Stat Comput (2018) 28:971–988
https://doi.org/10.1007/s11222-017-9773-3

Variational Bayes with synthetic likelihood

Victor M. H. Ong[1] · **David J. Nott**[1] · **Minh-Ngoc Tran**[2] · **Scott A. Sisson**[3] · **Christopher C. Drovandi**[4]

Received: 12 August 2016 / Accepted: 21 August 2017 / Published online: 31 August 2017

Abstract Synthetic likelihood is an attractive approach to likelihood-free inference when an approximately Gaussian summary statistic for the data, informative for inference about the parameters, is available. The synthetic likelihood method derives an approximate likelihood function from a plug-in normal density estimate for the summary statistic, with plug-in mean and covariance matrix obtained by Monte Carlo simulation from the model. In this article, we develop alternatives to Markov chain Monte Carlo implementations of Bayesian synthetic likelihoods with reduced computational overheads. Our approach uses stochastic gradient variational inference methods for posterior approximation in the synthetic likelihood context, employing unbiased estimates of the log likelihood. We compare the new method with a related likelihood-free variational inference technique in the literature, while at the same time improving the implementation of that approach in a number of ways. These new algorithms are feasible to implement in situations which are challenging for conventional approximate Bayesian computation methods, in terms of the dimensionality of the parameter and summary statistic.

Keywords Approximate Bayesian computation · Stochastic gradient ascent · Synthetic likelihoods · Variational Bayes

✉ Victor M. H. Ong
g0900757@u.nus.edu

David J. Nott
standj@nus.edu.sg

Minh-Ngoc Tran
M.Tran@econ.usyd.edu.au

Scott A. Sisson
scott.sisson@unsw.edu.au

Christopher C. Drovandi
c.drovandi@qut.edu.au

[1] Department of Statistics and Applied Probability, National University of Singapore, Singapore, Singapore

[2] Discipline of Business Analytics, The University of Sydney Business School, The University of Sydney, Sydney, Australia

[3] School of Mathematics and Statistics, University of New South Wales, Sydney 2052, Australia

[4] School of Mathematical Sciences, Queensland University of Technology, Brisbane 4000, Australia

1 Introduction

Synthetic likelihood (Wood 2010; Fasiolo et al. 2016a) is an attractive approach to likelihood-free inference in situations where an approximately Gaussian summary statistic for the data, informative about the parameters, is available. As explained in Price et al. (2016), the use of synthetic likelihood mitigates to some extent the curse of dimensionality associated with conventional approximate Bayesian computation (ABC) methods, and it is also convenient to apply with algorithmic parameters that are easy to tune. In this article we develop alternatives to Markov chain Monte Carlo (MCMC) implementations of Bayesian synthetic likelihoods, with reduced computational overheads. In particular, using unbiased estimates of the log likelihood, we implement stochastic gradient variational inference methods for posterior approximation that are more tolerant of noise in the likelihood estimate used. The main contributions of this work are: (1) to improve on the variational Bayes with intractable likelihood (VBIL) methodology of Tran et al. (2015) by considering certain reduced variance gradient estimates, adaptive learning rates and alternative parametrisations; (2) to modify the VBIL methodology to work with unbiased log likelihood estimates in the synthetic likelihood framework; and (3) to compare variational Bayes synthetic likelihood (VBSL) with

pseudo-marginal MCMC synthetic likelihood implementations (Price et al. 2016) and VBIL in a number of examples. The new methods introduced are feasible to implement in situations which are challenging for conventional ABC methods in terms of the dimensionality of both the parameter and summary statistic.

Suppose we have data y, a parameter θ of dimension p, a likelihood $p(y|\theta)$ which is computationally intractable, and a summary statistic $S = S(y)$ of dimension $d \geq p$ which is assumed to be approximately Gaussian conditional on each value of θ. Inference is to be based on the observed value s of the summary statistic, which is thought to be informative about θ. The likelihood for the summary statistic, if this statistic is assumed to be exactly Gaussian, is $\phi(s; \mu(\theta), \Sigma(\theta))$ where $\phi(z; \mu, \Sigma)$ is the multivariate normal density with mean vector μ and covariance matrix Σ, and where $\mu(\theta) = E(S|\theta)$ and $\Sigma(\theta) = \mathrm{Cov}(S|\theta)$. In general, however, $\mu(\theta)$ and $\Sigma(\theta)$ will be unknown. Synthetic likelihood (Wood 2010) replaces $\mu(\theta)$ and $\Sigma(\theta)$ by estimates obtained by simulation. For a given θ we may simulate summary statistics $S_1, \ldots, S_N$ under the model given θ, calculate

$$\hat{\mu}(\theta) = \frac{1}{N}\sum_{i=1}^{N} S_i,$$

$$\hat{\Sigma}(\theta) = \frac{1}{N-1}\sum_{i=1}^{N}(S_i - \hat{\mu}(\theta))(S_i - \hat{\mu}(\theta))^\top,$$

and approximate $\phi(s; \mu(\theta), \Sigma(\theta))$ by

$$\hat{p}_N(s|\theta) = \phi(s; \hat{\mu}(\theta), \hat{\Sigma}(\theta)). \tag{1}$$

As $N \to \infty$, $\hat{p}_N(s|\theta)$ will converge to $\phi(s; \mu(\theta), \Sigma(\theta))$ pointwise for each value of θ. Synthetic likelihood has been used in a wide range of challenging problems in ecology, epidemiology, forestry and social network modelling among other areas (Brown et al. 2014; Hartig et al. 2014; Fasiolo et al. 2016a; Everitt et al. 2017). It is perhaps most useful in problems where the summary statistic dimension is high, which can make traditional ABC approaches difficult to apply.

In many applications of synthetic likelihood, users choose N to be very large so that the effects of estimating $\mu(\theta)$ and $\Sigma(\theta)$ can be safely ignored. However, choosing N large incurs a high computational cost for each synthetic likelihood evaluation. One way to circumvent this difficulty is to somehow emulate the synthetic likelihood, and this has been considered by a number of authors using a variety of techniques (Meeds et al. 2014; Moores et al. 2015; Wilkinson 2014; Gutmann and Corander 2015). Recently, Price et al. (2016) considered a variation of synthetic likelihood which they call unbiased synthetic likelihood (uSL). In this approach (1) is replaced by a likelihood approximation obtained from an unbiased estimate of a normal density function due to Ghurye and Olkin (1969). Using similar notation to Ghurye and Olkin (1969) let

$$c(k, \nu) = \frac{(2\pi)^{-k\nu/2}\pi^{-k(k-1)/4}}{\prod_{i=1}^{k}\Gamma\left(\frac{1}{2}(\nu - i + 1)\right)},$$

and for a square matrix A write $\psi(A) = |A|$ if $A > 0$ and 0 otherwise, where $|A|$ is the determinant of A and $A > 0$ means that A is positive definite. Then in uSL (1) is replaced by

$$\hat{p}_N^U(s|\theta) = (2\pi)^{-\frac{d}{2}}\frac{c(d, N-2)}{c(d, N-1)(1-1/N)^{d/2}} \times |S_\theta|^{-\frac{N-d-2}{2}}\psi\left(S_\theta - \frac{(s - \hat{\mu}(\theta))(s - \hat{\mu}(\theta))^\top}{(1-1/N)}\right)^{\frac{N-d-3}{2}}, \tag{2}$$

where $S_\theta = (N-1)\hat{\Sigma}(\theta)$. The results of Ghurye and Olkin (1969) imply that $E(\hat{p}_N^U(s|\theta)) = \phi(s; \mu(\theta), \Sigma(\theta))$ if the summary statistic is Gaussian, provided that $N > d+3$. This unbiasedness property means that if (2) is used in a pseudo-marginal MCMC algorithm (Beaumont 2003; Andrieu and Roberts 2009) and if S is actually normally distributed, then the Markov chain converges to the exact posterior regardless of the value of N. However, even though the distribution targeted by such a pseudo-marginal algorithm does not depend on N, the mixing of the algorithm can be very poor unless N is chosen large enough to control the variance of the likelihood estimate. Doucet et al. (2015) suggest fixing the variance of the log likelihood estimate to be around 1 for pseudo-marginal Metropolis–Hastings algorithms, to achieve an optimal trade off between computational cost and precision.

An alternative approach to MCMC methods for Bayesian computation is variational approximation [see, for example, Bishop (2006) and Ormerod and Wand (2010)]. Although variational approximation is an approximate inference method, it can often be implemented with an order of magnitude less computational effort than the corresponding "exact" algorithms such as MCMC. Recently, Tran et al. (2015) considered the use of stochastic gradient variational inference when the likelihood is computationally intractable, and only an unbiased estimate of the likelihood is available. This includes situations where conventional ABC methods (Marin et al. 2012; Blum et al. 2013) are usually applied. In standard ABC, a nonparametric approximation to the likelihood is used. With $K_\epsilon(\cdot, \cdot)$ a kernel function in which $\epsilon > 0$ is a bandwidth parameter, ABC considers the likelihood approximation

$$\tilde{p}(s|\theta) = \int K_\epsilon(s, S(y'))p(y'|\theta)\mathrm{d}y', \tag{3}$$

which is estimated unbiasedly by

$$\hat{p}(s|\theta) = \frac{1}{N}\sum_{i=1}^{N} K_\epsilon(s, S(y_i')), \tag{4}$$

where $y_1', \ldots, y_N'$ are iid draws from $p(y|\theta)$.

In principle, we can use the estimate (2) to give a synthetic likelihood version of the VBIL method of Tran et al. (2015)—this is discussed further in Sect. 3. This may be beneficial compared to unbiased estimation of (3), since the parametric assumptions made in the synthetic likelihood mean that the synthetic likelihood can be estimated more precisely for a given number of model simulations, N, than the corresponding ABC likelihood. However, for implementing stochastic gradient variational Bayes (VB) methods, it is much more convenient to work with unbiased estimates of the log-likelihood function (see Sect. 4.1). Unbiased estimation of the log likelihood corresponding to (3) cannot be achieved directly. Furthermore, the VBIL method using an unbiased likelihood estimate is not easy to apply in some ABC problems, as the user needs to tune the variance of the log likelihood estimator to be constant across the parameter space—see Sect. 3 for further details. However, stochastic gradient VB methods, which use unbiased estimates of a log likelihood, have no such requirement. Unbiased estimators of the log of a normal density function are available from the pattern recognition literature (Ripley 1996, p. 56). Hence, assuming that the summary statistic is Gaussian, unbiased estimates of the log likelihood are available in the synthetic likelihood context. This makes the implementation of stochastic gradient VB methods very easy.

The next section reviews stochastic gradient VB methods, and Sect. 3 explains the VBIL method of Tran et al. (2015). Our VBSL algorithm is described in Sect. 4, as well as some refinements of the basic stochastic gradient optimisation approach that apply both to VBIL and VBSL. Section 5 compares VBSL with VBIL and pseudo-marginal synthetic likelihood approaches in some challenging examples. We conclude with a discussion.

2 Stochastic gradient variational Bayes

Consider a Bayesian inference problem with data y, a p-dimensional parameter θ, prior distribution $p(\theta)$ and likelihood function $p(y|\theta)$, so that the posterior density is $p(\theta|y) \propto p(\theta)p(y|\theta)$. In variational inference, the posterior density is approximated by a density within some tractable family. Here, we consider a parametric family with typical element $q_\lambda(\theta)$, where λ is a variational parameter to be chosen. The Kullback–Leibler divergence from $q_\lambda(\theta)$ to $p(\theta|y)$ is given by

$$\mathrm{KL}(\lambda) = \mathrm{KL}(q_\lambda(\theta)||p(\theta|y)) = \int \log\frac{q_\lambda(\theta)}{p(\theta|y)} q_\lambda(\theta)\mathrm{d}\theta. \tag{5}$$

Denote the marginal likelihood by $p(y) = \int p(\theta)p(y|\theta)\mathrm{d}\theta$. Minimising $\mathrm{KL}(\lambda)$ with respect to λ is equivalent to maximising

$$\mathcal{L}(\lambda) = \int \log\frac{p(\theta)p(y|\theta)}{q_\lambda(\theta)} q_\lambda(\theta)\mathrm{d}\theta,$$

and it can be shown that $\mathcal{L}(\lambda)$ is a lower bound on the log marginal likelihood $\log p(y)$. For introductory discussion of VB methods, see e.g. Bishop (2006) and Ormerod and Wand (2010). In non-conjugate settings $\mathcal{L}(\lambda)$ may not be directly computable. In this setting, stochastic gradient methods (Robbins and Monro 1951; Bottou 2010) have been developed which can optimise $\mathcal{L}(\lambda)$ effectively even when it cannot be calculated analytically, provided simulation from $q_\lambda(\theta)$ is possible (Ji et al. 2010; Nott et al. 2012; Paisley et al. 2012; Salimans and Knowles 2013; Kingma and Welling 2013; Hoffman et al. 2013; Rezende et al. 2014; Titsias and Lázaro-Gredilla 2015).

The most general approaches to using stochastic gradient methods in VB have been based on the "log derivative trick". Observe that

$$\nabla_\lambda q_\lambda(\theta) = q_\lambda(\theta)\nabla_\lambda \log q_\lambda(\theta),$$

and that $E(\nabla_\lambda \log q_\lambda(\theta)) = 0$ (where the expectation is with respect to $q_\lambda(\theta)$). This last identity follows from differentiating both sides of the equation $\int q_\lambda(\theta)\mathrm{d}\theta = 1$ with respect to λ. Writing $h(\theta) = p(\theta)p(y|\theta)$, then

$$\begin{aligned}
\nabla_\lambda \mathcal{L}(\lambda) &= \nabla_\lambda \int \{\log h(\theta) - \log q_\lambda(\theta)\} q_\lambda(\theta)\mathrm{d}\theta \\
&= \int \log h(\theta)\nabla_\lambda \log q_\lambda(\theta) q_\lambda(\theta)\mathrm{d}\theta \\
&\quad - \int \log q_\lambda(\theta)\nabla_\lambda \log q_\lambda(\theta) q_\lambda(\theta)\mathrm{d}\theta \\
&= \int \nabla_\lambda \log q_\lambda(\theta)\{\log h(\theta) - \log q_\lambda(\theta)\} q_\lambda(\theta)\mathrm{d}\theta.
\end{aligned} \tag{6}$$

The last expression is an expectation with respect to $q_\lambda(\theta)$, which is easily estimated unbiasedly if we can simulate from $q_\lambda(\theta)$. This then permits implementation of a stochastic gradient algorithm for optimising $\mathcal{L}(\lambda)$. In the original lower bound expression, some terms (e.g. $E(\log q_\lambda(\theta))$) can sometimes be calculated analytically, in which case the estimate (6) can be modified appropriately, although this may not always be beneficial (Salimans and Knowles 2013). It is well known that gradient estimates obtained by the log derivative trick are highly variable, and a variety of additional methods for variance reduction have also been considered in the above

references. Titsias and Lázaro-Gredilla (2015) recently considered an interesting approach that can be implemented in a model independent fashion.

For large datasets, it is convenient to replace the log-likelihood term in $\log h(\theta)$ by an unbiased estimate—this still results in an unbiased estimate of the gradient of $\mathcal{L}(\lambda)$. Such estimates of the log-likelihood are usually obtained by subsampling. Variational schemes that use both subsampling and sampling from the variational posterior to generate gradient estimates have been termed "doubly stochastic" by Titsias and Lázaro-Gredilla (2014) (see also Kingma and Welling (2013) and Salimans and Knowles (2013) for similar approaches). The variational Bayes with intractable log-likelihood (VBILL) methodology of Gunawan et al. (2016) considers unbiased estimation of log likelihoods within stochastic gradient variational inference using difference estimators for variance reduction.

3 Variational Bayes with intractable likelihood (VBIL)

We now describe the VBIL method of Tran et al. (2015) since we build on this approach in Sect. 4. VBIL is the first attempt to apply stochastic gradient variational inference methods to a class of problems that includes likelihood-free inference, and uses black box variational inference methods (Ranganath et al. 2014). However, a related expectation propagation approach to likelihood-free inference has been considered previously by Barthelmé and Chopin (2014). More recently Moreno et al. (2016) have considered an automatic variational ABC approach based on stochastic gradient VB with attractive methods for gradient estimation, which apply when the forward simulation model can be written as a differentiable function of both model parameters and random variables, and when the model code is written in an automatic differentiation environment.

The VBIL approach works with an unbiased estimate of the likelihood which we denote by $\hat{p}_N(y|\theta)$. Here, N is an algorithmic parameter controlling the accuracy of the approximation, such as the number of Monte Carlo samples used. Following Pitt et al. (2012) and Tran et al. (2015), we refer to N as the number of particles. Write $z = \log \hat{p}_N(y|\theta) - \log p(y|\theta)$, and $g_N(z|\theta)$ for the distribution of z given θ. Since $\hat{p}_N(y|\theta)$ is unbiased, we must have

$$\int \exp(z) g_N(z|\theta) \mathrm{d}\theta = 1. \tag{7}$$

Tran et al. (2015) consider implementing VB in the augmented space (θ, z), inspired by similar ideas in the literature on pseudo-marginal MCMC algorithms (Beaumont 2003; Andrieu and Roberts 2009), and in particular, consider the target distribution

$$p_N(\theta, z) = p(\theta|y) \exp(z) g_N(z|\theta).$$

Using (7), we see that the θ marginal of $p_N(\theta, z)$ is the posterior distribution of interest, $p(\theta|y)$. Consider a family of approximating distributions of the form

$$q_\lambda(\theta, z) = q_\lambda(\theta) g_N(z|\theta),$$

where λ is a variational parameter to be chosen. The θ marginal of $q_\lambda(\theta, z)$ is $q_\lambda(\theta)$. Performing the VB optimisation in the augmented space, by choosing λ to minimise $\mathrm{KL}(q_\lambda(\theta, z)||p_N(\theta, z))$, then the gradient of the objective function can be shown to be

$$E(\nabla_\lambda \log q_\lambda(\theta)(\log(p(\theta)\hat{p}_N(y|\theta)) - \log q_\lambda(\theta))), \tag{8}$$

where the expectation is with respect to $q_\lambda(\theta, z)$. The expression in (8) is easily obtained from (6) and is easily approximated by simulation, since all that is required is simulation of θ from $q_\lambda(\theta)$ and calculation of the likelihood estimate $\hat{p}_N(y|\theta)$. Knowledge of z, which depends on the unknown $p(y|\theta)$, is not required.

Minimisation of $\mathrm{KL}(q_\lambda(\theta, z)||p_N(\theta, z))$ is not the same in general as minimisation of $\mathrm{KL}(\lambda)$ given by (5). However, Tran et al. (2015) show that if (a) there is a function $\gamma^2(\theta) > 0$ such that $E(z|\theta) = -\gamma^2(\theta)/(2N)$ and $\mathrm{Var}(z|\theta) = \gamma^2(\theta)/N$, and (b) for a given $\sigma^2 > 0$, N can be chosen as a function of θ and σ^2 so that $\mathrm{Var}(z|\theta) \equiv \sigma^2$, then the minimisers of $\mathrm{KL}(q_\lambda(\theta, z)||p_N(\theta, z))$ and $\mathrm{KL}(\lambda)$ correspond. The lower bound in the augmented space is

$$\begin{aligned}\mathcal{L}_a(\lambda) &= \int \log \frac{p(\theta)p(y|\theta)\exp(z)g_N(z|\theta)}{q_\lambda(\theta)g_N(z|\theta)} q_\lambda(\theta, z)\mathrm{d}z\mathrm{d}\theta \\ &= \mathcal{L}(\lambda) + \int z g_N(z|\theta) q_\lambda(\theta)\mathrm{d}\theta,\end{aligned}$$

which is $\mathcal{L}(\lambda)$ plus a constant which is independent of λ if N has been tuned so that $E(z|\theta)$ does not depend on θ. If the log-likelihood estimator is asymptotically normal, so that z is normal, this implies that asymptotically $z|\theta \sim N(E(z|\theta), -2E(z|\theta))$ by the unbiasedness condition. Hence, tuning $E(z|\theta)$ to not depend on θ is equivalent to tuning the variance of the log-likelihood estimator to not depend on θ in this case. The resulting lower bound in the augmented space is

$$\mathcal{L}_a(\lambda) = \mathcal{L}(\lambda) - \frac{\tau^2}{2}, \tag{9}$$

Initialise $\lambda^{(0)} = (\lambda_1^{(0)}, \lambda_2^{(0)})$, $t = 0$, $\lambda^{(1)} = \lambda^{(0)}$. N is the number of particles, S the number of θ samples used in the gradient estimates.

1. (a) Generate $(\theta^{(t)}, z_s^{(t)}) \sim q_{\lambda^{(t)},N}(\theta, z)$, $s = 1, \ldots, S$. Note that the $z_s^{(t)}$ can be generated only implicitly through computation of estimates $\hat{p}_N^S(y|\theta^{(t)})$, $s = 1, \ldots, S$.

   (b) Set

   $$c^{(t)} = \frac{\mathrm{Cov}(\hat{h}(\theta, z)\nabla_\lambda \log q_\lambda(\theta), \nabla_\lambda \log q_\lambda(\theta))}{\mathrm{Var}(\nabla_\lambda \log q_\lambda(\theta))}$$

   where $\mathrm{Cov}(\cdot)$ and $\mathrm{Var}(\cdot)$ are sample estimates of covariance and variance based on the samples $(\theta_s^{(t)}, z_s^{(t)})$, $s = 1, \ldots, S$, and $\hat{h}(\theta, z) = \log p(\theta)\hat{p}_N(y|\theta)$.

   (c) $t = t + 1$.

2. Repeat

   (a) Generate $(\theta_s^{(t)}, z_s^{(t)}) \sim q_{\lambda^{(t)},N}(\theta, z)$, $s = 1, \ldots, S$.

   (b) $\hat{H}^{(t)} = \frac{1}{S}\sum_{s=1}^S (\hat{h}(\theta_s^{(t)}, z_s^{(t)}) - \log q_\lambda(\theta_s^{(t)}) - c^{(t-1)})\nabla_\lambda \log q_\lambda(\theta_s^{(t)})$.

   (c) Estimate $c^{(t)}$ as in step 1 (b).

   (d) $\tilde{\lambda}^{(t+1)} = \lambda^{(t)} + \rho_t I_F(\lambda^{(t)})^{-1}\hat{H}^{(t)}$

   (e) If $\Sigma(\tilde{\lambda}^{(t+1)})$ is not positive definite $\lambda^{(t+1)} = \lambda^{(t)}$ else $\lambda^{(t+1)} = \tilde{\lambda}^{(t+1)}$.

   (f) Set $\mathrm{LB}^{(t)} = \left\{\frac{1}{S}\sum_{s=1}^S \hat{h}(\theta_s^{(t)}, z_s^{(t)}) - \log q_{\lambda^{(t)}}(\theta_s^{(t)})\right\}$.

   (g) $t = t + 1$

until some stopping rule is satisfied.

Algorithm 1: VBIL algorithm with Gaussian variational posterior distribution. Further details of the parametrisation of the variational distribution and computation of $I_F(\lambda)$ are provided in the "Appendix".

where τ^2 is the targeted variance for the log-likelihood estimator. Tran et al. (2015) show that this approach is more tolerant of noise in the likelihood estimate than pseudo-marginal MCMC algorithms which use similar unbiased estimates of the likelihood.

The VBIL method of Tran et al. (2015) is useful in a number of settings, such as state space models and random effects models, where it is convenient to obtain unbiased estimates of the likelihood. It is also useful for ABC since it is trivial to estimate (3) unbiasedly. Crucial to the VBIL method is the use of variance reduction methods in the gradient estimates in the stochastic gradient procedure. In this article, we consider only multivariate normal approximations to the posterior; exploiting the fact that such approximations are in the exponential family allows the use of natural gradient methods (Amari 1998) as described in Tran et al. (2015). Using these ideas as well as the control variates approach to variance reduction described in Tran et al. (2015) results in Algorithm 1. Further justifications for the details of the algorithm are given in Sect. 3 of Tran et al. (2016). In Algorithm 1, λ denotes the natural parameters in the normal variational posterior distribution $q_\lambda(\theta)$ and $I_F(\lambda) = \mathrm{Cov}(\nabla_\lambda \log q_\lambda(\theta))$. Details of the parametrisation and form of $I_F(\lambda)$ are given in "Appendix A". In Algorithm 1, we also write n for a sample size parameter that scales the lower bound, and S is the number of samples used in the gradient estimate. Finally, ρ_t, $t \geq 0$, is a learning rate sequence satisfying the Robbins–Monro conditions $\sum_t \rho_t = \infty$, $\sum_t \rho_t^2 < \infty$ (Robbins and Monro 1951).

We note that there are two differences between Algorithm 1 based on Tran et al. (2016), and the earlier approach described in Tran et al. (2015). Firstly, it is suggested in Tran et al. (2016) that the values $\theta^{(s)}$, $s = 1, \ldots, S$ in step 1 can be generated using randomised quasi Monte Carlo, and this can be helpful for reducing the variance of the gradient estimates in some problems. Secondly, Algorithm 1 follows Tran et al. (2016) in estimating all parts of the lower bound expression using Monte Carlo with the same θ samples to reduce variance of gradient estimates, rather than calculating certain parts of the lower bound analytically (see Tran et al. (2016) for further discussion).

In Algorithm 1, N is treated as fixed. However, we would like N to be chosen adaptively so that the variance of the log-likelihood estimator is approximately constant with θ (or at least approximately constant over the high posterior probability region). Hence, in practice, we adapt N by first setting some minimum value N' for the number of simulations in the likelihood estimation. Then, if some target value for the log-likelihood variance is exceeded based on an empirical estimate, an additional number of particles (50, say) is

repeatedly simulated, until the target accuracy is achieved. This adaptive procedure does not bias the likelihood estimate obtained.

4 Variational Bayes synthetic likelihood (VBSL)

We now consider some extensions of Algorithm 1—in particular, we incorporate the use of the synthetic likelihood, resulting in the VBSL algorithm. Additionally, we develop an adaptive method for determining the algorithm learning rates, and reparametrisations that may be helpful in cases where ensuring the positive definiteness of the variational posterior covariance matrix is difficult. Our parametrisation of the Gaussian variational posterior covariance uses the Cholesky factor of the precision matrix. While Cholesky factor parametrisations are also used by other authors in the literature on Gaussian variational approximation (Titsias and Lázaro-Gredilla 2014; Kucukelbir et al. 2017; Tan and Nott 2017) unlike these authors we implement a natural gradient algorithm and the faster convergence this brings can be particularly valuable in the likelihood-free setting with expensive simulation models. Most adaptive learning rates used in the literature are not designed for use in conjunction with the natural gradient, and we adapt a method of Ranganath et al. (2013) to the current setting.

4.1 Unbiased synthetic log-likelihood estimation

Following Ripley (1996, p. 56), when the summary statistics are normally distributed, an unbiased estimate of the log of a normal density $\log \phi(s; \mu(\theta), \Sigma(\theta))$ based on a random sample of size N from it leading to sample mean and covariance matrix $\hat{\mu}(\theta)$ and $\hat{\Sigma}(\theta)$, respectively, is

$$\begin{aligned}\hat{l}_N^U(s|\theta) = &-\frac{d}{2}\log 2\pi \\ &-\frac{1}{2}\left\{\log|\hat{\Sigma}(\theta)| + d\log\left(\frac{N-1}{2}\right) - \sum_{i=1}^{d}\psi\left(\frac{N-i}{2}\right)\right\} \\ &-\frac{1}{2}\left\{\frac{N-d-2}{N-1}(s-\hat{\mu}(\theta))^{\mathrm{T}}\hat{\Sigma}(\theta)^{-1}(s-\hat{\mu}(\theta)) - \frac{d}{N}\right\},\end{aligned} \tag{10}$$

provided that $N > d + 2$, where $\psi(\cdot)$ denotes the digamma function. Hence, although unbiased estimation of the logarithm of (3) for the nonparametric ABC likelihood approximation cannot be achieved directly, in the context of synthetic likelihood, where the summary statistic is assumed to follow a Gaussian distribution, it is straightforward to use (10) as an unbiased estimate of the log likelihood. To implement a stochastic gradient VB algorithm for approximation of the posterior, the only change required in Algorithm 1 is to replace $\log \hat{p}_N(y|\theta)$ wherever it appears by the expression (10) above.

However, note that the previous requirements for minimisation of $\mathrm{KL}(q_\lambda(\theta, z)||p_N(\theta, z))$ correspond to minimisation of $\mathrm{KL}(\lambda)$ in VBIL can now be dropped—it is no longer necessary to tune N as a function of θ so that the variance of the log-likelihood estimator is approximately constant. In addition, the parametric assumptions used in the synthetic likelihood enable us to both reduce the variance of the log likelihood estimator for a given number of simulations, and also that of the stochastic gradients in Algorithm 1 and our refinements.

In many situations, the assumptions made in the synthetic likelihood are reasonable—the statistics can often be chosen, perhaps after transformation, so that they satisfy some central limit theorem (Wood 2010). For example, if we use an indirect inference approach to summary statistic choice (Drovandi et al. 2011), then maximum likelihood estimators of parameters in misspecified models are asymptotically normal under fairly general conditions. Price et al. (2016) find that the Bayesian synthetic likelihood posterior generally seems to be not very sensitive to violations of the Gaussian assumption, but there is also recent work on relaxing the Gaussian assumption in synthetic likelihood in a variety of ways (Fasiolo et al. 2016b; Dutta et al. 2016). It is also often possible to check the reasonableness of the final model and posterior inference through Bayesian model checking (Box 1980; Gelman et al. 1996), and we implement this in a later real example. The synthetic likelihood approach may be particularly helpful for large datasets where the forward model simulations are expensive. For large datasets, the normal variational posterior approximation will often be very reasonable, as well as the normal distributional assumption of the summary statistics. The VBSL approach can work very efficiently in this situation without much loss of accuracy.

Perhaps the most important advantage of the VBSL algorithm, however, is that its tuning parameters are much easier to set than for VBIL. In particular, for VBIL the ABC tolerance ϵ must be chosen beforehand, and in general the accuracy of the approximation as well as the variance of the gradient estimates within the algorithm are very sensitive to this choice. Practically, as a result, multiple implementations of VBIL with different ϵ values will be required to establish a reasonable computation time and accuracy trade off. The analogous parameter in the VBSL algorithm is N, the number of Monte Carlo samples used in the empirical estimation of the mean and covariance matrix of the summary statistics. If the summary statistic is exactly Gaussian distributed, the solution to the variational optimisation problem does not depend on N, and in practice, if the distribution is close to Gaussian there is very little sensitivity to this choice.

4.2 Adaptive learning rate

A second refinement of Algorithm 1 applicable to both VBSL and VBIL is to use an adaptive learning rate. In Tran et al. (2015) the learning rate ρ_t is chosen to be some sequence satisfying the Robbins-Monro conditions $\sum_t \rho_t = \infty$, $\sum_t \rho_t^2 < \infty$ where the sequence has a specified form with parameters that need to be manually tuned. However, suitable adaptive choices of the step sizes can avoid manual tuning, improve convergence and make algorithm stability and performance less sensitive to starting values. We propose an adaptive learning rate choice based on previous work by Ranganath et al. (2013) in the context of stochastic variational inference (SVI) (Hoffman et al. 2013). Similar to Algorithm 1, SVI is a stochastic natural gradient ascent algorithm, but one where the stochasticity of the gradient estimates derives from subsampling. The arguments provided by Ranganath et al. (2013) justifying their adaptive learning rate carry over to the current setting, where the stochasticity in the estimate of the natural gradient comes from sampling the variational distribution and from estimation of the log likelihood itself.

Let $\hat{n}_t$ be the natural gradient estimate for the lower bound at time t, $\hat{n}_t = I_F(\lambda^{(t)})^{-1}\hat{H}^{(t)}$. A running average of the values of $\hat{n}_t$ and $\hat{n}_t^\top \hat{n}_t$ can be maintained as

$$\bar{n}_t = (1-\alpha_t)\bar{n}_{t-1} + \alpha_t \hat{n}_t$$
$$\bar{c}_t = (1-\alpha_t)\bar{c}_{t-1} + \alpha_t \hat{n}_t^T \hat{n}_t,$$

where α_t is a discounting factor. The learning rate ρ_t is then given by

$$\rho_t = \frac{\bar{n}_t^\top \bar{n}_t}{\bar{c}_t},$$

with α_t also adapted as

$$\alpha_{t+1}^{-1} = \alpha_t^{-1}(1-\rho_t) + 1.$$

The initial values $\bar{n}_0$ and $\bar{c}_0$ are chosen based on computation of K independent gradient estimates at the starting value for the variational parameters, and α_0 is initialised as $1/K$. Intuition behind the choice of ρ_t is that $\bar{n}_t^\top \bar{n}_t$ represents the "signal" in the noisy gradient estimates, whereas $\bar{c}_t$ represents the extent of the total variation, including both signal and noise. So large steps will be taken when the magnitude of the gradient is large compared to the noise, whereas if the noise dominates the signal small steps are chosen. The adaptation of the discounting factors α_t is implemented in such a way that more weight is given to the current iteration following a big step. The rationale for the approach is based on minimising some loss function, which measures how well one step of the approach mimics the approach with noise free gradient (see Ranganath et al. (2013) for further discussion). However, we find that in some of our applications, using the proposed adaptive learning rate may still lead to instability at early iterations. We find it helpful to set a maximum step size in the early iterations, which in our examples we choose as $\rho_t \leq \sqrt{d/\bar{c}_t}$.

4.3 Cholesky parametrisation of the covariance matrix

Our final modification of Algorithm 1 is to parametrise the normal variational distribution in terms of the Cholesky factor of the precision matrix. Implementing natural gradient steps can still be performed conveniently for this parametrisation. In the natural parametrisation of the normal distribution used in Algorithm 1, it is possible for an update to result in a parameter value λ for which Σ is not positive definite. In Algorithm 1 such updates are rejected, however for high-dimensional problems and with poor choices of starting values or noisy gradients, such rejection steps may occur frequently resulting in slow convergence. Reparametrisation in terms of the Cholesky factor avoids this.

In describing the implementation of the Cholesky parameterisation, we require some notation, similar to that found in Magnus and Neudecker (1999) and Wand (2014). For a $d \times d$ matrix A, write $\mathrm{vec}(A)$ for the vector of length d^2 obtained by stacking the columns one underneath another moving from left to right. When A is symmetric, write $\mathrm{vech}(A)$ for the vector with $d(d+1)/2$ elements obtained by stacking the lower triangular elements of A.

We parametrise the normal variational posterior distribution in terms of the mean μ and the (lower triangular) Cholesky factor C of Σ^{-1} so that $\Sigma^{-1} = CC^\top$. We do not enforce the constraint that the diagonal elements of C be positive as such non-uniqueness is not a concern in the present context. Our variational parameters are now

$$\lambda = \begin{bmatrix} \mu \\ \mathrm{vech}(C) \end{bmatrix}. \tag{11}$$

We then have

$$\log q_\lambda(\theta) = -\frac{d}{2}\log 2\pi + \log|C| - \frac{1}{2}(\theta-\mu)^T CC^\top(\theta-\mu),$$

and upon differentiation with respect to μ and $\mathrm{vech}(C)$

$$\nabla_\lambda \log q_\lambda(\theta) = \begin{bmatrix} CC^\top(\theta-\mu) \\ \mathrm{vech}(\mathrm{diag}(1/C) - (\theta-\mu)(\theta-\mu)^\top C) \end{bmatrix},$$

where $\mathrm{diag}(1/C)$ denotes the diagonal matrix with the same dimensions as C with ith diagonal entry $1/C_{ii}$. This expression for $\nabla_\lambda \log q_\lambda(\theta)$ allows us to construct an unbiased gradient estimate from (6). However, Algorithm 1 uses the

natural gradient, and we would like to construct a natural gradient algorithm in the new parametrisation. To do this we need

$$I_F(\lambda) = \mathrm{Cov}_\lambda(\nabla_\lambda \log q_\lambda(\theta)).$$

Writing $I_F(\lambda)$ in block form, corresponding to the partition in (11), then

$$I_F(\lambda) = \begin{bmatrix} I_{11}(\lambda) & I_{21}(\lambda)^\top \\ I_{21}(\lambda) & I_{22}(\lambda) \end{bmatrix}.$$

Write L_d for the elimination matrix of order d (Magnus and Neudecker 1999) which for a (not necessarily symmetric) $d \times d$ matrix A, transforms $\mathrm{vec}(A)$ into $\mathrm{vech}(A)$, and write $\otimes$ for the Kronecker product. We also denote by D_d the duplication matrix of order d, which is the unique $d^2 \times d(d+1)/2$ matrix of zeros and ones such that $D_d \mathrm{vech}(A) = \mathrm{vec}(A)$ for symmetric $d \times d$ matrices A, and its Moore-Penrose inverse is written as $D_d^+ = (D_d^\top D_d)^{-1} D_d^\top$. Then, we get

$$\begin{aligned} I_{22}(\lambda) &= \mathrm{Cov}(\mathrm{vech}((\theta-\mu)(\theta-\mu)^\top C)) \\ &= \mathrm{Cov}(L_d \mathrm{vec}((\theta-\mu)(\theta-\mu)^\top C)) \\ &= \mathrm{Cov}(L_d(C^T \otimes I)\mathrm{vec}((\theta-\mu)(\theta-\mu)^\top)) \\ &= L_d(C^\top \otimes I)\mathrm{Cov}(\mathrm{vec}((\theta-\mu)(\theta-\mu)^T)(C \otimes I){L_d}^\top \\ &= L_d(C^\top \otimes I)D_d\mathrm{Cov}(\mathrm{vech}((\theta-\mu)(\theta-\mu)^T)) \\ &\quad D_d^T(C \otimes I){L_d}^\top \\ &= 2L_d(C^\top \otimes I)D_d D_d^+(\Sigma \otimes \Sigma){D_d^+}^T D_d^T(C \otimes I){L_d}^\top, \end{aligned}$$

where in the final line we have used the expression for $\mathrm{Cov}(\mathrm{vech}(xx^\top))$ for normal x derived in the proof of Theorem 1 c) of Wand (2014). Finally

$$I_{11}(\lambda) = \mathrm{Cov}(CC^\top(\theta-\mu)) = CC^\top \Sigma CC^\top = \Sigma^{-1},$$

and

$$\begin{aligned} I_{21}(\lambda) &= -\mathrm{Cov}(\mathrm{vech}((\theta-\mu)(\theta-\mu)^\top C), CC^\top(\theta-\mu)) \\ &= -L_d\mathrm{Cov}(\mathrm{vec}((\theta-\mu)(\theta-\mu)^T C), \theta-\mu)CC^\top \\ &= -L_d\mathrm{Cov}((C^\top \otimes I)\mathrm{vec}((\theta-\mu)(\theta-\mu)^T), \theta-\mu)CC^\top \\ &= -L_d(C^\top \otimes I)\mathrm{Cov}(\mathrm{vec}((\theta-\mu)(\theta-\mu)^T), \theta-\mu)CC^\top \\ &= 0, \end{aligned}$$

where in the last line we have used the fact that odd order central moments of the multivariate normal distribution are zero. That is, we can compute $I_F(\lambda)$ in the new parametrisation, allowing for a natural gradient implementation. In our application in Sect. 5.3, we directly compare the natural gradient approach with the use of the ordinary gradient in the Cholesky parametrisation, with a per parameter adaptive learning rate determined according to the ADADELTA approach of Zeiler (2012).

5 Applications

We investigate the performance of the VBSL approach using four different models. In the first experiment, we consider a toy example using data generated from a Gaussian distribution. This example permits direct comparison with the VBIL method, since the calculations can be performed analytically, and the effects of the finite ABC tolerance ϵ can be separated from the inaccuracy of the variational approximation itself in the VBIL algorithm. In the next two examples, we investigate α-stable and multivariate g-and-k models, which do not have closed form expressions for the density. The α-stable analysis is used to demonstrate the importance of adaptive learning rates, and the g-and-k analysis is used to compare our adaptive natural gradient optimisation scheme with a method based on the ordinary gradient and an adaptive per parameter learning rate [the ADADELTA method of Zeiler (2012)]. Since the multivariate g-and-k model possesses a fairly high-dimensional parameter, it gives some insight into how to implement the VBSL methodology in an efficient and stable way in this setting. Finally, our last example considers the case of a very high-dimensional summary statistic, using a real problem from cell biology.

5.1 Toy example: normal location model

We consider data, $y_1, \ldots, y_n$, from a Gaussian distribution with unknown mean θ and unit variance. We assume that the observed data is $y = (0, \ldots, 0)$ and adopt a standard normal distribution $N(0,1)$ for the prior on θ so that the posterior distribution is $\theta|y \sim N(n/(1+n)\bar{y}, 1/(1+n))$ where $\bar{y}$ denotes the sample mean. We ignore the fact that $\bar{y}$ is a sufficient statistic and take the entire data set y as the summary statistic. This allows us to explore the effect of increasing dimension of the summary statistic on the likelihood-free methods. For the VBIL approach, we use the ABC likelihood (4) with a Gaussian kernel defined as

$$K_\epsilon(s, s') = (2\pi\epsilon)^{-d/2} \exp\left\{-\frac{1}{2\epsilon}\left[(s-s')^\top(s-s')\right]\right\}. \tag{12}$$

With this kernel, the ABC likelihood (4) can be computed analytically, and the corresponding posterior distribution for θ is

$$p_{ABC,\epsilon}(\theta|y) = N\left(\frac{n/(1+\epsilon)}{1+n/(1+\epsilon)}\bar{y}, \frac{1}{1+n/(1+\epsilon)}\right). \tag{13}$$

Being able to compute the targeted posterior analytically for the VBIL approach is important. This is because the use of a finite ϵ inflates the targeted posterior variance compared to the truth, whereas the VB approximation can result in an error in the opposite direction (underestimation of variance will occur in this example if we have not perfectly tuned the variance of log likelihood estimates to be constant across the parameter space). So apparent good performance of VBIL can sometimes result simply from a fortuitous cancellation of these errors in different directions, so it is important to understand what distribution is being targeted by the VBIL algorithm.

We consider $d = n = 4, 8$ and set $S = 100$. For VBSL we fixed $N = 50$. For VBIL, we set the ABC tolerance parameter ϵ in (12) as 0.1282 and 0.1139 for $d = 4, 8$ respectively. These values are chosen to ensure that (13) only overestimates the true posterior standard deviation by 10%, which is a reasonable standard of accuracy. Of course, since the summary statistic is exactly Gaussian here the synthetic likelihood method is exact. We set the minimum value of N in VBIL to be 50, but implement the adaptive sample size approach described in Sect. 3 to tune N to target variances of $\log \hat{p}(y|\theta)$ of 0.1 and 0.5 and denote these two methods by $\text{VBIL}_{0.1}$ and $\text{VBIL}_{0.5}$, respectively. On average, for $d = 4$, we required approximately $N = 60$ and $N = 400$ simulations to achieve $\text{Var}(\log \hat{p}(y|\theta)) \leq 0.5$ and 0.1 respectively, and an average of $N = 250$ and $N = 6500$ simulations for $d = 8$.

Note that with these specifications one iteration of the VBSL algorithm takes either the same or less computational effort than the VBIL approaches, so that faster convergence of VBSL implies less computational effort overall. We set the learning rate $\rho_t = \frac{1}{5+t}$ where t is the iteration number; this form satisfies the Robbins–Monro conditions with constants hand tuned for good performance in the VBIL approaches. The effects of adaptive learning rates are investigated further in later examples. We initialise our starting point for $q(\theta)$ to be $N(\mu^{(0)}, \sigma^{(0)})$ where $\mu^{(0)}$ is the mean of the observed data and $\sigma^{(0)} = 1$. We fixed the number of iterations to 100.

In this example, the optimised variational lower bound value can be calculated analytically. In the case of VBIL, this assumes that it is properly tuned so that the variance of the log-likelihood estimate is constant. In particular, considering $1/n \log p(y) = (1/n) \log \int p(y|\theta) p(\theta) \mathrm{d}\theta$ and replacing $p(y|\theta)$ with the ABC or synthetic likelihood, the (scaled) lower bound is

$$\text{LB}_{\text{VBSL}} = -\frac{1}{2}\log(2\pi) - \frac{1}{2n}\sum_{i=1}^{n} y_i^2 - \frac{1}{2n}\log(n+1) + \frac{1}{2n(n+1)}\left(\sum_{i=1}^{n} y_i\right)^2,$$

for the VBSL approach and

$$\text{LB}_{\text{VBIL}} = -\frac{1}{2}\log(2\pi) - \frac{1}{2}\log(1+\epsilon) - \frac{1}{2n(1+\epsilon)}\sum_{i=1}^{n} y_i^2 - \frac{1}{2n}\log\left(\frac{n}{1+\epsilon}+1\right) + \frac{(n/(1+\epsilon))^2}{2n(1+n/(1+\epsilon))}\bar{y}^2 - \frac{\tau^2}{2n},$$

for the VBIL approach, where τ^2 is the (assumed constant) targeted variance for the log-likelihood estimate (for the VBIL method we have used (9) to derive this expression). How close we come to attaining these analytically calculated lower bound expressions is a measure of the accuracy of the algorithm taking into account the different likelihoods implicitly being used, and also helps assess convergence of the algorithm.

Figures 1 and 2 illustrate the variational distribution of θ and the realised lower bound using VBSL and VBIL. For both $d = 4$ and 8, the VBSL approach matches the true posterior distribution (represented by the black dotted curve) and attains its analytic lower bound (represented by the black horizontal line). For $\text{VBIL}_{0.1}$ and $\text{VBIL}_{0.5}$, their means match but variances differ slightly for $d = 4$ and $\text{VBIL}_{0.5}$ has not really converged within 100 iterations for $d = 8$. Unsurprisingly, the performance of the VBIL approach deteriorates when the dimension of the summary statistics increases. The estimated posterior distribution deviates from the true posterior distribution, by having a smaller variance, when we set $\text{Var}(\log \hat{p}(y|\theta)) \leq 0.5$ for $d = 8$. The performance greatly improves if we set $\text{Var}(\log \hat{p}(y|\theta)) \leq 0.1$. However, we observe this method requires $N = 6200$ simulations per likelihood estimate on average, which in turn would imply a much larger computational effort. In fact, we found that for $d = 8$, the synthetic likelihood with $N = 50$ and $\text{VBIL}_{0.1}$ require 3 and 13 min respectively for 100 iterations. This reflects the advantage of the parametric assumptions made in the synthetic likelihood and we are able to achieve reasonable answers for less computational effort.

5.2 α-stable model

We now examine the importance of adaptive learning rates within the VBSL algorithm. α-stable models (see, for example, Adler et al. (1998), Section VII) are a convenient family of heavy-tailed distributions used in a number of applications. Inference is challenging, since for distributions in this family there is no closed form expression for the density function. The most common parametrisation of these distributions is in terms of a parameter $\theta = (\alpha, \beta, \gamma, \delta)^\top$, where α is a parameter controlling tail behaviour, β controls skew-

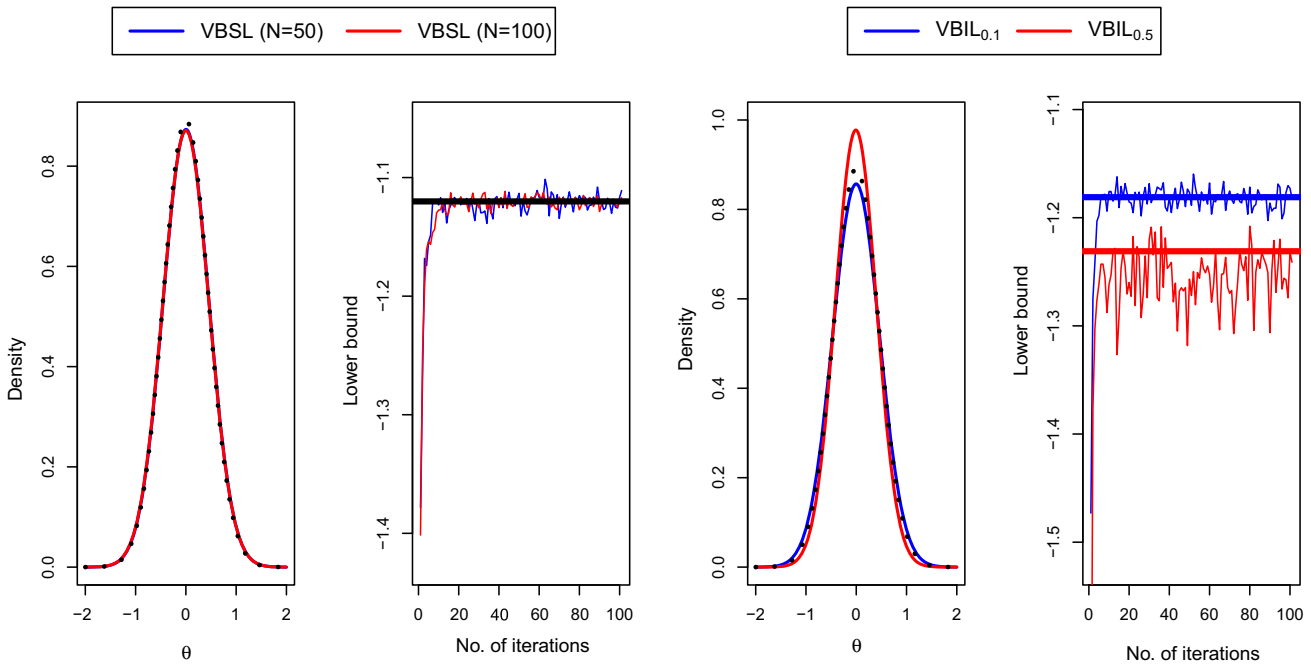

Fig. 1 Posterior distribution and variational lower bound for $d = 4$. The *black dotted line* on the density plot represents the true posterior distribution. The *horizontal black*, *blue* and *red lines* in the lower bound plot represent the analytically calculated lower bound for VBSL, $\text{VBIL}_{0.1}$ and $\text{VBIL}_{0.5}$ respectively. (Color figure online)

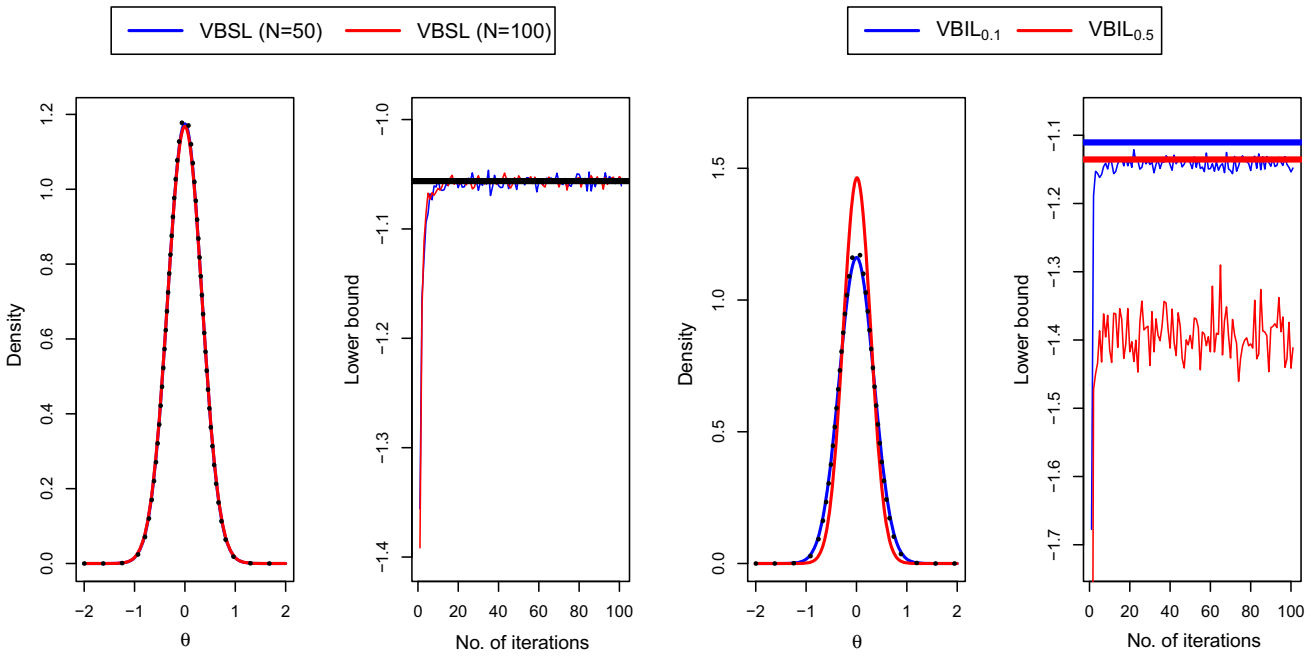

Fig. 2 Posterior distribution and variational lower bound for $d = 8$. The *black dotted line* on the density plot represents the true posterior distribution. The *horizontal black*, *blue* and *red lines* in the lower bound plot represents the analytically calculated lower bound for VBSL, $\text{VBIL}_{0.1}$ and $\text{VBIL}_{0.5}$ respectively. (Color figure online)

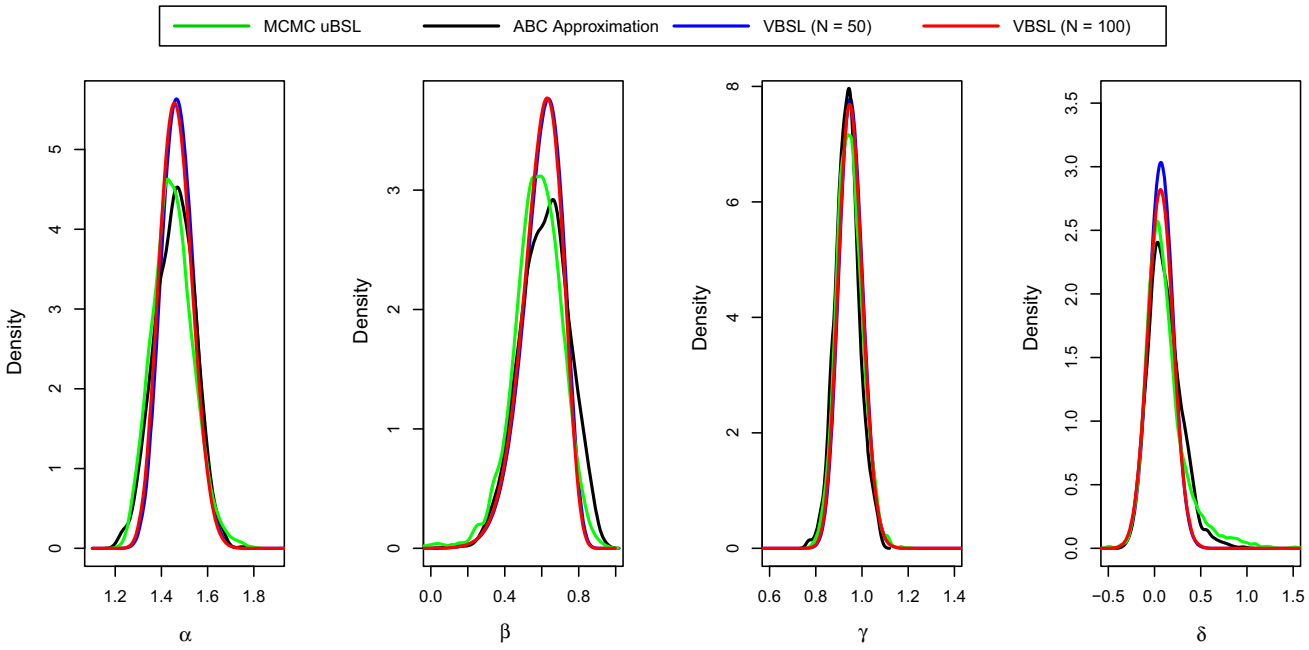

Fig. 3 Marginal variational posterior distributions for the four parameters $(\alpha, \beta, \gamma, \delta)$ for the α-stable model under $N = 50$ and $N = 100$

ness, γ is a scale parameter and δ a location parameter. The characteristic function is

$$\phi(t) = \begin{cases} \exp\{i\delta t - \gamma^{\alpha}|t|^{\alpha} & \alpha \neq 1 \\ \left(1 + i\beta \tan\frac{\pi\alpha}{2}\text{sgn}(t)\left(|\gamma t|^{1-\alpha} - 1\right)\right)\} & \\ \exp\{i\delta t - \gamma|t| & \alpha \neq 1\,, \\ \left(1 + i\beta\frac{2}{\pi}\text{sign}(t)\log(\gamma|t|)\right)\} & \end{cases}$$

where $\text{sgn}(t)$ is the sign function which is 1 if $t > 0$, 0 if $t = 0$ and -1 if $t < 0$. ABC methods for inference in this model were considered by Peters et al. (2012), who exploit the fact that convenient simulation algorithms are available for these models. Here we use a univariate model, but Peters et al. (2012) also consider the multivariate case. We follow Tran et al. (2015) who apply VBIL on a dataset of size 500 simulated from an α-stable model with $(\alpha, \beta, \gamma, \delta) = (1.5, 0.5, 1, 0)$. Here, we compare the performance of VBSL with the uBSL pseudo-marginal approach of Price et al. (2016).

Similar to Tran et al. (2015), we enforce constraints that $\alpha \in [1.1, 2]$, $\beta \in [-1, 1]$ and $\gamma > 0$ (e.g. Peters et al. 2012) through the reparametrisation $\tilde{\theta} = (\tilde{\alpha}, \tilde{\beta}, \tilde{\gamma}, \tilde{\delta})^{\top}$, with

$$\tilde{\alpha} = \log\frac{\alpha - 1.1}{2 - \alpha}, \quad \tilde{\beta} = \log\frac{\beta + 1}{1 - \beta} \quad \tilde{\gamma} = \log\gamma \quad \text{and} \quad \tilde{\delta} = \delta.$$

We consider a normal prior on $\tilde{\theta}$, $N(0, I_4)$, and approximate the posterior distribution of $\tilde{\theta}$ with a multivariate normal distribution, but report results for the posterior distribution for θ by inversion of the transformation from θ to $\tilde{\theta}$. For summary statistics, we consider a point estimator of θ due to McCulloch (1986) and then transform this point estimator to an estimator of $\tilde{\theta}$.

In implementing VBSL, there are a number of algorithmic parameters to be set. We choose $S = 500$ and use $N = 50$ and $N = 100$ to inspect the sensitivity of the VBSL towards the choice of N. In this analysis, we first consider the adaptive learning rate sequence described in Sect. 3. Figure 3 shows the variational distribution of the four parameters α, β, γ and δ. The true parameter values that are used to generate the data are recovered well—the variational distributions are quite close to those estimated by a "gold standard" ABC approximation with local linear adjustment, which is based on 1,000,000 generated samples, Epanechnikov kernel and a tolerance of $\epsilon = 0.001$. Furthermore, we observe that the variational distribution of the parameters is quite insensitive to N.

Figure 4 shows the convergence of the algorithm for the adaptive learning rate sequence (black line) and three fixed learning rate sequences, as a function of different starting values of the variational means for the four parameters (one "good" starting value (a), and two poor values). The starting variational covariance matrix is fixed at $0.04I_4$. The "good" variational means starting value (a) uses estimated summary statistics from the observed data. For the second and third starting values, we consider a starting variational mean of $(\alpha, \beta, \gamma, \delta) = (1.5, 0.5, 3, 0)$ and $(1.5, 0, 2, 0)$.

Figure 4 demonstrates that except for the "good" starting value (where the different learning rates perform similarly), the adaptive learning rate sequence converges much faster than all the fixed learning rates. It is generally the case that the adaptive learning rate sequence is more robust to an inferior starting point. To support this, Fig. 5 illustrates the step

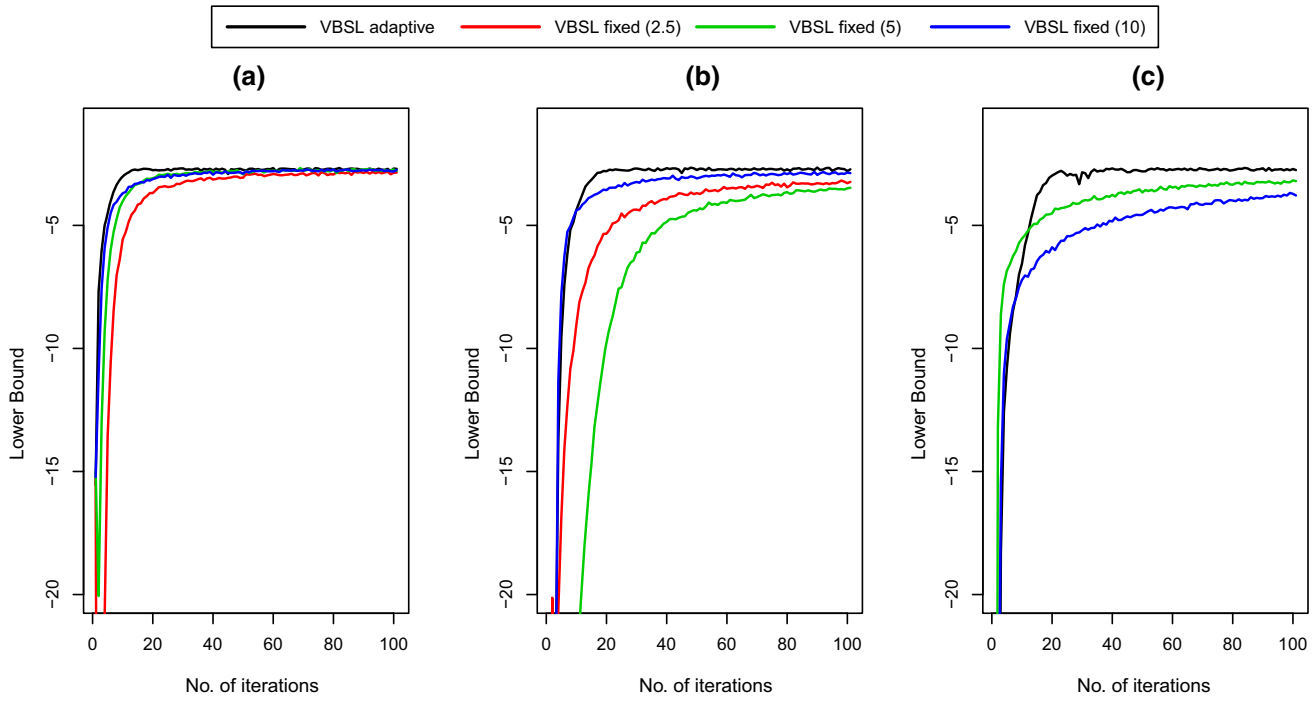

Fig. 4 Convergence of VBSL algorithm for adaptive versus three fixed learning rates. VBSL fixed (2.5) uses $\rho_t = 1/(2.5 + t)$, VBSL fixed (5) uses $\rho_t = 1/(5 + t)$ and VBSL fixed (10) uses $\rho_t = 1/(10 + t)$. Convergence speed is shown for three different starting values of the variational mean for $(\alpha, \beta, \gamma, \delta)$. Namely: **a** the estimated summary statistics for the observed data, **b** (1.5, 0.5, 3, 0) and **c** (1.5, 0, 2, 0)

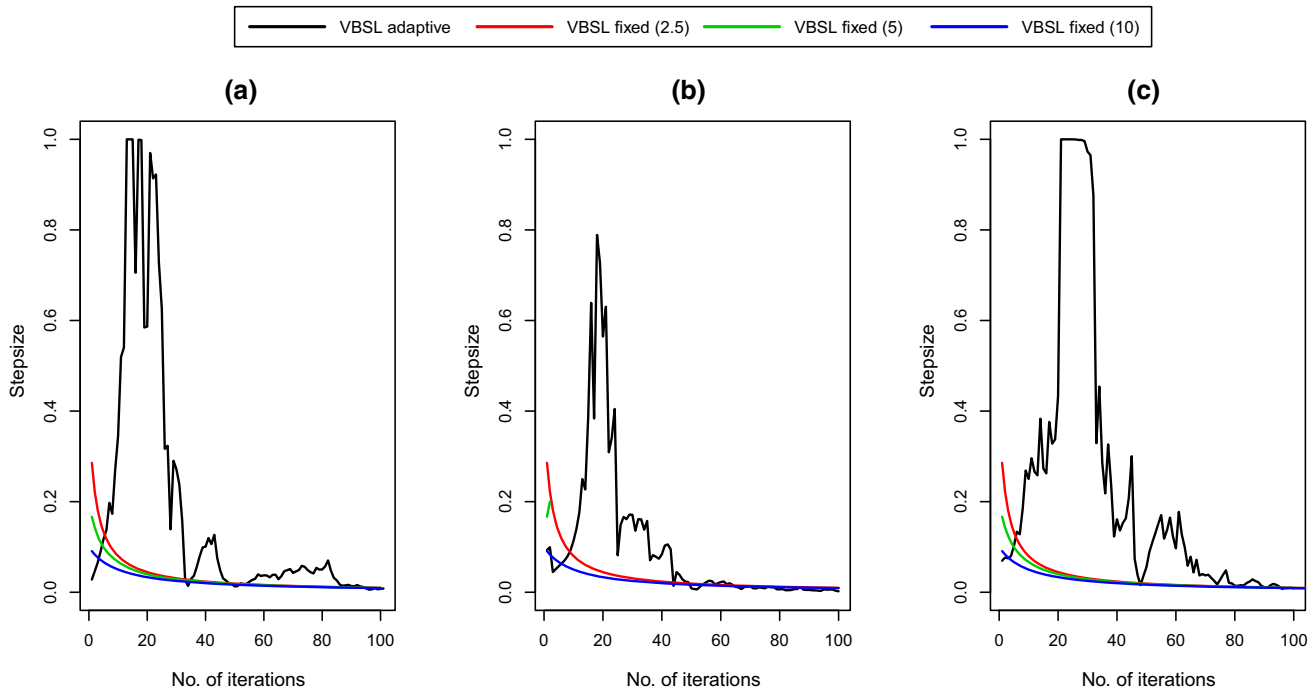

Fig. 5 Step size of VBSL algorithm against iteration number for adaptive versus three fixed learning rates for the α-stable model. VBSL fixed (2.5) uses $\rho_t = 1/(2.5 + t)$, VBSL fixed (5) uses $\rho_t = 1/(5 + t)$ and VBSL fixed (10) uses $\rho_t = 1/(10+t)$. Step sizes are shown for three different starting values of the variational mean for $(\alpha, \beta, \gamma, \delta)$. Namely: **a** the estimated summary statistics for the observed data, **b** (1.5, 0.5, 3, 0) and **c** (1.5, 0, 2, 0)

size of the four learning rate sequences against the number of iterations. We observe that the adaptive learning rate frequently takes larger steps for a greater number of iterations than the fixed-rate sequences, particularly when using an inferior starting point.

5.3 Multivariate *g*-and-*k* model

The g-and-k distribution (Rayner and MacGillivray 2002) is another flexible family of distributions for which inference can be challenging due to the lack of a closed form density function. The g-and-k distribution is defined through its quantile function, $Q(p)$, $p \in (0, 1)$, where

$$Q(p) = A + B\left[1 + c\frac{1-\exp(-gz(p))}{1+\exp(-gz(p))}\right](1+z(p)^2)^k z(p), \tag{14}$$

where $z(p) = \Phi^{-1}(p)$ and $\Phi(\cdot)$ is the standard normal distribution function. The parameter c is conventionally fixed at 0.8, and the parameters A, $B > 0$, g and $k > -0.5$ control, respectively, the location, scale, skewness and kurtosis. Simulation from the g-and-k model is easily done, since for $U \sim U[0, 1]$, $Q(U)$ is a draw from the corresponding distribution with quantile function $Q(p)$. This makes ABC methods for inference attractive (Allingham et al. 2009).

Following Drovandi and Pettitt (2011) and Li et al. (2015), we consider a multivariate g-and-k model in which the copula of the distribution is a Gaussian copula. In particular, suppose we have independent and identically distributed multivariate observations $y_1, \ldots, y_n$ where $y_i = (y_{i1}, \ldots, y_{iq})^\top$. Each y_{ir} follows a univariate g-and-k distribution marginally, $F(x; \theta_r)$ say with parameters $\theta_r = (A_r, B_r, g_r, k_r)^\top$, $r = 1, \ldots, q$. The density function and quantile function corresponding to $F(x; \theta_r)$ are written respectively as $f(x; \theta_r)$ and $Q(p; \theta_r)$. Dependence between components of y_i is modelled using a Gaussian copula (Drovandi and Pettitt 2011; Joe 1997). Let Σ be a $q \times q$ correlation matrix. Then, the density of y_i is

$$f(y_i; \theta) = |\Sigma|^{-1/2}\exp\left(\eta_i^\top(I - \Sigma^{-1})\eta_i\right)\prod_{j=1}^{q} f(y_{ij}; \theta_j), \tag{15}$$

where $\eta_i = (\eta_{i1}, \ldots, \eta_{iq})^\top$ with $\eta_{ir} = \Phi^{-1}(F(y_{ir}; \theta_r))$. This density cannot be computed in closed form because the g-and-k marginals are not available in closed form. However, it is easy to simulate from the model. Simulation from the Gaussian copula based model (15) is easily achieved by generating $Z \sim N(0, \Sigma)$ and transforming Z to $(Q(\Phi(Z_1); \theta_1), \ldots, Q(\Phi(Z_q); \theta_q))^\top$. For summary statistics, we follow Drovandi and Pettitt (2011) and use

$$S_{A_r} = E_4^{(r)}, \quad S_{B_r} = E_6^{(r)} - E_2^{(r)},$$
$$S_{g_r} = \frac{E_7^{(r)} - E_5^{(r)} + E_3^{(r)} - E_1^{(r)}}{S_{B_r}},$$
$$S_{k_r} = \frac{E_6^{(r)} + E_2^{(r)} - 2E_4^{(r)}}{S_{B_r}},$$

where $E_j^{(r)}$ is the j-th octile of the data $(y_{1r}, \ldots, y_{nr})$, for the model parameters and the robust normal scores correlation coefficient (Fisher and Yates 1948) for each of the correlation parameters in the off-diagonal entries of the copula correlation matrix Σ.

The model (15) has marginal parameters $\theta_1, \ldots, \theta_q$, as well as the copula correlation matrix Σ. It will be convenient to work with an unconstrained parametrisation of Σ. We will use a spherical parametrisation (see Pinheiro and Bates (1996), Section 2.3) and only consider the cases $q = 2, 3$. For $q = 2$, we let $w^{(2)}$ be an unconstrained real parameter and $\gamma^{(2)} = \pi/(1+\exp(-w^{(2)}))$. We parametrise Σ in terms of $\gamma^{(2)}$ by considering the Cholesky factorisation of Σ, $\Sigma = LL^\top$, and letting

$$L = \begin{bmatrix} 1 & 0 \\ \cos(\gamma^{(2)}) & \sin(\gamma^{(2)}) \end{bmatrix}.$$

For $q = 3$, we let $w^{(3)} = (w_1^{(3)}, w_2^{(3)}, w_3^{(3)})^\top$ where the elements of $w^{(3)}$ are unconstrained real parameters, define $\gamma_j^{(3)} = \pi/(1+\exp(-w_j^{(3)}))$, $j = 1, 2, 3$ and parametrise the Cholesky factor L of Σ as

$$L = \begin{bmatrix} 1 & 0 & 0 \\ \cos(\gamma_1^{(3)}) & \sin(\gamma_1^{(3)}) & \\ \cos(\gamma_2^{(3)}) & \sin(\gamma_2^{(3)})\cos(\gamma_3^{(3)}) & \sin(\gamma_2^{(3)})\sin(\gamma_3^{(3)}) \end{bmatrix}.$$

For both $q = 2$ and $q = 3$, the entries of $w^{(q)}$ are given independent normal priors, $N(0, 1.75^2)$. For the marginal parameters θ_r, we adopt independent priors for different components r. Reparametrising as $\tilde{\theta}_r = (\tilde{A}_r, \tilde{B}_r, \tilde{g}_r, \tilde{k}_r)^\top$ where

$$\tilde{A}_r = 10\log\frac{A_r + 0.1}{0.1 - A_r}, \tilde{B}_r = \log\frac{B_r}{0.05 - B_r},$$
$$\tilde{g}_r = \log\frac{g_r + 1}{1 - g_r}, \tilde{k}_r = \log\frac{k_r + 0.2}{0.5 - k_r},$$

we adopt a normal prior, $N(0, 4I_4)$ for $\tilde{\theta}_r$.

We fit models with $q = 1, 2, 3$ dimensions, with corresponding dimensions of the parameter space being 4, 9 and 15 respectively, to investigate how two different implementations of VBSL perform as the dimension increases. We parametrise the variational distribution in terms of the Cholesky factor of the precision matrix and compare the natural gradient implementation and an adaptive step size, with

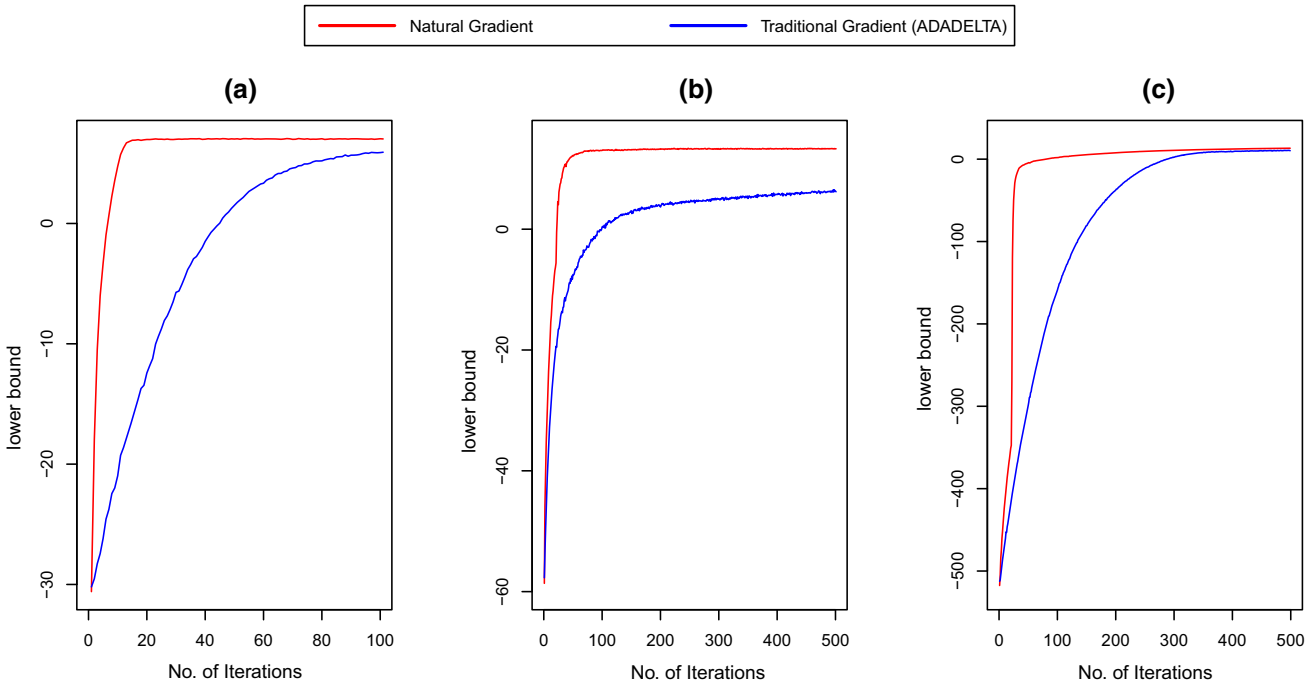

Fig. 6 Lower bound against the number of iterations using the ADADELTA traditional gradient approach and our proposed adaptive natural gradient approach for the multivariate g-and-k model. We set $N = 100$ and $S = 500$ for $q = 1, 2$, and $N = 500$ and $S = 500$ for $q = 3$

the approach based on the ordinary gradient and per parameter adaptive step sizes chosen according to the ADADELTA method of Zeiler (2012). The data we use consist of foreign currency exchange log daily returns against the Australian dollar (AUD) for 1,757 trading days between June 1, 2007, and 31 December, 2013 (Reserve Bank of Australia 2014). We consider data for 3 foreign currencies, the US dollar (USD), Japanese Yen (JY) and the Euro (EUR). Our univariate model uses just the USD, the $q = 2$ model uses the USD and JY, and the $q = 3$ model uses all 3 currencies.

We set the starting values for the variational means of $(\tilde{A}_r, \tilde{B}_r, \tilde{g}_r, \tilde{k}_r)$ as $(0, -1.5, -0.5, 0)$ and the corresponding variational variances as $(0.0001, 0.001, 0.1, 0.1)$ for $r = 1, 2$. In the $q = 3$ dimensional model (a 15 dimensional parameter), the starting value is based on the variational optimisation for the $q = 2$ model. In particular, we use the final variational mean and covariance matrix from $q = 2$ and set the starting value for the variational mean of $(\tilde{A}_3, \tilde{B}_3, \tilde{g}_3, \tilde{k}_3)$ as $(0, -1.5, -0.5, 0)$ and the corresponding variational posterior variances as $(0.0002, 0.001, 0.1, 0.1)$. For the other algorithmic parameters, we set $N = 100$ and $S = 500$ for $q = 1, 2$ and $N = 500$ and $S = 500$ for the highest dimensional example, $q = 3$. A larger N seems to be required when dealing with higher dimensional summary statistics, particularly in the initial stages, when trying to estimate likelihoods for many parameter values out in the tails of the likelihood can result in highly variable estimates. The natural gradient approach is more sensitive to this effect than the ordinary gradient approach, although the natural gradient converges faster if a large enough N is used.

Figure 6 shows the progress of the lower bounds for the two different schemes. We found that the adaptive natural gradient approach converges quite rapidly for all models, while the ordinary gradient requires a much larger number of iterations.

5.4 Cell motility example

Price et al. (2016) consider an analysis involving a stochastic model of collective cell spreading. The model contains two parameters: $P_m \in (0, 1)$ (the probability that a cell moves to a neighbouring location in a small time step) and $P_p \in (0, 1)$ (the probability that a cell gives birth to a daughter that is placed in a neighbouring location in a small time step). Price et al. (2016) consider a simulated dataset involving a time series of binary matrices where a 1 denotes the presence of a cell at a particular location. This dataset is condensed into a 145 dimensional summary statistic, which is difficult to accommodate in conventional ABC settings. They obtain significant computational advancements using a pseudo-marginal synthetic likelihood approach—however, the posterior inference remains time consuming. For more details about this application see Price et al. (2016) and the references therein.

For the variational distribution, we use a bivariate normal distribution on the logit of the parameter space. We run

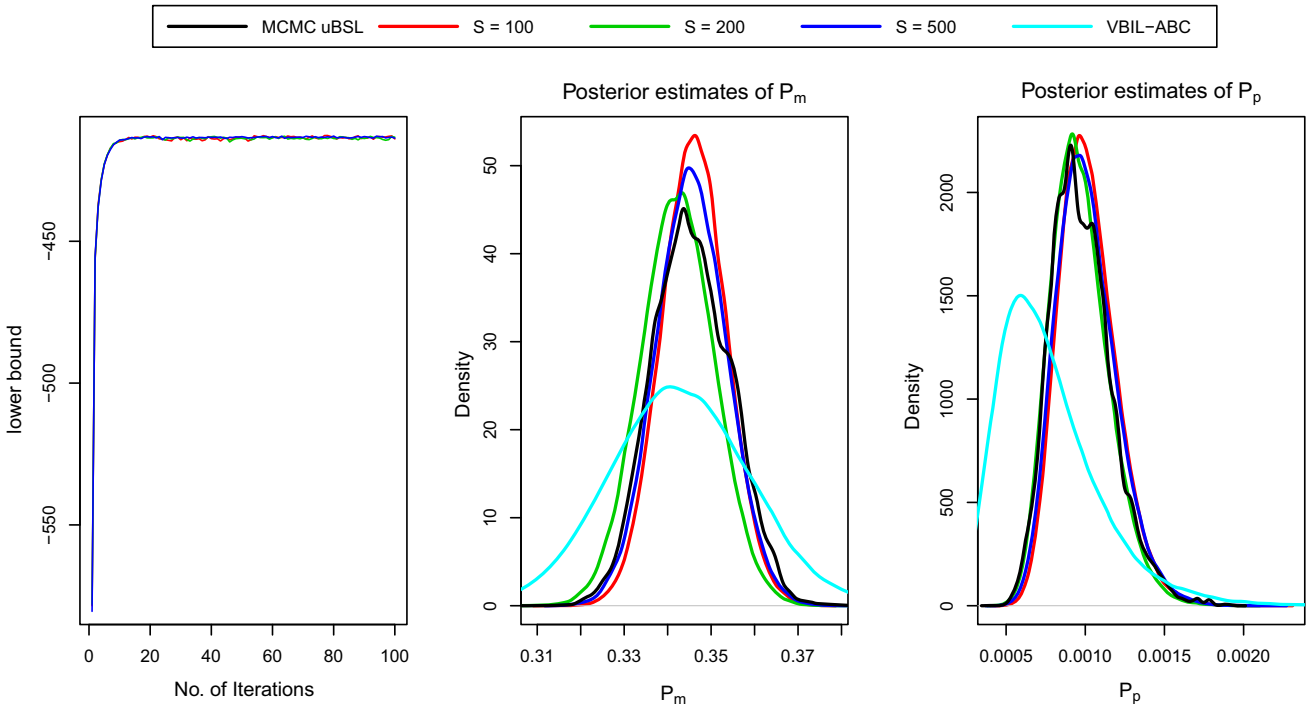

Fig. 7 Convergence of the VBSL algorithm for the cell motility analysis with $N = 1000$ and different values of S

VBSL with $N = 1000$ using our adaptive natural gradient algorithm and $S = 100, 200, 500$. We note that Price et al. (2016) find that the best choice of N in terms of computational efficiency in the context of their pseudo-marginal algorithm is $N = 5000$ (out of the trialled values of 2500, 3750, 5000, 7500 and 10,000). Figure 7 shows plots of the variational lower bound against algorithm iteration and the posterior density of P_m and P_p. We have also shown the estimated posterior distributions for the pseudo-marginal synthetic likelihood approach and the VBIL algorithm of Tran et al. (2015) with $N = 1000$ and $S = 500$. For VBIL, in the ABC likelihood approximation we have used the kernel $K_\epsilon(s, s') \propto \exp(-\frac{\rho(s,s')^2}{2\epsilon})$, where ϵ is a scaling parameter and $\rho(s, s')$ is a squared Mahalanobis distance based on the correlation matrix of the summary statistics at the true parameter value. The scaling parameter ϵ was chosen to be 600, which ensures the variance of the gradient estimates is small enough with the other algorithmic parameter settings to ensure stability of the stochastic gradient ascent algorithm.

We observe that with the adaptive scheme the VBSL methods converge rapidly and their posterior estimates are similar to the pseudo-marginal synthetic likelihood approach. However, the total computational effort involved is much reduced compared to the MCMC application considered in Price et al. (2016). In the MCMC scheme, 50,000 iterations with $N = 5000$ requires 250 million simulations of the summary statistics. On the other hand, with $S = 100$, and given that our VBSL scheme converges in about 20 iterations (and taking into account a further 5 iterations used in initialisation of the adaptive step size) the number of summary statistic simulations required is about 2.5 million for VBSL, so that the computational requirement is about 100 times less. The results for the VBIL method are also far from those given by the VBSL method with comparable computational effort (i.e. using the same S and N). The time taken to convergence is roughly the same for VBIL and VBSL. Although the VBIL method uses a different likelihood, the synthetic likelihood based answers here are closer to those based on the ABC likelihood using MCMC shown in Fig. 3 of Price et al. (2016). It is possible to improve the agreement of VBIL with the other methods by decreasing ϵ, but this requires a larger N to control the variability of the likelihood estimates.

We also show in Fig. 8 results of some posterior predictive checks (Gelman et al. 1996) based on the VBSL posterior distribution.

Here, we take a sample from the VBSL posterior distribution, $\theta^{(i)}$, $i = 1, \ldots, 100$, and for each sample and some component of the summary statistic vector S^* we generate $S^{*(i)} \sim p(S^*|\theta^{(i)})$ to get a sample from the posterior predictive distribution for S^* for a hypothetical replicate. For an adequate model, the data simulated under the fitted model should somehow look like the observed data. Figure 8 shows, for 20 randomly chosen components of the summary statistic vector, kernel estimates of the replicate posterior predictive density and the corresponding observed value. None of the observed summary statistics are lying out in the tails of the

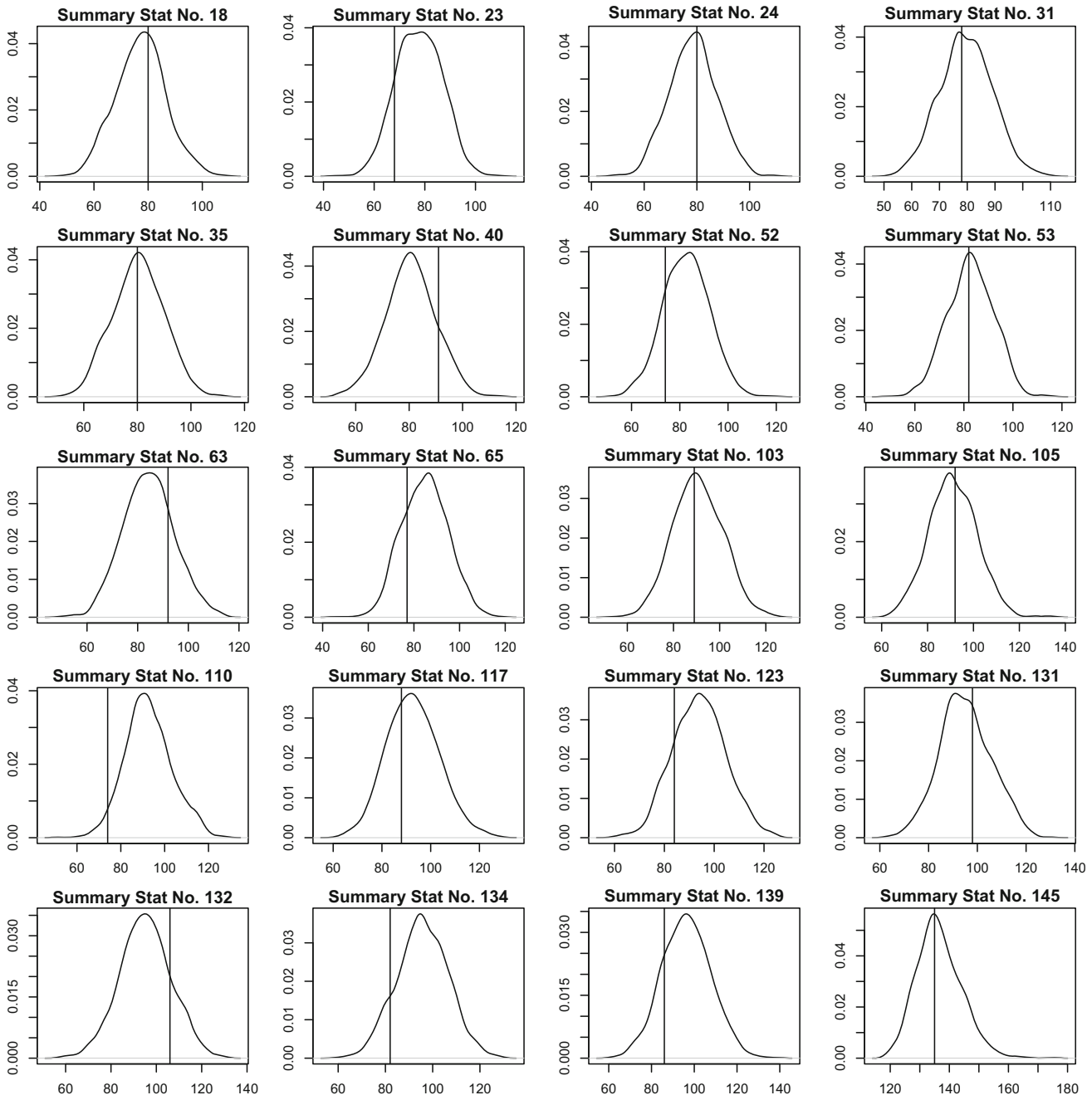

Fig. 8 Estimated posterior predictive distributions for randomly chosen summary statistic components together with corresponding observed summary statistic values (*vertical lines*)

posterior predictive distributions based on the fitted VBSL model, which provides some validation that both the model and the characterisation of uncertainty provided by the VBSL posterior are reasonable here.

6 Discussion

We have introduced a new VB approach to likelihood-free inference based on unbiased estimation of the log likelihood in the situation where the summary statistic is approximately Gaussian. In situations where the approximate Gaussian assumption holds, the methods are able to achieve good accuracy with much less computational effort than conventional ABC or synthetic likelihood methods. A focus of our future work will be making the form of the variational posterior more flexible (i.e. non-Gaussian) and implementing suitable variance reduction methods in estimating stochastic gradients in this situation. The local expectation gradients (LEG)

framework of Titsias and Lázaro-Gredilla (2015) may be particularly useful here.

Acknowledgements Victor Ong and David Nott were supported by a Singapore Ministry of Education Academic Research Fund Tier 2 grant (R-155-000-143-112). Christopher Drovandi was supported by an Australian Research Council's Discovery Early Career Researcher Award funding scheme (DE160100741). Scott Sisson was supported by the Australian Research Council through the Discovery Scheme (DP160102544) and the ACEMS Centre of Excellence (CE140100049).

Appendix A

This appendix explains the parametrisation of the variational distribution and computation of the information matrix $I_F(\lambda)$ in Algorithm 1. Most of the notations, i.e. vec, vech, D_d^+ and D_d, can be found in Sect. 4.3. We also write $\text{vec}^{-1}(a)$ for the inverse operation that takes a vector a of length d^2 and makes a $d \times d$ matrix by filling up the columns from left to right from the elements of the vector.

Suppose that $q_\lambda(\theta)$ represents our multivariate normal variational posterior approximation. λ will denote the natural parameters in the exponential family representation of the density, given below. Writing μ and Σ for the mean and covariance matrix of $q_\lambda(\theta)$, we have (Wand 2014)

$$\lambda = \begin{bmatrix} \lambda_1 \\ \lambda_2 \end{bmatrix} = \begin{bmatrix} \Sigma^{-1}\mu \\ -\frac{1}{2} D_d^\top \text{vec}(\Sigma^{-1}) \end{bmatrix},$$

and we can write μ and Σ in terms of λ as

$$\mu = \mu(\lambda) = -\frac{1}{2}\left\{\text{vec}^{-1}(D_d^{+\top}\lambda_2)\right\}^{-1}\lambda_1,$$
$$\Sigma = \Sigma(\lambda) = -\frac{1}{2}\left\{\text{vec}^{-1}(D_d^{+\top}\lambda_2)\right\}^{-1}.$$

The exponential family representation is

$$q_\lambda(\theta) = \exp\left(T(\theta)^\top\lambda - Z(\lambda)\right),$$

where $T(\theta)$ is the sufficient statistic

$$T(\theta) = \begin{bmatrix} \theta \\ \text{vech}(\theta\theta^\top) \end{bmatrix},$$

and $Z(\lambda)$ is the appropriate normalising constant. Wand (2014) shows that with $I_F(\lambda)$ defined as $\text{Cov}_\lambda(T(\theta))$, where Cov_λ denotes the covariance computed using expectation with respect to $q_\lambda(\theta)$, then [again using similar notation to Wand (2014)]

$$I_F(\lambda)^{-1} = \begin{bmatrix} \Sigma^{-1} + M^\top S^{-1} M & -M^\top S^{-1} \\ -S^{-1}M & S^{-1} \end{bmatrix},$$

where $M = 2D_d^+(\mu \otimes I_d)$ and $S = 2D_d^+(\Sigma \otimes \Sigma)D_d^{+\top}$ and $\otimes$ denotes the Kronecker product. Finally

$$\nabla_\lambda \log q_\lambda(\theta) = \begin{bmatrix} \theta - \mu \\ \text{vech}(\theta\theta^\top - \Sigma - \mu\mu^\top) \end{bmatrix}.$$

References

Adler, R.J., Feldman, R.E., Taqqu, M.S. (eds.): A Practical Guide to Heavy Tails: Statistical Techniques and Applications. Birkhauser Boston Inc., Cambridge (1998)

Allingham, D.R., King, A.R., Mengersen, K.L.: Bayesian estimation of quantile distributions. Stat. Comput. **19**, 189–201 (2009)

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

Andrieu, C., Roberts, G.O.: The pseudo-marginal approach for efficient Monte Carlo computations. Ann. Stat. **37**(2), 697–725 (2009)

Barthelmé, S., Chopin, N.: Expectation propagation for likelihood-free inference. J. Am. Stat. Assoc. **109**(505), 315–333 (2014)

Beaumont, M.A.: Estimation of population growth or decline in genetically monitored populations. Genetics **164**(3), 1139–1160 (2003)

Bishop, C.M.: Pattern Recognition and Machine Learning. Springer, Berlin (2006)

Blum, M.G.B., Nunes, M.A., Prangle, D., Sisson, S.A.: A comparative review of dimension reduction methods in approximate Bayesian computation. Stat. Sci. **28**(2), 189–208 (2013)

Bottou, L.: Large-scale machine learning with stochastic gradient descent. In: Lechevallier, Y., Saporta, G. (eds.) Proceedings of the 19th International Conference on Computational Statistics (COMPSTAT'2010), pp 177–187. Springer, Berlin (2010)

Box, G.E.P.: Sampling and Bayes' inference in scientific modelling and robustness (with discussion). J. R. Stat. Soc. Ser. A **143**, 383–430 (1980)

Brown, V.L., Drake, J.M., Barton, H.D., Stallknecht, D.E., Brown, J.D., Rohani, P.: Neutrality, cross-immunity and subtype dominance in avian influenza viruses. PLOS ONE **9**(2), 1–10 (2014)

Doucet, A., Pitt, M.K., Deligiannidis, G., Kohn, R.: Efficient implementation of Markov chain Monte Carlo when using an unbiased likelihood estimator. Biometrika **102**(2), 295–313 (2015). doi:10.1093/biomet/asu075

Drovandi, C.C., Pettitt, A.N.: Likelihood-free Bayesian estimation of multivariate quantile distributions. Comput. Stat. Data Anal. **55**, 2541–2556 (2011)

Drovandi, C.C., Pettitt, A.N., Faddy, M.J.: Approximate Bayesian computation using indirect inference. J. R. Stat. Soc. Ser. C (Appl. Stat.) **60**(3), 503–524 (2011)

Dutta, R., Corander, J., Kaski, S., Gutmann, M.U.: Likelihood-Free Inference by Penalised Logistic Regression. arXiv:1611.10242 (2016)

Everitt, R.G., Johansen, A.M., Rowing, E., Evdemon-Hogan, M.: Bayesian model comparison with un-normalised likelihoods. Stat. Comput. **27**(2), 403–422 (2017)

Fasiolo, M., Pya, N., Wood, S.N.: A comparison of inferential methods for highly nonlinear state space models in ecology and epidemiology. Stat. Sci. **31**, 96–118 (2016a)

Fasiolo, M., Wood, S.N., Hartig, F., Bravington, M.V.: An extended empirical saddlepoint approximation for intractable likelihoods. arXiv:1601.01849 (2016b)

Fisher, R.A., Yates, F.: Statistical Tables for Biological, Agricultural and Medical Research. Hafner, New York (1948)

Gelman, A., Meng, X.L., Stern, H.: Posterior predictive assessment of model fitness via realized discrepancies. Stat. Sin. **6**, 733–807 (1996)

Ghurye, S.G., Olkin, I.: Unbiased estimation of some multivariate probability densities and related functions. Ann. Math. Stat. **40**(4), 1261–1271 (1969)

Gunawan, D., Tran, M.N., Kohn, R.: Fast inference for intractable likelihood problems using variational Bayes. Working Paper, Discipline of Business Analytics, University of Sydney. http://hdl.handle.net/2123/14594 (2016)

Gutmann, M.U., Corander, J.: Bayesian optimization for likelihood-free inference of simulator-based statistical models. J. Mach. Learn. Res. **17**(125), 1–47 (2015)

Hartig, F., Dislich, C., Wiegand, T., Huth, A.: Technical note: approximate Bayesian parameterization of a process-based tropical forest model. Biogeosciences **11**, 1261–1272 (2014)

Hoffman, M.D., Blei, D.M., Wang, C., Paisley, J.: Stochastic variational inference. J. Mach. Learn. Res. **14**(1), 1303–1347 (2013)

Ji, C., Shen, H., West, M.: Bounded approximations for marginal likelihoods. Technical Report 10-05, Institute of Decision Sciences, Duke University. http://ftp.stat.duke.edu/WorkingPapers/10-05.html (2010)

Joe, H.: Multivariate models and dependence concepts. Chapman & Hall, London (1997)

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

Kucukelbir, A., Tran, D., Ranganath, R., Gelman, A., Blei, D.M.: Automatic differentiation variational inference. J. Mach. Learn. Res. **18**(14), 1–45 (2017)

Li, J., Nott, D.J., Fan, Y., Sisson, S.A.: Extending approximate Bayesian computation methods to high dimensions via Gaussian copula. Comput. Stat. Data Anal. **106**, 77–89 (2017)

Magnus, J.R., Neudecker, H.: Matrix Differential Calculus with Applications in Statistics and Econometrics. Wiley, New York (1999)

Marin, J.M., Pudlo, P., Robert, C.P., Ryder, R.J.: Approximate Bayesian computational methods. Stat. Comput. **22**(6), 1167–1180 (2012)

McCulloch, J.: Simple consistent estimators of stable distribution parameters. Commun. Stat. Simul. Comput. **15**(4), 1109–1136 (1986)

Meeds, E., Welling, M.: GPS-ABC: Gaussian process surrogate approximate Bayesian computation. In: Proceedings of the Thirtieth Conference Annual Conference on Uncertainty in Artificial Intelligence (UAI-14), pp. 593–602 (2014)

Moores, M.T., Drovandi, C.C., Mengersen, K.L., Robert, C.P.: Preprocessing for approximate Bayesian computation in image analysis. Stat. Comput. **25**(1), 23–33 (2015)

Moreno, A., Adel, T., Meeds, E., Rehg, J.M., Welling, M.: Automatic variational ABC. arXiv:1606.08549 (2016)

Nott, D., Tan, S., Villani, M., Kohn, R.: Regression density estimation with variational methods and stochastic approximation. J. Comput. Graph. Stat. **21**(3), 797–820 (2012)

Ormerod, J., Wand, M.: Explaining variational approximations. Am. Stat. **64**, 140–153 (2010)

Paisley, J.W., Blei, D.M., Jordan, M.I.: Variational Bayesian inference with stochastic search. In: Proceedings of the 29th International Conference on Machine Learning (ICML-12) (2012)

Peters, G.W., Sisson, S.A., Fan, Y.: Likelihood-free Bayesian inference for α-stable models. Comput. Stat. Data Anal. **56**, 3743–3756 (2012)

Pinheiro, J.C., Bates, D.M.: Unconstrained parametrizations for variance-covariance matrices. Stat. Comput. **6**(3), 289–296 (1996)

Pitt, M.K., Silva, RdS, Giordani, P., Kohn, R.: On some properties of Markov chain Monte Carlo simulation methods based on the particle filter. J. Econ. **171**(2), 134–151 (2012)

Price, L.F., Drovandi, C.C., Lee, A.C., Nott, D.J.: Bayesian synthetic likelihood. J. Comput. Graph. Stat. (2016) (to appear) (2016)

Ranganath, R., Wang, C., Blei, D.M., Xing, E.P.: An adaptive learning rate for stochastic variational inference. In: Proceedings of the 30th International Conference on Machine Learning (ICML-13), pp. 298–306 (2013)

Ranganath, R., Gerrish, S., Blei, D.M.: Black box variational inference. Int. Conf. Artif. Intell. Stat. **33**, 814–822 (2014)

Rayner, G., MacGillivray, H.: Weighted quantile-based estimation for a class of transformation distributions. Comput. Stat. Data Anal. **39**(4), 401–433 (2002)

Reserve Bank of Australia (2014) Historical data. http://www.rba.gov.au/statistics/historical-data.html. Accessed 16 Sept 2014

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

Ripley, B.D.: Pattern Recognition and Neural Networks. Cambridge University Press, Cambridge (1996)

Robbins, H., Monro, S.: A stochastic approximation method. Ann. Math. Stat. **22**(3), 400–407 (1951)

Salimans, T., Knowles, D.A.: Fixed-form variational posterior approximation through stochastic linear regression. Bayesian Anal. **8**(4), 837–882 (2013)

Tan, L.S.L., Nott, D.J.: Gaussian variational approximation with sparse precision matrices. Stat. Comput. (2017). doi:10.1007/s11222-017-9729-7

Titsias, M., Lázaro-Gredilla, M.: Doubly stochastic variational Bayes for non-conjugate inference. In: Proceedings of the 31st International Conference on Machine Learning (ICML-14), pp. 1971–1979 (2014)

Titsias, M., Lázaro-Gredilla, M.: Local expectation gradients for black box variational inference. In: Cortes, C., Lawrence, N.D., Lee, D.D., Sugiyama, M., Garnett, R. (eds.) Advances in Neural Information Processing Systems, vol. 28, pp. 2638–2646. Curran Associates Inc, Red Hook (2015)

Tran, M.N., Nott, D.J., Kohn, R.: Variational Bayes with intractable likelihood. arXiv:1503.08621v1 (2015)

Tran, M.N., Nott, D.J., Kohn, R.: Variational Bayes with intractable likelihood J. Comput. Graph. Stat. (2016) (to appear)

Wand, M.P.: Fully simplified multivariate normal updates in non-conjugate variational message passing. J. Mach. Learn. Res. **15**, 1351–1369 (2014)

Wilkinson, R.: Accelerating ABC methods using Gaussian processes. J. Mach. Learn. Res. **33**, 1015–1023 (2014)

Wood, S.N.: Statistical inference for noisy nonlinear ecological dynamic systems. Nature **466**, 1102–1107 (2010)

Zeiler, M.D.: ADADELTA: an adaptive learning rate method. arXiv:1212.5701 (2012)