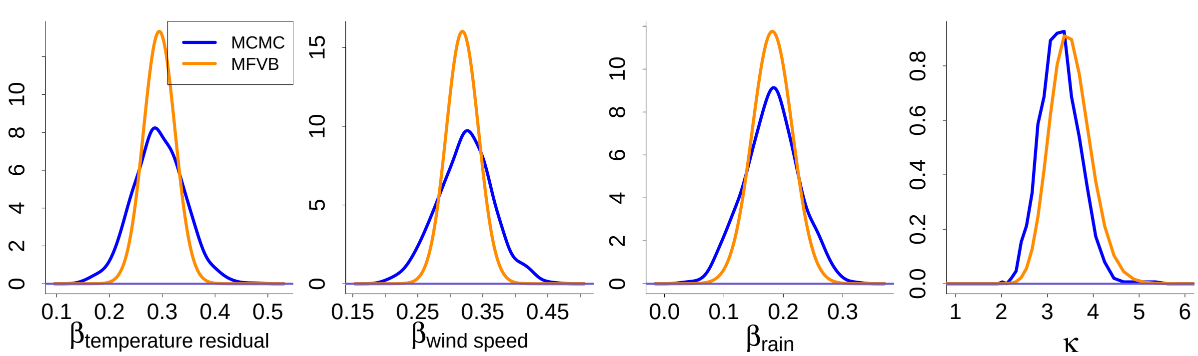

Figure 5: *Comparison of posterior density function and probability mass function approximations based on structured mean field variational Bayes (MFVB) and Markov chain Monte Carlo (MCMC) for four of the parameters in model (15).*

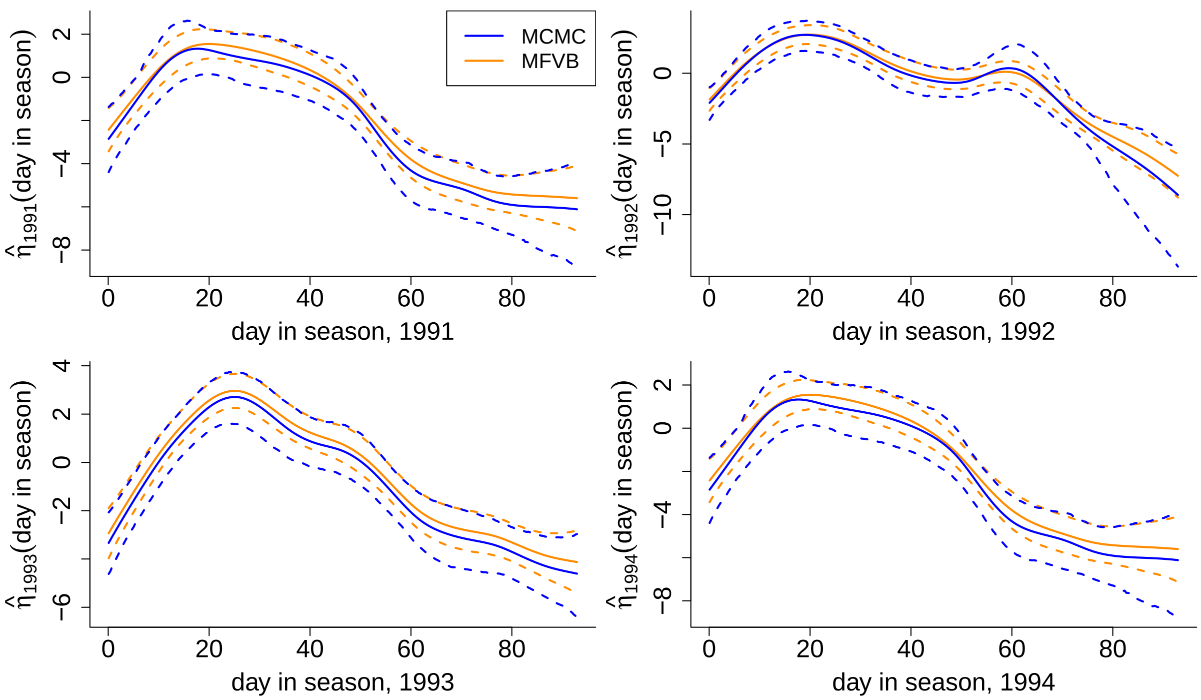

Figure 6: *Structured mean field variational Bayes (MFVB) and Markov chain Monte Carlo (MCMC) posterior means (solid lines) and 95% pointwise credible intervals (dashed lines) for estimation of the functions* $\eta_{1991}, \ldots, \eta_{1994}$ *in the model conveyed by (14) and (15).*

Figure 5 shows the posterior density functions for the parameters associated with the linear effects of the quantitative explanatory variables considered in the model. On the right, the posterior probability distribution function of κ is represented. It can be noticed that all the the coefficient parameters are significantly different from zero. The variational approximate density functions for the linear effects appear narrower than the Markov chain Monte Carlo ones; the approximate posterior distribution of κ appears slightly shifted to the right compared to the exact one. Both distributions assign posterior mass to values ranging from 2 to 5, supporting the suitability of the Negative-Binomial response model.

The trend of the linear predictor over the days of the season, shown separately for each year, is displayed in Figure 6. Solid lines indicate the posterior mean, and dashed lines the posterior pointwise credible intervals at level 95%. The four curves exhibit similar behavior, with a peak around the 20th day of the season, followed by a decreasing trend.

6 Conclusion

Our convex solution to the count response semiparametric regression problem described here offers stability and real-time processing advantages. The inferential accuracy is often reasonable. However, it also prone to some inaccuracy and this aspect needs to be taken into account when trading off against speed. Even though we have focussed on semiparametric regression models, the same general approach applies to numerous other count response settings.

Acknowledgement

We are grateful to Emanuele Aliverti for discussions related to this article. This research was partially supported by Australian Research Council grant DP230101179.

References

Bishop, C.M. (2006). *Pattern Recognition and Machine Learning*. New York: Springer.

Durante, D. & Rigon, T. (2019). Conditionally conjugate mean-field variational Bayes for logistic models. *Statistical Science*, **34**, 472–485.

Jaakkola, T.S. & Jordan, M.I. (2000). Bayesian parameter estimation via variational methods. *Statistics and Computing*, **10**, 25–37.

Lee, C.Y.Y. and Wand, M.P. (2016). Variational inference for fitting complex Bayesian mixed effects models to health data. *Statistics in Medicine*, **35**, 165–188.

Luts, J., Broderick, T. and Wand, M.P. (2014). Real-time semiparametric regression. *Journal of Computational and Graphical Statistics*, **23**, 589–615.

Luts, J. and Wand, M.P. (2015). Variational inference for count response semiparametric regression. *Bayesian Analysis*, **10**, 991–1023.

Miao, Y., Kook, J.H., Lu, Y., Guindani, M. & Vannucci, M. (2020). Scalable Bayesian variable selection models for count data. In Y. Fan, D.J. Nott, M.S. Smith, J.-C. Dortet-Bernadet, editors, *Flexible Bayesian Modelling*, pp. 187–219, London: Academic Press.

Ormerod, J.T. and Wand, M.P. (2010). Explaining variational approximations. *The American Statistician*, **64**, 140–153.

Pillow, J.W. & Scott, J.G. (2012). Fully Bayesian inference for neural models with negative-binomial spiking. In Bartlett, P., Pereira, F.C.N., Burges, C.J.C., Bottou, L. & Weinberger, K.Q., editors, In *Advances in Neural Information Processing Systems 25*, pp. 1898–1906. Cambridge, Massachusetts: MIT Press.

Plummer, M. (2025). rjags: Bayesian graphical models using Markov chain Monte Carlo. R package version 4-17. `CRAN.R-project.org/package=rjags`.

Polson, N.G., Scott, J.G. & Windle, J. (2013). Bayesian inference for logistic models using Pólya-Gamma latent variables. *Journal of the American Statistical Association*, **108**, 1339–1349.

R Core Team (2025). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. `https://www.R-project.org/`.

Saul, L.K. & Jordan, M.I. (1996). Exploiting tractable substructures in intractable networks. In Mozer, M.C., Jordan, M.I. & Petsche T., editors, *Advances in Neural Information Processing Systems 9*, pp. 435–442. Cambridge, Massachusetts: MIT Press.

Stark, P.C., Ryan, L.M., McDonald, J.L. & Burge, H.A. (1997). Using meteorologic data to model and predict daily ragweed pollen levels. *Aerobiologia*, **13**, 177–184.

Wainwright, M.J. & Jordan, M.I. (2008). Graphical models, exponential families and variational inference. *Foundations and Trends in Machine Learning*, **1**, 1–305.

Wand, M.P. and Ormerod, J.T. (2008). On semiparametric regression with O'Sullivan penalized splines. *Australian and New Zealand Journal of Statistics*, **50**, 179–198.

Wand, M.P., Ormerod, J.T., Padoan, S.A. & Frühwirth, R. (2011). Mean field variational Bayes for elaborate distributions. *Bayesian Analysis*, **6**, 847–900.

Wand, M. & Ripley, B. (2024). KernSmooth: Functions for kernel smoothing supporting Wand & Jones (1995). R package version 4-17. `CRAN.R-project.org/package=KernSmooth`.

Wand, M.P. and Yu, J.C.F. (2022). Density estimation via Bayesian inference engines. *Advances in Statistical Analysis*, **106**, 199–216.

Zhao, Y., Staudenmayer, J., Coull, B.A. and Wand, M.P. (2006). General design Bayesian generalized linear mixed models. *Statistical Science*, **21**, 35–51.

Zhou, M. Li, L., Dunson, D. & Carin, L. (2012). Lognormal and Gamma mixed Negative Binomial regression. In *Proceedings of the 29th International Conference on Machine Learning*, pp. 1343–1350.

Supplement for:

Variational Inference for Count Response Semiparametric Regression: A Convex Solution

VIRGINIA MURRU AND MATT P. WAND

Università di Padova and University of Technology Sydney

S.1 Pólya-Gamma Distribution Definitions and Results

The Pólya-Gamma distribution plays an important role in this article's variational inference approach. In this section we provide relevant definitions and results.

S.1.1 The Gamma Distribution

A random variable x has a Gamma distribution with shape parameter $\alpha > 0$ and rate parameter $\beta > 0$, written

$$x \sim \text{Gamma}(\alpha, \beta),$$

if and only if its probability density function is

$$\mathfrak{p}(x; \alpha, \beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} \, x^{\alpha-1} \exp(-\beta x), \quad x > 0.$$

S.1.2 The Pólya-Gamma Distribution

A random variable x has a Pólya-Gamma distribution with shape parameter $b > 0$ and tilting parameter $c > 0$, written

$$x \sim \text{Pólya-Gamma}(b, c),$$

if and only if

$$x \quad \text{is equal in distribution to} \quad \frac{1}{2\pi^2} \sum_{k=1}^{\infty} \frac{g_k}{(k - 1/2)^2 + c^2/(4\pi^2)}$$

where

$$g_k \overset{\text{ind.}}{\sim} \text{Gamma}(b, 1), \quad k = 1, 2, \ldots$$

Note that there is no closed form expression for the density function of a Pólya-Gamma(b, c) random variable.

S.1.2.1 A Decomposition of the General Pólya Gamma Density Function

Let $\mathfrak{p}_{\text{PG}}(\,\cdot\,; b, c)$ denote the density function of a Pólya-Gamma(b, c) random variable. Then for all $x, b > 0$ and $c \in \mathbb{R}$:

$$\mathfrak{p}_{\text{PG}}(x; b, c) = \cosh^b(c/2) \exp\left(-\tfrac{1}{2}c^2 x\right) \mathfrak{p}_{\text{PG}}(x; b, 0). \tag{S.1}$$

This result corresponds to equation (5) of Polson *et al.* (2013).

S.1.2.2 The Mean of a Pólya Gamma Random Variable

From Section 2.3 of Polson *et al.* (2013), if

$$x \sim \text{Pólya-Gamma}(b, c) \quad \text{then} \quad E(x) = \frac{b}{2c}\tanh(c/2) = 2b\lambda_{\text{JJ}}(c). \tag{S.2}$$

S.2 Impracticality of Ordinary Mean Field Variational Bayes

The impracticality of ordinary mean field variational Bayes for model (2) stems from the following fact mentioned in Section S.1: Pólya Gamma density functions do not admit closed forms. In this section we provide relevant details.

From Figure 1, the Markov blanket of κ is $\{\boldsymbol{y}, \boldsymbol{\beta}, \boldsymbol{u}, \boldsymbol{\alpha}\}$. Therefore

$$\mathfrak{p}(\kappa|\text{rest}) = \mathfrak{p}(\kappa|\boldsymbol{y}, \boldsymbol{\beta}, \boldsymbol{u}, \boldsymbol{\alpha}) \propto \mathfrak{p}(\boldsymbol{y}|\boldsymbol{\beta}, \boldsymbol{u}, \kappa)\mathfrak{p}(\boldsymbol{\alpha}|\boldsymbol{y}, \boldsymbol{\beta}, \boldsymbol{u}, \kappa)\mathfrak{p}(\kappa).$$

The factors $\mathfrak{p}(\boldsymbol{y}|\boldsymbol{\beta}, \boldsymbol{u}, \kappa)$ and $\mathfrak{p}(\kappa)$ have simple closed form expressions, but

$$\mathfrak{p}(\boldsymbol{\alpha}|\boldsymbol{y}, \boldsymbol{\beta}, \boldsymbol{u}, \kappa) = \prod_{i=1}^{n} \mathfrak{p}_{\text{PG}}\big(\alpha_i; y_i + \kappa, (\boldsymbol{X\beta} + \boldsymbol{Zu})_i + \log(\kappa)\big)$$

depends on intractable Pólya-Gamma density functions. This hinders practical mean field variational Bayes for model (2).

S.3 The Structured Mean Field Variational Bayes Alternative

The structured mean field variational Bayes alternative goes back to machine learning articles such as Saul & Jordan (1996) and Jaakkola (2001) with application to, for example, coupled hidden Markov models. Section 3.1 of Wand *et al.* (2011) describes a structured mean field variational Bayes paradigm for Bayesian hierarchical models within the field of statistics. The derivations and subsequent optimal q-density formulae given in Section 3.1 of Wand *et al.* (2011) are the basis for Algorithm 1.

S.4 Algorithm 1 Justification

We now justify the steps given in Algorithm 1. There are two main components: (1) the optimal $\mathfrak{q}$-density derivations which lead to the Algorithm 1's coordinate ascent scheme and (2) the explicit expression for the marginal log-likelihood conditional on κ, which form the basis for the structured mean field variational Bayes posterior density approximations.

Throughout this section we let 'rest' denote all random variable in the model other then the random vector of current interest. Also, $E_{\mathfrak{q}(-\boldsymbol{\theta})}$ signifies expectation with respect to the joint $\mathfrak{q}$-density function of all model parameters but with $\boldsymbol{\theta}$ omitted.

S.4.1 Optimal $\mathfrak{q}$-Density Derivations

We now provide the derivation of each of the optimal $\mathfrak{q}$-density functions.

Derivation of $\mathfrak{q}^*(\boldsymbol{\alpha}|\kappa)$

It follows from the second line of (2) that

$$\log\{\mathfrak{p}(\boldsymbol{\alpha}|\text{rest})\} = \sum_{i=1}^{n} \log\big\{\mathfrak{p}_{\text{PG}}\big(\alpha_i; y_i + \kappa, (\boldsymbol{X\beta} + \boldsymbol{Zu})_i - \log(\kappa)\big)\big\}.$$

In view of (S.1) we then have

$$\log\{\mathfrak{p}(\alpha_i|\text{rest})\} = \log\left\{\mathfrak{p}_{\text{PG}}\big(\alpha_i; y_i+\kappa, (\boldsymbol{X\beta}+\boldsymbol{Zu})_i - \log(\kappa)\big)\right\}$$
$$= -\tfrac{1}{2}\big\{(\boldsymbol{X\beta}+\boldsymbol{Zu})_i - \log(\kappa)\big\}^2\alpha_i + \log\{\mathfrak{p}_{\text{PG}}(\alpha_i; y_i+\kappa, 0)\} + \text{const}$$

where 'const' denotes terms that do not depend on α_i. Therefore,

$$E_{\mathfrak{q}(-(\alpha_i,\kappa))}\big[\log\{\mathfrak{p}(\alpha_i|\text{rest})\}\big] = -\tfrac{1}{2}E_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}\big[\big\{(\boldsymbol{X\beta}+\boldsymbol{Zu})_i - \log(\kappa)\big\}^2\big]\alpha_i$$
$$+ \log\{\mathfrak{p}_{\text{PG}}(\alpha_i; y_i+\kappa, 0)\} + \text{const}.$$

It quickly follows that, for each $1 \le i \le n$,

$$\mathfrak{q}^*(\alpha_i|\kappa) \ \text{ is the Pólya-Gamma}\big(y_i+\kappa, c_{\mathfrak{q}(\alpha_i|\kappa)}\big) \ \text{ density function}$$

where

$$c_{\mathfrak{q}(\alpha_i|\kappa)} \equiv \sqrt{E_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}\big[\big\{(\boldsymbol{X\beta}+\boldsymbol{Zu})_i - \log(\kappa)\big\}^2\big]}$$

and that

$$\mathfrak{q}^*(\boldsymbol{\alpha}|\kappa) = \prod_{i=1}^n \mathfrak{q}^*(\alpha_i|\kappa).$$

Next note that

$$E_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}\big[\big\{(\boldsymbol{X\beta}+\boldsymbol{Zu})_i - \log(\kappa)\big\}^2 = \text{Var}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}\big(\boldsymbol{C}[\boldsymbol{\beta}^T\ \boldsymbol{u}^T]^T)_i\big)$$
$$+\big\{\big(\boldsymbol{C}E_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}[\boldsymbol{\beta}^T\ \boldsymbol{u}^T]^T\big)_i - \log(\kappa)\big\}^2$$
$$= \big(\boldsymbol{C}\boldsymbol{\Sigma}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}\boldsymbol{C}^T\big)_{ii} + \big\{\big(\boldsymbol{C}\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}\big)_i - \log(\kappa)\big\}^2$$
$$= \Big(\text{diagonal}\big(\boldsymbol{C}\boldsymbol{\Sigma}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}\boldsymbol{C}^T\big) + \big(\boldsymbol{C}\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)} - \log(\kappa)\boldsymbol{1}\big)^2\Big)_i.$$

Hence

$$c_{\mathfrak{q}(\alpha_i|\kappa)} = \sqrt{\Big(\text{diagonal}\big(\boldsymbol{C}\boldsymbol{\Sigma}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}\boldsymbol{C}^T\big) + \big(\boldsymbol{C}\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)} - \log(\kappa)\boldsymbol{1}\big)^2\Big)_i}.$$

From (S.2),

$$\mu_{\mathfrak{q}(\alpha_i)} = 2(y_i+\kappa)\lambda_{\text{JJ}}\big(c_{\mathfrak{q}(\alpha_i|\kappa)}\big), \quad 1 \le i \le n,$$

and so

$$\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)} = 2(\boldsymbol{y}+\kappa\boldsymbol{1}) \odot \lambda_{\text{JJ}}\big(c_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)}\big)$$

with

$$\boldsymbol{c}_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)} = \sqrt{\text{diagonal}\big(\boldsymbol{C}\boldsymbol{\Sigma}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}\boldsymbol{C}^T\big) + \big(\boldsymbol{C}\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)} - \log(\kappa)\boldsymbol{1}\big)^2}. \qquad \text{(S.3)}$$

Derivation of $\mathfrak{q}^*(\boldsymbol{\beta},\boldsymbol{u}|\kappa)$

First note that the ith contribution to the likelihood part of (2), $1 \le i \le n$, can be written

$$\mathfrak{p}(y_i|\boldsymbol{\beta},\boldsymbol{u},\kappa) = \frac{\kappa^\kappa\Gamma(y_i+\kappa)\exp\big((\boldsymbol{X\beta}+\boldsymbol{Zu})_i\big)^{y_i}}{\Gamma(\kappa)\Big(\kappa+\exp\big((\boldsymbol{X\beta}+\boldsymbol{Zu})_i\big)\Big)^{y_i+\kappa}\Gamma(y_i+1)}$$
$$= \frac{\Gamma(y_i+\kappa)}{\Gamma(\kappa)\Gamma(y_i+1)}\,\frac{\exp\big((\boldsymbol{X\beta}+\boldsymbol{Zu})_i - \log(\kappa)\big)^{y_i}}{\big\{1+\exp\big((\boldsymbol{X\beta}+\boldsymbol{Zu})_i - \log(\kappa)\big)\big\}^{\kappa+y_i}}$$
$$= \frac{\Gamma(y_i+\kappa)}{2^{y_i+\kappa}\Gamma(\kappa)\Gamma(y_i+1)}\,\frac{\exp\big\{\frac{1}{2}(y_i-\kappa)\big((\boldsymbol{X\beta}+\boldsymbol{Zu})_i - \log(\kappa)\big)\big\}}{\cosh^{y_i+\kappa}\Big(\frac{1}{2}\big((\boldsymbol{X\beta}+\boldsymbol{Zu})_i - \log(\kappa)\big)\Big)}.$$

Then the full conditional density function of $(\boldsymbol{\beta},\boldsymbol{u})$ is

$$
\begin{aligned}
\mathfrak{p}(\boldsymbol{\beta},\boldsymbol{u}|\text{rest}) &= \mathfrak{p}(\boldsymbol{\beta},\boldsymbol{u}|\boldsymbol{y},\boldsymbol{\alpha},\boldsymbol{\sigma}^2,\kappa) \propto \mathfrak{p}(\boldsymbol{y}|\boldsymbol{\beta},\boldsymbol{u},\kappa)\mathfrak{p}(\boldsymbol{\alpha}|\boldsymbol{y},\boldsymbol{\beta},\boldsymbol{u},\kappa)\mathfrak{p}(\boldsymbol{\beta})\mathfrak{p}(\boldsymbol{u}|\boldsymbol{\sigma}^2)\\
&= \left(\prod_{i=1}^{n}\mathfrak{p}(y_i|\boldsymbol{\beta},\boldsymbol{u},\kappa)\mathfrak{p}_{\text{PG}}\big(\alpha_i;y_i+\kappa,(\boldsymbol{X\beta}+\boldsymbol{Zu})_i-\log(\kappa)\big)\right)\mathfrak{p}(\boldsymbol{\beta})\mathfrak{p}(\boldsymbol{u}|\boldsymbol{\sigma}^2)\\
&\propto \left(\prod_{i=1}^{n}\frac{\exp\left\{\frac{1}{2}(y_i-\kappa)\big((\boldsymbol{X\beta}+\boldsymbol{Zu})_i-\log(\kappa)\big)\right\}}{\cosh^{y_i+\kappa}\left(\frac{1}{2}\big((\boldsymbol{X\beta}+\boldsymbol{Zu})_i-\log(\kappa)\big)\right)}\right.\\
&\qquad\times\cosh^{y_i+\kappa}\left(\tfrac{1}{2}\big((\boldsymbol{X\beta}+\boldsymbol{Zu})_i-\log(\kappa)\big)\right)\\
&\qquad\left.\times\exp\left[-\tfrac{1}{2}\big\{(\boldsymbol{X\beta}+\boldsymbol{Zu})_i-\log(\kappa)\big\}^2\alpha_i\right]\mathfrak{p}_{\text{PG}}(\alpha_i;y_i+\kappa,0)\right)\mathfrak{p}(\boldsymbol{\beta})\mathfrak{p}(\boldsymbol{u}|\boldsymbol{\sigma}^2)
\end{aligned}
$$

where the last step follows from (S.1). We then have

$$
\begin{aligned}
\mathfrak{p}(\boldsymbol{\beta},\boldsymbol{u}|\text{rest}) &\propto \left(\prod_{i=1}^{n}\exp\left[\tfrac{1}{2}(y_i-\kappa)(\boldsymbol{X\beta}+\boldsymbol{Zu})_i-\tfrac{1}{2}\big\{(\boldsymbol{X\beta}+\boldsymbol{Zu})_i-\log(\kappa)\big\}^2\alpha_i\right]\right)\\
&\qquad\times\exp\left(-\frac{\|\boldsymbol{\beta}\|^2}{2\sigma_\beta^2}-\sum_{j=1}^{r}\frac{\|\boldsymbol{u}_j\|^2}{2\sigma_j^2}\right)\\
&\propto \left(\prod_{i=1}^{n}\exp\left[\tfrac{1}{2}(y_i-\kappa)(\boldsymbol{X\beta}+\boldsymbol{Zu})_i+\alpha_i\big\{\log(\kappa)(\boldsymbol{X\beta}+\boldsymbol{Zu})_i-\tfrac{1}{2}(\boldsymbol{X\beta}+\boldsymbol{Zu})_i^2\big\}\right]\right)\\
&\qquad\times\exp\left(-\frac{\|\boldsymbol{\beta}\|^2}{2\sigma_\beta^2}-\sum_{j=1}^{r}\frac{\|\boldsymbol{u}_j\|^2}{2\sigma_j^2}\right)\\
&=\exp\left[\big\{\tfrac{1}{2}(\boldsymbol{y}-\kappa\boldsymbol{1})+\log(\kappa)\boldsymbol{\alpha}\big\}^T\boldsymbol{C}\begin{bmatrix}\boldsymbol{\beta}\\ \boldsymbol{u}\end{bmatrix}-\tfrac{1}{2}\begin{bmatrix}\boldsymbol{\beta}\\ \boldsymbol{u}\end{bmatrix}^T\boldsymbol{C}^T\text{diag}(\boldsymbol{\alpha})\boldsymbol{C}\begin{bmatrix}\boldsymbol{\beta}\\ \boldsymbol{u}\end{bmatrix}\right.\\
&\qquad\qquad\left.-\frac{\|\boldsymbol{\beta}\|^2}{2\sigma_\beta^2}-\sum_{j=1}^{r}\frac{\|\boldsymbol{u}_j\|^2}{2\sigma_j^2}\right]\\
&=\exp\left\{\begin{bmatrix}\begin{bmatrix}\boldsymbol{\beta}\\ \boldsymbol{u}\end{bmatrix}\\ \text{vec}\left(\begin{bmatrix}\boldsymbol{\beta}\\ \boldsymbol{u}\end{bmatrix}\begin{bmatrix}\boldsymbol{\beta}\\ \boldsymbol{u}\end{bmatrix}^T\right)\end{bmatrix}^T\begin{bmatrix}\boldsymbol{C}^T\big\{\frac{1}{2}(\boldsymbol{y}-\kappa\boldsymbol{1})+\log(\kappa)\boldsymbol{\alpha}\big\}\\ -\frac{1}{2}\text{vec}\left(\boldsymbol{C}^T\text{diag}(\boldsymbol{\alpha})\boldsymbol{C}+\widetilde{\boldsymbol{M}}\right)\end{bmatrix}\right\}.
\end{aligned}
$$

where

$$
\widetilde{\boldsymbol{M}}\equiv\text{blockdiag}(\sigma_\beta^{-2}\boldsymbol{I}_p,\sigma_1^{-2}\boldsymbol{I}_{K_1},\ldots,\sigma_r^{-2}\boldsymbol{I}_{K_r}).
$$

Therefore,

$$
\begin{aligned}
&E_{\mathfrak{q}(-(\boldsymbol{\beta},\boldsymbol{u},\kappa))}\big[\log\{\mathfrak{p}(\boldsymbol{\beta},\boldsymbol{u}|\text{rest})\}\big]\\
&=\begin{bmatrix}\begin{bmatrix}\boldsymbol{\beta}\\ \boldsymbol{u}\end{bmatrix}\\ \text{vec}\left(\begin{bmatrix}\boldsymbol{\beta}\\ \boldsymbol{u}\end{bmatrix}\begin{bmatrix}\boldsymbol{\beta}\\ \boldsymbol{u}\end{bmatrix}^T\right)\end{bmatrix}^T\begin{bmatrix}\boldsymbol{C}^T\big\{\frac{1}{2}(\boldsymbol{y}-\kappa\boldsymbol{1})+\log(\kappa)\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)}\big\}\\ -\frac{1}{2}\text{vec}\left(\boldsymbol{C}^T\text{diag}\big(\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)}\big)\boldsymbol{C}+\boldsymbol{M}_{\mathfrak{q}(1/\boldsymbol{\sigma}^2|\kappa)}\right)\end{bmatrix}+\text{const}
\end{aligned}
$$

where $\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)}$ denotes the mean of the optimal $\mathfrak{q}$-density of $\boldsymbol{\alpha}|\kappa$ and 'const' denotes terms that do not depend on $(\boldsymbol{\beta},\boldsymbol{u})$. It follows quickly that

$$
\mathfrak{q}^*(\boldsymbol{\beta},\boldsymbol{u}|\kappa)\text{ is the }N\big(\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)},\boldsymbol{\Sigma}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}\big)\text{ density function}
$$

where

$$\boldsymbol{\Sigma}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)} \equiv \Big\{\boldsymbol{C}^T\mathrm{diag}\big(\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)}\big)\boldsymbol{C} + \boldsymbol{M}_{\mathfrak{q}(1/\boldsymbol{\sigma}^2)}\Big\}^{-1}$$

and

$$\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)} \equiv \boldsymbol{\Sigma}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}\boldsymbol{C}^T\big\{\tfrac{1}{2}(\boldsymbol{y}-\kappa\boldsymbol{1}) + \log(\kappa)\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)}\big\}.$$

Derivation of $\mathfrak{q}^*(\boldsymbol{\sigma}^2|\kappa)$

Arguments similar to those given in Appendix C of Wand & Ormerod (2011) lead to

$$\mathfrak{q}^*(\boldsymbol{\sigma}^2|\kappa) = \prod_{j=1}^{r}\mathfrak{q}^*(\sigma_j^2|\kappa)$$

where, for $1 \le j \le r$,

$$\mathfrak{q}^*(\sigma_j^2|\kappa) \quad \text{is the} \quad \text{Inverse-Gamma}\Big(\tfrac{1}{2}(K_j+1), \mu_{\mathfrak{q}(1/a_j|\kappa)} + \tfrac{1}{2}\big\{\|\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{u}_j|\kappa)}\|^2 + \mathrm{tr}(\boldsymbol{\Sigma}_{\mathfrak{q}(\boldsymbol{u}_j|\kappa)})\big\}\Big)$$

density function. Note that $\mu_{\mathfrak{q}(1/a_j|\kappa)}$ is the mean of $1/a_j$ according to the optimal $\mathfrak{q}^*$-density described next. Also, $\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{u}_j|\kappa)}$ and $\boldsymbol{\Sigma}_{\mathfrak{q}(\boldsymbol{u}_j|\kappa)}$ are the sub-matrices of $\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}$ and $\boldsymbol{\Sigma}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}$ according to the partition of $\boldsymbol{u}$ given at (8). The reciprocal moment of σ_j^2, according to $\mathfrak{q}^*(\sigma_j^2|\kappa)$, is

$$\mu_{\mathfrak{q}(1/\sigma_j^2|\kappa)} = \frac{K_j+1}{2\mu_{\mathfrak{q}(1/a_j|\kappa)} + \|\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{u}_j|\kappa)}\|^2 + \mathrm{tr}(\boldsymbol{\Sigma}_{\mathfrak{q}(\boldsymbol{u}_j|\kappa)})}.$$

Derivation of $\mathfrak{q}^*(\boldsymbol{a}|\kappa)$

Steps provided by Appendix C of Wand & Ormerod (2011) lead to

$$\mathfrak{q}^*(\boldsymbol{a}|\kappa) = \prod_{j=1}^{r}\mathfrak{q}^*(a_j|\kappa)$$

where, for $1 \le j \le r$,

$$\mathfrak{q}^*(a_j|\kappa) \quad \text{is the} \quad \text{Inverse-Gamma}\Big(1, \mu_{\mathfrak{q}(1/\sigma_j^2|\kappa)} + s_\sigma^{-2}\Big)$$

density function. The reciprocal moment of a_j, according to $\mathfrak{q}^*(a_j|\kappa)$, is

$$\mu_{\mathfrak{q}(1/a_j|\kappa)} = \frac{1}{\mu_{\mathfrak{q}(1/\sigma_j^2|\kappa)} + s_\sigma^{-2}}.$$

S.4.2 The Approximate Marginal Log-Likelihood

The approximate marginal log-likelihood, conditional on κ, is

$$\begin{aligned}
\log\{\underline{\mathfrak{p}}(\boldsymbol{y}|\kappa)\} &= E_{\mathfrak{q}(-\kappa)}\big[\log\{\mathfrak{p}(\boldsymbol{y},\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{u},\boldsymbol{\sigma}^2,\boldsymbol{a}|\kappa)\} - \log\{\mathfrak{q}(\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{u},\boldsymbol{\sigma}^2,\boldsymbol{a}|\kappa)\}\big] \\
&= E_{\mathfrak{q}(-\kappa)}\big[\log\{\mathfrak{p}(\boldsymbol{y}|\boldsymbol{\beta},\boldsymbol{u},\kappa)\mathfrak{p}(\boldsymbol{\alpha}|\boldsymbol{y},\boldsymbol{\beta},\boldsymbol{u},\kappa)\mathfrak{p}(\boldsymbol{\beta},\boldsymbol{u}|\boldsymbol{\sigma}^2,\kappa)\mathfrak{p}(\boldsymbol{\sigma}^2|\boldsymbol{a},\kappa)\mathfrak{p}(\boldsymbol{a}|\kappa)\} \\
&\qquad - \log\{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)\mathfrak{q}(\boldsymbol{\sigma}^2|\kappa)\mathfrak{q}(\boldsymbol{a}|\kappa)\}\big] \\
&= E_{\mathfrak{q}(-\kappa)}\big[\log\{\mathfrak{p}(\boldsymbol{y}|\boldsymbol{\beta},\boldsymbol{u},\kappa)\} + \log\{\mathfrak{p}(\boldsymbol{\alpha}|\boldsymbol{y},\boldsymbol{\beta},\boldsymbol{u},\kappa)\} - \log\{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)\} \\
&\qquad + \log\{\mathfrak{p}(\boldsymbol{\beta},\boldsymbol{u}|\boldsymbol{\sigma}^2,\kappa)\} - \log\{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)\} + \log\{\mathfrak{p}(\boldsymbol{\sigma}^2|\boldsymbol{a},\kappa)\} - \log\{\mathfrak{q}(\boldsymbol{\sigma}^2|\kappa)\} \\
&\qquad + \log\{\mathfrak{p}(\boldsymbol{a}|\kappa)\} - \log\{\mathfrak{q}(\boldsymbol{a}|\kappa)\}\big].
\end{aligned}$$

Simplification of $E_{\mathfrak{q}(-\kappa)}\big[\log\{\mathfrak{p}(\boldsymbol{y}|\boldsymbol{\beta},\boldsymbol{u},\kappa)\}\big]$

Since

$$\mathfrak{p}(y_i|\boldsymbol{\beta},\boldsymbol{u},\kappa)=\frac{\Gamma(y_i+\kappa)}{2^{y_i+\kappa}\Gamma(\kappa)\Gamma(y_i+1)}\;\frac{\exp\left\{\tfrac{1}{2}(y_i-\kappa)\big((\boldsymbol{X\beta}+\boldsymbol{Zu})_i-\log(\kappa)\big)\right\}}{\cosh^{y_i+\kappa}\left(\tfrac{1}{2}\big((\boldsymbol{X\beta}+\boldsymbol{Zu})_i-\log(\kappa)\big)\right)}$$

we have

$$\begin{aligned}\log\{\mathfrak{p}(\boldsymbol{y}|\boldsymbol{\beta},\boldsymbol{u},\kappa)\}=\sum_{i=1}^{n}\Big[&\log\{\Gamma(y_i+\kappa)\}-(y_i+\kappa)\log(2)-\log\{\Gamma(y_i+1)\}\\&+\tfrac{1}{2}(y_i-\kappa)\big((\boldsymbol{X\beta}+\boldsymbol{Zu})_i-\log(\kappa)\big)\\&-(y_i+\kappa)\log\left\{\cosh\left(\tfrac{1}{2}\big((\boldsymbol{X\beta}+\boldsymbol{Zu})_i-\log(\kappa)\big)\right)\right\}\Big]-n\log\{\Gamma(\kappa)\}.\end{aligned}$$

This leads to

$$\begin{aligned}E_{\mathfrak{q}(-\kappa)}\big[\log\{\mathfrak{p}(\boldsymbol{y}|\boldsymbol{\beta},\boldsymbol{u},\kappa)\}\big]=&-(\boldsymbol{y}^T\boldsymbol{1})\log(2)-\boldsymbol{1}^T\log\{\Gamma(\boldsymbol{y}+\boldsymbol{1})\}+\boldsymbol{1}^T\log\{\Gamma(\boldsymbol{y}+\kappa\boldsymbol{1})\}\\&+n\big[\tfrac{1}{2}\kappa\log(\kappa)-\log(2)\kappa-\log\{\Gamma(\kappa)\}\big]\\&-\tfrac{1}{2}(\boldsymbol{y}^T\boldsymbol{1})\log(\kappa)+\tfrac{1}{2}\boldsymbol{\mu}^T_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}\big(\boldsymbol{C}^T\boldsymbol{y}-\kappa\,\boldsymbol{C}^T\boldsymbol{1}\big)\\&-\sum_{i=1}^{n}E_{\mathfrak{q}(-\kappa)}\left[(y_i+\kappa)\log\left\{\cosh\left(\tfrac{1}{2}\big((\boldsymbol{X\beta}+\boldsymbol{Zu})_i-\log(\kappa)\big)\right)\right\}\right].\end{aligned}$$

Simplification of $E_{\mathfrak{q}(-\kappa)}\big[\log\{\mathfrak{p}(\boldsymbol{\alpha}|\boldsymbol{y},\boldsymbol{\beta},\boldsymbol{u},\kappa)\}-\log\{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)\}\big]$

First note that

$$\begin{aligned}&\log\{\mathfrak{p}(\boldsymbol{\alpha}|\boldsymbol{y},\boldsymbol{\beta},\boldsymbol{u},\kappa)\}-\log\{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)\}\\&\quad=\sum_{i=1}^{n}\Big[\log\left\{\mathfrak{p}_{\text{PG}}(\alpha_i;y_i+\kappa,(\boldsymbol{X\beta}+\boldsymbol{Zu})_i-\log(\kappa))\right\}-\log\left\{\mathfrak{p}_{\text{PG}}(\alpha_i;y_i+\kappa,c_{\mathfrak{q}(\alpha_i|\kappa)})\right\}\Big].\end{aligned}$$

Since

$$\begin{aligned}&\log\{\mathfrak{p}_{\text{PG}}(\alpha_i;y_i+\kappa,(\boldsymbol{X\beta}+\boldsymbol{Zu})_i-\log(\kappa))\}\\&\quad=(y_i+\kappa)\log\left\{\cosh\left(\tfrac{1}{2}\big((\boldsymbol{X\beta}+\boldsymbol{Zu})_i-\log(\kappa)\big)\right)\right\}-\tfrac{1}{2}\big((\boldsymbol{X\beta}+\boldsymbol{Zu})_i-\log(\kappa)\big)^2\alpha_i\\&\qquad-\log\{\mathfrak{p}_{\text{PG}}(\alpha_i;y_i+\kappa,0)\}\end{aligned}$$

and

$$\begin{aligned}&\log\{\mathfrak{p}_{\text{PG}}(\alpha_i;y_i+\kappa,c_{\mathfrak{q}(\alpha_i|\kappa)}\}\\&\quad=(y_i+\kappa)\log\left\{\cosh\left(\tfrac{1}{2}c_{\mathfrak{q}(\alpha_i|\kappa)}\right)\right\}-\tfrac{1}{2}c^2_{\mathfrak{q}(\alpha_i|\kappa)}\alpha_i-\log\{\mathfrak{p}_{\text{PG}}(\alpha_i;y_i+\kappa,0)\}\end{aligned}$$

we get the cancellation of the $\log\{\mathfrak{p}_{\text{PG}}(\alpha_i;y_i+\kappa,0)\}$ terms and the following explicit expression for the log-density difference:

$$\begin{aligned}&\log\left\{\mathfrak{p}_{\text{PG}}(\alpha_i;y_i+\kappa,(\boldsymbol{X\beta}+\boldsymbol{Zu})_i-\log(\kappa))\right\}-\log\left\{\mathfrak{p}_{\text{PG}}(\alpha_i;y_i+\kappa,c_{\mathfrak{q}(\alpha_i|\kappa)})\right\}\\&\quad=(y_i+\kappa)\log\left\{\cosh\left(\tfrac{1}{2}\big((\boldsymbol{X\beta}+\boldsymbol{Zu})_i-\log(\kappa)\big)\right)\right\}-(y_i+\kappa)\log\left\{\cosh\left(\tfrac{1}{2}c_{\mathfrak{q}(\alpha_i|\kappa)}\right)\right\}\\&\qquad+\tfrac{1}{2}\alpha_i\left\{c^2_{\mathfrak{q}(\alpha_i|\kappa)}-\Big((\boldsymbol{X\beta}+\boldsymbol{Zu})_i-\log(\kappa)\Big)^2\right\}.\end{aligned}$$

The cancellation of the $\log\{\mathfrak{p}_{\text{PG}}(\alpha_i;y_i+\kappa,0)\}$ terms is very important from a practical standpoint since $\mathfrak{p}_{\text{PG}}(\alpha_i;y_i+\kappa,0)$ does not admit a closed form.

We then have

$$
\begin{aligned}
&E_{\mathfrak{q}(-\kappa)}\big[\log\{\mathfrak{p}(\boldsymbol{\alpha}|\boldsymbol{y},\boldsymbol{\beta},\boldsymbol{u},\kappa)\}-\log\{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)\}\big]\\
&\quad=\sum_{i=1}^{n}E_{\mathfrak{q}(-\kappa)}\Big[(y_i+\kappa)\log\Big\{\cosh\Big(\tfrac{1}{2}\big((\boldsymbol{X}\boldsymbol{\beta}+\boldsymbol{Z}\boldsymbol{u})_i-\log(\kappa)\big)\Big)\Big\}\Big]\\
&\qquad-(\boldsymbol{y}+\kappa\boldsymbol{1})^T\log\Big\{\cosh\Big(\tfrac{1}{2}c_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)}\Big)\Big\}+\tfrac{1}{2}\boldsymbol{c}_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)}^T\mathrm{diag}\{\mu_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)}\}\boldsymbol{c}_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)}\\
&\qquad-\tfrac{1}{2}\sum_{i=1}^{n}E_{\mathfrak{q}(-\kappa)}\Big\{\alpha_i\big((\boldsymbol{X}\boldsymbol{\beta}+\boldsymbol{Z}\boldsymbol{u})_i-\log(\kappa)\big)^2\Big\}.
\end{aligned}
$$

Simplification of $E_{\mathfrak{q}(-\kappa)}\big[\log\{\mathfrak{p}(\boldsymbol{\beta},\boldsymbol{u}|\boldsymbol{\sigma}^2,\kappa)\}-\log\{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)\}\big]$

For this contribution, we have

$$
\begin{aligned}
\log\{\mathfrak{p}(\boldsymbol{\beta},\boldsymbol{u}|\boldsymbol{\sigma}^2,\kappa)\}-\log\{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)\}&=-\tfrac{1}{2}p\log(\sigma_\beta^2)-\tfrac{1}{2}\sum_{j=1}^{r}K_j\log(\sigma_j^2)\\
&\quad-\frac{\|\boldsymbol{\beta}\|^2}{2\sigma_\beta^2}-\sum_{j=1}^{r}\frac{\|\boldsymbol{u}_j\|^2}{2\sigma_j^2}+\tfrac{1}{2}\log|\boldsymbol{\Sigma}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}|\\
&\quad+\tfrac{1}{2}\left(\begin{bmatrix}\boldsymbol{\beta}\\\boldsymbol{u}\end{bmatrix}-\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}\right)^T\boldsymbol{\Sigma}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}^{-1}\left(\begin{bmatrix}\boldsymbol{\beta}\\\boldsymbol{u}\end{bmatrix}-\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}\right).
\end{aligned}
$$

We then obtain

$$
\begin{aligned}
&E_{\mathfrak{q}(-\kappa)}\big[\log\{\mathfrak{p}(\boldsymbol{\beta},\boldsymbol{u}|\boldsymbol{\sigma}^2,\kappa)\}-\log\{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)\}\big]\\
&\quad=-\tfrac{1}{2}p\log(\sigma_\beta^2)-\tfrac{1}{2}\sum_{j=1}^{r}K_jE_{\mathfrak{q}(-\kappa)}\{\log(\sigma_j^2)\}\\
&\qquad-\frac{\|\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}\|^2+\mathrm{tr}(\boldsymbol{\Sigma}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)})}{2\sigma_\beta^2}-\tfrac{1}{2}\sum_{j=1}^{r}\mu_{\mathfrak{q}(1/\sigma_j^2|\kappa)}\big\{\|\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{u}_j|\kappa)}\|^2+\mathrm{tr}\big(\boldsymbol{\Sigma}_{\mathfrak{q}(\boldsymbol{u}_j|\kappa)}\big)\big\}\\
&\qquad+\tfrac{1}{2}\log|\boldsymbol{\Sigma}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}|+\tfrac{1}{2}p+\tfrac{1}{2}\sum_{j=1}^{r}K_j.
\end{aligned}
$$

Simplification of $\log\{\mathfrak{p}(\boldsymbol{\sigma}^2|\boldsymbol{a},\kappa)\}-\log\{\mathfrak{q}(\boldsymbol{\sigma}^2|\kappa)\}$

Simple manipulations give

$$
\begin{aligned}
\log\{\mathfrak{p}(\boldsymbol{\sigma}^2|\boldsymbol{a},\kappa)\}-\log\{\mathfrak{q}(\boldsymbol{\sigma}^2|\kappa)\}=\sum_{j=1}^{r}\Big[&-\tfrac{1}{2}\log(a_j)-\tfrac{1}{2}\log(\pi)-1/(\sigma_j^2a_j)\\
&-\tfrac{1}{2}(K_j+1)\log\big(\lambda_{\mathfrak{q}(\sigma_j^2|\kappa)}\big)+\log\big\{\Gamma\big(\tfrac{1}{2}(K_j+1)\big)\big\}\\
&+\tfrac{1}{2}K_j\log(\sigma_j^2)+\lambda_{\mathfrak{q}(\sigma_j^2|\kappa)}/\sigma_j^2\Big]
\end{aligned}
$$

where

$$
\lambda_{\mathfrak{q}(\sigma_j^2|\kappa)}\equiv\mu_{\mathfrak{q}(1/a_j|\kappa)}+\tfrac{1}{2}\big\{\|\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{u}_j|\kappa)}\|^2+\mathrm{tr}(\boldsymbol{\Sigma}_{\mathfrak{q}(\boldsymbol{u}_j|\kappa)})\big\}.
$$

Therefore

$$
\begin{aligned}
E_{\mathfrak{q}(-\kappa)}\big[\log\{\mathfrak{p}(\boldsymbol{\sigma}^2|\boldsymbol{a},\kappa)\}-\log\{\mathfrak{q}(\boldsymbol{\sigma}^2|\kappa)\}\big]=&-\tfrac{1}{2}r\log(\pi)+\sum_{j=1}^{r}\log\left\{\Gamma\big(\tfrac{1}{2}(K_j+1)\big)\right\}\\
&-\tfrac{1}{2}\sum_{j=1}^{r}E_{\mathfrak{q}(-\kappa)}\{\log(a_j)\}+\tfrac{1}{2}\sum_{j=1}^{r}K_jE_{\mathfrak{q}(-\kappa)}\{\log(\sigma_j^2)\}\\
&+\sum_{j=1}^{r}\Big[\lambda_{\mathfrak{q}(\sigma_j^2|\kappa)}\mu_{\mathfrak{q}(1/\sigma_j^2|\kappa)}-\mu_{\mathfrak{q}(1/\sigma_j^2|\kappa)}\mu_{\mathfrak{q}(1/a_j|\kappa)}\\
&\qquad-\tfrac{1}{2}(K_j+1)\log\left(\lambda_{\mathfrak{q}(\sigma_j^2|\kappa)}\right)\Big].
\end{aligned}
$$

Simplification of $E_{\mathfrak{q}(-\kappa)}\big[\log\{\mathfrak{p}(\boldsymbol{a}|\kappa)\}-\log\{\mathfrak{q}(\boldsymbol{a}|\kappa)\}\big]$

Lastly, we have

$$
\begin{aligned}
\log\{\mathfrak{p}(\boldsymbol{a}|\kappa)\}-\log\{\mathfrak{q}(\boldsymbol{a}|\kappa)\}=\sum_{j=1}^{r}\Big[&-\log(s_\sigma)-\tfrac{1}{2}\log(\pi)+\tfrac{1}{2}\log(a_j)-1/(a_js_\sigma^2)\\
&-\log\left(\lambda_{\mathfrak{q}(a_j|\kappa)}\right)+\lambda_{\mathfrak{q}(a_j|\kappa)}\mu_{\mathfrak{q}(1/a_j|\kappa)}\Big]
\end{aligned}
$$

where

$$
\lambda_{\mathfrak{q}(a_j|\kappa)}\equiv\mu_{\mathfrak{q}(1/\sigma_j^2|\kappa)}+s_\sigma^{-2}.
$$

The required $\mathfrak{q}$-density expectation is

$$
\begin{aligned}
E_{\mathfrak{q}(-\kappa)}\big[\log\{\mathfrak{p}(\boldsymbol{a}|\kappa)\}-\log\{\mathfrak{q}(\boldsymbol{a}|\kappa)\}\big]=&-r\log(s_\sigma)-\tfrac{1}{2}\,r\log(\pi)+\tfrac{1}{2}\sum_{j=1}^{r}E_{\mathfrak{q}(-\kappa)}\{\log(a_j)\}\\
&+\sum_{j=1}^{r}\Big[\lambda_{\mathfrak{q}(a_j|\kappa)}\mu_{\mathfrak{q}(1/a_j|\kappa)}-\mu_{\mathfrak{q}(1/a_j|\kappa)}/s_\sigma^2-\log\left(\lambda_{\mathfrak{q}(a_j|\kappa)}\right)\Big].
\end{aligned}
$$

Fully Simplified $\log\{\underline{\mathfrak{p}}(\boldsymbol{y}|\kappa)\}$ *Expression*

Combining each of the simplified contributions, we obtain

$$
\begin{aligned}
\log\{\underline{\mathfrak{p}}(\boldsymbol{y}|\kappa)\}=\;&\mathbf{1}^T\log\{\Gamma(\boldsymbol{y}+\kappa\mathbf{1})\}+n\big[\tfrac{1}{2}\kappa\log(\kappa)-\log(2)\kappa-\log\{\Gamma(\kappa)\}\big]\\
&-\tfrac{1}{2}(\boldsymbol{y}^T\mathbf{1})\log(\kappa)+\tfrac{1}{2}\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}^T\big(\boldsymbol{C}^T\boldsymbol{y}-\kappa\,\boldsymbol{C}^T\mathbf{1}\big)\\
&-(\boldsymbol{y}+\kappa\mathbf{1})^T\log\left\{\cosh\left(\tfrac{1}{2}c_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)}\right)\right\}-\frac{\|\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{\beta}|\kappa)}\|^2+\mathrm{tr}(\boldsymbol{\Sigma}_{\mathfrak{q}(\boldsymbol{\beta}|\kappa)})}{2\sigma_\beta^2}\\
&-\tfrac{1}{2}\sum_{j=1}^{r}\mu_{\mathfrak{q}(1/\sigma_j^2|\kappa)}\big\{\|\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{u}_j|\kappa)}\|^2+\mathrm{tr}\big(\boldsymbol{\Sigma}_{\mathfrak{q}(\boldsymbol{u}_j|\kappa)}\big)\big\}+\tfrac{1}{2}\log|\boldsymbol{\Sigma}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}|\\
&+\sum_{j=1}^{r}\Big\{\lambda_{\mathfrak{q}(\sigma_j^2|\kappa)}\mu_{\mathfrak{q}(1/\sigma_j^2|\kappa)}-\mu_{\mathfrak{q}(1/\sigma_j^2|\kappa)}\mu_{\mathfrak{q}(1/a_j|\kappa)}-\tfrac{1}{2}(K_j+1)\log\left(\lambda_{\mathfrak{q}(\sigma_j^2|\kappa)}\right)\\
&\qquad+\lambda_{\mathfrak{q}(a_j|\kappa)}\mu_{\mathfrak{q}(1/a_j|\kappa)}-\mu_{\mathfrak{q}(1/a_j|\kappa)}/s_\sigma^2-\log\left(\lambda_{\mathfrak{q}(a_j|\kappa)}\right)\Big\}
\end{aligned}
$$

$$+\tfrac{1}{2}\boldsymbol{c}_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)}^T\mathrm{diag}\{\mu_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)}\}\boldsymbol{c}_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)} - \tfrac{1}{2}\sum_{i=1}^n E_{\mathfrak{q}(-\kappa)}\Big\{\big((\boldsymbol{X\beta}+\boldsymbol{Zu})_i - \log(\kappa)\big)^2\Big\}$$

$$-(\boldsymbol{y}^T\boldsymbol{1})\log(2) - \boldsymbol{1}^T\log\{\Gamma(\boldsymbol{y}+\boldsymbol{1})\} - \tfrac{1}{2}p\log(\sigma_\beta^2) + \tfrac{1}{2}p + \tfrac{1}{2}\sum_{j=1}^r K_j$$

$$-r\log(\pi) - r\log(s_\sigma) + \sum_{j=1}^r \log\left\{\Gamma\big(\tfrac{1}{2}(K_j+1)\big)\right\}$$

which leads to the expression

$$\log\{\underline{\mathfrak{p}}(\boldsymbol{y}|\kappa)\} = \underline{\ell}(\kappa) + \tfrac{1}{2}\boldsymbol{c}_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)}^T\mathrm{diag}\{\mu_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)}\}\boldsymbol{c}_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)} - \tfrac{1}{2}\sum_{i=1}^n E_{\mathfrak{q}(-\kappa)}\Big\{\alpha_i\big((\boldsymbol{X\beta}+\boldsymbol{Zu})_i - \log(\kappa)\big)^2\Big\}$$
$$+\text{const}$$

where 'const' denotes terms that do not involve κ or $\mathfrak{q}$-density parameters and

$$\begin{aligned}\underline{\ell}(\kappa) \equiv\ & \boldsymbol{1}^T\log\{\Gamma(\boldsymbol{y}+\kappa\boldsymbol{1})\} + n\big[\tfrac{1}{2}\kappa\log(\kappa) - \log(2)\kappa - \log\{\Gamma(\kappa)\}\big]\\
&-\tfrac{1}{2}(\boldsymbol{y}^T\boldsymbol{1})\log(\kappa) + \tfrac{1}{2}\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}^T\big(\boldsymbol{C}^T\boldsymbol{y} - \kappa\,\boldsymbol{C}^T\boldsymbol{1}\big)\\
&-(\boldsymbol{y}+\kappa\boldsymbol{1})^T\log\left\{\cosh\left(\tfrac{1}{2}c_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)}\right)\right\} - \frac{\|\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{\beta}|\kappa)}\|^2 + \mathrm{tr}(\boldsymbol{\Sigma}_{\mathfrak{q}(\boldsymbol{\beta}|\kappa)})}{2\sigma_\beta^2}\\
&-\tfrac{1}{2}\sum_{j=1}^r \mu_{\mathfrak{q}(1/\sigma_j^2|\kappa)}\big\{\|\boldsymbol{\mu}_{\mathfrak{q}(\boldsymbol{u}_j|\kappa)}\|^2 + \mathrm{tr}\big(\boldsymbol{\Sigma}_{\mathfrak{q}(\boldsymbol{u}_j|\kappa)}\big)\big\} + \tfrac{1}{2}\log|\boldsymbol{\Sigma}_{\mathfrak{q}(\boldsymbol{\beta},\boldsymbol{u}|\kappa)}|\\
&+\sum_{j=1}^r\Big\{\lambda_{\mathfrak{q}(\sigma_j^2|\kappa)}\mu_{\mathfrak{q}(1/\sigma_j^2|\kappa)} - \mu_{\mathfrak{q}(1/\sigma_j^2|\kappa)}\mu_{\mathfrak{q}(1/a_j|\kappa)} - \tfrac{1}{2}(K_j+1)\log\big(\lambda_{\mathfrak{q}(\sigma_j^2|\kappa)}\big)\\
&\qquad+\lambda_{\mathfrak{q}(a_j|\kappa)}\mu_{\mathfrak{q}(1/a_j|\kappa)} - \mu_{\mathfrak{q}(1/a_j|\kappa)}/s_\sigma^2 - \log\big(\lambda_{\mathfrak{q}(a_j|\kappa)}\big)\Big\}.\end{aligned}$$

Lastly, we note that the

$$\boldsymbol{c}_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)}^T\mathrm{diag}\{\mu_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)}\}\boldsymbol{c}_{\mathfrak{q}(\boldsymbol{\alpha}|\kappa)} \quad \text{and} \quad \sum_{i=1}^n E_{\mathfrak{q}(-\kappa)}\Big\{\alpha_i\big((\boldsymbol{X\beta}+\boldsymbol{Zu})_i - \log(\kappa)\big)^2\Big\}$$

terms cancel with each other in the $\mathfrak{q}$-density updates, and we have the simpler marginal log-likelihood expression

$$\log\{\underline{\mathfrak{p}}(\boldsymbol{y}|\kappa)\} = \underline{\ell}(\kappa) + \text{const}.$$

Additional References

Jaakkola, T.S. (2001). Tutorial on variational approximation methods. In Opper, M. & Saad, D., editors, *Advanced Mean Field Methods: Theory and Practice*, 129–160. Cambridge, Massachusetts: MIT Press.

Wand, M.P. and Ormerod, J.T. (2011). Penalized wavelets: embedding wavelets into semiparametric regression. *Electronic Journal of Statistics*, **5**, 1654–1717.