Manifold Gaussian Variational Bayes on the Precision Matrix

Martin Magris  
magris@ece.au.dk  
Department of Electrical and Computer Engineering, Aarhus University, Aarhus 8200, Denmark, and Instituto Tecnológico Autónomo de México (ITAM), 01080 Ciudad de México, México

Mostafa Shabani  
mshabani@ece.au.dk  
Alexandros Iosifidis  
ai@ece.au.dk  
Department of Electrical and Computer Engineering, Aarhus University, Aarhus 8200, Denmark

We propose an optimization algorithm for variational inference (VI) in complex models. Our approach relies on natural gradient updates where the variational space is a Riemann manifold. We develop an efficient algorithm for gaussian variational inference whose updates satisfy the positive definite constraint on the variational covariance matrix. Our manifold gaussian variational Bayes on the precision matrix (MGVBP) solution provides simple update rules, is straightforward to implement, and the use of the precision matrix parameterization has a significant computational advantage. Due to its black-box nature, MGVBP stands as a ready-to-use solution for VI in complex models. Over five data sets, we empirically validate our feasible approach on different statistical and econometric models, discussing its performance with respect to baseline methods.

1 Introduction

Although Bayesian principles are not new to machine learning (ML) (Mackay, 1992, 1995; Lampinen & Vehtari, 2001), it is only with the recent methodological developments that we are witnessing a growing use of Bayesian techniques in the field (Zhang et al., 2018; Trusheim et al., 2018; Osawa et al., 2019; Khan et al., 2018; Khan & Nielsen, 2018). In typical ML settings, the applicability of sampling methods for the challenging computation of the posterior is prohibitive; however, approximate methods such as variational inference (VI) have been proved suitable and successful (Saul et al., 1996; Wainwright & Jordan, 2008; Hoffman et al., 2013; Blei et al., 2017). VI is generally performed with stochastic gradient descent.
(SGD) methods (Robbins & Monro, 1951; Hoffman et al., 2013; Salimans & Knowles, 2014), boosted by the use of natural gradients (Hoffman et al., 2013; Wierstra et al., 2014; Khan et al., 2018), and the updates often take a simple form (Khan & Nielsen, 2018; Osawa et al., 2019; Magris et al., 2022).

Most VI algorithms rely on the extensive use of models’ gradients, and the form of the variational posterior implies additional model-specific derivations that are not easy to adapt to a general plug-and-play optimizer. Black box methods (Ranganath et al., 2014) are straightforward to implement and versatile as they avoid model-specific derivations by relying on stochastic sampling (Salimans & Knowles, 2014; Paisley et al., 2012; Kingma & Welling, 2013). The increased variance in the gradient estimates as opposed to, for example, methods relying on the reparameterization trick (Blundell et al., 2015; Xu et al., 2019) can be alleviated with variance reduction techniques.

Furthermore, most existing algorithms do not directly address parameter constraints. Under the typical gaussian variational assumption, granting positive-definiteness of the covariance matrix is an acknowledged problem (Tran, Nguyen, & Nguyen, 2021; Khan et al., 2018; Lin et al., 2020). Only a few algorithms directly tackle the problem (Osawa et al., 2019; Lin et al., 2020), see section 4. A recent approximate approach based on manifold optimization is found in Tran, Nguyen, and Nguyen (2021). For a review of the various algorithms for performing VI, see Magris and Iosifidis (2023a).

On the results of Tran, Nguyen, and Nguyen (2021) and on their manifold gaussian variational Bayes (MGVB) method, we develop a variational inference algorithm that explicitly tackles the positive-definiteness constraint for the variational covariance matrix, resembles the readily applicable natural-gradient black-box framework of Magris et al. (2022) and has computational advantages. We bridge a theoretical issue for the use of symmetric and positive-definite manifold retraction and parallel transport for gaussian VI, leading to our manifold gaussian variational Bayes on the precision matrix (MGVBP) algorithm. Our solution, based on the precision matrix parameterization of the variational gaussian distribution, furthermore has a computational advantage over the implementation of the usual canonical parameterization on the covariance matrix, as the form of the relevant gradients in our update rule is greatly simplified. We distinguish and apply two forms of the stochastic gradient estimator that are applicable in a wider context and show how to exploit certain forms of the prior/posterior further to reduce the variance of the stochastic gradient estimators. We show that MGVBP is straightforward to implement, discuss recommendations and practicalities in this regard, and demonstrate its feasibility in extensive experiments over 5 data sets, fourteen models, 3 competing VI optimizers, and a Markov chain Monte Carlo baseline.

In section 2, we review the basis of VI. In section 3, manifold optimization, discussed in section 3, is an attractive possibility. In section 4, we review the manifold gaussian variational Bayes approach and other related works. Section 5 describes the proposed approach. Section 6 discusses
implementation aspects, results are reported in section 7, and section 8 concludes this letter. Appendixes expand the experiments and provide proofs.

2 Variational Inference

Variational inference (VI) is a convenient and feasible approximate method for Bayesian inference. Let \( y \) denote the data \( p(y|\theta) \) the likelihood of the data based on some model whose \( d \)-dimensional parameter is \( \theta \). Let \( p(\theta) \) be the prior distribution on \( \theta \). In standard Bayesian inference, the posterior is retrieved via the Bayes theorem as \( p(\theta|y) = p(\theta)p(y|\theta)/p(y) \). As the marginal likelihood \( p(y) \) is generally intractable, Bayesian inference is often difficult for complex models. Though sampling techniques can tackle the problem, nonparametric and asymptotically exact Monte Carlo methods may be slow, especially in high-dimensional applications (Salimans et al., 2015).

Fixed-form VI approximates the true unknown posterior with a probability density \( q \) chosen within a tractable class of distributions \( Q \), such as the exponential family. VI turns the Bayesian inference problem into that of finding the best variational distribution \( q^* \in Q \), minimizing the Kullback-Leibler (KL) divergence from \( q \) to \( p(\theta|y) \): \( q^* = \arg\min_{q \in Q} D_{KL}(q||p(\theta|y)) \). It can be shown that the KL minimization problem is equivalent to the maximization of the so-called lower bound (LB) on \( \log p(y) \), for example, (Tran, Nguyen, & Dao, 2021). The optimization problem accounts for finding the optimal variational parameter \( \xi \) parameterizing \( q = q_\xi \) that maximizes the LB; \( \mathcal{L} \), that is, \( \xi^* = \arg\max_{\xi \in \mathcal{Z}} \mathcal{L}(\xi) \), with

\[
\mathcal{L}(\xi) := \int q_\xi(\theta) \log \frac{p(\theta)p(y|\theta)}{q_\xi(\theta)} d\theta = \mathbb{E}_{q_\xi} \left[ \log \frac{p(\theta)p(y|\theta)}{q_\xi(\theta)} \right] = \mathbb{E}_{q_\xi} [h_\xi(\theta)],
\]

where \( \mathbb{E}_{q_\xi} \) means that the expectation is taken with respect to the distribution \( q_\xi \), and \( \mathcal{Z} \) is the parameter space for \( \xi \).

The maximization of the LB is generally addressed with a gradient-descent method such as SGD (Robbins & Monro, 1951) or Adam (Kingma & Ba, 2014). The learning of the parameter \( \xi \) based on standard gradient descent is, however, problematic as it ignores the information geometry of the distribution \( q_\xi \), is not scale invariant, is unstable, and is very susceptible to the initial values (Wierstra et al., 2014). SGD implicitly relies on the Euclidean norm for capturing the dissimilarity between two distributions, which can be a poor and misleading measure of discrepancy (Khan & Nielsen, 2018). By using the KL divergence in place of the Euclidean norm, the SGD update results in the following natural gradient update,

\[
\xi_{t+1} = \xi_t + \beta_t \left[ \nabla_\xi \mathcal{L}(\xi) \right]_{\xi=\xi_t},
\]

where \( \beta_t \) is a possibly adaptive learning rate and \( t \) denotes the iteration. The above update results in improved steps toward the maximum of the
LB when optimizing it for the variational parameter \( \xi \). The natural gradient \( \hat{\nabla}_\xi \mathcal{L}(\xi) \) is obtained by rescaling the Euclidean gradient \( \nabla_\xi \mathcal{L}(\xi) \) by the inverse of the Fisher information matrix (FIM),

\[
\hat{\nabla}_\xi \mathcal{L}(\xi) = \mathcal{I}_\xi^{-1} \nabla_\xi \mathcal{L}(\xi).
\]

where \( \mathcal{I}_\xi \) denotes the FIM. A significant issue in following this approach is that \( \xi \) is unconstrained. Think of a gaussian variational posterior: in the above setting, there is no guarantee that the covariance matrix updates onto a symmetric and positive-definite matrix. As discussed in section 1, manifold optimization is an attractive possibility.

### 3 Elements of Manifold Optimization

We wish to optimize the function \( \mathcal{L} \) of the variational parameter \( \xi \) with an update like equation 2.1, where the variational parameter \( \xi \) lies in a manifold. The usual approach for unconstrained optimization reduces to finding the descent direction and performing a step in that direction to obtain function decrease. The notion of gradient is extended to manifolds through the tangent space. At a point \( \xi \) on the manifold, the tangent space \( T_\xi \) is the approximating vector space; thus given a descent direction \( \xi_\xi \in T_\xi \), a step is performed along the smooth curve on the manifold in this direction.

A Riemannian manifold is a real, smooth manifold equipped with a positive-definite inner product \( g_\xi(\cdot, \cdot) \) on the tangent space at each point \( \xi \) (see Absil et al., 2008, for a rigorous definition). A Riemann manifold, hereafter simply called a manifold, is thus a pair \( (\mathcal{S}, g) \), where \( \mathcal{S} \) is a certain set (e.g., of certain matrices). For Riemannian manifolds, the Riemann gradient denoted by \( \text{grad}_f(\xi) \) is defined as a direction on the tangent space, where the inner product of the Riemann gradient and any direction in the tangent space gives the directional derivative of the function,

\[
\langle \text{grad}_f(\xi), \xi_\xi \rangle = Df(\xi)[\eta],
\]

where \( Df(\xi)[\eta] \) denotes the directional derivative of \( f \) at \( \xi \) in the direction \( \eta \) (see Figure 1, left panel). The gradient has the property that the direction of \( \text{grad}_f(\xi) \) is the steepest-ascent direction of \( f \) at \( \xi \) (Absil et al., 2008) is important for the scope of optimization.

For a descent direction on the tangent space, the map that gives the corresponding point on the manifold is called the exponential map. The exponential map \( \text{Exp}_\xi(\xi_\xi) \) thus projects a tangent vector \( \xi_\xi \in T_\xi \) back to the manifold, generalizing the usual concept \( \xi + \xi_\xi \) in Euclidean spaces. In fact, \( \text{Exp}_\xi(\xi_\xi) \) can be thought of as the point on the manifold reached by leaving from \( \xi \) and moving in the direction \( \xi_\xi \) while remaining on the manifold (see Figure 1, middle panel). Therefore, in analogy with the usual
gradient descent approach $\zeta \leftarrow \zeta + \beta \nabla f(\zeta)$ with $\beta$ being the learning rate, on manifolds, the update is performed through retraction following the steepest direction provided by the Riemann gradient as $\text{Exp}_\zeta(\beta \text{ grad } f(\zeta))$.

In practice, exponential maps are cumbersome to compute; retractions are used as first-order approximations. A Riemannian manifold also has a natural way of transporting vectors. Parallel transport moves tangent vectors from one tangent space to another while preserving the original length and direction (see Figure 1, right panel), extending the use of momentum gradients to manifolds. As for the exponential map, a parallel transport is in practice approximated by the so-called vector transport. Note that the forms of retraction and vector transport, as much as that of the Riemann gradient, depend on the specific metric adopted in the tangent space.

Thinking of $\zeta$ as the parameter of a gaussian distribution, $\zeta$ involves elements related to $\mu$, unconstrained over $\mathbb{R}^d$, and elements related to the covariance matrix, constrained to define a valid covariance matrix: the product space of Riemannian manifolds is itself a Riemannian manifold. The exponential map, gradient, and parallel transport are defined as the Cartesian product of the individual ones, while the inner product is defined as the sum of the inner product of the components in their respective manifolds (Hosseini & Sra, 2015).

3.1 Two Manifolds. By $\mathcal{S} = \{ P \in \mathbb{R}^{d \times d} : P = P^T, P > 0 \}$, the set of symmetric and positive-definite (SPD) $d \times d$ matrices, we denote by $\mathcal{M} = (\mathcal{S}, g_P)$ the corresponding manifold, with the metric $g_P$ between two vectors $\zeta$ and $\xi$ at $P$ defined as $g_P = \text{Tr}(P^{-1} \zeta P^{-1} \xi)$. In the remainder of the letter, as the metric $g_P$ is derived from the Frobenius norm (i.e., the Euclidean norm of a matrix), we refer to this metric between tangent vectors as the Euclidean metric (Bhatia et al., 2019; Han et al., 2021). By denoting with $f$
a generic smooth function of $P$, from, for example, Hosseini and Sra (2015), Pennec (2020), and Boumal et al. (2014), the relationship between the usual Euclidean gradient $\nabla f(P)$ and the Riemann gradient is

$$\text{grad } f(P) = P \nabla f(P) P,$$

assuming $\nabla f(P)$ is a symmetric matrix. For the manifold $\mathcal{M}$, the retraction at point $P$ in the direction $\xi$ is computed as

$$R_P(\xi) = P + \xi + \frac{1}{2} \xi P^{-1} \xi, \quad \xi \in T_P \mathcal{M},$$

while the vector transport of the tangent vector from $P_1$ to $P_2$ is given by

$$T_{P_1 \to P_2}(\xi) = E \xi E^\top, \quad E = (P_2 P_1^{-1})^{\frac{1}{2}}.$$

An alternative to the manifold $\mathcal{M}$, as for the discussion in section 2, another metric that gained popularity is the Fisher-Rao metric. This metric depends on the Fisher information matrix (FIM) $I_P$, and for two vectors $\zeta, \xi$ at a point $P$ in the tangent space $T_P$, it defines the inner product as $\langle \zeta, \xi \rangle = \zeta^\top I_P \xi$. We refer to the manifold of the SPD matrices equipped with the Fisher-Rao metric as $\mathcal{F}$. Importantly, the manifold $\mathcal{F}$ is the one adopted by the work of Tran, Nguyen, and Nguyen (2021), the basis of this letter. Tran, Nguyen, and Nguyen (2021) furthermore show that the Riemann gradient in $\mathcal{F}$ corresponds to the natural gradient, that is,

$$\text{grad } f(P) = \tilde{\nabla} f(P).$$

From the form of the gaussian FIM, $\tilde{\nabla} f(P) = 2P \nabla f(P) P$ (see appendix C.3), note that

$$P \nabla f(P) P = \frac{1}{2} \tilde{\nabla} f(P).$$

The Riemann gradient in $\mathcal{M}$ is one-half of the natural gradient, which conversely is the Riemann gradient in $\mathcal{F}$. Note that the above also applies to $P^{-1}$ since the inverse of an SPD matrix is as well an SPD.

4 Related Work

Tran, Nguyen, and Nguyen (2021) adopt a fixed-form $d$-dimensional gaussian distribution with mean $\mu$ and covariance matrix $\Sigma$ for the variational approximation $q_\zeta$, where the variational parameter $\zeta^c$ collects all the elements of $\mu$ and $\Sigma$, that is, $\zeta^c = (\mu^\top, \text{vec}(\Sigma)^\top)^\top$ (canonical
parameterization). There are no restrictions on $\mu$, yet the covariance matrix $\Sigma$ is constrained to the manifold $\mathcal{F}$ of SPD matrices equipped with the Fisher-Rao metric.

For a multivariate gaussian $q_{\zeta}$, the exact form of the corresponding FIM reads (Mardia & Marshall, 1984)

$$
I_{\zeta} = \begin{pmatrix}
\Sigma^{-1} & 0 \\
0 & I_{\zeta}(\Sigma)
\end{pmatrix},
$$

with $I_{\zeta}(\Sigma)_{\sigma_{ij},\sigma_{kl}} = \frac{1}{2} \text{Tr} \left( \Sigma^{-1} \frac{\partial \Sigma}{\partial \sigma_{ij}} \Sigma^{-1} \frac{\partial \Sigma}{\partial \sigma_{kl}} \right)$ being the generic element of the $d^2 \times d^2$ matrix $I_{\zeta}(\Sigma)$. The manifold gaussian variational Bayes (MGVB) method (Tran, Nguyen, & Nguyen, 2021) relies on the approximation $I_{\zeta}(\Sigma) \approx \Sigma^{-1} \otimes \Sigma^{-1}$, where $\otimes$ denotes the Kronecker product. Accordingly, Tran, Nguyen, and Nguyen (2021) compute the convenient approximate form of the natural gradients of the LB with respect to $\mu$ and $\Sigma$, respectively, as

$$
\tilde{\nabla}_\mu L(\zeta^c) = \Sigma \nabla_\mu L(\zeta^c) \quad \text{and} \quad \tilde{\nabla}_\Sigma L(\zeta^c) \approx \Sigma \nabla_\Sigma L(\zeta^c) \Sigma.
$$

In virtue of the natural gradient definition, the first natural gradient is exact, while the second is approximate. Thus, Tran, Nguyen, and Nguyen (2021) adopt the following updates on the variational parameter:

$$
\mu \leftarrow \mu + \beta \tilde{\nabla}_\mu L(\zeta^c) \quad \text{and} \quad \Sigma \leftarrow R_\Sigma(\beta \tilde{\nabla}_\Sigma L(\zeta^c)),
$$

where $R_\Sigma(\cdot)$ denotes the retraction form, equation 3.2. Retractions are of central importance in manifold optimization, and for the scope of the letter they enable projecting vectors from the so-called tangent space back to the manifold. Momentum gradients can be used in place of plain natural ones. In particular, the momentum gradient for the update of $\Sigma$ relies on a vector transport, granting that at each iteration, the weighted gradient remains in the tangent space of the manifold (see section 3).

A method of handling the positivity constraint in diagonal covariance matrices is the variational online Gauss-Newton (VOGN) optimizer (Khan et al., 2018; Osawa et al., 2019). VOGN relates to the VON (Khan & Lin, 2017) update as it indirectly updates $\mu$ and $\Sigma$ from gaussian natural parameters. Following a non-black-box approach, VOGN uses some theoretical results on the gaussian distribution to recover an update for $\Sigma$ that involves the Hessian of the likelihood. The Hessian is estimated as the samples’ mean.

\footnote{We follow the presentation of Tran, Nguyen, and Nguyen (2021), where $\tilde{\nabla}_\Sigma L(\zeta^c)$ is approximated up to a scaling constant as $\Sigma \nabla_\Sigma \Sigma$. Indeed Lin et al. (2020) clarify that the exact form of $I_{\zeta}(\Sigma)$ should read $2(\Sigma^{-1} \otimes \Sigma^{-1})$ (see Barfoot, 2020), leading to the exact natural gradient $\tilde{\nabla}_\Sigma L(\zeta^c) = 2 \Sigma \nabla_\Sigma L(\zeta^c) \Sigma$.}
squared gradient, granting the nonnegativity of the diagonal covariance update. Osawa et al. (2019) devise the computation of the approximate Hessian in a block-diagonal fashion within the layers of a deep-learning model.

Lin et al. (2020) extend the above to handle the positive-definiteness constraint by adding a term to the update rule for $\Sigma$, applicable to certain partitioned structures of the FIM. The retraction map in Lin et al. (2020) is more general than Tran, Nguyen, and Nguyen (2021) and obtained through a different Riemann metric, from which MGVB is retrieved as a special case. As opposed to MGVB, the use of the reparameterization trick in Lin et al. (2020) requires model-specific computation or auto-differentiation. (See Lin et al., 2021, for an extension on stochastic, nonconvex problems.)

Alternative methods that rely on unconstrained transformations such as the Cholesky factor (Tan, 2021), or on the adaptive adjustment of the learning rate (Khan & Lin, 2017) lie outside the manifold context discussed here. Among the methods that do not control for the positive-definiteness constraint, the QBVI update (Magris et al., 2022) provides a comparable black-box method that despite other black-box VI algorithms, uses exact natural gradients updates obtained without the computation of the FIM.

5 Manifold Gaussian Variational Bayes on the Precision Matrix

5.1 Caveats with the MGVB Update. We identify three criticalities concerning the MGVB update as formulated and implemented by Tran, Nguyen, and Nguyen (2021) providing the motivation for this letter and the basis for developing our update.

1. Tran, Nguyen, and Nguyen (2021) adopt the manifold of symmetric and positive-definite matrices equipped with the Fisher-Rao metric. We denote such manifold by $F$. As of section 3, in manifold optimization, the update is carried out by employing the retraction function on the Riemann gradient, mapping elements of the tangent space onto the manifold. Indeed, the update (Tran, Nguyen, & Nguyen, 2021) suggest the variational covariance matrix $\Sigma_1$ for optimizing the lower-bound objective $L$ involves $R_\Sigma(\beta \text{grad} L)$. Interestingly, they show that for the manifold $F$, the Riemann gradient corresponds to the natural gradient (see their lemma 2.1). However, the form of the retraction they adopt is that of the manifold $M$ of the SPD matrices equipped with the Euclidean metric, not the retraction form for the manifold $\Sigma$ induced by the Fisher-Rao metric. The retraction form for the manifold $S$ is an open question. In practice, Tran, Nguyen, and Nguyen (2021) mix elements of the two manifolds as they apply to the retraction derived from $M$ on a Riemann gradient of the manifold $F$. This same point is also raised in Lin et al. (2020), which, in fact, underlines that in Tran, Nguyen, and Nguyen (2021), the chosen form of the retraction is not well justified as it is specific for the SPD matrix manifold, whereas the natural gradient is computed within a different manifold.
Tran, Nguyen, and Nguyen (2021) establish the equivalence between the Riemann gradient and the natural gradient in $\mathcal{F}$. However, in equations 5.2 and 5.4 of Tan (2021), and the implementation accompanying the paper, the natural gradient reads $\Sigma \nabla_{\Sigma} L \Sigma$ in place of $2 \Sigma \nabla_{\Sigma} L \Sigma$ (see appendix C.1 and note 1). This leads to the following two observations:

2. The halved natural gradient $\Sigma \nabla_{\Sigma} L \Sigma$ is a Riemann gradient for the manifold $\mathcal{M}$ (see equation 3.1). Therefore, the form of retraction, equation 3.2, is coherent and applicable; thus, the update, equation 4.3 on $\Sigma \nabla_{\Sigma} L \Sigma$ is formally correct. However, this is a fully consistent procedure for the manifold $\mathcal{M}$, and not for the manifold $\mathcal{F}$ adopted in Tran, Nguyen, and Nguyen (2021).

3. By adopting $\Sigma \nabla_{\Sigma} L \Sigma$ in place of $2 \Sigma \nabla_{\Sigma} L \Sigma$, in the above light, we recognize that the overall update in Tran, Nguyen, and Nguyen (2021) reads as a hybrid update: whereas $\mu$ is updated with a natural ingredient update, $\Sigma$ is updated with the gradient $\Sigma \nabla_{\Sigma} L \Sigma$, Riemannian in $\mathcal{F}$, but not natural.

Regarding point 3, as for equation 3.4 $\Sigma \nabla_{\Sigma} L \Sigma = \frac{1}{2} \tilde{\nabla}_{\Sigma} L$. Thus, the Riemann gradient $\text{grad} L$ shares the same direction in both $\mathcal{F}$ and $\mathcal{M}$. Therefore, in both manifolds, the respective Riemann gradients point in the direction of the natural gradient. So the update is, in practice, a natural gradient update. Equation 3.4, which does not appear in Tran, Nguyen, and Nguyen (2021) and in the earlier VI literature, therefore explains why MGVB actually works despite the above issues.

A remedy for the major points 1 and 2 above requires revising the MGVB update for $\Sigma$—specifically, to adopt retraction for $\mathcal{M}$ applied to an actual Riemann gradient for $\mathcal{M}$, i.e., $R_{\Sigma}(\beta \tilde{\nabla}_{\Sigma} L)$. Figure 2 depicts the difference between the MGVB update and such a revisited version. The three panels show a simple example of applying retraction 3.2 on the actual Riemann gradient on $\mathcal{M}$, highlighting a nonirrelevant impact on the steps of line search and the convergence of the algorithm toward the optimum.

The computation of the Riemann gradient $\frac{1}{2} \tilde{\nabla}_{\Sigma} L = \Sigma \nabla_{\Sigma} L \Sigma$, which involves two matrix multiplications of $O(d^3)$ operations, can, however, be simplified considering a parameterization of the variational distribution in terms of its covariance matrix $\Sigma^{-1}$. This leads to our suggested update in the next section.

5.2 Updating the Precision Matrix. Consider a variational gaussian distribution with mean $\mu$ and positive-definite covariance matrix $\Sigma$. The corresponding precision matrix $\Sigma^{-1}$ is thus well identified, and the following proposition establishes a central relationship in this regard.

---

2 This is the case of practical relevance, ruling out singular gaussian distributions for which the discussion here is out of scope. Positive-definiteness is not restrictive and is aligned with Tran, Nguyen, and Nguyen (2021).
Figure 2: Variational inference for the simple linear regression model, with 100 observations simulated according to $Y_i = X_i \varepsilon_i \sim N(0, 1)$, $X = [0, 0.05, 0.1, \ldots, 5]$, for the MGVB update and the revisited version (MGVB rev. label) applying retraction on the Riemannian gradient in $\mathcal{M}$. The red circle denotes the true posterior parameter computed with standard results in Bayesian linear regression.

Proposition 1. For a $d$-dimensional gaussian variational posterior whose mean is denoted by $\mu$ and covariance matrix by $\Sigma$, consider the following two parameterizations: the canonical parameterization $\zeta^c = (\mu^\top, \text{vec}(\Sigma^\top))^\top$ and the inverse parameterization $\zeta^i = (\mu^\top, \text{vec}(\Sigma^{-1})^\top)^\top$. It holds that

\begin{align}
\tilde{\nabla}_\mu \mathcal{L}(\zeta^i) &= \Sigma \nabla_\mu \mathcal{L}(\zeta^c). \\
\tilde{\nabla}_{\Sigma^{-1}} \mathcal{L}(\zeta^i) &= -2 \nabla_\Sigma \mathcal{L}(\zeta^c).
\end{align}

Equation 5.2 establishes an algebraic link between the natural gradient of the lower bound in the parameterization $\zeta^i$ with respect to the precision matrix $\Sigma^{-1}$—inverse (covariance) parameterization—and the Euclidean gradient of the lower bound with respect to the covariance matrix $\Sigma$. In particular, proposition 1 suggests that under the parameterization $\zeta^i$, one could update $\mu$ and $\Sigma^{-1}$, for which natural gradients are of a closed-form solution and correspond to the standard Euclidean gradients of $\mathcal{L}(\zeta^c)$ under the canonical parameterization. Compared to the canonical parameterization adopted by Tran, Nguyen, and Nguyen (2021) and the discussion in section 5.1, the advantage of using the parameterization $\zeta^i$ is clear: $\tilde{\nabla}_{\Sigma^{-1}} \mathcal{L}(\zeta^i)$ requires only the computation of $\nabla_\Sigma \mathcal{L}(\zeta^c)$, whereas $\tilde{\nabla}_{\Sigma^{-1}} \mathcal{L}(\zeta^i) = \Sigma \nabla_\Sigma \mathcal{L} \Sigma$ involves additional matrix multiplications that significantly increase the time complexity of the lower-bound optimization. Section 5.3 shows, furthermore, that such Euclidean gradients are straightforward to compute and that their variance can be conveniently controlled.
The matrix $\Sigma^{-1}$ is symmetric and positive definite; it thus lies in the manifold $\mathcal{M}$ and can be effectively updated with the retraction algorithm for $\Sigma$ in equation 4.3:

$$
\Sigma^{-1} \leftarrow R_{\Sigma^{-1}} (\beta \Sigma^{-1} \nabla_{\Sigma^{-1}} \mathcal{L}(\zeta^c) \Sigma^{-1}) = R_{\Sigma^{-1}} \left( \frac{\beta}{2} \tilde{\nabla}_{\Sigma^{-1}} \mathcal{L}(\zeta^c) \right) = R_{\Sigma^{-1}} (-\beta \nabla_{\Sigma} \mathcal{L}(\zeta^c)).
$$

(5.3)

As opposed to equation 4.3, updating $\Sigma$ upon the approximation $I_{\zeta^c}(\Sigma) \approx \Sigma^{-1} \otimes \Sigma^{-1}$ for tackling the positive-definite constraint, we update $\Sigma^{-1}$, for which the natural gradient is available in an exact form, avoids the computation of the FIM, and reduces to the simple and standard computation of the Euclidean gradient $\nabla_{\Sigma} \mathcal{L}(\zeta^c)$. For updating $\mu$, it is reasonable to adopt a plain SGD-like step driven by the natural parameter $\tilde{\nabla}_{\mu} \mathcal{L}(\zeta^c) = \nabla_{\mu} \mathcal{L}(\zeta^c)$, as in Tran, Nguyen, and Nguyen (2021). We refer to the following update rules as manifold gaussian variational Bayes on the precision matrix (MGVB):

$$
\mu_{t+1} = \mu_t + \beta \Sigma \nabla_{\mu} \mathcal{L}_t(\zeta^c),
$$

(5.4)

$$
\Sigma_{t+1}^{-1} = R_{\Sigma^{-1}} (-\beta \nabla_{\Sigma} \mathcal{L}_t(\zeta^c)),
$$

(5.5)

where the gradients are evaluated at the current value of the parameters (e.g., $\nabla_{\mu} \mathcal{L}_t(\zeta^c) = \nabla_{\mu} \mathcal{L}(\zeta^c)|_{\zeta^c = \zeta^c_t}$). Opposed to the MGVB update of Tran, Nguyen, and Nguyen (2021), the update for $\Sigma^{-1}$ is consistent for the manifold $\mathcal{M}$, as it employs the corresponding Riemann gradient, $\Sigma^{-1} \nabla_{\Sigma^{-1}} \mathcal{L}(\zeta^c) \Sigma^{-1} = \nabla_{\Sigma} \mathcal{L}(\zeta^c)$, which is $O(d^3)$ cheaper than computing $\tilde{\nabla}_{\Sigma} \mathcal{L}(\zeta^c) = \nabla_{\Sigma} \mathcal{L}(\zeta^c) \Sigma$.

### 5.3 Gradient Computations.

We elaborate on how to evaluate the gradients $\nabla_{\Sigma} \mathcal{L}(\zeta^c)$ and $\nabla_{\mu} \mathcal{L}(\zeta^c)$. We follow the black-box approach (Ranganath et al., 2014) under which such gradients are approximated via Monte Carlo (MC) sampling and rely on function queries only. Implementing MGVBP does not require the model’s gradient to be specified or to be computed numerically (e.g., with backpropagation). The so-called log-derivative trick (Ranganath et al., 2014) makes it possible to evaluate the gradients of the LB as an expectation with respect to the variational distribution. In particular,

$$
\nabla_{\zeta^c} \mathcal{L}(\zeta^c) = \mathbb{E}_{q_{\zeta^c}} [\nabla_{\zeta^c} [\log q_{\zeta^c}(\theta)] h_{\zeta^c}(\theta)],
$$

where $h_{\zeta^c}(\theta) = \log[p(\theta)p(y|\theta)/q_{\zeta^c}(\theta)]$. The gradient can be easily estimated using $S$ samples from the posterior through the unbiased estimator
\[ \nabla_c \mathcal{L}(\xi^c) \approx \frac{1}{S} \sum_{s=1}^{S} [\nabla_c [\log q_c(\theta_s)] h_c(\theta_s)], \quad (5.6) \]

with \( \theta_s \sim q_c \). For the gaussian variational case under consideration, it can be shown that (Wierstra et al., 2014; Magris et al., 2022)

\[ \nabla_\mu \log q_c(\theta) = \Sigma^{-1}(\theta - \mu) = \nu, \quad (5.7) \]

\[ \nabla_\Sigma \log q_c(\theta) = -\frac{1}{2}(\Sigma^{-1} - \nu\nu^\top). \quad (5.8) \]

Equations 5.7 and 5.8, along with equation 5.6 and proposition 1 immediately lead to the feasible natural gradients estimators:

\[ \hat{\nabla}_\mu \mathcal{L}(\xi_i^c) \approx \Sigma_i \hat{\nabla}_\mu \mathcal{L}(\xi^c) = \frac{1}{S} \sum_{s=1}^{S} [(\theta_s - \mu)h_c(\theta_s)], \]

\[ \hat{\nabla}_\Sigma^{-1} \mathcal{L}(\xi_i^c) \approx -\hat{\nabla}_\Sigma \mathcal{L}(\xi^c) = \frac{1}{2S} \sum_{s=1}^{S} [(\Sigma^{-1} - \nu_s\nu_s^\top)h_c(\theta_s)], \quad (5.9) \]

with \( \nu_s = \Sigma^{-1}(\theta_s - \mu) \). As for the MGVB update, MGVBP applies exclusively to gaussian variational posteriors, yet no constraints are imposed on the parametric form of the prior \( p(\theta) \). When considering a gaussian prior, the implementation of the MGVBP update can take advantage of some analytical results leading to MC estimators of reduced variance, implemented over the log-likelihood log \( p(y|\theta_s) \) rather than the \( h \)-function.

In appendix C.5, we show that under a gaussian prior specification, the above updates can also be implemented in terms of the model likelihood rather than the \( h \)-function. At iteration \( t \), the general form of the gradients evaluated at the current (epoch-\( t \)) values of the parameters read

\[ \hat{\nabla}_\mu \mathcal{L}_t(\xi^i) \approx c_{\mu_i} + \frac{1}{S} \sum_{s=1}^{S} [(\theta_s - \mu_i) \log f(\theta_s)], \quad (5.9) \]

\[ \hat{\nabla}_\Sigma^{-1} \mathcal{L}_t(\xi^i) \approx C_{\Sigma_i} + \frac{1}{2S} \sum_{s=1}^{S} [(\Sigma_i^{-1} - \nu_{s,i}\nu_{s,i}^\top) \log f(\theta_s)], \quad (5.10) \]

with \( \nu_{s,i} = \Sigma_i^{-1}(\theta_s - \mu_i) \), where in general (whether the prior is gaussian or not)

\[
\begin{cases}
C_{\Sigma_i} = 0, \\
c_{\mu_i} = 0, \\
\log f(\theta_s) = h_c(\theta_s),
\end{cases}
\quad (5.11)
\]
whereas for a gaussian prior, one can adopt

\[
\begin{align*}
C_{\Sigma} &= -\frac{1}{2}\Sigma^{-1} + \frac{1}{2}\Sigma_0^{-1}, \\
c_{\mu} &= -\Sigma_t\Sigma_0^{-1}(\mu_t - \mu_0), \\
\log f(\theta_s) &= \log p(y|\theta_s).
\end{align*}
\] (5.12)

That is, equation 5.11 holds for a generic prior (including a gaussian prior as a special case), whereas equation 5.12 holds with gaussian priors only. As shown in appendix C.5, under a gaussian prior, certain components of the $h$-function involved in equation 5.11 have an algebraic solution, and the MC estimator, equation 5.12, based on the log-likelihood function is of reduced variance. Thus, under a gaussian prior, equation 5.12 is preferred to equation 5.11. This alternative estimator is applicable in Tran, Nguyen, and Nguyen (2021) and, generally, in other black-box gaussian VI contexts. Note that the log-likelihood case does not involve an additional inversion for retrieving $1/\Sigma_1$ in $c_{\mu}$, as $\Sigma$ is required in the second-order retraction (for both MGVB and MGVBP). This aspect is further developed in section 5.7.

For inverting $1/\Sigma$ we suggest inverting the Cholesky factor $L$ of $\Sigma^{-1}$ and compute $\Sigma$ as $L^{-\top}L^{-1}$. The triangular form of $L$ can be inverted with back-substitution, requiring $d^3/3$ flops instead of $d^3$. $L^{-\top}$ is furthermore used for generating the draws $\theta_s$ as $\theta_s = \mu + L^{-\top}e$, with $e \sim N(0, 1)$. We suggest using control variates to reduce the variance of the stochastic gradient estimators (see section 6.2).

Though the lower bound is not directly involved in MGVBP updates, it can be naively estimated at each iteration as

\[
\hat{\mathcal{L}}_t(\xi^c) = \frac{1}{S}\sum_{s=1}^{S}\left[\log p(\theta_s) + \log p(y|\theta_s) - \log q_{\xi^c}(\theta_s)\right], \quad \theta \sim q_{\xi^c}.
\] (5.13)

$\hat{\mathcal{L}}_t(\xi^c)$ is required for terminating the optimization routine (see section 6.4), verifying anomalies in the algorithm (the LB should increase across the iterations and eventually and converge), and comparing MGVBP with MGVB, as in section 7.

### 5.4 Retraction and Vector Transport.

Aligned with Tran, Nguyen, and Nguyen (2021), we adopt the retraction method advanced in Jeuris et al. (2012) for the manifold $\mathcal{M}$, but on the actual Riemannian gradients for this manifold,

\[
R_{\Sigma^{-1}}(\xi) = \Sigma^{-1} + \frac{1}{2}\xi\Sigma\xi.
\] (5.14)
with \( \xi \in T_{\Sigma^{-1}} \mathcal{M} \) being the rescaled natural gradient \( \beta/2 \bar{\nabla}_{\Sigma^{-1}} \mathcal{L}(\zeta^c) = -\beta \nabla_{\Sigma} \mathcal{L}(\zeta^c) \). Vector transport is easily implemented by

\[
T_{\Sigma^{-1}_{t+1}} \rightarrow \Sigma^{-1}_{t+1} (\xi) = E \xi E^\top,
\]

with \( E = (\Sigma^{-1}_{t+1})^{1/2} \), \( \xi \in T_{\Sigma^{-1}} \mathcal{M} \). In implementations, for numerically granting the symmetric form of a matrix \( P \), we compute \( P \) as \( 1/2( P + P^\top) \).

We refer to the Manopt toolbox (Boumal et al., 2014) for further practical details on implementing the above two algorithms in a numerically stable fashion. The momentum gradients immediately follow:

\[
\bar{\nabla} \text{mom.}_{\Sigma^{-1}_{t+1}} L_{t+1}(\zeta^i) = \omega T_{\Sigma^{-1}_{t+1}} \rightarrow (\bar{\nabla} \text{mom.}_{\Sigma^{-1}} L_t(\zeta^i)) + (1 - \omega) \bar{\nabla}_{\Sigma^{-1}} L_{t+1}(\zeta^i),
\]

(5.16)

\[
\bar{\nabla} \text{mom.}_\mu L_{t+1}(\zeta^i) = \omega \bar{\nabla} \text{mom.}_\mu L_t(\zeta^i) + (1 - \omega) \bar{\nabla}_\mu L_t(\zeta^i),
\]

(5.17)

where the weight \( 0 < \omega < 1 \) is a hyperparameter. Algorithm 1 summarizes the MGVBP update for the gaussian prior-variational posterior case. Aspects of relevance in its implementation are discussed in section 6.

### 5.5 Isotropic Prior

For midsized to large-scale problems, the prior is commonly specified as an isotropic gaussian of mean \( \mu_0 \), often \( \mu_0 = 0 \), and covariance matrix \( \tau^{-1} I_d \), with \( \tau > 0 \) a scalar precision parameter. The covariance matrix of the variational posterior can be diagonal or not. Whether a full covariance specification (\( d^2 \) parameters) can provide additional degrees of freedom to gauge models’ predictive ability, a diagonal posterior (\( d \) parameters) can be practically and computationally convenient to adopt (e.g., in large-sized problems). The diagonal-posterior assumption is broadly adopted in Bayesian inference and VI (Blundell et al., 2015; Ganguly & Earp, 2021; Tran, Nguyen, and Dao, 2021) and Bayesian ML applications (Kingma & Welling, 2013; Graves, 2011; Khan et al., 2018; Osawa et al., 2019). In appendix A, we provide a block-diagonal variant.

#### 5.5.1 Isotropic Prior and Diagonal Gaussian Posterior

Assume a \( d \)-variate diagonal gaussian variational specification, that is, \( q \sim \mathcal{N}(\mu, \Sigma) \) with \( \text{diag}(\Sigma) = \sigma^2 \), \( \Sigma_{ij} = 0 \), for \( i, j = 1, \ldots, d \) and \( i \neq j \). In this case, \( \Sigma^{-1} = \text{diag}(1/\sigma^2) \), where the division intended element-wise is now a \( d \times 1 \) vector. By \( \zeta^c = (\mu^\top, (\sigma^2)^\top)^\top \) and \( \zeta^i = (\mu^\top, (\sigma^{-2})^\top)^\top \), \( \nabla_{\zeta^c} \mathcal{L}(\zeta^c) = \text{diag}(\nabla_{\zeta^c} \mathcal{L}(\zeta^c)) \).

Updating \( \Sigma^{-1} \) amounts to updating \( \sigma^{-2} \): the natural gradient retraction-based update for \( \sigma^{-2} \) is now based on the equality \( \bar{\nabla}_{\zeta^c} \mathcal{L}(\zeta^i) = -\nabla_{\zeta^c} \mathcal{L}(\zeta^c) \), so that the general-case MGVBP update reads

\[
\mu_{t+1} = \mu_t + \sigma_t^2 \odot \beta \nabla_\mu \mathcal{L}(\zeta^c),
\]

\[
\sigma_{t+1}^{-2} = R_{\sigma_{t+1}^{-2}} ( -\beta \nabla_{\zeta^c} \mathcal{L}(\zeta^c) ),
\]
where \( \odot \) denotes the element-wise product, and the retraction is adapted to

\[
R_{\sigma_t^{-2}}(\xi) = \sigma_t^{-2} + \xi + \frac{1}{2} \xi \odot \sigma_t^{-2} \odot \xi,
\]

where \( \xi \) is a \( d \)-dimensional vector. The corresponding MC estimators for the gradients are

\[
\hat{\nabla}_t \mathcal{L} \approx \sigma^2 \odot \hat{\nabla}_t \mathcal{L}
\]

\[
= c_{\mu_t} + \frac{1}{S} \sum_{s=1}^{S} ((\theta_s - \mu_t) \log p(y|\theta_s)),
\]
\[ \nabla_{\sigma^{-2}} \mathcal{L}_t(\xi^i) \approx -\nabla_{\sigma^{-2}} \mathcal{L}_t(\xi^c) \]

\[ = c_{\sigma_i^2} + \sigma_i^{-2} \otimes \frac{1}{2S} \sum_{s=1}^{S} \left[ (1_d - (\theta_s - \mu_i)^2 \otimes \sigma^{-2}_i) \log p(y|\theta_s) \right], \]

where \( c_{\sigma_i^2} = -1/2\sigma_i^{-2} + \tau/2 \), \( c_{\mu_i} = \tau \sigma_i^{-2} \otimes (\mu_i - \mu_0) \), \( \theta_s \sim \mathcal{N}(\mu_i, \text{diag}(\sigma_i^{-2})) \), \( s = 1 \ldots S \), \((\theta_s - \mu_i)^2\) is intended element-wise, and \( 1_d = (1, \ldots, 1)^\top \in \mathbb{R}^d \).

In the gaussian case with a general diagonal covariance matrix, retrieving \( \sigma^{-2} \) from the updated \( \sigma^{-2} \) is inexpensive as

\[ \sigma^{-2}_i = \frac{1}{\sigma_i^{-2}} + \frac{\tau}{2}, \]

\[ c_{\mu_i} = \tau \Sigma^{-1}_i (\mu_i - \mu_0) \] or \( c_{\Sigma_i} = 0 \), depending on the prior.

### 5.5.2 Isotropic Prior and Full Gaussian Posterior.

Because of the full form of the covariance matrix, this case is rather analogous to the general one. In particular, factors \( c_{\mu_i} \) and \( c_{\Sigma_i} \) in C.6 are replaced by

\[ c_{\Sigma_i} = -1/2 \Sigma_i^{-1} + \tau/2 \]

\[ c_{\mu_i} = \tau \Sigma_i^{-1} (\mu_i - \mu_0) \] or \( c_{\Sigma_i} = 0 \), respectively, under the gaussian-prior case (log likelihood).

The MC estimators, equations 5.7 and 5.8 apply but are of reduced variance.

### 5.6 Mean-Field Variant.

Assume that for a \( d \)-variate model, the gaussian variational posterior can be factorized as

\[ q_{\xi^c}(\theta) = q_{\xi_1^c}(\theta_1)q_{\xi_2^c}(\theta_2) \ldots q_{\xi_h^c}(\theta_h) = \prod_{j=1}^{h} q_{\xi_j^c}(\theta_j), \]

with \( h \leq d \). If \( h = d \), this corresponds to a full-diagonal case where each \( \theta_j \) is a scalar and \( \xi_j^c = (\mu_j, \sigma_j) \). If \( h < d \), the variational covariance matrix \( \Sigma \) of \( q_{\xi^c} \) corresponds to a block-diagonal matrix, and \( \xi_j^c = (\mu_j^\top, \text{vec}(\Sigma_j)^\top)^\top \), with \( \mu_j \), \( \Sigma_j (\Sigma_j^{-1}) \), respectively, denoting the mean and covariance (precision) matrix of the \( h \)th block of \( \Sigma (\Sigma^{-1}) \). Also, a gaussian prior’s covariance matrix can be diagonal, full, or block-diagonal with a structure matching or not that of \( \Sigma \).

Equations 5.9 and 5.10 with the condition FILL can be used as a starting point to derive case-specific MGVBP variants based on the form of the prior covariance.

Algorithm 2 in appendix A summarizes the case with an isotropic gaussian prior of zero-mean and precision matrix \( \tau I_d \), using the gradient estimator based on the log likelihood. In this case, the block-wise natural gradients are estimated as

\[ \Sigma \nabla_{\mu_j} \mathcal{L}(\xi_j^c) = -\tau \Sigma_j^{-1} \mu_j + \frac{1}{S} \sum_{s=1}^{S} \left[ (\theta_s - \mu_j) \log p(y|\theta_s) \right], \]

\[ -\nabla_{\Sigma_j} \mathcal{L}(\xi_j^c) = -\frac{1}{2} \Sigma_j^{-1} + \frac{\tau}{2} I + \frac{1}{2S} \sum_{s=1}^{S} \left[ (\Sigma_j^{-1} - v_j v_j^\top) \log p(y|\theta_s) \right]. \]
where $I$ is the identity matrix of appropriate size for the block $j$, $v_j = \Sigma_j^{-1}(\theta_{sj} - \mu_j)$, and $\theta_{sj} \sim q(\xi_j)$, $j = 1, \ldots, h$.

### 5.7 Computational Aspects.
In terms of computational complexity, the exact MGVBP implementation is at no additional cost. Actually, the cost of computing the natural gradient in MGVBP as $-\nabla_\Sigma L(\xi^c)$ is much cheaper than the one in MGVB, $\Sigma \nabla_\xi L(\xi^c) \Sigma$, involving $O(d^3)$ operations for each matrix multiplication. However, both MGVB and MGVBP share a cumbersome matrix inversion.

Going back to equations 5.10 and 5.9 it is noticeable that under the most general estimator based on the $h$-function, $\Sigma$ is not involved in any computation—neither in the gradient involved in the update for $\Sigma^{-1}$ nor in that for $\mu$—suggesting that implicitly the MGVBP optimization routine does not require the inversion of $\Sigma^{-1}$. The above point, however, ignores that the underlying retraction masks the update for $\Sigma^{-1}$. For the retraction form in equation 5.14, both $\Sigma^{-1}$ and its inverse $\Sigma$ are needed, thus implying a matrix inversion at every iteration. That is, the covariance matrix inversion is implicit in MGVB and MGVBP methods, which both require $\Sigma^{-1}$ and $\Sigma$ at every iteration (with little surprise, as the form of the retraction is a second-order approximation of the exponential map).

With the $h$-function estimator, even though neither equation 5.7 nor 5.8 involves $\Sigma$, the inversion of $\Sigma^{-1}$ is still necessary, as $\Sigma$ is required in retraction. Similarly, adopting the log-likelihood estimator under the gaussian regime in equation C.6, is not computationally more expensive than the $h$-function case, as $\Sigma$, involved in the computation of $c_\mu$, is, once again, required by retraction. In appendix B.4, we compare the running times of MGVBP and MGVB, showing that they are indeed rather aligned, despite the computational simplicity of the MGVBP natural gradient.

As outlined in this section, $\Sigma$ can be conveniently recovered from the Cholesky factor of $\Sigma^{-1}$, with fewer flips. Finally, if $\Sigma^{-1}$ (or $\Sigma$) is diagonal, the inversion is trivial, and, when applicable, equation 5.9 is preferred.

### 6 Implementation Aspects

#### 6.1 Classification versus Regression.
The MGVBP framework applies to both regression and classification problems. In generic classification problems, predictions are based on the class of maximum probability, obtained through a softmax function returning the $K \times 1$ probability vector $p_i$, whose $k$th entry interprets as the probability that the label of the $i$th sample is in class $k$, $k \in \{1, \ldots, K\}$. From these probabilities, it is straightforward to compute the model log likelihood as $\sum_{i=1}^{N} y_i \log p_i$, where $y_i$ denotes the one-hot-key encoded true label for the $i$th input: a $1 \times K$ vector with 1 at the element corresponding to the true label and 0 elsewhere.
For regression, the parametric form of log $p(y|\theta)$ is clearly different and model specific (e.g., regression with normal errors as opposed to Poisson regression, with the latter being feasible as the use of the score estimator does not require the likelihood to be differentiable). Note that, however, additional parameters may enter into play besides the ones involved in the backbone forward model. For instance, for a regression with normal errors tackled with an artificial neural network, the gaussian likelihood involves the regression variance, an additional parameter over the network’s ones, or the degree of freedom parameter $\nu$ (with the constraint $\nu > 2$) for Student-t errors. See the application in appendix B.3.

6.2 Variance Reduction. As MGVBP does not involve model gradients, the use of the reparameterization trick (RT) (Blundell et al., 2015) is not immediate. While equations 5.4 and 5.5 would generally hold, the form of the MGVBP gradient estimators under the RT would differ from equations 5.7 and 5.8. We develop MGVBP as a general and ready-to-use solution for VI that does not require model-specific derivations, yet one may certainly enable the RT within MGVBP. The use of the RT is quite popular in VI and ML as it empirically yields more accurate estimates of the gradient of the variational objective than alternative approaches (Mohamed et al., 2020). However, note that in general, the use of the RT estimator is not necessarily preferable, as its variance can be higher than that of the score-function estimator (Xu et al., 2019; Mohamed et al., 2020). Furthermore, the applicability of the adopted score estimator is broader, as it does not require log $p(y|\theta)$ to be differentiable.

Control variates (CV) is a simple and effective approach for reducing the variance of the MC gradient estimator. The CV estimator

$$
\frac{1}{S} \sum_{s=1}^{S} \nabla_{c}[\log q_{c}(\theta_s)](\log p(y|\theta_s) - c)
$$

is unbiased for the expected gradient but of equal or smaller variance than the naive MC one. The optimal $c$ minimizing the variance of the CV estimator is (Paisley et al., 2012; Ranganath et al., 2014)

$$
c^{\star} = \frac{\text{Cov}(\nabla_{c}[\log q_{c}(\theta)] \log p(y|\theta), \nabla_{c} \log q_{c}(\theta))}{\text{Var}(\nabla_{c} \log q_{c}(\theta))}.
$$

(6.1)

For a given $S$, the CV estimator reduces the variance of the gradient estimates or, conversely, reduces the number of MC samples required for attaining a desired level of variance. For instance, for our logistic regression experiment (see Table 7), values of $S$ as little as 10 appear satisfactory (see Table 8). Yet if the iterative computation of the log likelihood is not prohibitive, we suggest adopting a more generous value. Magris et al. (2022)
furthermore show that the denominator in equation 6.1 is analytically tractable for a gaussian choice for \( q \), reducing the variance of the estimated \( c^* \) and thus improving the overall efficiency of the CV estimator.

### 6.3 Constraints on Model Parameters

MGVBP assumes a gaussian variational posterior: the mean parameter is unbounded and defined over the entire real line. Assuming that a model parameter \( \theta \) is required to lie on a support \( S \), to impose such a constraint, it suffices to identify a feasible transform \( T : \mathbb{R} \to S \) and apply the MGVBP update to the unconstrained parameter \( \psi = T^{-1}(\theta) \). By applying VI on \( \psi \), we require that the variational posterior assumption holds for \( \psi \) rather than \( \theta \). The actual distribution for \( \theta \) under a gaussian variational distribution for \( \psi \) can be computed (or approximated with a sampling method) as \( \mathcal{N}(T^{-1}(\theta); \mu, \Sigma) \det(I_{T^{-1}}(\theta)) \), with \( I_{T^{-1}} \) the Jacobian of the inverse transform, and \( \mathcal{N} \) denoting the multivariate normal probability density function (Kucukelbir et al., 2015).

**Example.** For the GARCH(1,1) model, the intercept \( \omega \), the autoregressive coefficient of the lag-one squared return \( \alpha \), and the moving-average coefficient \( \beta \) of the lag-one conditional variances need to satisfy the stationarity condition \( \alpha + \beta < 1 \) with \( \omega > 0, \alpha \geq 0, \beta \geq 0 \). Such conditions are unfeasible under a gaussian variational approximation. As in Magris and Iosifidis (2023b), we estimate the unconstrained parameters \( \psi_\omega, \psi_\alpha, \psi_\beta \), where \( \omega = T(\psi_\omega), \alpha = T(\psi_\alpha)(1 - T(\psi_\beta)), \beta = T(\psi_\alpha)T(\psi_\beta) \) with \( T(x) = \exp(x)/(1 + \exp(x)) \) for \( x \) real, on which gaussian prior-posterior assumptions apply.

### 6.4 LB Smoothing and Stopping Criterion

The stochastic nature of the lower-bound estimator, equation 5.13, introduces some noise that can violate the expected nondecreasing behavior of the lower bound across the iterations. By setting a window of size \( w \), we compute the moving average \( \hat{\mathcal{L}}_t = 1/w \sum_{i=1}^{w} \hat{\mathcal{L}}_{t-i+1}(\xi^*) \), whose variance is reduced and behavior stabilized.

By keeping track of \( \mathcal{L}^* := \max \hat{\mathcal{L}}_t \), occurring at some iteration \( t^* \), \( 1 \leq t^* \leq t \), we terminate the optimization after \( \mathcal{L}^* \) did not improve for \( P \) iterations (patience parameter) or after a maximum number of iterations \( t_{\text{max}} \) is reached. Therefore, the termination of the algorithm is determined by an exit condition that depends on \( t, t_{\text{max}}, \) the distance \( t - t^* \), and the values of \( \hat{\mathcal{L}}_{t-w+1}, \ldots, \hat{\mathcal{L}}_t, \) and \( \mathcal{L}^* \). Here \( t_{\text{max}}, P, \) and \( w \) are hyperparameters. In algorithms 1 and 2, this exit function is compactly denoted by \( f_{\text{exit}}(t, \ldots) \).

### 6.5 Gradient Clipping

Especially for low values of \( S \), and even more, if a variance control method is not adopted, the stochastic gradient estimate may be poor and the offset from its actual value may be large. This can result in updates whose magnitude is too big in either a positive or negative direction. Especially at early iterations and with poor initial values, this issue
may, for example, cause complex roots in equation 5.15. At each iteration $t$, to control for the magnitude of the stochastic gradient $\hat{g}_t$, we rescale its $\ell_2$-norm $||\hat{g}_t||$ whenever it is larger than a fixed threshold $l_{\text{max}}$ by replacing $\hat{g}_t$ with $\hat{g}_t l_{\text{max}}/||\hat{g}_t||$, which bounds the norm of the rescaled gradient while preserving its direction. Gradient clipping can be applied to either the gradients $\hat{\nabla}_\mu L(\zeta^c)$, $\hat{\nabla}_\Sigma L(\zeta^c)$ or to the natural gradients $\Sigma \hat{\nabla}_\mu L(\zeta^c)$, $-\hat{\nabla}_\Sigma L(\zeta^c)$. We suggest applying gradient clipping on both gradients to promptly mitigate the impact that far-from-the-mean estimates may have on natural gradient computations and to control the norm of the product $\Sigma \hat{\nabla}_\mu L(\zeta^c)$.

6.6 Adaptive Learning Rate. Adopting an adaptive learning rate or scheduler for decreasing the learning rate $\beta$ after a certain number of iterations is convenient. Typical choices include multiplying $\beta$ by a certain factor (e.g., 0.2) every set number of iterations (say, 100) or dynamically updating it after a certain iteration $t'$. We adopt $\beta_t = \min(\beta, \beta_{t'}^{1/f})$, where $t'$ is a fraction (e.g., 0.7) of the maximum number of iterations $t_{\text{max}}$.

7 Experiments

We validate and explore our suggested optimizer’s empirical validity and feasibility over five data sets and 14 models. These include logistic regression (Labor data set), different volatility models on the S&P 500 returns (S&P 500 data set), linear regression on different stock indexes (Istanbul data), a neural network (Limit-Order Book (LOB) data), and a nondifferentiable model (Synthetic data). Details on the data sets and models used in the different experiments are in Table 15. Our experiments deal with both classification and regression tasks. For classification, we discuss binary-class prediction with logistic regression and time-series multiclass classification with a neural network. For regression, we propose a linear model and several GARCH-like models for volatility modeling. Finally, we exploit the advantage of the black-box setting for regression within a nondifferentiable model. Extended results, with additional data sets and models, appear in appendix B. The relevant codes are available at github.com/mmagris/MGVBP. The main baseline for model comparison is the MGVB optimizer and (sequential) MCMC estimates representative of the true posterior. Additionally, we also include results related to the QBVI optimizer (Magris et al., 2022), maximum likelihood (ML), and Adam when applicable.

7.1 Classification Tasks. The logistic regression experiments provide a sanity and qualitative check on the feasibility, robustness, and learning process of MGVBP. From the convex form of the likelihood, we expect to observe a tight alignment of the parameters’ estimates and performance metrics due to the similar optimum the different algorithms attain for the relatively simple form of the LB under this experimental setting. This is indeed the case for the results presented later. In such a setting, we can grasp
and largely comment on the qualitative differences in the learning process and final results obtained with the different optimizers.

The upper panels in Figure 3 depict the LB optimization process across the iterations for the logistic regression. MGVBP’s LB shows a steeper growth rate and improved convergence rate on the training set, irrespective of whether the $h$-function estimator is used. The central and left panels of Figure 3 furthermore highlight an excellent accordance of the learned MGVBP variational posterior with its MCMC counterpart, with means aligned with the ML estimates. This underlines that the gaussian variational assumption is well suited and that the moments of the variational marginal match those derived from the MCMC sampler. In Figure 4, we plot the dynamics of standard performance metrics for classification tasks. On both the training and test sets, at early iterations, MGVBP displays a very steep growth in model accuracy, precision, recall, and f1-score compared to MGVB. Eventually, at later iterations, MGVBP and MGVB performance metrics converge to a similar level. Yet, as depicted by the vertical lines in the left panel of Figure 3, MGVBP leads to LB convergence approximately 200 iterations earlier than MGVB, reflected in the values of the performance
metrics. That is, the training is completed in a much smaller number of iterations.

In fact, this is aligned with Table 1, showing that the LB reaches a similar maximum value corresponding to rather analogous performance metrics. As a consequence, the final variational solution MGVBP attains with respect to the existing optimizers is similar. Table 2 reports the estimated posterior means, Table 3 the variances, and Table 4 the estimated covariances.

Table 2 underlines the unbiasedness of the MGVBP posterior estimates of the variational parameter with respect to the MCMC baseline and the ML target for large samples where the prior is asymptotically negligible.

Similarly, the variational variances of the model parameters in Table 3 are closely aligned between MGVBP, the other models, and the MCMC and the ML counterparts. As well, the covariance matrix also closely replicates the one from the MCMC chain (see Table 4).

The above empirically validates the suitability of the gaussian approximation, the unbiasedness of the MGVBP posterior estimates of the variational parameter, and its overall consistency in terms of the asymptotic ML covariance matrix.

A diagonal constraint on the covariance harms the performance metrics as a consequence of a different value of the optimized LB as for Table 1. This results from the different overall levels of the estimated posterior means and variances under this setup (see Tables 2 and 3) and is expected considering the considerably smaller overall number of free parameters. However, the variational parameters and the corresponding performance metrics are still aligned between the different optimizers. As an insight, we comment that the diagonal assumption practically translates the variational gaussian as the posterior means indeed shift. Indeed, variational methods can be biased with respect to the true posterior distribution (Carbonetto & Stephens, 2012). This is, after all, justified, thinking that the LB and its gradients, see, e.g., equation 4.2, depend on the interaction of variational elements concerning both the variational mean and variational covariance elements. That is, the variational mean under the full case would differ from that under the diagonal case.

Furthermore, Figure 5 shows that the variational mean and covariance learning dynamics are smooth and steady, without wigglings or anomalies. At the same time, using the $h$-function estimator stabilizes the learning process but has a minor effect on the maximized LB (see Tables 1, 2, and 3). Indeed, the adoption of the gradient estimator based on the $h$-function has an overall minimal impact on the optimized results. In fact, the $h$-function alternative affects only the variance of the gradient estimator without affecting the expected direction of the steps in the SGD update.

Table 8 in appendix B addressed the impact the number of MC samples $S$ has on the posterior mean and performance measures, and the optimized LB is minor for both the training and test phases. We conclude that about 20 samples are sufficient (also for complex models such as the following
Table 1: Performance Metrics and Value of the Optimized Lower Bound ($L^*$) for All the Classification Experiments on the Labor Data Set.

|          | Train |          |          |          | Test |          |          |          |          |
|----------|-------|----------|----------|----------|------|----------|----------|----------|----------|
|          | $L^*$ | Accuracy | Precision | Recall  | $f_1$ | Accuracy | Precision | Recall  | $f_1$   |
| MGVBP    | -356.645 | 0.713    | 0.712    | 0.703   | 0.708 | 0.698    | 0.679    | 0.674   | 0.676   |
| MGVB     | -356.646 | 0.711    | 0.710    | 0.701   | 0.706 | 0.698    | 0.679    | 0.674   | 0.676   |
| QBVI     | -356.645 | 0.713    | 0.712    | 0.703   | 0.708 | 0.698    | 0.679    | 0.674   | 0.676   |
| MGVBP$_{h}$-func. | -356.635 | 0.711    | 0.710    | 0.701   | 0.706 | 0.698    | 0.679    | 0.674   | 0.676   |
| MGVB$_{h}$-func. | -356.635 | 0.711    | 0.710    | 0.701   | 0.706 | 0.698    | 0.679    | 0.674   | 0.676   |
| QBVI$_{h}$-func. | -356.635 | 0.711    | 0.710    | 0.701   | 0.706 | 0.698    | 0.679    | 0.674   | 0.676   |
| MGVBP$_{diag}$. | -358.605 | 0.713    | 0.712    | 0.703   | 0.708 | 0.672    | 0.650    | 0.647   | 0.649   |
| MGVB$_{diag}$. | -358.681 | 0.709    | 0.708    | 0.701   | 0.704 | 0.667    | 0.645    | 0.643   | 0.644   |
| QBVI$_{diag}$. | -358.594 | 0.713    | 0.712    | 0.703   | 0.708 | 0.672    | 0.650    | 0.647   | 0.649   |
| MGVBP$_{h}$-func. diag | -358.601 | 0.713    | 0.712    | 0.703   | 0.708 | 0.667    | 0.644    | 0.640   | 0.642   |
| MGVB$_{h}$-func. diag | -358.680 | 0.711    | 0.710    | 0.702   | 0.706 | 0.667    | 0.644    | 0.640   | 0.642   |
| QBVI$_{h}$-func. diag | -358.590 | 0.713    | 0.712    | 0.703   | 0.708 | 0.667    | 0.644    | 0.640   | 0.642   |

Note: $h$-func. denotes the use of the $h$-function gradient estimator, and $diag$: the use of a diagonal variational posterior.
Table 2: Posterior Variational Means for All the Classification Experiments on the Labor Data Set.

| Model | \( \beta_0 \) | \( \beta_1 \) | \( \beta_2 \) | \( \beta_3 \) | \( \beta_4 \) | \( \beta_5 \) | \( \beta_6 \) | \( \beta_7 \) |
|-------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|
| MGVBP | 0.679          | -1.490         | -0.083         | -0.574         | 0.494          | -0.640         | 0.607          | 0.048          |
| MGVB  | 0.679          | -1.489         | -0.083         | -0.574         | 0.493          | -0.639         | 0.607          | 0.048          |
| QBVI  | 0.679          | -1.489         | -0.083         | -0.574         | 0.493          | -0.639         | 0.607          | 0.048          |
| MGVBP\( h \)-func. | 0.678 | -1.487 | -0.084 | -0.574 | 0.492 | -0.638 | 0.609 | 0.048 |
| MGVB\( h \)-func. | 0.678 | -1.487 | -0.084 | -0.574 | 0.492 | -0.638 | 0.609 | 0.048 |
| QBVI\( h \)-func. | 0.678 | -1.487 | -0.084 | -0.574 | 0.492 | -0.638 | 0.609 | 0.048 |
| MGVBP\( \text{diag} \) | 0.544 | -1.414 | -0.045 | -0.529 | 0.494 | -0.629 | 0.583 | 0.119 |
| MGVB\( \text{diag} \) | 0.542 | -1.413 | -0.044 | -0.529 | 0.493 | -0.629 | 0.583 | 0.121 |
| QBVI\( \text{diag} \) | 0.546 | -1.415 | -0.045 | -0.530 | 0.494 | -0.629 | 0.583 | 0.119 |
| MGVBP\( \text{h-\text{func. diag}} \) | 0.544 | -1.413 | -0.047 | -0.531 | 0.492 | -0.628 | 0.585 | 0.121 |
| MGVB\( \text{h-\text{func. diag}} \) | 0.541 | -1.411 | -0.046 | -0.530 | 0.492 | -0.629 | 0.586 | 0.123 |
| QBVI\( \text{h-\text{func. diag}} \) | 0.545 | -1.414 | -0.047 | -0.531 | 0.492 | -0.628 | 0.585 | 0.121 |
| MCMC  | 0.679          | -1.487         | -0.084         | -0.574         | 0.493          | -0.638         | 0.607          | 0.047          |
| ML    | 0.679          | -1.476         | -0.085         | -0.571         | 0.485          | -0.625         | 0.599          | 0.045          |

Table 3: Variances of the Variational Approximation on the Parameters for the Labor Data Set.

| Model       | \( \beta_0 \) | \( \beta_1 \) | \( \beta_2 \) | \( \beta_3 \) | \( \beta_4 \) | \( \beta_5 \) | \( \beta_6 \) | \( \beta_7 \) |
|-------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|
| MGVBP       | 3.956          | 5.418          | 0.604          | 1.531          | 1.376          | 2.105          | 1.964          | 4.332          |
| MGVB        | 4.288          | 5.644          | 0.624          | 1.496          | 1.354          | 2.191          | 2.024          | 4.314          |
| QBVI        | 3.983          | 5.446          | 0.608          | 1.540          | 1.384          | 2.118          | 1.977          | 4.361          |
| MGVBP\( h \)-func. | 4.071 | 5.377 | 0.596 | 1.463 | 1.284 | 2.057 | 1.972 | 4.389 |
| MGVB\( h \)-func. | 4.043 | 5.384 | 0.594 | 1.457 | 1.284 | 2.070 | 1.963 | 4.372 |
| QBVI\( h \)-func. | 4.072 | 5.378 | 0.596 | 1.463 | 1.284 | 2.057 | 1.972 | 4.389 |
| MGVBP\( \text{diag} \) | 0.815 | 3.304 | 0.255 | 0.930 | 1.086 | 1.066 | 0.915 | 1.367 |
| MGVB\( \text{diag} \) | 0.808 | 3.299 | 0.255 | 0.882 | 1.107 | 1.059 | 0.913 | 1.321 |
| QBVI\( \text{diag} \) | 0.817 | 3.311 | 0.255 | 0.932 | 1.088 | 1.068 | 0.917 | 1.371 |
| MGVBP\( \text{h-\text{func. diag}} \) | 0.852 | 3.222 | 0.247 | 0.900 | 1.005 | 1.018 | 0.935 | 1.366 |
| MGVB\( \text{h-\text{func. diag}} \) | 0.853 | 3.228 | 0.248 | 0.878 | 1.013 | 1.012 | 0.940 | 1.348 |
| QBVI\( \text{h-\text{func. diag}} \) | 0.853 | 3.227 | 0.248 | 0.901 | 1.006 | 1.019 | 0.936 | 1.368 |
| MCMC        | 3.967          | 5.364          | 0.589          | 1.406          | 1.274          | 2.071          | 1.966          | 4.248          |
| ML          | 4.079          | 5.436          | 0.589          | 1.457          | 1.276          | 2.059          | 1.960          | 4.393          |

Note: Entries are multiplied by \(10^2\).

neural network) and are a feasible compromise between the quality of the MC approximation of the LB gradient and computational efficiency (also in the following experiments). Our experiments are, however, oriented toward discussing the improvements and advantages of MGVBP over MGVB. To this end, we require a precise estimation of the LB and thus employ a much higher value of \(S\) (see Table 15).
Table 4: Variational Covariance Matrices, MCMC Covariance Matrix, and ML Asymptotic Covariance Matrix.

|        | MGVBP |        |        |        |        |        |        |        |        |        |        |        |
|--------|-------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
|        | $\beta_0$ | $\beta_1$ | $\beta_2$ | $\beta_3$ | $\beta_4$ | $\beta_5$ | $\beta_6$ | $\beta_7$ |
| MGVBP  | $\beta_0$ | -1.710 | -0.877 | -0.674 | 0.163 | 0.478 | 0.031 | -2.716 |
|        | $\beta_1$ | -1.458 | 0.282 | 1.465 | -0.561 | -0.192 | 0.229 | -0.002 |
|        | $\beta_2$ | -0.790 | 0.218 | 0.385 | 0.089 | 0.035 | -0.024 | -0.090 |
|        | $\beta_3$ | -0.712 | 1.406 | 0.406 | -0.014 | 0.123 | -0.063 | -0.414 |
|        | $\beta_4$ | 0.135 | -0.395 | 0.100 | 0.046 | -0.104 | -0.280 | -0.131 |
|        | $\beta_5$ | 0.228 | -0.136 | 0.031 | 0.082 | -0.111 | -1.366 | -0.743 |
|        | $\beta_6$ | 0.235 | 0.177 | -0.057 | -0.176 | -0.313 | -1.305 | -0.039 |
|        | $\beta_7$ | -2.584 | -0.090 | -0.124 | -0.389 | -0.199 | -0.561 | -0.122 |

|        |        |        |        |        |        |        |        |        |        |        |        |        |
|--------|-------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
|        | $\beta_0$ | $\beta_1$ | $\beta_2$ | $\beta_3$ | $\beta_4$ | $\beta_5$ | $\beta_6$ | $\beta_7$ |
| ML     | $\beta_0$ | -1.578 | -0.800 | -0.684 | 0.067 | 0.318 | 0.194 | -2.701 |
|        | $\beta_1$ | -1.510 | 0.274 | 1.378 | -0.422 | -0.027 | 0.074 | -0.084 |
|        | $\beta_2$ | -0.791 | 0.249 | 0.393 | 0.102 | 0.024 | -0.058 | -0.093 |
|        | $\beta_3$ | -0.654 | 1.307 | 0.381 | 0.067 | 0.117 | -0.164 | -0.352 |
|        | $\beta_4$ | 0.047 | -0.414 | 0.111 | 0.072 | -0.172 | -0.265 | -0.109 |
|        | $\beta_5$ | 0.334 | -0.049 | 0.006 | 0.089 | -0.177 | -1.286 | -0.613 |
|        | $\beta_6$ | 0.172 | 0.111 | -0.036 | -0.137 | -0.266 | -1.282 | -0.097 |
|        | $\beta_7$ | -2.603 | -0.085 | -0.097 | -0.349 | -0.119 | -0.590 | -0.102 |

Note: Entries are multiplied by $10^2$.

Figure 5: Parameter learning across the iterations for the MGVBP algorithm on the Labour data set for some selected variational parameters. Dotted lines correspond to the diagonal case, dashed lines to the use of the $h$-function gradient estimator.

We additionally train a (bilinear) neural network with a TABL layer (Tran et al., 2019). The bottom panel of Table 5 reports the estimation results on the TABL neural network, referred to as the LOB experiment. The
Table 5: Performance Measures for the LOB Experiment.

| Method          | Train $\mathcal{L}^*$ | Train Accuracy | Train Precision | Train Recall | Train f1   | Test Accuracy | Test Precision | Test Recall | Test f1   |
|-----------------|------------------------|----------------|-----------------|--------------|------------|---------------|----------------|-------------|------------|
| MGVBP           | -91948.593             | 0.738          | 0.804           | 0.610        | 0.651      | 0.795         | 0.798          | 0.614       | 0.665      |
| QBVI(diag.)     | -104261.698            | 0.717          | 0.697           | 0.637        | 0.658      | 0.686         | 0.650          | 0.605       | 0.603      |
| VOGN(diag.)     | -100791.406            | 0.673          | 0.631           | 0.629        | 0.630      | 0.758         | 0.677          | 0.637       | 0.652      |
| MCMC            |                        | 0.689          | 0.773           | 0.536        | 0.567      | 0.761         | 0.779          | 0.536       | 0.585      |
| ADAM(non Bayesian) |                      | 0.755          | 0.668           | 0.640        | 0.652      |               |                |             |            |
prediction of high-frequency, midprice movements in limit-order book markets is known to be a challenging task, where Bayesian techniques are particularly relevant for implementing risk-aware trading strategies (Magris et al., 2023) and where the TABL architecture is very effective. Training MGVB on such a machine learning model empirically appears unstable and generally unfeasible with our own and the original implementation of Tran, Nguyen, and Nguyen (2021). Indeed we are unaware of any MGVB application for typical machine learning models. We adopt different baselines. Because of the inability of QBVI to grant the positive definite constraint, we train it under a diagonal assumption. Furthermore, we include the state-of-the-art VOGN optimizer (Osawa et al., 2019). VOGN also assumes a diagonal gaussian posterior and relies on natural gradient computations employing models’ (per sample) gradients and an approximate Hessian. Its second-order nature closely resembles the standard Adam optimizer, included as a non-Bayesian baseline.

The problem is highly nonconvex: indeed, we observe the algorithms converging at different optima and heterogeneous performance measures. Yet MGVB clearly outperforms the other Bayesian alternatives on both the training and test set, promoting its use beyond standard statistical models. Concerning the MCMC estimates, we observe ubiquitous evidence of multimodality, skewness, and asymmetry in the margins with an underlying nongaussian copula that any fixed-form gaussian VI cannot, by construction, deal with. However, the peak performance obtained with the gaussian assumption under MGVB suggests a main mode where most of the density is concentrated, captured by the VI approximation. The evolution of the learning process for the LB and performance metrics in Figure 6 shows the very stable behavior of the stochastic LB estimate ($N_s = 20$ only), along with a consistent and resolved improvement of the performance measures since early epochs. The performance under MGVB is also remarkably improved on the test set.
7.2 Regression Tasks. The GARCH-related models in the lower panels of Table 6 address the predictive ability of MGVBP under nonstandard and nonconvex likelihood functions. As performance measures, we adopt the mean squared error (MSE) and the QLIKE loss (Patton, 2011) computed with respect to the five-minute subsampled realized volatility (RV5SS) (Zhang et al., 2005) as a robust proxy for daily compared to squared daily returns. Additional information and extended results appear in appendix B.2, also involving performance measures computed with respect to the squared daily returns.

As for the logistic regression experiments, we observe that all the algorithms approach the very same ML optimum, indicating that both MGVBP and MGVB move toward the same LB maximum. Furthermore, all variational approximations are well aligned with the MCMC and ML estimates. Thus, the estimates, performance metrics, and value of the optimized LB are similar across the optimizers: they all converge to the minimum but in a qualitatively different way. Indeed, in Figure 7, the LB improvement across the iterations is steeper for MGVBP and also for a diagonal implementation. With respect to the dynamics of the performance metrics computed on the test depicted on the second and third columns of the plots in Figure 7, we observed that MGVBP dominates in terms of both MSE and QLIKE, the MGVB baseline for the HAR model. For the GARCH family models, whereas MGVBP clearly outperforms MGVB in terms of the QLIKE loss on both the training and test sets, this dominance is not so evident for the MSE. In fact, there is no direct link between the dynamics of the LB and that of the performance measures, that is, there are no trivial guarantees that, for example, as the LB is maximized, the MSE is minimized. In fact, while this is the case for the HAR model, for the GARCH modes, we observe an increase of the MSE across the iterations despite the fact that the LB approaches the optimum. The same holds for the test MSE: for the GARCH(1,1) it decreases across the iterations, but not for the GJR(1,1). This should not be surprising as the optimization objective is the overall minimization of the KL divergence between the true posterior and the variational one, achieved via LB maximization, and not the minimization of the MSE, Q-link, or other performance metrics. Figure 7 validates this interpretation and reminds us that VI has to be interpreted for what it is: a solution for approximating the posterior, not a performance-metric minimizing tool, as, for example, ordinary least squares (Nguyen, & Nguyen (2021) the squared loss) or ML (with regard to the model negative log likelihood). Consequently, the reference measure for an actual comparison of the VI algorithms should be the value of the optimized lower bound and its dynamics. The additional results in appendix B.2 confirm the improved ability of MGVBP in optimizing the LB and the overall alignment between the solution obtained with MGVB, MCMC, and ML on further GARCH-type models. See Tables 9 to 11 and Figures 8 to 10, comparing the posterior marginals obtained via VI and MCMC.
|         | $\omega$ | $\alpha$ | $\gamma$ | $\beta$ | $\mathcal{L}^*$ | $p(y|\mu^*)$ | $\text{MSE}_{rv}$ | $\text{QLIKE}_{rv}$ | $\text{MSE}_{rv}$ | $\text{QLIKE}_{rv}$ |
|---------|----------|----------|----------|---------|----------------|----------------|-----------------|-----------------|----------------|----------------|
| **GARCH(1,1)** |          |          |          |         |                |                |                 |                 |                 |                 |
| MGVBP   | 0.043    | 0.230    | 0.737    | $-2027.489$ | $-2017.633$ | 5.124          | 0.531           | 2.564           | 0.566           |                  |
| MGVB    | 0.043    | 0.229    | 0.737    | $-2027.493$ | $-2017.626$ | 5.113          | 0.531           | 2.561           | 0.566           |                  |
| QBVI    | 0.043    | 0.229    | 0.737    | $-2027.493$ | $-2017.626$ | 5.114          | 0.531           | 2.561           | 0.566           |                  |
| MCMC    | 0.042    | 0.231    | 0.738    | $-2017.713$ | 5.189       | 0.531           |                 | 2.565           | 0.566           |                  |
| ML      | 0.042    | 0.226    | 0.739    | $-2017.593$ | 5.073       | 0.531           |                 | 2.531           | 0.566           |                  |
| **GJR(1,1)** |          |          |          |         |                |                |                 |                 |                 |                 |
| MGVBP   | 0.043    | 0.108    | 0.294    | 0.722   | $-2003.542$ | $-1990.907$ | 7.039          | 0.310           | 1.556           | 0.346           |
| MGVB    | 0.044    | 0.108    | 0.292    | 0.722   | $-2003.550$ | $-1990.896$ | 6.969          | 0.310           | 1.551           | 0.347           |
| QBVI    | 0.044    | 0.108    | 0.292    | 0.722   | $-2003.550$ | $-1990.897$ | 6.971          | 0.310           | 1.551           | 0.347           |
| MCMC    | 0.043    | 0.108    | 0.298    | 0.724   | $-1991.095$ | 7.318          | 0.309           |                 | 1.576           | 0.344           |
| ML      | 0.042    | 0.108    | 0.291    | 0.723   | $-1990.853$ | 6.940          | 0.311           |                 | 1.548           | 0.349           |
| **HAR** |          |          |          |         |                |                |                 |                 |                 |                 |
| MGVBP   | 1.032    | 0.489    | 0.420    | $-0.009$ | $-5080.191$ | $-5058.113$ | 24.193         | 5.324           | 18.861          | 5.712           |
| MGVB    | 1.019    | 0.485    | 0.426    | $-0.011$ | $-5082.123$ | $-5058.167$ | 24.194         | 5.328           | 18.879          | 5.720           |
| QBVI$^{\text{diag.}}$ | 1.036    | 0.485    | 0.423    | $-0.007$ | $-5083.253$ | $-5058.143$ | 24.193         | 5.326           | 18.862          | 5.710           |
| MCMC    | 1.075    | 0.488    | 0.421    | $-0.012$ | $-5058.090$ | 24.192        | 5.328           | 18.858          | 5.706           |
| ML      | 1.080    | 0.488    | 0.421    | $-0.012$ | $-5058.086$ | 24.192        | 5.328           | 18.857          | 5.706           |
Figure 7: Progression of the lower-bound optimization and dynamics of the MSE and QLIKE, both computed with respect to RV5, across the iterations for the HAR model (top row), GARCH(1,1) model (middle row), and GJR(1,1) (bottom row). Vertical lines denote the iteration $t^*$, for which the line style and color match the one of the corresponding optimizer. For the GARCH and GJR, see appendix A for the corresponding plots of the parameters’ posterior margins.

Table 7: Results for the Nondifferentiable Model.

|          | $L^*$   | $p(y|\mu^*)$ | MSE  | MAD  | $p(y|\mu^*)$ | MSE  | MAD  |
|----------|---------|--------------|------|------|--------------|------|------|
| MGVB     | -2901.67| -2848.05     | 1.02 | 0.80 | -707.451     | 1.03 | 0.79 |
| MGVBP    | -2898.86| -2845.41     | 1.02 | 0.80 | -705.945     | 1.02 | 0.79 |
| QBVI diag.| -2904.48| -2846.67     | 1.02 | 0.80 | -706.78      | 1.03 | 0.79 |

7.3 Nondifferentiable Model. Table 7 reports the estimation results for a nondifferentiable model, where, for example, the reparameterization trick and methods generally relying on gradients and automatic differentiation are inapplicable. We adopt the reconstruction benchmark problem in Lyu
1774 M. Magris, M. Shabani, and A. Iosifidis

and Tsang (2021):

\[ y_i = f(x_i) + \varepsilon_i, \quad \varepsilon_i \sim \mathcal{N}(0, \sigma^2), \]
\[ f(x_i) = ||\text{sign}(x_i - 0.5) - \beta||_2^2 - ||\text{sign}(\beta) - \beta||_2^2, \quad x_i \in \{0, 1\}^d. \]

We simulate \( i = 1, \ldots, 5000 \) observations for the above model with \( d = 20, \beta \sim \text{Uniform}(0, 1), \ p = 0.5, \ \sigma^2 = 1, \) and \( x \) sampled from \( d \) independent Bernoulli distributions. MGVBP and MGVB are both implemented over the \( h \)-function estimator and for a full specification of the variational covariance matrix. For reference, the initial value of the lower bound for all the experiments is \(-11,591.339\), and we include the mean absolute deviation (MAD).

Details on the hyperparameters are provided in Table 15.

From Table 7, we confirm the overall accordance of the three black-box approaches considered in this work and the improved performance of MGVBP also for the above nondifferentiable model.

8 Conclusion

We propose an algorithm based on manifold optimization to guarantee the positive-definite constraint on the covariance matrix of the gaussian variational posterior. Extending the baseline method of Tran, Nguyen, and Nguyen (2021), we exploit the computational advantage of the gaussian parameterization in terms of the precision matrix and employ simple and computationally convenient analytical solutions for the natural gradients. Furthermore, we provide a theoretically consistent setup justifying the use of the SPD-manifold retraction form. Our MGVBP black-box optimizer results in a ready-to-use solution for VI, scalable to structured covariance matrices, that can take advantage of alternative forms of the stochastic gradient estimator, control variates, and momentum. We show our solution’s feasibility on many statistical, econometric, and ML models over different baseline optimizers. Our results align with sampling methods and highlight the advantages of the suggested approach over state-of-the-art baselines in terms of convergence and performance metrics. Future research may investigate the applicability of our approach to a broader set of variational distributions and the use of the reparameterization trick in place of black-box gradients.
Appendix A: Block-Diagonal Implementation

Algorithm 2: MGVB for a Block-Diagonal Covariance Matrix (Prior with-
Zero Mean and Covariance Matrix $\tau^{-1}I$).

1: Set hyper-parameters: $0 < \beta, \omega < 1$, $S$
2: Set the type of gradient estimator, i.e., function $f(\theta_s)$
3: Set prior $p(\theta; 0, \tau I)$, likelihood $p(y|\theta)$, and initial values $\mu, \Sigma^{-1}$
4: $t = 1$, Stop = false
5: Generate: $\theta_{s \leq S} \sim q_{\mu, \Sigma}$, $s = 1, \ldots, S$, $i = 1, \ldots, h$
6: for $i = 1, \ldots, h$ do
7: Compute: $\hat{g}_{\mu_i} = \Sigma_i \nabla_{\mu_i} \mathcal{L}$, $\hat{g}_{\Sigma_i} = -\nabla_{\Sigma_i} \mathcal{L}$
8: $m_{\mu_i} = \hat{g}_{\mu_i}$, $m_{\Sigma_i} = \hat{g}_{\Sigma_i}$
9: end for
10: while Stop = false do
11: for $i = 1, \ldots, h$ do
12: $\mu_i = \mu_i + \beta m_{\mu_i}$
13: $\Sigma_{i, \text{old}} = \Sigma_i^{-1}$, $\Sigma_i^{-1} = R_{\Sigma_{i, \text{old}}} (\beta m_{\Sigma_i}^{-1})$
14: end for
15: Generate: $\theta_{s \leq S} \sim q_{\mu, \Sigma}$, $s = 1, \ldots, S$, $i = 1, \ldots, h$
16: Set: $\log q_s = 0$, $s = 1, \ldots, S$
17: for $i = 1, \ldots, h$ do
18: Compute: $\hat{g}_{\mu_i}$, $\hat{g}_{\Sigma_i}$
19: $m_{\mu_i} = \omega m_{\mu_i} + (1 - \omega) \hat{g}_{\mu_i}$
20: $m_{\Sigma_i} = T_{\Sigma_{i, \text{old}}} \Sigma_i^{-1} (m_{\Sigma_i}^{-1}) + (1 - \omega) \hat{g}_{\Sigma_i}$
21: $\log q_s = \log q_s + \log q_{\mu, \Sigma} (\theta_{s \leq S})$
22: end for
23: $\hat{\mathcal{L}} = \frac{1}{3} \sum_{s=1}^{S} \log p(\theta_s) + \log p(y|\theta_s) + \log q_s$
24: $t = t + 1$, Stop = $f_{\text{exit}} (t, \ldots)$
25: end while

Appendix B: Experiments

B.1 Additional Results for the Labour Data. Table 8 presents estimated parameters and performance Measures on the Labor data set for MGVB (full posterior) for different sizes of the number of MC draws for the estimation of the stochastic gradients $S$.

B.2 Volatility Models. Our second set of experiments involves the estimation of several GARCH-family volatility models. The models in Tables 9, 10, and 11 differ for the number of estimated parameters, the form of the likelihood function (which can be quite complex as for the FIGARCH models), and constraints imposed on the parameters. Besides the GARCH-type models, we include the well-known linear HAR model for realized volatility (Corsi, 2009). We performed a preliminary study for retaining only relevant models, for example, we observed that for a GARCH(1,0,2) $\beta_2$ is not
Table 8: \( t \) Refers to the Run-Time per Iteration (in Milliseconds), \( \mathcal{L}(\theta_0) \) to the LB Evaluated at the Initial Parameters.

| \( S \) | \( t \) | \( \beta_0 \) | \( \beta_1 \) | \( \beta_2 \) | \( \beta_3 \) | \( \beta_4 \) | \( \beta_5 \) | \( \beta_6 \) | \( \beta_7 \) | \( \mathcal{L}(\zeta_0) \) |
|-------|------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| 10    | 0.002 | 0.674  | -1.478 | -0.080 | -0.569 | 0.500  | -0.634 | 0.593  | 0.041  | -430.216|
| 20    | 0.002 | 0.680  | -1.490 | -0.083 | -0.573 | 0.491  | -0.637 | 0.602  | 0.044  | -429.626|
| 30    | 0.003 | 0.680  | -1.491 | -0.083 | -0.573 | 0.492  | -0.635 | 0.602  | 0.045  | -434.863|
| 50    | 0.004 | 0.679  | -1.489 | -0.083 | -0.573 | 0.495  | -0.639 | 0.607  | 0.045  | -435.890|
| 75    | 0.006 | 0.679  | -1.489 | -0.083 | -0.574 | 0.493  | -0.639 | 0.607  | 0.048  | -436.282|
| 100   | 0.008 | 0.679  | -1.490 | -0.084 | -0.575 | 0.494  | -0.639 | 0.608  | 0.049  | -436.924|
| 150   | 0.011 | 0.679  | -1.488 | -0.085 | -0.575 | 0.493  | -0.639 | 0.609  | 0.048  | -436.759|
| 200   | 0.013 | 0.680  | -1.489 | -0.085 | -0.575 | 0.493  | -0.638 | 0.609  | 0.047  | -437.036|
| 300   | 0.020 | 0.679  | -1.487 | -0.084 | -0.575 | 0.493  | -0.638 | 0.609  | 0.047  | -436.529|

Note: For each \( S \), a common random seed is used.
Table 9: Estimation Results and Performance Metrics for Some GARCH Variants.

| Model Type      | MGVBP | MGVB | QBVI | MCMC | ML   |
|-----------------|-------|------|------|------|------|
| \( \omega \)    | 0.519 | 0.520| 0.520| 0.519| 0.520|
| \( \alpha \)    | 0.640 | 0.639| 0.639| 0.640| 0.628|
| \( \gamma \)    |       |      |      |      |      |
| \( \beta_1 \)   |       |      |      |      |      |
| \( \beta_2 \)   |       |      |      |      |      |
| \( \mathcal{L}^* \) | -2262.243 | -2262.244 | -2262.244 | -2262.136 | -2256.136 |
| \( p(y|\mu^*) \) | 29.659 | 29.647| 29.647| 29.660| 29.538|
| MSE (Train)     | 8.164 | 8.146| 8.146| 8.165| 7.993|
| MSE (Test)      | 1.918 | 1.918| 1.918| 1.918| 1.918|
| QLIKE (Train)   | 0.531 | 0.531| 0.531| 0.531| 0.531|
| QLIKE (Test)    | 0.566 | 0.566| 0.566| 0.566| 0.566|
| GARCH(1,0,1)    |       |      |      |      |      |
| \( \omega \)    | 0.043 | 0.043| 0.043| 0.042| 0.042|
| \( \alpha \)    | 0.230 | 0.229| 0.229| 0.231| 0.226|
| \( \gamma \)    | 0.737 | 0.737| 0.737| 0.738| 0.739|
| \( \beta_1 \)   |       |      |      |      |      |
| \( \beta_2 \)   |       |      |      |      |      |
| \( \mathcal{L}^* \) | -2027.489 | -2027.493 | -2027.493 | -2017.633 | -2017.593 |
| \( p(y|\mu^*) \) | 25.560 | 25.557| 25.557| 25.386| 25.939|
| MSE (Train)     | 5.124 | 5.113| 5.114| 5.189| 5.073|
| MSE (Test)      | 1.637 | 1.637| 1.637| 1.637| 1.637|
| QLIKE (Train)   | 0.330 | 0.330| 0.330| 0.331| 0.331|
| QLIKE (Test)    | 0.382 | 0.382| 0.382| 0.382| 0.382|
| GJR(1,1,1)      |       |      |      |      |      |
| \( \omega \)    | 0.043 | 0.044| 0.044| 0.042| 0.042|
| \( \alpha \)    | 0.108 | 0.108| 0.108| 0.108| 0.108|
| \( \gamma \)    | 0.294 | 0.292| 0.292| 0.298| 0.291|
| \( \beta_1 \)   |       |      |      |      |      |
| \( \beta_2 \)   |       |      |      |      |      |
| \( \mathcal{L}^* \) | -2003.542 | -2003.550 | -2003.550 | -1990.907 | -1990.853 |
| \( p(y|\mu^*) \) | 27.125 | 27.071| 27.073| 27.339| 27.048|
| MSE (Train)     | 7.039 | 6.969| 6.971| 7.188| 6.940|
| MSE (Test)      | 1.606 | 1.606| 1.606| 1.606| 1.606|
| QLIKE (Train)   | 0.310 | 0.310| 0.310| 0.309| 0.309|
| QLIKE (Test)    | 0.346 | 0.347| 0.347| 0.344| 0.344|
| GJR(1,1,2)      |       |      |      |      |      |
| \( \omega \)    | 0.045 | 0.045| 0.045| 0.042| 0.044|
| \( \alpha \)    | 0.116 | 0.114| 0.114| 0.108| 0.111|
| \( \gamma \)    | 0.323 | 0.319| 0.319| 0.298| 0.307|
| \( \beta_1 \)   |       |      |      |      |      |
| \( \beta_2 \)   |       |      |      |      |      |
| \( \mathcal{L}^* \) | -2003.193 | -2003.247 | -2003.247 | -1990.907 | -1990.853 |
| \( p(y|\mu^*) \) | 27.657 | 27.499| 27.301| 27.339| 27.048|
| MSE (Train)     | 7.517 | 7.291| 7.294| 7.188| 6.940|
| MSE (Test)      | 1.606 | 1.606| 1.606| 1.606| 1.606|
| QLIKE (Train)   | 0.310 | 0.310| 0.310| 0.309| 0.309|
| QLIKE (Test)    | 0.348 | 0.347| 0.347| 0.344| 0.344|
Table 10: Estimation Results and Performance Metrics for EGARCH and FIGARCH Models.

| Model       | ω    | α    | γ    | β₁   | L*  | p(y|µ*) | MSEᵣ | MSEᵥ | QLIKEᵣ | QLIKEᵥ | p(y|µ*) | MSEᵣ | MSEᵥ | QLIKEᵣ | QLIKEᵥ |
|-------------|------|------|------|------|-----|--------|-------|-------|---------|---------|--------|-------|-------|---------|---------|
| **EGARCH(1,0,1)** |      |      |      |      |     |        |       |       |         |         |        |       |       |         |         |
| MGVBP       | -0.003 | 0.413 | 0.929 | -2048.242 | -2032.756 | 26.713 | 5.156 | 1.655 | 0.337 | -610.894 | 4.740 | 1.667 | 1.428 | 0.381 |
| MGVB        | -0.003 | 0.413 | 0.929 | -2048.242 | -2032.757 | 26.713 | 5.157 | 1.655 | 0.337 | -610.895 | 4.740 | 1.667 | 1.428 | 0.381 |
| QBVI        | -0.003 | 0.416 | 0.929 | -2048.247 | -2032.779 | 26.709 | 5.154 | 1.655 | 0.337 | -610.911 | 4.742 | 1.666 | 1.428 | 0.381 |
| MCMC        | -0.003 | 0.413 | 0.929 | -2032.752 | 26.711 | 5.155 | 1.655 | 0.337 | -610.878 | 4.739 | 1.667 | 1.428 | 0.381 |
| ML          | -0.003 | 0.405 | 0.932 | -2032.723 | 26.732 | 5.169 | 1.655 | 0.337 | -610.840 | 4.730 | 1.672 | 1.427 | 0.381 |
| **EGARCH(1,1,1)** |      |      |      |      |     |        |       |       |         |         |        |       |       |         |         |
| MGVBP       | -0.014 | 0.350 | -0.172 | 0.930 | -2010.242 | -1989.666 | 26.039 | 4.413 | 1.604 | 0.303 | -600.116 | 4.972 | 1.356 | 1.377 | 0.327 |
| MGVB        | -0.014 | 0.350 | -0.172 | 0.930 | -2010.242 | -1989.667 | 26.040 | 4.413 | 1.604 | 0.303 | -600.119 | 4.973 | 1.356 | 1.377 | 0.327 |
| QBVI        | -0.015 | 0.364 | -0.175 | 0.926 | -2010.403 | -1989.914 | 26.081 | 4.454 | 1.604 | 0.304 | -600.654 | 5.022 | 1.363 | 1.379 | 0.328 |
| MCMC        | -0.015 | 0.349 | -0.171 | 0.930 | -1989.665 | 26.051 | 4.419 | 1.604 | 0.303 | -600.112 | 4.970 | 1.356 | 1.377 | 0.327 |
| ML          | -0.014 | 0.340 | -0.170 | 0.932 | -1989.610 | 26.011 | 4.386 | 1.604 | 0.303 | -599.682 | 4.935 | 1.352 | 1.375 | 0.327 |
| **FIGARCH(0,d,1)** |      |      |      |      |     |        |       |       |         |         |        |       |       |         |         |
| MGVBP       | 0.099 | 0.644 | 0.415 | -2022.562 | -2015.130 | 25.533 | 5.540 | 1.634 | 0.326 | -609.607 | 4.789 | 1.574 | 1.422 | 0.382 |
| MGVB        | 0.099 | 0.643 | 0.414 | -2022.562 | -2015.130 | 25.532 | 5.538 | 1.634 | 0.326 | -609.604 | 4.788 | 1.574 | 1.422 | 0.382 |
| QBVI        | 0.099 | 0.643 | 0.414 | -2022.562 | -2015.130 | 25.532 | 5.539 | 1.634 | 0.326 | -609.607 | 4.789 | 1.574 | 1.422 | 0.382 |
| MCMC        | 0.099 | 0.655 | 0.422 | -2015.186 | 25.537 | 5.566 | 1.634 | 0.327 | -609.886 | 4.800 | 1.574 | 1.423 | 0.383 |
| ML          | 0.102 | 0.653 | 0.428 | -2015.094 | 25.582 | 5.570 | 1.634 | 0.326 | -609.345 | 4.780 | 1.574 | 1.420 | 0.381 |
| **FIGARCH(1,d,1)** |      |      |      |      |     |        |       |       |         |         |        |       |       |         |         |
| MGVBP       | 0.100 | 0.060 | 0.662 | 0.481 | -2022.666 | -2014.747 | 25.639 | 5.480 | 1.634 | 0.326 | -609.772 | 4.802 | 1.570 | 1.422 | 0.380 |
| MGVB        | 0.100 | 0.060 | 0.661 | 0.480 | -2022.667 | -2014.746 | 25.637 | 5.478 | 1.634 | 0.326 | -609.761 | 4.802 | 1.570 | 1.422 | 0.380 |
| QBVI        | 0.100 | 0.060 | 0.661 | 0.480 | -2022.667 | -2014.746 | 25.638 | 5.478 | 1.634 | 0.326 | -609.762 | 4.802 | 1.570 | 1.422 | 0.380 |
| MCMC        | 0.100 | 0.059 | 0.668 | 0.482 | -2014.805 | 25.628 | 5.495 | 1.634 | 0.326 | -610.007 | 4.813 | 1.570 | 1.423 | 0.381 |
| ML          | 0.100 | 0.064 | 0.653 | 0.479 | -2014.721 | 25.649 | 5.458 | 1.634 | 0.326 | -609.489 | 4.791 | 1.571 | 1.421 | 0.379 |
Table 11: Additional Results for the HAR Model Involving the $h$-Function Gradient Estimation and the Diagonal Posterior Form.

| Method       | $\beta_0$ | $\beta_1$ | $\beta_2$ | $\beta_3$ | $\sigma_\epsilon$ | $\mathcal{L}^*$ | $p(y|\mu^*)$ | MSE<sub>rV</sub> | $10^2 \times $QLIKE<sub>rV</sub> | Train | Test       | p(y|μ*) | MSE<sub>rV</sub> | $10^2 \times $QLIKE<sub>rV</sub> |
|--------------|-----------|-----------|-----------|-----------|-------------------|-----------------|--------------|--------------|-------------------|-------|------------|--------|--------------|-------------------|
| MGVBP        | 1.032     | 0.489     | 0.420     | −0.009    | 4.922             | −5080.191       | −5058.113    | 24.193       | 5.324             | 1.032 | 0.489      | 0.420  | −0.009       | 4.922             | −5080.191       | −5058.113    | 24.193       | 5.324             |
| MGVB         | 1.019     | 0.485     | 0.426     | −0.011    | 4.935             | −5082.123       | −5058.167    | 24.194       | 5.328             | 1.019 | 0.485      | 0.426  | −0.011       | 4.935             | −5082.123       | −5058.167    | 24.194       | 5.328             |
| QBVI         | 1.023     | 0.489     | 0.420     | −0.008    | 4.924             | −5080.679       | −5058.127    | 24.193       | 5.323             | 1.023 | 0.485      | 0.426  | −0.008       | 4.924             | −5080.679       | −5058.127    | 24.193       | 5.323             |
| MGVBP<sub>h</sub>-func. | 1.032     | 0.489     | 0.420     | −0.009    | 4.922             | −5080.192       | −5058.113    | 24.193       | 5.324             | 1.032 | 0.489      | 0.420  | −0.009       | 4.922             | −5080.192       | −5058.113    | 24.193       | 5.324             |
| MGVB<sub>h</sub>-func. | 1.019     | 0.485     | 0.426     | −0.011    | 4.934             | −5082.127       | −5058.166    | 24.194       | 5.328             | 1.019 | 0.485      | 0.426  | −0.011       | 4.934             | −5082.127       | −5058.166    | 24.194       | 5.328             |
| QBVI<sub>h</sub>-func. | 1.023     | 0.489     | 0.420     | −0.008    | 4.924             | −5080.652       | −5058.126    | 24.193       | 5.323             | 1.023 | 0.485      | 0.426  | −0.008       | 4.924             | −5080.652       | −5058.126    | 24.193       | 5.323             |
| MGVBP<sub>diag</sub> | 1.042     | 0.487     | 0.423     | −0.010    | 4.925             | −5082.818       | −5058.111    | 24.193       | 5.327             | 1.042 | 0.487      | 0.423  | −0.010       | 4.925             | −5082.818       | −5058.111    | 24.193       | 5.327             |
| MGVB<sub>diag</sub> | 1.020     | 0.485     | 0.430     | −0.015    | 4.944             | −5084.909       | −5058.230    | 24.195       | 5.331             | 1.020 | 0.485      | 0.430  | −0.015       | 4.944             | −5084.909       | −5058.230    | 24.195       | 5.331             |
| QBVI<sub>diag</sub> | 1.036     | 0.485     | 0.423     | −0.007    | 4.931             | −5083.253       | −5058.143    | 24.193       | 5.326             | 1.036 | 0.485      | 0.423  | −0.007       | 4.931             | −5083.253       | −5058.143    | 24.193       | 5.326             |
| MCMC         | 1.075     | 0.488     | 0.421     | −0.012    | 4.922             | −5058.090       | 24.192       | 5.328             | −1220.130 | 18.858 | 5.706     | 1.075 | 0.488      | 0.421  | −0.012       | 4.922             | −5058.090       | 24.192       | 5.328             |
| ML           | 1.080     | 0.488     | 0.421     | −0.012    | 4.919             | −5058.086       | 24.192       | 5.328             | −1220.053 | 18.857 | 5.706     | 1.080 | 0.488      | 0.421  | −0.012       | 4.919             | −5058.086       | 24.192       | 5.328             |
significant, so we trained a GARCH(1,0,1), or that the autoregressive coefficient of the squared innovations is always significant only at lag one, so we did not consider further lags for $\alpha$. For $\alpha, \beta, \gamma$, we restricted the search up to lag 2. Except for HAR’s parameter $\beta_3$, all the parameters of all the models are statistically significant under standard ML at 5%. Note that the aim of this experiment is to perform VI to the above class of models, not to discuss their empirical performance or forecasting ability. We refer readers unfamiliar with the above (standard) econometric models, discussion, and notation to the accessible introduction of Teräsvirta (2009).

We report the values of the smoothed lower bound computed at the optimized parameter $L(\xi^*)$, the model’s log likelihood in the estimates posterior parameter $p(y|\theta^*)$, and the MSE and QLIKE between the fitted values and squared daily returns and subsampled realized variances, used as a volatility proxy. Details on the data and hyperparameters are provided in appendix B.4.

A visual inspection of the marginal densities in Figures 8 to 10 reveals that, in general, both MGVBP and MGVB perform quite well compared to MCMC sampling and that the variational gaussian assumption is feasible for all the volatility models. Note that the skews observed in the figures are due to the parameter transformation: VI is applied on the unconstrained parameters ($\psi_\omega, \psi_\phi$), and such variational gaussians are back-transformed on the original constrained parameter space where the distributions are generally no longer gaussian.

B.3 Istanbul Data Set: Block-Diagonal Covariance. In this section, we apply MGVBP under different assumptions for the structure of the variational covariance matrix. We use the Istanbul stock exchange data set of
Akbilgic et al. (2014; details are provided in appendix B.4). To demonstrate the feasibility of the block-diagonal estimation under the mean-field framework outlined in section 5.6, we consider the following model for the daily returns of the Istanbul Stock Exchange National 100 index (ISE):

\[
ISE_t = \beta_0 + \beta_1 SP_t + \beta_2 NIK_t + \beta_3 BOV_t + \beta_4 DAX_t + \beta_5 FTSE_t + \beta_6 EU_t + \beta_7 EM_t + \epsilon_t,
\]

with \( \epsilon_t \sim N(0, \sigma^2) \) (conditionally on the regressors at time \( t \)). The covariates respectively correspond to daily returns of the S&P 500 index, the Japanese Nikkei index, the Brazilian Bovespa index, the German DAX index, the UK FTSE index, the MSCI European index, and of the MSCI emerging market
We estimate the coefficients $\beta_0, \ldots, \beta_7$ and the transformed parameter $\psi_\sigma = \log(\sigma)$, from which $\sigma$ is computed as $\sigma = \exp(\psi_\sigma) + \text{Var}(\psi_\sigma)/2$, with $\text{Var}(\psi_\sigma)$ taken from the variational posterior covariance matrix.

We consider the following structures for the variational posterior: (i) full covariance matrix (Full), (ii) diagonal covariance matrix (Diagonal), (iii) block-diagonal structure with two blocks of sizes $8 \times 8$ and $1 \times 1$ (Block 1), and (iv) block diagonal structure with blocks of sizes $1 \times 1$, $3 \times 3$, $2 \times 2$, $2 \times 2$ and $1 \times 1$ (Block 2). Case iii, models the covariance between the regression coefficients ($\beta$s) but neglects their covariance with the variance of the error $\sigma^2$. Case iv groups the indices traded in non-European stock exchanges in a $3 \times 3$ block, and in the remaining $2 \times 2$ blocks, the indices referring to European exchanges and the two MSCI indexes. As for the earlier case, the covariances between the regression parameters and $\sigma^2$ are ignored.

Note that the purpose of this application is to provide an example for algorithm 2 and for the discussion in appendix 5.6. To this end, structures iii and iv correspond to an intuitive and economically motivated grouping of the variables. Providing an effective predictive model supported by a solid econometric rationale is here out of scope.

Tables 12 and 13 summarize the estimation results. Table 12 shows that the impact of the different structures of the covariance matrix is somewhat marginal in terms of the performance measures, with respect to each other and with respect to the ML estimates. As for the logistic regression example, in the most constrained cases (ii and iii), we observe that the estimates of certain posterior means slightly deviate from the others, indicating that the algorithm perhaps reaches a different maximum. Regarding the variational covariances reported in Table 13, there is remarkable accordance between the covariance structures (i and iii) and ML, while for the diagonal structure (ii) and the block-diagonal structure (iv), the covariances are misaligned with the ML and full-diagonal case, further suggesting the convergence of the algorithms at different maxima of the LB.

From a theoretical perspective, if $\Sigma$ is the covariance matrix of the joint gaussian distribution of the variates (case i), estimates of the block-diagonal entries (or main diagonal) should match the corresponding elements in $\Sigma$. However, the elements in the submatrices (e.g., cases ii and iv), deviate from those $\Sigma$ (case i). Indeed, the results refer to independent optimizations of alternative models (over the same data set) that are not granted to converge at the same LB maximum (and thus variational gaussian). Across the covariance structures (i to iv) the optimal variational parameters correspond to different multivariate distributions that independently maximize the lower bound and are not constrained to be related to each other. This is indeed confirmed by the differences in the maximized lower bound $L(\hat{\zeta}^*)$ in Table 12, and in the different levels at which the curves in Figure 11 are observed to converge. In this light, the ML estimates’ variances in Table 14 can be compared to those of case i, but are misleading for the other cases, as the covariance matrix of the asymptotic (gaussian) distribution of the ML estimator is implicitly a full matrix.
Table 12: Posterior Means, ML Estimates, and Performance Measures on the Train and Test Set.

|            | $\beta_0$ | $\beta_1$ | $\beta_2$ | $\beta_3$ | $\beta_4$ | $\beta_5$ | $\beta_6$ | $\beta_7$ | $\sigma_e$ |
|------------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
| Full (case i) | 0.001     | 0.098     | 0.079     | -0.271    | -0.167    | -0.354    | 1.164     | 0.944     | 0.014     |
| Diagonal (case ii) | 0.001     | 0.074     | 0.092     | -0.239    | 0.093     | -0.015    | 0.555     | 0.935     | 0.014     |
| Block 1 (case iii) | 0.001     | 0.098     | 0.079     | -0.272    | -0.167    | -0.353    | 1.164     | 0.943     | 0.014     |
| Block 2 (case iv) | 0.001     | 0.067     | 0.115     | -0.218    | 0.201     | 0.162     | 0.293     | 0.871     | 0.014     |
| ML         | 0.001     | 0.099     | 0.078     | -0.273    | -0.174    | -0.363    | 1.179     | 0.946     | 0.014     |

|            | $\mathcal{L}^*$ | $p(y|\mu^*)$ | $10^2 \times \text{MSE}$ | $p(y|\mu^*)$ | $10^2 \times \text{MSE}$ |
|------------|----------------|-------------|-----------------|-------------|-----------------|
| Full       | 1186.082       | 1223.70     | 19.41           | 316.84      | 4.15            |
| Diagonal   | 1173.662       | 1214.83     | 20.23           | 316.71      | 4.12            |
| Block 1    | 1186.087       | 1223.70     | 19.41           | 316.86      | 4.15            |
| Block 2    | 1172.580       | 1212.28     | 20.48           | 316.72      | 4.10            |
| ML         | 1223.73        | 19.41       |                 | 316.87      | 4.15            |
Table 13: Posterior Covariance Matrix under the Full Specification (Case i), and Covariances of the ML Estimates and Posterior Covariance Matrices under the Block 1 and Block 2 Structures.

|       | $\beta_1$ | $\beta_2$ | $\beta_3$ | $\beta_4$ | $\beta_5$ | $\beta_6$ | $\beta_7$ | $\beta_8$ | $\sigma_0$ |
|-------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|------------|
| **ML** |           |           |           |           |           |           |           |           |            |
| $\beta_0$ | 0.001    | 0.002    | -0.001   | -0.002   | -0.001   | 0.006     | -0.008    | 0.000     |            |
| $\beta_1$ | 0.001    | 0.008    | -2.947   | -1.915   | -0.767   | -0.094    | 1.520     | -0.001    |            |
| $\beta_2$ | 0.003    | -0.032   | 1.094    | 0.442    | 0.768    | -0.640    | -4.141    | 0.012     |            |
| $\beta_3$ | -0.001   | -2.992   | 1.094    | 1.018    | 0.683    | -1.213    | -4.710    | 0.017     |            |
| $\beta_4$ | -0.002   | -1.906   | 0.297    | 0.961    | 3.742    | -20.149   | -0.748    | 0.028     |            |
| $\beta_5$ | -0.002   | -0.729   | 0.679    | 0.622    | 3.836    | -28.773   | -1.594    | 0.052     |            |
| $\beta_6$ | 0.007    | -0.170   | -0.393   | -1.065   | -20.477  | -28.813   | -2.984    | -0.142    |            |
| $\beta_7$ | -0.009   | 1.605    | -4.148   | -4.700   | -0.563   | -1.386    | -3.324    | 0.030     |            |

|       | $\beta_1$ | $\beta_2$ | $\beta_3$ | $\beta_4$ | $\beta_5$ | $\beta_6$ | $\beta_7$ | $\beta_8$ | $\sigma_0$ |
|-------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|------------|
| **Block 1** |           |           |           |           |           |           |           |           |            |
| $\beta_0$ | 0.001    | 0.002    | -0.001   | -0.003   | -0.002   | 0.008     | -0.009    |            |            |
| $\beta_1$ | -0.064   | -2.966   | -2.005   | -0.728   | -0.049   | 1.665     |            |            |            |
| $\beta_2$ | 0.322    | 1.079    | 0.296    | 0.690    | -0.382   | -4.075    |            |            |            |
| $\beta_3$ | -3.327   | -0.525   | 0.950    | 0.512    | -1.010   | -4.558    |            |            |            |
| $\beta_4$ |          | 3.841    | -20.608  | -0.584   |          |            |            |            |            |
| $\beta_5$ |          | -8.708   | -28.509  | -1.107   |          |            |            |            |            |
| $\beta_6$ |          |          | -3.584   |          |          |            |            |            |            |
| $\beta_7$ |          |          | -4.452   |          |          |            |            |            |            |

Note: All the entries are multiplied by $10^4$. 
Figure 11: Lower-bound optimization for the Istanbul data under the different posterior variational covariance matrix specifications.

Table 14: Standard Deviations of the Posterior Marginals for the Different Cases, along with ML Standard Errors.

|      | $\beta_1$ | $\beta_2$ | $\beta_3$ | $\beta_4$ | $\beta_5$ | $\beta_6$ | $\beta_7$ | $\beta_8$ | $\sigma$ |
|------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|----------|
| Full | 0.069     | 7.472     | 5.641     | 7.322     | 12.915    | 16.663    | 23.056    | 12.395    | 3.463    |
| Diagonal | 0.073   | 4.662     | 4.056     | 4.477     | 4.343     | 5.587     | 4.733     | 5.976     | 3.599    |
| Block 1 | 0.068   | 7.470     | 5.629     | 7.274     | 13.072    | 16.534    | 23.092    | 12.257    | 3.421    |
| Block 2 | 0.071   | 6.530     | 4.451     | 6.611     | 9.333     | 10.620    | 7.001     | 8.724     | 3.367    |
| ML    | 0.068    | 7.489     | 5.631     | 7.322     | 12.997    | 16.636    | 23.133    | 12.363    |          |

Note: Entries are multiplied by $10^2$.

B.4 Data Sets and Hyperparameters. Table 15 summarizes some information about the data sets and the setup used across the experiments. For the experiments on the Labour and S&P 500 data sets, the same set of hyperparameters applies to MGVBP and MGVB (and QBVI). While the Labour$^3$ and Istanbul$^4$ data set are readily available, the S&P 500 data set is extracted from the Oxford-Man Institute realized volatility library.$^5$ We use daily close-to-close demeaned returns for the GARCH-family models and five-minute subsampled daily measures of realized volatilities (further annualized) for the HAR model. For information on the publicly available limit-order book (LOB) data set, see Ntakaris et al. (2018). MGVBP and all

---

$^3$Publicly available at key2stats.com/data-set/view/140. See Mroz (1984) for details. The data are also adopted in other VI applications (e.g., Tran, Nguyen, and Dao, 2021; Magris et al., 2022).

$^4$Publicly available at the UCI repository, archive.ics.uci.edu/ml/datasets/istanbul+stock+exchange. See Akbilgic et al. (2014) for details.

$^5$realized.oxford-man.ox.ac.uk.
Table 15: Details on the Data Sets and Corresponding Models.

| Data Set   | Model               | Number of variational parameters | Number of samples | Samples in train set | Samples in test set | Period                      |
|------------|---------------------|----------------------------------|-------------------|----------------------|---------------------|-----------------------------|
| Labour     | Logistic regression | 72                               | 753               | 564 (75%)            | 189 (25%)           | 3-Jan-2014 / 28-Jun-2022    |
| S&P 500    | ARCH                | 6                                | 2123              | 1689 (80%)           | 425                 | 3-Jan-2014 / 28-Jun-2022    |
|            | GARCH(1,0,1)        | 12                               | 2123              | 1689                 | 425                 | 3-Jan-2014 / 28-Jun-2022    |
|            | GJR(1,1,1)          | 20                               | 2123              | 1689                 | 425                 | 3-Jan-2014 / 28-Jun-2022    |
|            | GJR(1,1,2)          | 30                               | 2123              | 1689                 | 425                 | 3-Jan-2014 / 28-Jun-2022    |
|            | EGARCH(1,0,1)       | 12                               | 2123              | 1689                 | 425                 | 3-Jan-2014 / 28-Jun-2022    |
|            | EGARCH(1,1,1)       | 20                               | 2123              | 1689                 | 425                 | 3-Jan-2014 / 28-Jun-2022    |
|            | EGARCH(1,1,2)       | 30                               | 2123              | 1689                 | 425                 | 3-Jan-2014 / 28-Jun-2022    |
|            | FIGARCH(0,1,1)      | 12                               | 2123              | 1689                 | 425                 | 3-Jan-2014 / 28-Jun-2022    |
|            | FIGARCH(1,1,2)      | 20                               | 2123              | 1689                 | 425                 | 3-Jan-2014 / 28-Jun-2022    |
|            | HAR (Linear regr.)  | 30                               | 2102              | 1681                 | 421                 | 4-Feb-2014 / 28-Jun-2022    |
| Istanbul   | Linear regression   | 90                               | 536               | 428 (80%)            | 108                 | 5-Jan-2009 / 22-Feb-2022    |
| LOB       | Neural network      | 54,990                           | 557,297           | 256,461 (65%)        | 150,418             | 1-Jun-2010 / 14-Jun-2010    |
| Synthetic | Non-diff. model     | 420                              | 5000              | 4000 (80%)           | 1000                |                             |
Table 15: (Continued) Details on the Hyperparameters Used in the Experiments.

| Experiment          | $\beta$ | Grad. clip | Grad. clip init. | $\omega$ | $w$  | $t_{\text{max}}$ | $t'_t$ | $P$  | $S$  | Initial values | Prior          |
|---------------------|---------|------------|------------------|---------|-----|------------------|-------|-----|-----|----------------|---------------|
| Labour data         | 0.01    | 3000       | 1000             | 0.4     | 30  | 1200             | 1000  | 500 | 75  | $\mathcal{N}(0, \Sigma_1)$ | 0.05          |
| ARCH-GARCH-GJR      | 0.01    | 1000       | 1000             | 0.4     | 30  | 1200             | 1000  | 500 | 150 | ML             | 0.05          |
| EGARCH              | 0.01    | 1000       | 1000             | 0.4     | 30  | 3000             | 2500  | 500 | 150 | ML             | 0.05          |
| FIGARCH             | 0.01    | 1000       | 1000             | 0.4     | 30  | 1200             | 1000  | 500 | 150 | ML             | 0.05          |
| HAR                 | 0.001   | 50,000     | 1000             | 0.4     | 30  | 1000             | 750   | 100 | 150 | ML             | 0.01          |
| İstanbul data       | 0.07    | 50,000     | 500              | 0.4     | 30  | 1200             | 1000  | 500 | 100 | $\mathcal{N}(0, \Sigma_1)$ | 0.01          |
| LOB data            | 0.1     | 5000       | 100              | 0.4     | 30  | 4000             | 3000  | 500 | 20  | ADAM           | 0.01          |
| Non-diff. model     | 0.1     | 150        | 1000             | 0.4     | 30  | 5000             | 5000  | 500 | 15  | $\mathcal{N}(0, \Sigma_1)$ | 0.01          |
optimizers appearing in Table 5 are initialized after 30 ADAM iterations for the neural network experiments. VOGN is implemented on the LOB data following the scheme presented in Magris et al. (2023). The synthetic data used for the nondifferentiable model are discussed in section 7.

**B.4.1 Starting Values.** As a robustness check, in Tables 16 and 17, we provide summary statistics of the estimation results and performance metrics for two representative models. With both a random and constant initialization of $\theta_0$, we see a tight consistency in the results, underlining the robustness of the estimation routine with respect to the choice of the starting values.

**B.4.2 Runtime.** Table 18 compares the running times of the adopted variational Bayes algorithms for two representative models. As expected, MGVBP and MGVB show a rather aligned performance (the factor 2 in equation 5.3 compared to equation 4.3 is indeed irrelevant in terms of running time). On the other hand, QBVI relies on update rules that are completely unrelated and different from those of MGVBP and MGVB; as such, the corresponding running times are provided as a reference since they are not directly comparable. Irrespective of the adopted optimizers, the major factor deriving the runtime appears to be the complexity of the log likelihood (with little surprise the iterative nature of the expensive computations in the GARCH likelihood severely affects the runtime) the log $p$ estimator variant, equation 5.12, is consistently outperforming the classical approach based on the $h$-function estimator equation 5.11.

Table 19 reports the runtime for a linear regression experiment (simulated data, $N_s = 100$, averaged across 50 replications) per iteration with a growing number of variational parameters. It can be seen that MGVBP is slightly but consistently faster than MGVB. As discussed in section 5.7, despite the simple natural gradient form the MGVBP has, the actual computational effort is a retraction, which requires inverting the variational covariance matrix at each iteration. Nevertheless, the experiments we performed show a steeper LB optimization and an improved optimum.

Appendix C: Proofs

**C.1 Preliminaries: The Gaussian FIM.** From Barfoot (2020) the natural gradients for a $d$-variate gaussian distribution with mean $\mu$ and covariance matrix $\Sigma$, in the parameterizations $\xi^c$ and $\xi^t$ are respectively given by

$$
\mathcal{I}^{-1}_{\xi^c} = \begin{bmatrix} \Sigma & 0 \\ 0 & 2(\Sigma \otimes \Sigma) \end{bmatrix}, \quad \mathcal{I}^{-1}_{\xi^t} = \begin{bmatrix} \Sigma & 0 \\ 0 & 2(\Sigma \otimes \Sigma)^{-1} \end{bmatrix}.
$$

(C.1)
Table 16: Estimation Results for the Logistic Regression Model (Labor Data Set) under Different Starting Values.

| $\theta_0 \sim N(\mu, \sigma^2 I)$ | Mean | Std. Dev. ($\times 100$) |
|-------------------------------------|------|--------------------------|
|                                     | $\mathcal{L}(\zeta^* \gamma)$ | Accuracy | Precision | Recall | F1     | $\mathcal{L}(\zeta^* \gamma)$ | Accuracy | Precision | Recall | F1     |
| $\mu$                               |      |                         |           |         |       |       |                       |           |         |       |       |
| 0                                   | $-356.640$ | 71.179                  | 71.108    | 70.216  | 70.659 | $0.091$ | 0.090 | 0.103     | 0.084 | 0.093 |
| 0                                   | $-356.640$ | 71.152                  | 71.076    | 70.194  | 70.632 | $0.081$ | 0.083 | 0.096     | 0.076 | 0.086 |
| 0                                   | $-356.640$ | 71.117                  | 71.037    | 70.160  | 70.595 | $0.117$ | 0.079 | 0.087     | 0.077 | 0.082 |
| 0                                   | $-356.641$ | 71.126                  | 71.047    | 70.175  | 70.605 | $0.065$ | 0.065 | 0.074     | 0.060 | 0.067 |
| 0                                   | $-356.640$ | 71.117                  | 71.035    | 70.161  | 70.596 | $0.111$ | 0.055 | 0.063     | 0.049 | 0.056 |
| 0.1                                 | $-356.640$ | 71.126                  | 71.047    | 70.165  | 70.606 | $0.117$ | 0.087 | 0.096     | 0.084 | 0.089 |
| -0.5                                | $-356.640$ | 71.137                  | 71.061    | 70.175  | 70.616 | $0.081$ | 0.095 | 0.105     | 0.092 | 0.098 |
| 5                                   | $-356.640$ | 71.117                  | 71.111    | 70.222  | 70.664 | $0.072$ | 0.091 | 0.105     | 0.082 | 0.093 |
| -5                                  | $-356.640$ | 71.165                  | 71.035    | 70.161  | 70.596 | $0.113$ | 0.055 | 0.063     | 0.049 | 0.056 |
| -5                                   | $-356.640$ | 71.126                  | 71.045    | 70.170  | 70.605 | $0.090$ | 0.065 | 0.075     | 0.059 | 0.067 |
| -5                                   | $-356.640$ | 71.152                  | 71.078    | 70.192  | 70.632 | $0.098$ | 0.101 | 0.113     | 0.097 | 0.104 |
| -5                                   | $-356.640$ | 71.146                  | 71.068    | 70.188  | 70.625 | $0.086$ | 0.080 | 0.092     | 0.073 | 0.082 |

| $\theta_0 = c \mathbf{1}_d$ | Mean | Std. Dev. ($\times 100$) |
|-------------------------------|------|--------------------------|
|                              | $\mathcal{L}(\zeta^* \gamma)$ | Accuracy | Precision | Recall | F1     | $\mathcal{L}(\zeta^* \gamma)$ | Accuracy | Precision | Recall | F1     |
| $c$                           |      |                         |           |         |       |       |                       |           |         |       |       |
| 0                             | $-356.640$ | 71.165                  | 71.092    | 70.203  | 70.644 | $0.064$ | 0.088 | 0.100     | 0.082 | 0.090 |
| 1                             | $-356.640$ | 71.146                  | 71.068    | 70.188  | 70.625 | $0.080$ | 0.080 | 0.092     | 0.073 | 0.082 |
| -1                            | $-356.640$ | 71.144                  | 71.067    | 70.184  | 70.623 | $0.126$ | 0.098 | 0.108     | 0.093 | 0.100 |
| 5                             | $-356.640$ | 71.137                  | 71.059    | 70.177  | 70.616 | $0.080$ | 0.095 | 0.105     | 0.091 | 0.098 |
| -5                            | $-356.640$ | 71.144                  | 71.066    | 70.186  | 70.623 | $0.077$ | 0.079 | 0.091     | 0.071 | 0.081 |
| 20                            | $-356.641$ | 71.135                  | 71.057    | 70.176  | 70.614 | $0.112$ | 0.073 | 0.083     | 0.067 | 0.075 |
| -20                           | $-356.641$ | 71.179                  | 71.108    | 70.216  | 70.659 | $0.114$ | 0.107 | 0.119     | 0.102 | 0.110 |

Note: Statistics are computed across 200 replications. Upper table: random initialization of $\theta_0$ from a multivariate gaussian with diagonal covariance matrix. Lower table: initialization of $\theta_0$ as a vector of constants.
Table 17: Estimation Results for the GARCH(1,0,1) Model under Different Starting Values.

| $\theta_0 \sim N(\mu, \sigma^2 I)$ Mean | Std. Dev. ($\times 100$) |
|----------------------------------------|-------------------------|
| $\mu$                                  | $\sigma^2$              | $\mathcal{L}(\xi^*)$ | $p(y|\theta^*)$ | MSE | $\mathcal{L}(\xi^*)$ | $p(y|\theta^*)$ | MSE |
| 0                                      | 0.1                     | $-2012.404$            | 2002.560        | 25.690 | 0.150 | 0.243 | 0.122 |
| 0                                      | 1                       | $-2012.404$            | 2002.560        | 25.690 | 0.148 | 0.252 | 0.129 |
| 0                                      | 5                       | $-2012.404$            | 2002.561        | 25.691 | 0.144 | 0.259 | 0.112 |
| 0                                      | 10                      | $-2012.412$            | 2002.564        | 25.689 | 3.046 | 0.744 | 1.245 |
| 5                                      | 0.1                     | $-2012.404$            | 2002.562        | 25.692 | 0.156 | 0.345 | 0.160 |
| 5                                      | 1                       | $-2012.404$            | 2002.562        | 25.691 | 0.156 | 0.348 | 0.188 |
| 5                                      | 5                       | $-2012.404$            | 2002.563        | 25.692 | 0.146 | 0.603 | 0.280 |
| 5                                      | 10                      | $-2012.404$            | 2002.562        | 25.691 | 0.179 | 0.557 | 0.315 |
| -5                                     | 0.1                     | $-2012.404$            | 2002.563        | 25.691 | 0.136 | 0.292 | 0.129 |
| -5                                     | 1                       | $-2012.404$            | 2002.563        | 25.691 | 0.134 | 0.646 | 0.240 |
| -5                                     | 5                       | $-2012.404$            | 2002.563        | 25.691 | 0.147 | 0.513 | 0.092 |
| -5                                     | 10                      | $-2012.405$            | 2002.565        | 25.693 | 0.216 | 1.042 | 0.450 |

Note: Statistics are computed across 200 replications. Upper table: random initialization of $\theta_0$ from a multivariate gaussian with diagonal covariance matrix. Lower table: initialization of $\theta_0$ as a vector of constants.

C.2 Preliminaries: A Useful Relation. For a generic SPD matrix $P$ and a function $f$ of $P$ that can be as well reparameterized in terms of $P^{-1}$,

$$P \nabla_P f P = -\nabla_{P^{-1}} f,$$

(C.2)

which can be shown to hold according to matrix calculus with Jacobian transformations of the gradient $\nabla_P f$.

C.3 Proof of Proposition 1. From the natural gradient definition:

$$\nabla_{\xi^i} \mathcal{L}^{i \defeq \mathcal{L}^{i \left( C.1 \right)}} = \mathcal{L}^{i \left( C.1 \right)} \nabla_{\Sigma^{-1}} \mathcal{L}^{i \left( C.1 \right)} = \left[ \Sigma \nabla_{\mu} \mathcal{L}^{i \left( C.1 \right)} \
\frac{2(\Sigma \otimes \Sigma)^{-1} \nabla_{\Sigma} \mathcal{L}^{i \left( C.1 \right)}} \right],$$
Table 18: Running Time for the Logistic Regression (Labor Data Set).

|                      | Logistic regression | GARCH(1,0,1)         | GJR(1,1,1)         |
|----------------------|---------------------|----------------------|-------------------|
|                      | Mean | Median | Std. | Mean | Median | Std. | Mean | Median | Std. | Mean | Median | Std. |
| h-func. estimator    |      |        |      |      |        |      |      |        |      |      |        |      |
| Full cov.            |      |        |      |      |        |      |      |        |      |      |        |      |
| MGVB                 | 5.411| 5.387  | 1.137| 5.375| 5.346  | 0.181| 32.76| 33.05  | 0.78 | 30.73| 30.45  | 2.44 |
| MGVBP                | 5.375| 5.368  | 1.483| 5.321| 5.328  | 0.127| 33.02| 32.90  | 0.45 | 30.91| 32.07  | 2.30 |
| QBVI                 | 5.340| 5.305  | 0.884| 5.425| 5.408  | 0.254| 32.70| 32.67  | 0.66 | 30.79| 31.03  | 2.53 |
|                     | Diagonal cov.       |                      | Full cov.          |                      |
| MGVB                 | 5.883| 5.877  | 0.121| 5.791| 5.733  | 0.419| 30.94| 31.18  | 1.52 | 29.18| 30.18  | 1.89 |
| MGVBP                | 5.929| 5.904  | 0.184| 5.666| 5.656  | 0.743| 31.47| 31.69  | 0.82 | 29.21| 29.79  | 1.72 |
| QBVI                 | 5.879| 5.858  | 0.119| 5.543| 5.551  | 0.353| 31.14| 31.39  | 0.94 | 28.86| 29.98  | 2.39 |

Note: Statistics in seconds per 1000 iterations averaged over 200 replications ($N_s = 75$).
Table 19: Running Times (Seconds per Iteration) for a Linear Regression Problem of Increasing Complexity.

| Number of parameters | 6   | 12  | 30  | 110 | 240 | 650 | 2550 | 10,100 | 22,650 |
|----------------------|-----|-----|-----|-----|-----|-----|------|--------|--------|
| MGVB                 | 0.0365 | 0.0174 | 0.0194 | 0.0247 | 0.0323 | 0.0766 | 0.2112 | 0.6997 | 1.5176 |
| MGVBP                | 0.0355 | 0.0169 | 0.0185 | 0.0237 | 0.0308 | 0.0760 | 0.2044 | 0.6795 | 1.5037 |
| Difference           | 2.8% | 2.8% | 4.6% | 4.0% | 4.7% | 0.8% | 3.2% | 2.9% | 0.9% |

MGVB and MGVBP are compared, showing the running times for different numbers of parameters. The differences are calculated as the percentage difference between the times of MGVB and MGVBP.
removing the vectorization from the term related to $\Sigma^{-1}$:

$$\tilde{\nabla}_{\Sigma^{-1}} L^i = 2 \Sigma^{-1} \nabla_{\Sigma^{-1}} L^i \Sigma^{-1} = -2 \nabla_{\Sigma} L^i.$$  

(C.3)

### C.4 Derivation of the MGVBP Update

For the precision matrix, the Riemann gradient from the manifold $M$ is $\Sigma^{-1} \nabla_{\Sigma^{-1}} \Sigma^{-1}$. Therefore, the following equivalent representations hold:

$$\tilde{\nabla}_{\Sigma^{-1}} L (C.2) \equiv \Sigma^{-1} \nabla_{\Sigma^{-1}} \Sigma^{-1} (C.2) \equiv - \nabla_{\Sigma} L \equiv \frac{1}{2} \tilde{\nabla}_{\Sigma^{-1}} L.$$  

Note that the last equality is not general; it holds because of the particular $(\Sigma \otimes \Sigma)^{-1}$ form of the second block of $I^{-1}_\iota$, specific for the gaussian log likelihood. As a consequence, the following three updates are equivalent:

$$\Sigma^{-1} \leftarrow R_{\Sigma^{-1}} (\beta \Sigma^{-1} \nabla_{\Sigma^{-1}} \Sigma^{-1}),$$  

$$\Sigma^{-1} \leftarrow R_{\Sigma^{-1}} (-\beta \nabla_{\Sigma} L),$$  

(C.4)  

$$\Sigma^{-1} \leftarrow R_{\Sigma^{-1}} \left( \beta \frac{1}{2} \tilde{\nabla}_{\Sigma^{-1}} L \right).$$  

(C.5)

All three correspond to an update of a Riemann gradient on the manifold $M$ based on the retraction form derived from the manifold $M$ (specific for the equipped Euclidean metric). The caveat with equation C.5 alone, without proposition 1 is the following: by removing the factor $\frac{1}{2}$, the argument of the retraction form, equation C.5, derived from the SPD manifold equipped with the Euclidean metric, is no longer the Riemann gradient for this manifold, but the Riemann gradient of a different manifold (the SPD manifold equipped with the Fisher-Rao metric $\mathcal{F}$). This point is central and discussed in the main text.

In support of our argument and derivation, by unfolding the retraction in equation C.5, the update

$$\Sigma^{-1} \leftarrow \Sigma^{-1} + \frac{\beta}{2} [\tilde{\nabla}_{\Sigma^{-1}} L] + \frac{\beta^2}{2} [\tilde{\nabla}_{\Sigma^{-1}} L] \Sigma [\tilde{\nabla}_{\Sigma^{-1}} L],$$

aligns with the update form of Lin et al. (2020) as a special case of their Bayesian learning rule, since the gaussian FIM satisfies their block-coordinate natural parameterization assumption.

Working on $\Sigma$, one analogously obtains $R_{\Sigma} (\Sigma \nabla_{\Sigma} L \Sigma) = R_{\Sigma} (\frac{1}{2} \tilde{\nabla}_{\Sigma} L)$; however, the latter does not correspond to a simpler algebraic form leading to a computationally convenient update like equation C.4.
C.5 General Form of the MGVBP Update. For a prior $p \sim \mathcal{N}(\mu_0, \Sigma_0)$ and a variational posterior $q \sim \mathcal{N}(\mu, \Sigma)$, by rewriting the LB as
\[
\mathbb{E}_{q(t)} [h(t)(\theta)] = \mathbb{E}_{q(t)} [\log p(\theta) - \log q(t)(\theta) + \log p(y|\theta)] \\
= \mathbb{E}_{q(t)} [\log p(\theta) - \log q(t)(\theta)] + \mathbb{E}_{q(t)} [\log p(y|\theta)],
\]
we decompose $\nabla_{t}^e \mathcal{L}(\xi^t)$ as $\nabla_{t}^e \mathbb{E}_{q(t)} [\log p(\theta) - \log q(t)(\theta)] + \nabla_{t}^e \mathbb{E}_{q(t)} [\log p(y|\theta)]$. As in equation 5.6, we apply the log-derivative trick on the last term and write $\nabla_{t}^e \mathbb{E}_{q(t)} [\log p(y|\theta)] = \mathbb{E}_{q(t)} [\nabla_{t}^e \log q(t) \log p(y|\theta)]$. On the other hand, it is easy to show that up to a constant that does not depend on $\mu$ and $\Sigma$,
\[
\mathbb{E}_{q(t)} [\log p(\theta) - \log q(t)(\theta)] = -\frac{1}{2} \log |\Sigma| + \frac{1}{2} \log |\Sigma_0| + \frac{1}{2} \log |\Sigma_0| - \frac{1}{2} \log |\Sigma_0| - \frac{1}{2} (\mu - \mu_0)^\top \Sigma_0^{-1} (\mu - \mu_0),
\]
so that
\[
\nabla_\Sigma \mathbb{E}_{q(t)} [\log p(\theta) - \log q(t)(\theta)] = \frac{1}{2} \Sigma^{-1} - \frac{1}{2} \Sigma_0^{-1},
\]
\[
\nabla_\mu \mathbb{E}_{q(t)} [\log p(\theta) - \log q(t)(\theta)] = -\Sigma_0^{-1} (\mu - \mu_0).
\]
For the natural gradients, we have
\[
\nabla_{\Sigma^{-1}} \mathbb{E}_{q(t)} [\log p(\theta) - \log q(t)(\theta)] = -\nabla_\Sigma \mathbb{E}_{q(t)} [\log p(\theta) - \log q(t)(\theta)] \\
= -\frac{1}{2} \Sigma^{-1} + \frac{1}{2} \Sigma_0^{-1},
\]
\[
\nabla_\mu \mathbb{E}_{q(t)} [\log p(\theta) - \log q(t)(\theta)] = \Sigma \nabla_\mu \mathbb{E}_{q(t)} [\log p(\theta) - \log q(t)(\theta)] \\
= -\Sigma_0^{-1} (\mu - \mu_0),
\]
while the naive estimators for $\nabla_\mu \mathbb{E}_{q(t)} [\log p(y|\theta)]$ and $\nabla_{\Sigma^{-1}} \mathbb{E}_{q(t)} [\log p(y|\theta)]$ turn analogous to feasible natural gradient estimators presented in section 5.3, with $h(t)$ replaced by $\log p(y|\theta)$. This leads to the general form of the MGVBP update, based either on the $h$-function gradient estimator (generally applicable) or the above decomposition (applicable under a gaussian prior):
\[
\nabla_\mu \mathcal{L}(\xi) \approx \mathbb{E}_{\xi}(\theta) + \frac{1}{\mathcal{S}} \sum_{s=1}^{\mathcal{S}} ((\theta_s - \mu_t) \log f(\theta_s)),
\]
\[
\nabla_{\Sigma^{-1}} \mathcal{L}(\xi(t)) \approx \mathbb{E}_{\xi}(\theta) + \frac{1}{2\mathcal{S}} \sum_{s=1}^{\mathcal{S}} ((\Sigma^{-1} - \Sigma^{-1}_t) (\theta_s - \mu_t)(\theta_s - \mu_t)^\top \Sigma_0^{-1}) \log f(\theta_s)).
\]
where
\[
\begin{cases}
C_{\Sigma_i} = -\frac{1}{2} \Sigma_i^{-1} + \frac{1}{2} \Sigma_0^{-1}, \\
c_{\mu_i} = -\Sigma_i \Sigma_0^{-1} (\mu_i - \mu_0), & \text{if } p \text{ is gaussian,} \\
\log f(\theta_s) = \log p(y|\theta_s),
\end{cases}
\]
\[(C.6)\]

The short-hand notation \(\log f(\theta_s)\) embeds the fact that the function is meant to be evaluated at the current value of the parameter, \(\log f(\theta_s) \equiv \log f(\theta_s|\mu = \mu_t, \Sigma = \Sigma_t)\).

Acknowledgments

This project has received funding from the European Union’s Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant agreement 890690 and the Independent Research Fund Denmark project DISPA (project. 9041-00004).

References

Absil, P.-A., Mahony, R., & Sepulchre, R. (2008). Optimization algorithms on matrix manifolds. Princeton University Press.

Akibilgic, O., Bozdogan, H., & Balaban, M. E. (2014). A novel hybrid RBF neural networks model as a forecaster. Statistics and Computing, 24(3), 365–375. 10.1007/s11222-013-9375-7

Barfoot, T. D. (2020). Multivariate gaussian variational inference by natural gradient descent. arXiv:2001.10025.

Bhatia, R., Jain, T., & Lim, Y. (2019). On the Bures–Wasserstein distance between positive definite matrices. Expositiones Mathematicae, 37(2), 165–191. 10.1016/j.exmath.2018.01.002

Blei, D. M., Kucukelbir, A., & McAuliffe, J. D. (2017). Variational inference: A review for statisticians. Journal of the American Statistical Association, 112(518), 859–877. 10.1080/01621459.2017.1285773

Blundell, C., Cornebise, J., Kavukcuoglu, K., & Wierstra, D. (2015). Weight uncertainty in neural network. In Proceedings of the International Conference on Machine Learning (pp. 1613–1622).

Boumal, N., Mishra, B., Absil, P.-A., & Sepulchre, R. (2014). Manopt, a Matlab toolbox for optimization on manifolds. Journal of Machine Learning Research, 15(42), 1455–1459.
Carbonetto, P., & Stephens, M. (2012). Scalable variational inference for Bayesian variable selection in regression, and its accuracy in genetic association studies. *Bayesian Analysis, 7*(1), 73–108. 10.1214/12-BA703

Corsi, F. (2009). A simple approximate long-memory model of realized volatility. *Journal of Financial Econometrics, 7*(2), 174–196. 10.1093/jjfinec/nbp001

Ganguly, A., & Earp, S. W. F. (2021). An introduction to variational inference. arXiv:2108.13083.

Graves, A. (2011). Practical variational inference for neural networks. In J. Shawe-Taylor, R. Zemel, Bartlett, F. Pereira, & K. Q. Weinberger (Eds.), *Advances in neural information processing systems, 24*. Curran.

Han, A., Mishra, B., Jawanpuria, P. K., & Gao, J. (2021). On Riemannian optimization over positive definite matrices with the Bures-Wasserstein geometry. In M. A. Beygelzimer, Y. Dauphin, P. S. Liang, & J. Wortman Vaughan (Eds.), *Advances in neural information processing systems, 34* (pp. 8940–8953). Curran.

Hoffman, M. D., Blei, D. M., Wang, C., & Paisley, J. (2013). Stochastic variational inference. *Journal of Machine Learning Research, 14*(40), 1303–1347.

Hosseini, R., & Sra, S. (2015). Matrix manifold optimization for gaussian mixtures. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, & R. Garnett (Eds.), *Advances in neural information processing systems, 28*. Curran.

Jeuris, B., Vandebril, R., & Vandereycken, B. (2012). A survey and comparison of contemporary algorithms for computing the matrix geometric mean. *Electronic Transactions on Numerical Analysis, 39*, 379–402.

Khan, M., & Lin, W. (2017). Conjugate-computation variational inference: Converting variational inference in non-conjugate models to inferences in conjugate models. In *Artificial Intelligence and Statistics* (pp. 878–887).

Khan, M. E., & Nielsen, D. (2018). Fast yet simple natural-gradient descent for variational inference in complex models. In *Proceedings of the 2018 International Symposium on Information Theory and Its Applications* (pp. 31–35).

Khan, M., Nielsen, D., Tangkaratt, V., Lin, W., Gal, Y., & Srivastava, A. (2018). Fast and scalable Bayesian deep learning by weight-perturbation in Adam. In *Proceedings of the International Conference on Machine Learning* (pp. 2611–2620).

Kingma, D. P., & Ba, J. (2014). *Adam: A method for stochastic optimization*. arXiv:1412.6980.

Kingma, D. P., & Welling, M. (2013). *Auto-encoding variational Bayes*. arXiv:1312.6114.

Kucukelbir, A., Ranganath, R., Gelman, A., & Blei, D. (2015). Automatic variational inference in Stan. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, & R. Garnett (Eds.), *Advances in neural information processing systems, 28*. Curran.

Lampinen, J., & Vehtari, A. (2001). Bayesian approach for neural networks: Review and case studies. *Neural Networks, 14*(3), 257–274. 10.1016/S0893-6080(00)00098-8

Lin, W., Nielsen, F., Khan, M. E., & Schmidt, M. (2021). *Structured second-order methods via natural gradient descent*. arXiv:2107.10884.

Lin, W., Schmidt, M., & Khan, M. E. (2020). Handling the positive-definite constraint in the Bayesian learning rule. In *Proceedings of the International Conference on Machine Learning* (pp. 6116–6126).

Lyu, Y., & Tsang, I. W. (2021). Black-box optimizer with stochastic implicit natural gradient. In D. Koutra, M. Rodriguez, E. Baralis, & F. Bonchi (Eds.), *Machine learning and knowledge discovery in databases* (pp. 217–232). Springer.
Mackay, D. J. C. (1992). *Bayesian methods for adaptive models*. PhD diss., California Institute of Technology.

Mackay, D. J. C. (1995). Probable networks and plausible predictions: A review of practical Bayesian methods for supervised neural networks. *Network: Computation in Neural Systems*, 6, 469–505. 10.1088/0954-898X_6_3_011

Magris, M., & Iosifidis, A. (2023a). Bayesian learning for neural networks: An algorithmic survey. *Artificial Intelligence Review*, 56(10), 1–52.1–51. 10.1007/s10462-023-10443-1

Magris, M., & Iosifidis, A. (2023b). Variational inference for GARCH-family models. In *Proceedings of the Fourth ACM International Conference on AI in Finance* (pp. 541–548).

Magris, M., Shabani, M., & Iosifidis, A. (2022). *Quasi black-box variational inference with natural gradients for Bayesian learning*. arXiv:2205.11568.

Magris, M., Shabani, M., & Iosifidis, A. (2023). Bayesian bilinear neural network for predicting the mid-price dynamics in limit-order book markets. *Journal of Forecasting*, 42(6), 1407–1428. 10.1002/for.2955

Mardia, K. V., & Marshall, R. J. (1984). Maximum likelihood estimation of models for residual covariance in spatial regression. *Biometrika*, 71(1), 135–146. 10.1093/biomet/71.1.135

Mohamed, S., Rosca, M., Figurnov, M., & Mnih, A. (2020). Monte Carlo gradient estimation in machine learning. *Journal of Machine Learning Research*, 21(132), 1–62.

Mroz, T. A. (1984). *The sensitivity of an empirical model of married women’s hours of work to economic and statistical assumptions*. PhD diss., Stanford University.

Ntakaris, A., Magris, M., Kannaiainen, J., Gabouj, M., & Iosifidis, A. (2018). Benchmark dataset for mid-price forecasting of limit order book data with machine learning methods. *Journal of Forecasting*, 37(8), 852–866. 10.1002/for.2543

Osawa, K., Swaroop, S., Khan, M. E., Jain, A., Eschenhagen, R., Turner, R. E., & Yokota, R. (2019). Practical deep learning with Bayesian principles. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d’Alché-Buc, E. Fox, & R. Garnett (Eds.), *Advances in neural information processing systems*, 32 (pp. 1–13). Curran.

Paisley, J., Blei, D. M., & Jordan, M. I. (2012). Variational Bayesian inference with stochastic search. In *International Conference on Machine Learning*. 10.5555/3042573.3042748

Patton, A. J. (2011). Volatility forecast comparison using imperfect volatility proxies. *Journal of Econometrics*, 160(1), 246–256. 10.1016/j.jeconom.2010.03.034

Pennec, X. (2020). Manifold-valued image processing with SPD matrices. In X. Pennec, S. Sommer, & T. Fletcher (Eds.), *Riemannian geometric statistics in medical image analysis* (pp. 75–134). Elsevier.

Ranganath, R., Gerrish, S., & Blei, D. M. (2014). Black box variational inference. In *Proceedings of the 17th International Conference on Artificial Intelligence and Statistics* (pp. 814–822).

Robbins, H., & Monro, S. (1951). A stochastic approximation method. *Annals of Mathematical Statistics*, 22(3), 400–407. 10.1214/aoms/1177729586

Salimans, T., Kingma, D., & Welling, M. (2015). Markov chain Monte Carlo and variational inference: Bridging the gap. In *Proceedings of the International Conference on Machine Learning* (pp. 1218–1226).
Salimans, T., & Knowles, D. A. (2014). On using control variates with stochastic approximation for variational Bayes and its connection to stochastic linear regression. arXiv:1401.1022.

Saul, L. K., Jaakkola, T., & Jordan, M. I. (1996). Mean field theory for sigmoid belief networks. *Journal of Artificial Intelligence Research*, 4, 61–76. 10.1613/jair.251

Tan, L. S. (2021). Natural gradient updates for Cholesky factor in gaussian and structured variational inference. arXiv:2109.00375.

Teräsvirta, T. (2009). An introduction to univariate Garch models. In T. Mikosch, J.-P. Kreiss, R. A. Davis, & T. G. Andersen (Eds.), *Handbook of financial time series* (pp. 17–42). Springer.

Tran, D. T., Iosifidis, A., Kanniainen, J., & Gabbouj, M. (2019). Temporal attention-augmented bilinear network for financial time-series data analysis. *IEEE Transactions on Neural Networks and Learning Systems*, 30(5), 1407–1418. 10.1109/TNNLS.2018.2869225

Tran, M.-N., Nguyen, T.-N., & Dao, V.-H. (2021). A practical tutorial on variational Bayes. arXiv:2103.01327.

Tran, M.-N., Nguyen, D. H., & Nguyen, D. (2021). Variational Bayes on manifolds. *Statistics and Computing*, 31(6), 71. 10.1007/s11222-021-10047-1

Trusheim, F., Condurache, A., & Mertins, A. (2018). Boosting black-box variational inference by incorporating the natural gradient. In *Proceedings of the 24th International Conference on Pattern Recognition* (pp. 19–24).

Wainwright, M. J., & Jordan, M. I. (2008). Graphical models, exponential families, and variational inference. *Foundations and Trends in Machine Learning*, 1(1–2), 1–305. 10.1561/220000001

Wierstra, D., Schaul, T., Glasmachers, T., Sun, Y., Peters, J., & Schmidhuber, J. (2014). Natural evolution strategies. *Journal of Machine Learning Research*, 15(1), 949–980.

Xu, M., Quiroz, M., Kohn, R., & Sisson, S. A. (2019). Variance reduction properties of the reparameterization trick. In *Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics* (pp. 2711–2720).

Zhang, G., Sun, S., Duvenaud, D., & Grosse, R. (2018). Noisy natural gradient as variational inference. In *Proceedings of the International Conference on Machine Learning* (pp. 5852–5861).

Zhang, L., Mykland, P. A., & Aït-Sahalia, Y. (2005). A tale of two time scales: Determining integrated volatility with noisy high-frequency data. *Journal of the American Statistical Association*, 100(472), 1394–1411. 10.1198/01621450500000169

Received August 23, 2023; accepted April 16, 2024.