# Explicit natural gradient updates for Cholesky factor in Gaussian variational approximation

Linda S. L. Tan (statsll@nus.edu.sg)

National University of Singapore



**Abstract**

Stochastic gradient methods have enabled variational inference for high-dimensional models and large data. However, the steepest ascent direction in the parameter space of a statistical model is given not by the commonly used Euclidean gradient, but the natural gradient which premultiplies the Euclidean gradient by the inverted Fisher information matrix. Use of natural gradients can improve convergence significantly, but inverting the Fisher information matrix is daunting in high-dimensions. In Gaussian variational approximation, natural gradient updates of the natural parameters (expressed in terms of the mean and precision matrix) of the Gaussian distribution can be derived analytically, but do not ensure the precision matrix remains positive definite. To tackle this issue, we consider Cholesky decomposition of the covariance or precision matrix and derive explicit natural gradient updates of the Cholesky factor by finding the inverse of the Fisher information matrix analytically. Natural gradient updates of the Cholesky factor as compared to natural parameters, depend only on the first instead of the second derivative of the log posterior density and reduces computational cost. Sparsity constraints incorporating posterior independence structure can be imposed by fixing relevant entries in the Cholesky factor to zero.

**Keywords:** Gaussian variational approximation; Natural gradients; Cholesky factor; Covariance matrix; Sparse precision matrix; Positive definite constraint.

## 1 Introduction

Variational inference is fast and provides an attractive alternative to Markov chain Monte Carlo (MCMC) methods for approximating intractable posterior distributions in the Bayesian framework. Use of stochastic gradient methods (Robbins and Monro, 1951) has further enabled variational inference for high-dimensional models and large data sets (Hoffman et al., 2013; Salimans and Knowles, 2013). While Euclidean gradients are commonly used in the optimization of the variational objective function, the direction of steepest ascent in the parameter space of statistical models, where distance between probability distributions is measured using the Kullback-Leibler (KL) divergence, is actually given by the natural gradient (Amari, 1998). Stochastic optimization based on natural gradients has been found to be more robust with the ability to avoid or escape plateaus, resulting in faster convergence (Rattray et al., 1998). Martens (2020) shows that natural

gradient descent can be seen as a second order optimization method, with the Fisher information matrix taking the place of the Hessian and having more favorable properties.

The natural gradient is computed by premultiplying the Euclidean gradient with the inverse of the Fisher information matrix, the computation of which can be highly complex. However, in some cases, natural gradient updates can be simpler than those based on Euclidean gradients, such as for the conjugate exponential family models considered in Hoffman et al. (2013). If the variational approximation employs a distribution in the minimal exponential family (Wainwright and Jordan, 2008), then the natural gradient of the variational objective function (evidence lower bound) with respect to the natural parameter is just given by the gradient of the evidence lower bound with respect to the mean of the sufficient statistics (Hensman et al., 2012; Amari, 2016; Khan and Lin, 2017).

In Gaussian variational approximation (Opper and Archambeau, 2009), the true posterior is approximated by a multivariate Gaussian which belongs to the minimal exponential family. Natural gradient updates of the natural parameter can thus be derived analytically as described above. Combined with the theorems of Bonnet (1964) and (Price, 1958), these simplify to updates of the mean and precision matrix which depend on the first and second order derivatives of the log posterior density (Khan et al., 2018). However, the update for the precision matrix does not ensure that it remains positive definite.

Various approaches have been proposed to deal with the positive definite constraint. Khan and Lin (2017) use a back-tracking line search, but that can lead to slow convergence. Ong et al. (2018b) parametrize the Gaussian in terms of the mean and Cholesky factor of the precision matrix and derive the Fisher information matrix analytically, but compute the natural gradients by solving a linear system numerically. Using chain rule, Salimbeni et al. (2018) show that the inverse of the Fisher information matrix in alternative parametrizations (which are one-one transformations of the natural parameters) can be computed as a Jacobian-vector product using automatic differentiation. Ong et al. (2018), Ong et al. (2018a) and Tran et al. (2020) consider a factor structure for the covariance matrix, and Tran et al. (2020) compute the natural gradients using a conjugate gradient linear solver based on a block diagonal approximation of the Fisher information matrix. Lin et al. (2020) use Riemannian gradient descent with a retraction map (derived using a second-order approximation of the geodesic) to compute a modified update of the precision matrix, that includes an additional term to ensure positive definiteness. Tran et al. (2020) optimize the covariance matrix on the manifold of symmetric positive definite matrices and derive an update for the covariance based on an approximation of the natural gradient and a popular retraction for the manifold.

In this article, we consider Cholesky decompositions of either the covariance or precision matrix, and derive the inverse of the Fisher information matrix for these parametrizations in closed form. Explicit natural gradient updates for the Cholesky factor are then presented in both cases. In contrast to natural gradient updates of the natural parameter (involving the mean and precision matrix), our updates depend only on the first

order derivative of the log posterior density, thus reducing storage and computational costs. Updates of the mean and Cholesky factor based on Euclidean gradients have been presented in Titsias and Lázaro-Gredilla (2014), and we demonstrate that corresponding natural gradient updates only require minor modifications with minimal additional costs.

Gaussian variational approximation has been widely applied in many contexts such as likelihood-free inference using the synthetic likelihood approach (Ong et al., 2018b), Bayesian neural networks in deep learning (Khan et al., 2018), exponential random graph models for network modeling (Tan and Friel, 2020) and factor copula models (Nguyen et al., 2020) which seek to capture the dependence structure of high-dimensional variables using a small number of latent variables via bivariate links. For greater flexibility in accommodating variables which are constrained, skewed or heavy-tailed, a Gaussian variational approximation can be specified for variables which have first undergone independent parametric transformations, resulting in a Gaussian copula variational approximation for the original variables. Han et al. (2016) use a Bernstein polynomial transformation while Smith et al. (2020) employ the transformation of Yeo and Johnson (2000) and the Tukey g-and-h distribution (Yan and Genton, 2019) to improve the normality and symmetry of the original variables.

In high-dimensional models, sparsity constraints can be imposed on the covariance matrix by assuming a diagonal or block-diagonal structure according to the variational Bayes restriction (see, e.g. Titsias and Lázaro-Gredilla, 2014; Tan, 2021). Alternatively, the precision matrix can be assumed to adopt a structure that reflects the conditional independence structure in the true posterior, as demonstrated in state space models and generalized linear mixed models by Tan and Nott (2018). The ADVI (automatic differentiation variational inference) algorithm (Kucukelbir et al., 2017) in Stan (Stan Development Team, 2019) allows the user to fit Gaussian variational approximations with either a diagonal or full covariance matrix and provides a library of transformations to convert constrained variables onto the real line. However, it does not permit specification of other sparsity structures and uses Euclidean gradients to update the Cholesky factor in stochastic gradient ascent. Our natural gradient updates of the mean and Cholesky factor can be applied to improve convergence in stochastic gradient ascent in any context where a Gaussian density is used as an approximating density (such as those discussed above), and relevant entries in the Cholesky factor of the covariance or precision matrix can be fixed as zero to impose different sparsity constraints.

This article is organized as follows. Section 2 introduces the notation used in this article and Section 3 describes stochastic variational inference based on Euclidean gradients. In Section 4, we define the natural gradient and discuss its use in stochastic variational inference. Section 5 presents the natural gradient updates of the mean and Cholesky factor of either the covariance or precision matrix in Gaussian variational approximation. We conclude with a discussion in Section 6.

## 2 Notation

Let $A$ be a $d \times d$ matrix. We use $\text{vec}(A)$ to denote the $d^2 \times 1$ vector obtained by stacking the columns of $A$ in order from left to right, and $K$ the $d^2 \times d^2$ commutation matrix such that $K\text{vec}(A) = \text{vec}(A^T)$. Let $N = (K + I_{d^2})/2$.

Let $\text{vech}(A)$ denote the $d(d+1)/2 \times 1$ vector obtained from $\text{vec}(A)$ by omitting supradiagonal elements. If $A$ is symmetric, then $D\text{vech}(A) = \text{vec}(A)$ where $D$ denotes the $d^2 \times d(d+1)/2$ duplication matrix, and $D^+\text{vec}(A) = \text{vech}(A)$ where $D^+ = (D^T D)^{-1} D^T$ denotes the Moore-Penrose inverse of $D$. Let $L$ denote the $d(d+1)/2 \times d^2$ elimination matrix where $L\text{vec}(A) = \text{vech}(A)$. Note that $L^T\text{vech}(A) = \text{vec}(A)$ if $A$ is lower triangular. More details and identities on the commutation, duplication and elimination matrices can be found in Magnus and Neudecker (1980) and Magnus and Neudecker (2019).

Let $\otimes$ denote the Kronecker product such that $\text{vec}(AXB) = (B^T \otimes A)\text{vec}(X)$ and $\odot$ the elementwise Hadamard product. We use $\bar{A}$ to denote the lower triangular matrix derived from $A$ by replacing all supradiagonal elements by zero. Let $\text{diag}(A)$ denote the $d \times 1$ vector containing the diagonal elements of $A$, and $\text{dg}(A)$ denote the diagonal matrix derived from $A$ by replacing all non-diagonal elements by zero. If $a$ is a vector, then $\text{diag}(a)$ denotes the diagonal matrix having $a$ as the diagonal. Let $\nabla_\lambda \mathcal{L}$, $\nabla^2_\lambda \mathcal{L}$ and $\nabla^2_{\lambda,\alpha} \mathcal{L}$ denote $\partial \mathcal{L}/\partial \lambda$, $\partial^2 \mathcal{L}/\partial \lambda \partial \lambda^T$ and $\partial^2 \mathcal{L}/\partial \lambda \partial \alpha^T$ respectively for vectors $\lambda$ and $\alpha$.

## 3 Stochastic variational inference

Let $p(y|\theta)$ denote the likelihood of unknown variables $\theta \in \mathbb{R}^d$ given observed data $y$. Suppose a prior distribution $p(\theta)$ is specified and the true posterior distribution $p(\theta|y) = p(y|\theta)p(\theta)/p(y)$ is intractable. In variational inference, $p(\theta|y)$ is approximated by a density $q_\lambda(\theta)$ with parameters $\lambda \in \Omega$, which are chosen to minimize the KL divergence between $q_\lambda(\theta)$ and $p(\theta|y)$. As

$$\log p(y) = \underbrace{\int q_\lambda(\theta) \log \frac{q_\lambda(\theta)}{p(\theta|y)} d\theta}_{\text{KL divergence}} + \underbrace{\int q_\lambda(\theta) \log \frac{p(y,\theta)}{q_\lambda(\theta)} d\theta}_{\text{Evidence lower bound}},$$

minimizing the KL divergence is equivalent to maximizing the evidence lower bound on the log marginal likelihood. If we let $h_\lambda(\theta) = \log[p(y,\theta)/q_\lambda(\theta)]$, then the evidence lower bound,

$$\mathcal{L}(\lambda) = \text{E}_{q_\lambda(\theta)}[\log p(y,\theta) - \log q_\lambda(\theta)] = \text{E}_{q_\lambda(\theta)}[h_\lambda(\theta)]$$

is the variational objective function to be maximized with respect to $\lambda$. When $\mathcal{L}$ is intractable, stochastic gradient ascent can be used for optimization. Starting with some

initial estimate $\lambda^{(1)}$, an update

$$\lambda^{(t+1)} = \lambda^{(t)} + \rho_t \, \widehat{\nabla}_\lambda \mathcal{L}(\lambda^{(t)}) \tag{1}$$

is performed at iteration $t$, where $\widehat{\nabla}_\lambda \mathcal{L}(\lambda^{(t)})$ is an unbiased estimate of the Euclidean gradient $\nabla_\lambda \mathcal{L}$ evaluated at $\lambda^{(t)}$. Under regularity conditions, $\lambda^{(t)}$ will converge to a local maximum of $\mathcal{L}$ if the stepsize $\rho_t$ satisfies $\sum_{t=1}^{\infty} \rho_t = \infty$ and $\sum_{t=1}^{\infty} \rho_t^2 < \infty$ (Spall, 2003). In practice, an adaptive stepsize sequence such as Adam (Kingma and Ba, 2015) is often used.

### 3.1 Euclidean gradient of evidence lower bound

The Euclidean gradient of $\mathcal{L}$ with respect to $\lambda$ is given by

$$\begin{aligned} \nabla_\lambda \mathcal{L} &= \int [\nabla_\lambda q_\lambda(\theta)] h_\lambda(\theta) d\theta - \int q_\lambda(\theta) \nabla_\lambda \log q_\lambda(\theta) d\theta \\ &= \int [\nabla_\lambda q_\lambda(\theta)] h_\lambda(\theta) d\theta \\ &= \nabla_\lambda \mathrm{E}_{q_\lambda(\theta)}[h(\theta)]. \end{aligned} \tag{2}$$

The second term in the first line of (2) is the expectation of the score function which is zero. In the last line, we have dropped the subscript $\lambda$ from $h_\lambda(\cdot)$ so that it is clearer that the gradient $\nabla_\lambda$ applies only to $q_\lambda(\theta)$ as can be seen from the second line. Estimates of $\nabla_\lambda \mathcal{L}$ can be computed in different ways and existing techniques can be broadly divided into two approaches.

The first is to apply the log derivative trick or score function method, which is based on the fact that $\nabla_\lambda q_\lambda(\theta) = q_\lambda(\theta) \nabla_\lambda \log q_\lambda(\theta)$. From (2), this enables us to write

$$\nabla_\lambda \mathcal{L} = \mathrm{E}_{q_\lambda(\theta)}[\nabla_\lambda \log q_\lambda(\theta) h(\theta)],$$

and an unbiased estimate of $\nabla_\lambda \mathcal{L}$ is $\widehat{\nabla}_\lambda \mathcal{L} = \nabla_\lambda \log q_\lambda(\theta) h(\theta)$ where $\theta$ is simulated from $q_\lambda(\theta)$. Despite being widely applicable, such gradient estimates tend to have high variance leading to slow convergence. Various techniques have been proposed to reduce their variance such as the use of control variates (Paisley et al., 2012), Rao-Blackwellization (Ranganath et al., 2014) and importance sampling (Ruiz et al., 2016).

The second approach is to apply the reparametrization trick (Kingma and Welling, 2014; Rezende et al., 2014; Titsias and Lázaro-Gredilla, 2014), writing $\theta = \mathcal{T}_\lambda(z)$, where $\mathcal{T}_\lambda(\cdot)$ is a differentiable function and $z$ are random variables whose distribution $p(z)$ is independent of $\lambda$. From (2), after applying chain rule, we obtain

$$\nabla_\lambda \mathcal{L} = \nabla_\lambda \int p(z) h(\mathcal{T}_\lambda(z)) dz = \mathrm{E}_{p(z)}[\nabla_\lambda \theta \, \nabla_\theta h(\theta)], \tag{3}$$

where $\theta = \mathcal{T}_\lambda(z)$. Hence an unbiased estimate is $\widehat{\nabla}_\lambda \mathcal{L} = \nabla_\lambda \theta \, \nabla_\theta h(\theta)$ where $z$ is simulated from $p(z)$. With the reparametrization trick, $h$ becomes a direct function of $\lambda$ and gradient information from $h(\cdot)$ can be harnessed more effectively. Gradients computed in this way typically have lower variance than in the score function approach and very often, only a single sample from $p(z)$ is required for computing the unbiased estimate $\widehat{\nabla}_\lambda \mathcal{L}$. For instance, if $q_\lambda(\theta)$ is $\mathrm{N}(\mu, CC^T)$, where $CC^T$ is the Cholesky decomposition of the covariance matrix, we can use the transformation $\theta = Cz + \mu$ where $z \sim \mathrm{N}(0, I_d)$. More generally, for distributions outside the location-scale family, let $F_\lambda(\theta)$ denote the cdf of $q_\lambda(\theta)$. If $F_\lambda(\theta)$ is differentiable, then we can use the transformation $\theta = F_\lambda^{-1}(z)$, where $z \sim U[0,1]$. This approach can be applied easily to distributions with tractable inverse cdf such as the exponential, Cauchy, logistics and Weibull distributions.

## 4 Natural gradient of evidence lower bound

In stochastic variational inference, we are interested in finding the parameter $\lambda$ of $q_\lambda(\theta)$ that maximizes the evidence lower bound $\mathcal{L}(\lambda)$. The Fisher information matrix of $q_\lambda(\theta)$ is defined as

$$F_\lambda = -\mathrm{E}_{q_\lambda(\theta)}[\nabla_\lambda^2 \log q_\lambda(\theta)].$$

Let the distance between probability distributions be measured using the KL divergence. Applying a second order Taylor series expansion,

$$\begin{aligned}\mathrm{KL}(q_\lambda(\theta) \| q_{\lambda+d\lambda}(\theta)) &= \int q_\lambda(\theta) \log \frac{q_\lambda(\theta)}{q_{\lambda+d\lambda}(\theta)} d\theta \\ &\approx \mathrm{E}_{q_\lambda(\theta)}\{\log q_\lambda(\theta) - [\log q_\lambda(\theta) + d\lambda^T \nabla \log q_\lambda(\theta) + \tfrac{1}{2} d\lambda^T \nabla_\lambda^2 \log q_\lambda(\theta) d\lambda]\} \\ &= \tfrac{1}{2} d\lambda^T F_\lambda d\lambda.\end{aligned}$$

From Amari (2016), if $d\lambda$ is sufficiently small, the distance between two points, $\lambda$ and $\lambda + d\lambda$, in the parameter space can be defined as

$$2\mathrm{KL}(q_\lambda(\theta) \| q_{\lambda+d\lambda}(\theta)) = d\lambda^T F_\lambda d\lambda = \|d\lambda\|_{F_\lambda}^2.$$

Thus, the distance between two nearby parameters $\lambda$ and $\lambda + d\lambda$ is not given by $d\lambda^T d\lambda$ as in a Euclidean space, but by $d\lambda^T F_\lambda d\lambda$. The set of all distributions $q_\lambda(\theta), \lambda \in \Omega$ is a manifold where each point $\lambda$ denotes a probability density function and the KL divergence provides the manifold with a Riemannian structure. We say that the manifold is Riemannian with norm $\|d\lambda\|_{F_\lambda} = \sqrt{d\lambda^T F_\lambda d\lambda}$ if the Riemannian metric $F_\lambda$ is positive definite.

Suppose we want to find the steepest ascent direction of $\mathcal{L}(\lambda)$ at $\lambda$. Amari (1998) defines this direction as the vector $a$ that minimizes $\mathcal{L}(\lambda + a)$ where $\|a\|_{F_\lambda} = \epsilon$ for a small

constant $\epsilon$. Using the method of Lagrange multipliers, let

$$\begin{aligned}\mathfrak{L} &= \mathcal{L}(\lambda + a) - \alpha(\|a\|_{F_\lambda}^2 - \epsilon^2) \\ &= \mathcal{L}(\lambda) + a^T \nabla \mathcal{L}(\lambda) - \alpha(a^T F_\lambda a - \epsilon^2).\end{aligned}$$

Setting $\nabla_a \mathfrak{L} = \nabla_\lambda \mathcal{L}(\lambda) - 2\alpha F_\lambda a$ to zero gives $a \propto F_\lambda^{-1} \nabla_\lambda \mathcal{L}(\lambda)$. Hence the steepest ascent direction for $\mathcal{L}(\lambda)$ in the parameter space of $q_\lambda(\theta)$ is given by the natural gradient,

$$\widetilde{\nabla}_\lambda \mathcal{L} = F_\lambda^{-1} \nabla_\lambda \mathcal{L},$$

which premultiplies the Euclidean gradient by the inverse of the Fisher information matrix, provided $F_\lambda$ is positive definite (see also Amari, 1998, 2016; Martens, 2020). Replacing the estimate of the Euclidean gradient in (1) with that of the natural gradient then results in the natural gradient update,

$$\lambda^{(t+1)} = \lambda^{(t)} + \rho_t \, F_{\lambda^{(t)}}^{-1} \, \widehat{\nabla}_\lambda \mathcal{L}(\lambda^{(t)}).$$

### 4.1 Variational approximation in exponential family

Suppose $q_\lambda(\theta)$ belongs to an exponential family and

$$q_\lambda(\theta) = H(\theta) \exp[\phi(\theta)^T \lambda - A(\lambda)], \tag{4}$$

where $\lambda \in \Omega$ is the natural parameter, $\phi(\theta)$ are the sufficient statistics and $A(\lambda)$ is the cumulant or log-partition function. Then

$$m = \mathrm{E}_{q_\lambda(\theta)}[\phi(\theta)] = \nabla_\lambda A(\lambda), \qquad \mathrm{Var}_{q_\lambda(\theta)}[\phi(\theta)] = \nabla_\lambda^2 A(\lambda),$$

and the Fisher information matrix,

$$F_\lambda = -\mathrm{E}_{q_\lambda(\theta)}[\nabla_\lambda^2 \log q_\lambda(\theta)] = \nabla_\lambda^2 A(\lambda) = \nabla_\lambda m.$$

$F_\lambda$ is positive definite and invertible if the exponential family representation in (4) is minimal, or equivalently, if the mapping $m : \Omega \to \mathcal{M}$ is one-one, where $\mathcal{M}$ is the set of realizable mean parameters (Wainwright and Jordan, 2008, Page 40, 62–64).

Applying chain rule, $\nabla_\lambda \mathcal{L} = \nabla_\lambda m \nabla_m \mathcal{L} = F_\lambda \nabla_m \mathcal{L}$. Hence the natural gradient,

$$\widetilde{\nabla}_\lambda \mathcal{L} = F_\lambda^{-1} \nabla_\lambda \mathcal{L} = \nabla_m \mathcal{L}, \tag{5}$$

can be computed by finding the gradient of $\mathcal{L}$ with respect to the mean parameter without computing the Fisher information matrix directly (Khan and Lin, 2017).

### 4.2 Gaussian variational approximation

A popular option for $q_\lambda(\theta)$ is the multivariate Gaussian $\mathrm{N}(\mu, \Sigma)$, with mean $\mu$ and covariance matrix $\Sigma$ (Opper and Archambeau, 2009). If some variables in $\theta$ are constrained, we can first transform them to be unconstrained using for instance, the library of transformations provided in Stan (Kucukelbir et al., 2017). Alternatively, we can consider Gaussian copula variational approximation (Han et al., 2016; Smith et al., 2020) if some variables in $\theta$ are skewed or heavy-tailed.

The multivariate Gaussian can be represented as a member of the exponential family in (4) by writing

$$q_\lambda(\theta) = (2\pi)^{-d/2} \exp\left\{\phi(\theta)^T\lambda - \tfrac{1}{2}\mu^T\Sigma^{-1}\mu - \tfrac{1}{2}\log|\Sigma|\right\},$$

where

$$\phi(\theta) = \begin{bmatrix} \theta \\ \mathrm{vech}(\theta\theta^T) \end{bmatrix}, \quad \lambda = \begin{bmatrix} \Sigma^{-1}\mu \\ -\frac{1}{2}D^T\mathrm{vec}(\Sigma^{-1}) \end{bmatrix}, \quad m = \begin{bmatrix} \mu \\ \mathrm{vech}(\Sigma + \mu\mu^T) \end{bmatrix}.$$

From (5), the natural gradient of $\mathcal{L}$ with respect to the natural parameters $\lambda$ can be obtained simply by finding the gradient of $\mathcal{L}$ with respect to the mean parameters $m$. Let $m_1 = \mu$ and $m_2 = \mathrm{vech}(\Sigma + \mu\mu^T)$, and introduce $\zeta = (\zeta_1^T, \zeta_2^T)^T$, where

$$\zeta_1 = \mu = m_1, \qquad \zeta_2 = \mathrm{vech}(\Sigma) = m_2 - \mathrm{vech}(m_1 m_1^T).$$

Then

$$\nabla_m\zeta = \begin{bmatrix} \nabla m_1\zeta_1 & \nabla m_1\zeta_2 \\ \nabla m_2\zeta_1 & \nabla m_2\zeta_2 \end{bmatrix} = \begin{bmatrix} I_d & -2(I_d \otimes \mu^T)(D^+)^T \\ 0_{d(d+1)/2\times d} & I_{d(d+1)/2} \end{bmatrix}.$$

More details are given in Appendix A. Applying chain rule, the natural gradient is

$$\widetilde{\nabla}_\lambda\mathcal{L} = \nabla_m\mathcal{L} = \nabla_m\zeta\nabla_\zeta\mathcal{L} = \begin{bmatrix} \nabla_\mu\mathcal{L} - 2(\nabla_\Sigma\mathcal{L})\mu \\ D^T\mathrm{vec}(\nabla_\Sigma\mathcal{L}) \end{bmatrix}, \tag{6}$$

where $\nabla_{\mathrm{vech}(\Sigma)}\mathcal{L} = D^T\nabla_{\mathrm{vec}(\Sigma)}\mathcal{L} = D^T\mathrm{vec}(\nabla_\Sigma\mathcal{L})$. From (2) and the Theorems of Bonnet (1964) and Price (1958) (see (Rezende et al., 2014) for more details),

$$\nabla_\mu\mathcal{L} = \int [\nabla_\mu q_\lambda(\theta)]h(\theta)d\theta = \mathrm{E}_{q_\lambda(\theta)}[\nabla_\theta h(\theta)],$$
$$\nabla_\Sigma\mathcal{L} = \int [\nabla_\Sigma q_\lambda(\theta)]h(\theta)d\theta = \tfrac{1}{2}\mathrm{E}_{q_\lambda(\theta)}[\nabla^2_\theta h(\theta)],$$

Substituting these results in (6),

$$\widetilde{\nabla}_\lambda \mathcal{L} = \mathrm{E}_{q_\lambda(\theta)} \begin{bmatrix} \nabla_\theta h(\theta) - \nabla^2_\theta h(\theta)\mu \\ \frac{1}{2} D^T \mathrm{vec}(\nabla^2_\theta h(\theta)) \end{bmatrix}.$$

Let $\theta^{(t)}$ denote a sample generated from $q_\lambda(\theta)$ at iteration $t$. The natural gradient update of $\lambda$ is

$$\begin{bmatrix} {\Sigma^{(t+1)}}^{-1}\mu^{(t+1)} \\ -\frac{1}{2} D^T \mathrm{vec}({\Sigma^{(t+1)}}^{-1}) \end{bmatrix} = \begin{bmatrix} {\Sigma^{(t)}}^{-1}\mu^{(t)} \\ -\frac{1}{2} D^T \mathrm{vec}({\Sigma^{(t)}}^{-1}) \end{bmatrix} + \rho_t \begin{bmatrix} \nabla_\theta h(\theta^{(t)}) - \nabla^2_\theta h(\theta^{(t)})\mu^{(t)} \\ \frac{1}{2} D^T \mathrm{vec}(\nabla^2_\theta h(\theta^{(t)})) \end{bmatrix},$$

which simplifies to

$$\begin{aligned} {\Sigma^{(t+1)}}^{-1} &= {\Sigma^{(t)}}^{-1} - \rho_t \nabla^2_\theta h(\theta^{(t)}), \\ \mu^{(t+1)} &= \mu^{(t)} + \rho_t \Sigma^{(t+1)} \nabla_\theta h(\theta^{(t)}). \end{aligned}$$

The natural gradient update of $\Sigma^{-1}$ derived in this manner does not ensure $\Sigma^{-1}$ remains positive definite and it also depends on the second derivative of $h(\theta)$.

# 5 Natural gradient updates for mean and Cholesky factor of Gaussian variational approximation

To ensure that the covariance or precision matrix remains positive definite in the optimization, we consider different parametrizations of $q_\lambda(\theta)$ based on Cholesky decompositions. Updating only the Cholesky factor instead of the full covariance or precision matrix may also reduce computation and storage costs, resulting in greater efficiency. The first parametrization is

$$\lambda_1 = (\mu^T, \mathrm{vech}(C)^T)^T \quad \text{where} \quad \Sigma = CC^T \tag{7}$$

and $C$ is a lower triangular matrix. The second parametrization is

$$\lambda_2 = (\mu^T, \mathrm{vech}(T)^T)^T \quad \text{where} \quad \Sigma^{-1} = TT^T \tag{8}$$

and $T$ is a lower triangular matrix. The first parametrization is useful if a block-diagonal covariance structure corresponding to the assumption in variational Bayes is desired as entries off the block-diagonal in $C$ can simply be fixed as zero (Tan, 2021). On the other hand, if the true posterior has any inherent conditional independence structure, then the second parametrization allows corresponding sparsity constraints to be imposed on $\Sigma^{-1}$ and hence $T$ (Tan and Nott, 2018). For these parametrizations, $\lambda_1$ and $\lambda_2$ are not the natural parameters of $q_\lambda(\theta)$. Hence the natural gradient cannot be computed simply by using (5) and it is necessary to find $F_\lambda^{-1}$. We show that $F_\lambda^{-1}$ and hence the natural gradient updates can be evaluated analytically for the parametrizations in (7) and (8).

First, we find the Euclidean gradients of $\mathcal{L}$ with respect to $\lambda_1$ and $\lambda_2$ using the reparametrization trick. Let $z \sim \mathrm{N}(0, I_d)$ and $\phi(z)$ denote the density of $z$. From (3), $\nabla_\lambda \mathcal{L} = \mathrm{E}_{\phi(z)}[\nabla_\lambda \theta \nabla_\theta h(\theta)]$. For $\lambda_1$, let $\theta = Cz + \mu$. Then

$$\nabla_{\lambda_1}\theta = \begin{bmatrix} \nabla_\mu \theta \\ \nabla_{\mathrm{vech}(C)}\theta \end{bmatrix} = \begin{bmatrix} I_d \\ L(z \otimes I_d) \end{bmatrix} \quad \text{and} \quad \nabla_{\lambda_1}\mathcal{L} = \mathrm{E}_{\phi(z)} \begin{bmatrix} \nabla_\theta h(\theta) \\ \mathrm{vech}(\bar{G}_1) \end{bmatrix},$$

since $L(z \otimes I_d)\nabla_\theta h(\theta) = L\mathrm{vec}(\nabla_\theta h(\theta) z^T) = \mathrm{vech}(G_1) = \mathrm{vech}(\bar{G}_1)$, where $G_1 = \nabla_\theta h(\theta) z^T$. For $\lambda_2$, let $\theta = T^{-T}z + \mu$. We have

$$\nabla_{\lambda_2}\theta = \begin{bmatrix} \nabla_\mu \theta \\ \nabla_{\mathrm{vech}(T)}\theta \end{bmatrix} = \begin{bmatrix} I_d \\ -L(T^{-1} \otimes T^{-T}z) \end{bmatrix} \quad \text{and} \quad \nabla_{\lambda_2}\mathcal{L} = \mathrm{E}_{\phi(z)} \begin{bmatrix} \nabla_\theta h(\theta) \\ \mathrm{vech}(\bar{G}_2) \end{bmatrix},$$

since $-L(T^{-1} \otimes T^{-T}z)\nabla_\theta h(\theta) = -L\mathrm{vec}(T^{-T}z[\nabla_\theta h(\theta)]^T T^{-T}) = \mathrm{vech}(G_2) = \mathrm{vech}(\bar{G}_2)$ where $G_2 = -T^{-T}z[\nabla_\theta h(\theta)]^T T^{-T}$.

Next, we find the Fisher information matrix $F_{\lambda_i}$ and its inverse for each parametrization $\lambda_i$, $i = 1, 2$. To find the inverse, we require Lemma 1, whose proof is given in the Appendix B. The natural gradient is then given by $\widehat{\nabla}_{\lambda_i}\mathcal{L} = F_{\lambda_i}^{-1}\nabla_{\lambda_i}\mathcal{L}$. These results are summarized in Theorem 1.

**Lemma 1.** *If $\Lambda$ is a $d \times d$ lower triangular matrix, then*

$$\mathfrak{I} = L\{(\Lambda^{-1} \otimes \Lambda^{-T})K + I_d \otimes \Lambda^{-T}\Lambda^{-1}\}L^T = 2L(I_d \otimes \Lambda^{-T})N(I_d \otimes \Lambda^{-1})L^T,$$

*and*

$$\mathfrak{I}^{-1} = \tfrac{1}{2}L(I_d \otimes \Lambda)L^T(LNL^T)^{-1}L(I_d \otimes \Lambda^T)L^T.$$

**Theorem 1.** *For $i = 1, 2$, the Fisher information matrix of $q_{\lambda_i}(\theta)$ is given by*

$$F_{\lambda_i} = \begin{bmatrix} \Sigma^{-1} & 0 \\ 0 & 2L(I_d \otimes \Lambda_i^{-T})N(I_d \otimes \Lambda_i^{-1})L^T \end{bmatrix}, \tag{9}$$

*where $\Lambda_1 = C$ and $\Lambda_2 = T$. The inverse is given by*

$$F_{\lambda_i}^{-1} = \begin{bmatrix} \Sigma & 0 \\ 0 & \frac{1}{2}L(I_d \otimes \Lambda_i)L^T(LNL^T)^{-1}L(I_d \otimes \Lambda_i^T)L^T \end{bmatrix}, \tag{10}$$

*and the natural gradient is*

$$\widetilde{\nabla}_{\lambda_i}\mathcal{L} = E_{\phi(z)} \begin{bmatrix} \Sigma\nabla_\theta h(\theta) \\ vech[\Lambda_i\{\bar{H}_i - dg(\bar{H}_i)/2\}] \end{bmatrix}.$$

*where $H_i = \Lambda_i^T \bar{G}_i$, $\theta = Cz + \mu$ for $\lambda_1$ and $\theta = T^{-T}z + \mu$ for $\lambda_2$.*

*Proof.* The Fisher information matrix is given by $F_\lambda = -\mathrm{E}_{q_\lambda(\theta)}[\nabla^2_\lambda \ell_q]$, where

$$\ell_q = \log q_\lambda(\theta) = -\tfrac{d}{2}\log(2\pi) - \tfrac{1}{2}\log|\Sigma| - \tfrac{1}{2}(\theta-\mu)^T\Sigma^{-1}(\theta-\mu).$$

If $\lambda = \lambda_1$, $\Sigma = CC^T$ and if $\lambda = \lambda_2$, $\Sigma^{-1} = TT^T$. We have

$$\nabla^2_{\lambda_1}\ell_q = -\begin{bmatrix} \Sigma^{-1} & (z^T\otimes\Sigma^{-1} + C^{-T}\otimes z^T C^{-1})L^T \\ \cdot & L\{[(zz^T - I_d)C^{-1}\otimes C^{-T} + C^{-1}\otimes C^{-T}zz^T]K + zz^T\otimes\Sigma^{-1}\}L^T \end{bmatrix},$$

$$\nabla^2_{\lambda_2}\ell_q = -\begin{bmatrix} \Sigma^{-1} & -(z^T\otimes I + T\otimes z^T T^{-1})L^T \\ \cdot & L\{(T^{-1}\otimes T^{-T})K + I_d\otimes T^{-T}zz^T T^{-1}\}L^T \end{bmatrix},$$

Since $\mathrm{E}_{\phi(z)}(z) = 0$ and $\mathrm{E}_{\phi(z)}(zz^T) = I_d$, we obtain $F_{\lambda_i}$ in (9) by applying Lemma 1. Since $F_{\lambda_i}$ is a block-diagonal matrix and each of the blocks is invertible, applying Lemma 1 again gives $F^{-1}_{\lambda_i}$ in (10). Premultiplying the Euclidean gradient $\nabla_{\lambda_i}\mathcal{L}$ by $F^{-1}_{\lambda_i}$ and simplifying, we obtain the natural gradient $\widetilde{\nabla}_{\lambda_i}\mathcal{L}$. More details are given in Appendix C. □

The stochastic variational inference algorithms for updating $\mu$ and $C$ where $\Sigma = CC^T$ are outlined in Figure 1. For comparison, Algorithm 1a which is based on Euclidean gradients is given on the left, while Algorithm 1b which is based on natural gradients is given on the right. Similarly, the algorithms for updating $\mu$ and $T$ where $\Sigma^{-1} = TT^T$ are given in Figure 2, with Algorithm 2a based on Euclidean gradients on the left and Algorithm 2b based on natural gradients on the right.

**Algorithm 1a** (Update $\mu$ and $C$ using Euclidean gradients)

Initialize $\mu^{(1)}$ and $C^{(1)}$. For $t = 1, 2, \ldots, N$,

1. Generate $z \sim \mathrm{N}(0, I_d)$ and compute $\theta^{(t)} = C^{(t)}z + \mu^{(t)}$.
2. Compute $\bar{G}_1$, where $G_1 = \nabla_\theta h(\theta^{(t)})z^T$.
3. Update $\mu^{(t+1)} = \mu^{(t)} + \rho_t \nabla_\theta h(\theta^{(t)})$.
4. Update $C^{(t+1)} = C^{(t)} + \rho_t \bar{G}_1$.

**Algorithm 1b** (Update $\mu$ and $C$ using natural gradients)

Initialize $\mu^{(1)}$ and $C^{(1)}$. For $t = 1, 2, \ldots, N$,

1. Generate $z \sim \mathrm{N}(0, I_d)$ and compute $\theta^{(t)} = C^{(t)}z + \mu^{(t)}$.
2. Compute $\bar{G}_1$, where $G_1 = \nabla_\theta h(\theta^{(t)})z^T$.
3. Compute $\bar{H}_1$ where $H_1 = C^{(t)T}\bar{G}_1$.
4. Update $\mu^{(t+1)} = \mu^{(t)} + \rho_t C^{(t)}C^{(t)T}\nabla_\theta h(\theta^{(t)})$.
5. Update $C^{(t+1)} = C^{(t)} + \rho_t C^{(t)}\{\bar{H}_1 - \mathrm{dg}(\bar{H}_1)/2\}$.

Figure 1: Stochastic variational inference algorithms for updating $\mu$ and $C$ where $\Sigma = CC^T$.

## 6 Conclusion

In Gaussian variational approximation, the natural gradient update of the precision matrix does not ensure positive definiteness. To tackle this issue, we consider Cholesky decompositions of the covariance or precision matrix and derive natural gradient updates of the Cholesky factor in each case. As these parametrizations are not given in terms of

**Algorithm 2a** (Update $\mu$ and $T$ using Euclidean gradients)

Initialize $\mu^{(1)}$ and $T^{(1)}$. For $t = 1, 2, \ldots, N$,

1. Generate $z \sim \mathrm{N}(0, I_d)$ and compute $\theta^{(t)} = T^{(t)-T} z + \mu^{(t)}$.
2. Compute $\bar{G}_2$, where $G_2 = -T^{(t)-T} z[\nabla_\theta h(\theta^{(t)})]^T T^{(t)-T}$.
3. Update $\mu^{(t+1)} = \mu^{(t)} + \rho_t \nabla_\theta h(\theta^{(t)})$.
4. Update $T^{(t+1)} = T^{(t)} + \rho_t \bar{G}_2$.

**Algorithm 2b** (Update $\mu$ and $T$ using natural gradients)

Initialize $\mu^{(1)}$ and $T^{(1)}$. For $t = 1, 2, \ldots, N$,

1. Generate $z \sim \mathrm{N}(0, I_d)$ and compute $\theta^{(t)} = T^{(t)-T} z + \mu^{(t)}$.
2. Compute $\bar{G}_2$, where $G_2 = -T^{(t)-T} z[\nabla_\theta h(\theta^{(t)})]^T T^{(t)-T}$.
3. Compute $\bar{H}_2$ where $H_2 = T^{(t)T} \bar{G}_2$.
4. Update $\mu^{(t+1)} = \mu^{(t)} + \rho_t T^{(t)-T} T^{(t)-1} \nabla_\theta h(\theta^{(t)})$.
5. Update $T^{(t+1)} = T^{(t)} + \rho_t T^{(t)} \{\bar{H}_2 - \mathrm{dg}(\bar{H}_2)/2\}$.

Figure 2: Stochastic variational inference algorithms for updating $\mu$ and $T$ where $\Sigma^{-1} = TT^T$.

the natural parameter, we need to find the inverse of the Fisher information matrix. We demonstrate that this inverse can be found analytically and present the natural gradient updates of the Cholesky factors in closed form. These natural gradient updates can potentially improve convergence in stochastic gradient ascent and can be used in any context where the variational approximation is a multivariate Gaussian. Sparsity constraints can also be imposed to incorporate assumptions in variational Bayes or conditional independence structure in the posterior by fixing relevant entries in the Cholesky factor to zero.

## Acknowledgment

Linda Tan was supported by the start-up grant R-155-000-190-133.

## References

Amari, S. (1998). Natural gradient works efficiently in learning. *Neural Computation 10*, 251–276.

Amari, S. (2016). *Information Geometry and Its Applications*. Japan: Springer.

Bonnet, G. (1964). Transformations des signaux aléatoires a travers les systemes non linéaires sans mémoire. In *Annales des Télécommunications*, Volume 19, pp. 203–220. Springer.

Han, S., X. Liao, D. Dunson, and L. Carin (2016). Variational Gaussian copula inference. In A. Gretton and C. C. Robert (Eds.), *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, Volume 51 of *Proceedings of Machine Learning Research*, Cadiz, Spain, pp. 829–838. PMLR.

Hensman, J., M. Rattray, and N. D. Lawrence (2012). Fast variational inference in the conjugate exponential family. In *Proceedings of the 25th International Conference on Neural*

*Information Processing Systems - Volume 2*, NIPS'12, Red Hook, NY, USA, pp. 2888–2896. Curran Associates Inc.

Hoffman, M. D., D. M. Blei, C. Wang, and J. Paisley (2013). Stochastic variational inference. *Journal of Machine Learning Research 14*, 1303–1347.

Khan, M. and W. Lin (2017). Conjugate-Computation Variational Inference : Converting Variational Inference in Non-Conjugate Models to Inferences in Conjugate Models. In A. Singh and J. Zhu (Eds.), *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, Volume 54 of *Proceedings of Machine Learning Research*, pp. 878–887. PMLR.

Khan, M., D. Nielsen, V. Tangkaratt, W. Lin, Y. Gal, and A. Srivastava (2018). Fast and scalable Bayesian deep learning by weight-perturbation in Adam. In J. Dy and A. Krause (Eds.), *Proceedings of the 35th International Conference on Machine Learning*, Volume 80 of *Proceedings of Machine Learning Research*, pp. 2611–2620. PMLR.

Kingma, D. P. and J. Ba (2015). Adam: A method for stochastic optimization. In Y. Bengio and Y. LeCun (Eds.), *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Kingma, D. P. and M. Welling (2014). Auto-encoding variational bayes. In Y. Bengio and Y. LeCun (Eds.), *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*.

Kucukelbir, A., D. Tran, R. Ranganath, A. Gelman, and D. M. Blei (2017). Automatic differentiation variational inference. *Journal of Machine Learning Research 18*, 1–45.

Lin, W., M. Schmidt, and M. E. Khan (2020, 13–18 Jul). Handling the positive-definite constraint in the Bayesian learning rule. In H. D. III and A. Singh (Eds.), *Proceedings of the 37th International Conference on Machine Learning*, Volume 119 of *Proceedings of Machine Learning Research*, pp. 6116–6126. PMLR.

Magnus, J. R. and H. Neudecker (1980). The elimination matrix: Some lemmas and applications. *SIAM J. Algebraic Discret. Methods 1*, 422–449.

Magnus, J. R. and H. Neudecker (2019). *Matrix Differential Calculus with Applications in Statistics and Econometrics* (Third ed.). Chichester: John Wiley & Sons.

Martens, J. (2020). New insights and perspectives on the natural gradient method. *Journal of Machine Learning Research 21*(146), 1–76.

Nguyen, H., M. C. Ausín, and P. Galeano (2020). Variational inference for high dimensional structured factor copulas. *Computational Statistics & Data Analysis 151*, 107012.

Ong, V. M.-H., D. J. Nott, and M. S. Smith (2018). Gaussian variational approximation with a factor covariance structure. *Journal of Computational and Graphical Statistics 27*, 465–478.

Ong, V. M.-H., D. J. Nott, M.-N. Tran, S. A. Sisson, and C. C. Drovandi (2018a). Likelihood-free inference in high dimensions with synthetic likelihood. *Computational Statistics & Data Analysis 128*, 271–291.

Ong, V. M. H., D. J. Nott, M.-N. Tran, S. A. Sisson, and C. C. Drovandi (2018b). Variational bayes with synthetic likelihood. *Statistics and Computing 28*, 971–988.

Opper, M. and C. Archambeau (2009). The variational gaussian approximation revisited. *Neural computation 21*, 786–792.

Paisley, J., D. M. Blei, and M. I. Jordan (2012). Variational bayesian inference with stochastic search. In *Proceedings of the 29th International Coference on International Conference on Machine Learning*, Madison, WI, USA, pp. 1363–1370. Omnipress.

Price, R. (1958). A useful theorem for nonlinear devices having gaussian inputs. *IRE Transactions on Information Theory 4*, 69–72.

Ranganath, R., S. Gerrish, and D. Blei (2014). Black Box Variational Inference. In S. Kaski and J. Corander (Eds.), *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Statistics*, Volume 33 of *Proceedings of Machine Learning Research*, Reykjavik, Iceland, pp. 814–822. PMLR.

Rattray, M., D. Saad, and S.-i. Amari (1998). Natural gradient descent for on-line learning. *Phys. Rev. Lett. 81*, 5461–5464.

Rezende, D. J., S. Mohamed, and D. Wierstra (2014). Stochastic backpropagation and approximate inference in deep generative models. In *Proceedings of the 31st International Conference on International Conference on Machine Learning - Volume 32*, ICML'14, pp. II–1278–II–1286. JMLR.org.

Robbins, H. and S. Monro (1951). A stochastic approximation method. *The Annals of Mathematical Statistics 22*, 400–407.

Ruiz, F. J. R., M. K. Titsias, and D. M. Blei (2016). Overdispersed black-box variational inference. In *Proceedings of the Thirty-Second Conference on Uncertainty in Artificial Intelligence*, Arlington, Virginia, USA, pp. 647–656. AUAI Press.

Salimans, T. and D. A. Knowles (2013). Fixed-form variational posterior approximation through stochastic linear regression. *Bayesian Analysis 8*, 837–882.

Salimbeni, H., S. Eleftheriadis, and J. Hensman (2018). Natural gradients in practice: Non-conjugate variational inference in gaussian process models. In *International Conference on Artificial Intelligence and Statistics*, pp. 689–697. PMLR.

Smith, M. S., R. Loaiza-Maya, and D. J. Nott (2020). High-dimensional copula variational approximation through transformation. *Journal of Computational and Graphical Statistics 29*, 729–743.

Spall, J. C. (2003). *Introduction to stochastic search and optimization: estimation, simulation and control*. New Jersey: Wiley.

Stan Development Team (2019). Stan modeling language users guide and reference manual. Version 2.28.

Tan, L. S. L. (2021). Use of model reparametrization to improve variational bayes†. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 83*, 30–57.

Tan, L. S. L. and N. Friel (2020). Bayesian variational inference for exponential random graph models. *Journal of Computational and Graphical Statistics 29*, 910–928.

Tan, L. S. L. and D. J. Nott (2018). Gaussian variational approximation with sparse precision matrices. *Statistics and Computing 28*, 259–275.

Titsias, M. and M. Lázaro-Gredilla (2014, 22–24 Jun). Doubly stochastic variational bayes for non-conjugate inference. In E. P. Xing and T. Jebara (Eds.), *Proceedings of the 31st International Conference on Machine Learning*, Volume 32 of *Proceedings of Machine Learning Research*, Bejing, China, pp. 1971–1979. PMLR.

Tran, M.-N., N. Nguyen, D. Nott, and R. Kohn (2020). Bayesian deep net glm and glmm. *Journal of Computational and Graphical Statistics 29*, 97–113.

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

Yan, Y. and M. G. Genton (2019). The tukey g-and-h distribution. *Significance 16*, 12–13.

Yeo, I.-K. and R. A. Johnson (2000). A new family of power transformations to improve normality or symmetry. *Biometrika 87*, 954–959.

# A Natural gradient of $\mathcal{L}$ with respect to natural parameters

Differentiating $\zeta_2 = m_2 - \text{vech}(m_1 m_1^T)$ with respect to $m_1$, we obtain

$$\begin{aligned} d\zeta_2 &= -D^+\text{vec}(m_1 dm_1^T + dm_1 m_1^T) \\ &= -D^+(I_{d^2} + K)(I_d \otimes m_1)dm_1 \\ &= -2D^+DD^+(I_d \otimes m_1)dm_1 \\ &= -2D^+(I_d \otimes m_1)dm_1 \end{aligned} \implies \nabla_{m_1}\zeta_2 = -2(I_d \otimes m_1^T)(D^+)^T.$$

We have used the properties $I_{d^2} + K = 2DD^+$ and $D^+D = I_{d(d+1)/2}$.

## B Proof of Lemma 1

To prove Lemma 1, we require several results regarding the elimination matrix $L$ from Magnus and Neudecker (1980), which are stated in Lemma 2 for ease of reference.

**Lemma 2.** *If $P$ and $Q$ are lower triangular $d \times d$ matrices, then*

(i) $LL^T = I_{d(d+1)/2}$;

(ii) $(LNL^T)^{-1} = 2I_{d(d+1)/2} - LKL^T$;

(iii) $N = DLN$;

(iv) $L^T L(P^T \otimes Q)L^T = (P^T \otimes Q)L^T$ *or the transpose,* $L(P \otimes Q^T)L^T L = L(P \otimes Q^T)$;

(v) $L(P^T \otimes Q)L^T = D^T(P^T \otimes Q)L^T$ *or the transpose,* $L(P \otimes Q^T)L^T = L(P \otimes Q^T)D$.

*Proof.* The proofs can be found respectively in Lemma 3.2(ii), Lemma 3.4 (ii), Lemma 3.5 (ii) and Lemma 4.2 (i) and (iii), of Magnus and Neudecker (1980) . □

*Proof of Lemma 1.* From the left-hand side,

$$\begin{aligned}
\mathfrak{I} &= L\{K(\Lambda^{-T} \otimes \Lambda^{-1}) + I_d \otimes \Lambda^{-T}\Lambda^{-1}\}L^T \\
&= L\{K(\Lambda^{-T} \otimes I_d)(I_d \otimes \Lambda^{-1}) + (I_d \otimes \Lambda^{-T})(I_d \otimes \Lambda^{-1})\}L^T \\
&= L\{(I_d \otimes \Lambda^{-T})K + (I_d \otimes \Lambda^{-T})\}(I_d \otimes \Lambda^{-1})L^T \\
&= L(I_d \otimes \Lambda^{-T})(K + I_{d^2})(I_d \otimes \Lambda^{-1})L^T \\
&= 2L(I_d \otimes \Lambda^{-T})N(I_d \otimes \Lambda^{-1})L^T.
\end{aligned}$$

Using the results in Lemma 2, we have

$$\begin{aligned}
&\{2L(I_d \otimes \Lambda^{-T})N(I_d \otimes \Lambda^{-1})L^T\}\left\{\tfrac{1}{2}L(I_d \otimes \Lambda)L^T(LNL^T)^{-1}L(I_d \otimes \Lambda^T)L^T\right\} \\
&= L(I_d \otimes \Lambda^{-T})(DLN)(I_d \otimes \Lambda^{-1})(I_d \otimes \Lambda)L^T(LNL^T)^{-1}L(I_d \otimes \Lambda^T)L^T && \text{[(iii) \& (iv)]} \\
&= L(I_d \otimes \Lambda^{-T})L^T(LNL^T)(LNL^T)^{-1}L(I_d \otimes \Lambda^T)L^T && \text{[(v)]} \\
&= L(I_d \otimes \Lambda^{-T})L^T L(I_d \otimes \Lambda^T)L^T \\
&= L(I_d \otimes \Lambda^{-T})(I_d \otimes \Lambda^T)L^T && \text{[(iv)]} \\
&= LL^T = I_{d(d+1)/2}. && \text{[(i)]}
\end{aligned}$$

The roman letters in square brackets on the right indicate which parts of Lemma 2 are used. □

## C Proof of Theorem 1

*Proof.* In each case, $\nabla_\mu \ell_q = \Sigma^{-1}(\theta - \mu)$ and $\nabla^2_\mu \ell_q = -\Sigma^{-1}$.

If $\lambda = \lambda_1$, then $z = C^{-1}(\theta - \mu)$, differentiating $z$ with respect to $\text{vech}(C)$,

$$dz = -C^{-1}(dC)C^{-1}(\theta - \mu) = -C^{-1}(dC)z.$$

Differentiating $\nabla_\mu \ell_q = C^{-T}C^{-1}(\theta - \mu)$ w.r.t. $C$,

$$\begin{aligned} d(\nabla_\mu \ell_q) &= -[C^{-T}(dC^T)C^{-T}C^{-1} + C^{-T}C^{-1}(dC)C^{-1}](\theta - \mu) \\ &= -[C^{-T}(dC^T)C^{-T}z + C^{-T}C^{-1}(dC)z] \\ &= -[(z^TC^{-1} \otimes C^{-T})K + (z^T \otimes C^{-T}C^{-1})]L^T d\text{vech}(C). \\ \therefore \nabla^2_{\mu,\text{vech}(C)}\ell_q &= -[(C^{-T} \otimes z^TC^{-1}) + (z^T \otimes \Sigma^{-1})]L^T. \end{aligned}$$

Differentiating $\ell_q$ w.r.t. $C$,

$$\begin{aligned} d\ell_q &= -\text{tr}(C^{-1}dC) - z^T dz \\ &= -\text{vec}(C^{-T})^T d\text{vec}(C) + z^TC^{-1}(dC)z \\ &= [-\text{vec}(C^{-T}) + \text{vec}(C^{-T}zz^T)]^T L^T d\text{vech}(C) \\ \therefore \nabla_{\text{vech}(C)}\ell_q &= L\text{vec}[C^{-T}(zz^T - I_d)] = \text{vech}[C^{-T}(zz^T - I_d)]. \end{aligned}$$

Differentiating $\nabla_{\text{vech}(C)}\ell_q = \text{vech}[C^{-T}(zz^T - I_d)]$ w.r.t. $C$,

$$\begin{aligned} &d\nabla_{\text{vech}(C)}\ell_q = L\text{vec}[-C^{-T}dC^TC^{-T}(zz^T - I_d) + C^{-T}\{z(dz^T) + dz(z^T)\}] \\ &= -L\{[(zz^T - I_d)C^{-1} \otimes C^{-T}]Kd\text{vec}(C) + \text{vec}[C^{-T}zz^T(dC^T)C^{-T} + C^{-T}C^{-1}dCzz^T]\} \\ &= -L\{[(zz^T - I_d)C^{-1} \otimes C^{-T}]K + (C^{-1} \otimes C^{-T}zz^T)K + (zz^T \otimes \Sigma^{-1})\}L^T d\text{vech}(C). \end{aligned}$$

$$\therefore \nabla^2_{\text{vech}(C)}\ell_q = -L\{[(zz^T - I_d)C^{-1} \otimes C^{-T} + C^{-1} \otimes C^{-T}zz^T]K + zz^T \otimes \Sigma^{-1}\}L^T.$$

Taking negative expectations with respect to $q_{\lambda_1}(\theta)$, we have by Lemma 1,

$$\begin{aligned} -\text{E}_{q_{\lambda_1}(\theta)}[\nabla^2_{\text{vech}(C)}\ell_q] &= L\{(C^{-1} \otimes C^{-T})K + (I_d \otimes C^{-T}C^{-1})\}L^T \\ &= 2L(I_d \otimes C^{-T})N(I_d \otimes C^{-1})L^T. \end{aligned}$$

If $\lambda = \lambda_2$, then $z = T^T(\theta - \mu)$. Differentiating $z$ w.r.t. $T$, $dz = (dT^T)(\theta - \mu) = (dT^T)T^{-T}z$. Differentiating $\nabla_\mu \ell_q = TT^T(\theta - \mu)$ w.r.t. $T$,

$$\begin{aligned} d(\nabla_\mu \ell_q) &= [(dT)T^T + T(dT^T)](\theta - \mu) \\ &= (dT)z + T(dT^T)T^{-T}z \\ &= [(z^T \otimes I_d) + (z^TT^{-1} \otimes T)K]L^T d\text{vech}(T) \\ \therefore \nabla^2_{\mu,\text{vech}(T)}\ell_q &= (z^T \otimes I_d + T \otimes z^TT^{-1})L^T. \end{aligned}$$

Differentiating $\ell_q$ with respect to $T$,

$$\begin{aligned} d\ell_q &= \mathrm{tr}(T^{-1}dT) - z^T dz \\ &= \mathrm{vec}(T^{-T})^T d\mathrm{vec}(T) - z^T (dT^T) T^{-T} z \\ &= [\mathrm{vec}(T^{-T}) - K\mathrm{vec}(zz^T T^{-1})]^T L^T d\mathrm{vech}(T) \\ \therefore \nabla_{\mathrm{vech}(T)} \ell_q &= L\mathrm{vec}[T^{-T}(I_d - zz^T)] = \mathrm{vech}[T^{-T}(I_d - zz^T)]. \end{aligned}$$

Differentiating $\nabla_{\mathrm{vech}(T)} \ell_q = \mathrm{vech}[T^{-T}(I_d - zz^T)]$ w.r.t. $T$,

$$\begin{aligned} d\nabla_{\mathrm{vech}(T)} \ell_q &= -L\mathrm{vec}[T^{-T}(dT^T)T^{-T}(I_d - zz^T) + T^{-T}\{z(dz^T) + dz(z^T)\}] \\ &= -L\{[(I_d - zz^T)T^{-1} \otimes T^{-T}]Kd\mathrm{vec}(T) + T^{-T}zz^T T^{-1}(dT) + T^{-T}(dT^T)T^{-T}zz^T\} \\ &= -L\{[(I_d - zz^T)T^{-1} \otimes T^{-T} + zz^T T^{-1} \otimes T^{-T}]K + I_d \otimes T^{-T}zz^T T^{-1}\} L^T d\mathrm{vech}(T) \\ &= -L\{(T^{-1} \otimes T^{-T})K + I_d \otimes T^{-T}zz^T T^{-1}\} L^T d\mathrm{vech}(T). \end{aligned}$$

$$\therefore \nabla^2_{\mathrm{vech}(T)} \ell_q = -L\{(T^{-1} \otimes T^{-T})K + I_d \otimes T^{-T}zz^T T^{-1}\} L^T.$$

Taking negative expectations with respect to $q_{\lambda_2}(\theta)$, we have by Lemma 1,

$$\begin{aligned} -\mathrm{E}_{q_{\lambda_2}(\theta)}[\nabla^2_{\mathrm{vech}(T)} \ell_q] &= L\{(T^{-1} \otimes T^{-T})K + I_d \otimes T^{-T}T^{-1}\} L^T \\ &= 2L(I_d \otimes T^{-T})N(I_d \otimes T^{-1})L^T. \end{aligned}$$

Finally, the natural gradient is given by

$$\begin{aligned} \widetilde{\nabla}_{\lambda_i} \mathcal{L} &= F^{-1}_{\lambda_i} \nabla_{\lambda_i} \mathcal{L} \\ &= \begin{bmatrix} \Sigma & 0 \\ 0 & \frac{1}{2} L(I_d \otimes \Lambda_i) L^T (LNL^T)^{-1} L(I_d \otimes \Lambda_i^T) L^T \end{bmatrix} \mathrm{E}_{\phi(z)} \begin{bmatrix} \nabla_\theta h(\theta) \\ \mathrm{vech}(\bar{G}_i) \end{bmatrix} \\ &= \mathrm{E}_{\phi(z)} \begin{bmatrix} \Sigma \nabla_\theta h(\theta) \\ \frac{1}{2} L(I_d \otimes \Lambda_i) L^T (LNL^T)^{-1} L(I_d \otimes \Lambda_i^T) L^T \mathrm{vech}(\bar{G}_i) \end{bmatrix}. \end{aligned}$$

If $H_i = \Lambda_i^T \bar{G}_i$, then

$$\begin{aligned} &\tfrac{1}{2} L(I_d \otimes \Lambda_i) L^T (LNL^T)^{-1} L(I_d \otimes \Lambda_i^T) L^T \mathrm{vech}(\bar{G}_i) \\ &= \tfrac{1}{2} L(I_d \otimes \Lambda_i) L^T (LNL^T)^{-1} L(I_d \otimes \Lambda_i^T) \mathrm{vec}(\bar{G}_i) \\ &= \tfrac{1}{2} L(I_d \otimes \Lambda_i) L^T (2I_{d(d+1)/2} - LKL^T) L\mathrm{vec}(\Lambda_i^T \bar{G}_i) \qquad \text{[Lemma 2(ii)]} \\ &= \tfrac{1}{2} L(I_d \otimes \Lambda_i)(2I_{d^2} - L^T LK) L^T \mathrm{vech}(\bar{H}_i) \\ &= \tfrac{1}{2} L(I_d \otimes \Lambda_i)(2I_{d^2} - L^T LK) \mathrm{vec}(\bar{H}_i) \\ &= L(I_d \otimes \Lambda_i)\mathrm{vec}(\bar{H}_i) - \tfrac{1}{2} L(I_d \otimes \Lambda_i) L^T LK \mathrm{vec}(\bar{H}_i) \\ &= L\mathrm{vec}(\Lambda_i \bar{H}_i) - \tfrac{1}{2} L(I_d \otimes \Lambda_i) L^T \mathrm{vech}(\bar{H}_i^T) \\ &= \mathrm{vech}(\Lambda_i \bar{H}_i) - \tfrac{1}{2} L(I_d \otimes \Lambda_i) L^T \mathrm{vech}(\mathrm{dg}(\bar{H}_i)) \end{aligned}$$

$$= \text{vech}(\Lambda_i \bar{H}_i) - \tfrac{1}{2} L (I_d \otimes \Lambda_i) \text{vec}(\text{dg}(\bar{H}_i))$$
$$= \text{vech}(\Lambda_i \bar{H}_i) - \tfrac{1}{2} \text{vech}(\Lambda_i \text{dg}(\bar{H}_i))$$
$$= \text{vech}[\Lambda_i \{\bar{H}_i - \text{dg}(\bar{H}_i)/2\}].$$

□