Approximate natural gradient in Gaussian processes with non-log-concave likelihoods

Marcelo Hartmann [1]

Abstract

Approximate Bayesian inference on Gaussian process models with non-log-concave likelihoods is a challenging problem. When the log-likelihood function lacks concavity, finding the maximum a posterior estimate of the Gaussian process posterior becomes troublesome. Additionally, the lack of concavity complicates computer implementations and may increase computational load. In this work, we propose using an approximate Fisher information matrix as an alternative for defining a variant of the natural gradient update in the context of Gaussian process modeling, achieving this without incurring additional costs and with less analytical derivations. Moreover, experiments show that the approximate natural gradient works efficiently when the log-likelihood function strongly lacks concavity.

1. Introduction

Gaussian processes (GP) are stochastic processes used in Bayesian modelling as a prior distribution over some unknown function of interest. GP models have been commonly used in regression problems where the data is taken as a noisy evaluation of the unknown function, and whose noise distribution is taken as Gaussian. In many cases this assumption can be inappropriate in practical data analysis, where inference may be sensitive to outliers. To address this, several variants of GP models for robust regression have been proposed as in Taylor & Verbyla (2004), Kuss (2006), Vanhatalo et al. (2009), Jylänki et al. (2011) and Hartmann (2018). The core idea is to consider that the noisy evaluation of such underlying true function is associated with a probabilistic model in which the likelihood function is not log-concave, and it can accommodate the influence of points that are far away from the bulk of the remaining. In other words, it is resistant to outliers. Besides, in this class of GP models the posterior distribution is high-dimensional, non-conjugate and lacks concavity.

Precisely, if the logarithm of the posterior distribution were concave, it would guarantee a unique maximizer, and second-order optimization routines would work well in practice (see Rasmussen & Williams, 2006, page 42, Section 3.4.1). However, the lack of concavity introduces additional difficulties in approximate inference, such as finding the maximum a posteriori (MAP) estimate for posterior analysis. In this case, the posterior may not have a unique maximizer, and optimization routines may struggle to find the best optima. Therefore, alternative approximation techniques are of central importance in applied Bayesian analysis for more general scenarios.

In this work, we present a variant of the natural gradient update to find the MAP estimate of the Gaussian process posterior. Our method approximates the exact Fisher information matrix using only the gradient of the likelihood function, requiring no Hessian computation. In Section 2, we review the basics of Gaussian processes. Section 3 revisits the inference problem when the log-likelihood function is not concave. In Sections 4 and 5, we present a variant of the natural gradient update and a case study using the Student-t model to impose strong non-log-concave likelihood functions. Finally, in Section 6, we discuss our findings and pose the open question of why the approximate Fisher matrix used to define a variant of the natural gradient updates improved the inference scheme compared to the exact natural gradient updates when the lack of concavity may be present.

2. Gaussian processes and the MAP estimate

Let $f \sim \mathcal{GP}(m, k)$ denote the random function f following a Gaussian process with mean function m and covariance function k (see O'Hagan, 1978; Rasmussen & Williams, 2006, for details). For a data set $\boldsymbol{y} = \{y_n\}_{n=1}^N$ that is conditionally independent given the function values $\boldsymbol{f} = \{f(x_n)\}_{n=1}^N$, the likelihood function of $\boldsymbol{f}$ factorizes

[1]Department of Computer Science, University of Helsinki, Finland, Helsinki. Correspondence to: Marcelo Hartmann <marcelo.hartmann@helsinki.fi>.

Proceedings of the Geometry-grounded Representation Learning and Generative Modeling at 41st International Conference on Machine Learning, Vienna, Austria. PMLR Vol Number, 2024.

as $\pi(\boldsymbol{y} \mid \boldsymbol{f}) = \prod_{n=1}^{N} \pi(y_n|f_n)$ where $f_n = f(\boldsymbol{x}_n)$ is the unknown function value at the input point (or covariate) $\boldsymbol{x}_n$. In the Bayesian approach for statistical inference the posterior distribution of $\boldsymbol{f}$ given $\boldsymbol{y}$ becomes,

$$\pi(\boldsymbol{f} \mid \boldsymbol{y}) = \frac{\prod_{n=1}^{N} \pi(y_n|f_n)\pi(\boldsymbol{f})}{\pi(\boldsymbol{y})} \tag{1}$$

where $\pi(\boldsymbol{f}) = \mathcal{N}(\boldsymbol{f} \mid \boldsymbol{\mu}, \boldsymbol{K})$ is a multivariate Gaussian playing the role of the prior distribution. $\boldsymbol{\mu}$ is the mean vector whose elements are $\boldsymbol{\mu}_n = m(\boldsymbol{x}_n)$ and $\boldsymbol{K}$ is a covariance matrix whose (r, s) entry is formed from the covariance function as $\boldsymbol{K}_{r,s} = k(\boldsymbol{x}_r, \boldsymbol{x}_s)$ for $r, s = 1, \ldots, N$. It is also common in the Bayesian methodology to summarize the posterior distribution with a point estimate such as the MAP, which is usually obtained via some type of gradient-based optimisation or Newton's method. Here we focus on the latter, which has been first presented by Rasmussen & Williams (2006) in the Algorithm 3.1 (page 46, Chapter 3), with numerically stable implementation of the computational algorithms.

Algorithm 1 Natural gradient for finding the MAP estimate

input $\boldsymbol{y}$, K, $\boldsymbol{\mu}$, ℓ (log-likelihood function of $\boldsymbol{f}$)

$\boldsymbol{f} := \boldsymbol{\mu}$
repeat
  $G := \mathbb{E}(-\nabla^2 \ell)$ ▷ exact Fisher matrix
  $L := \text{Cholesky}(I_N + G^{\frac{1}{2}} K G^{\frac{1}{2}})$
  $\boldsymbol{b} := G(\boldsymbol{f} - \boldsymbol{\mu}) + \nabla \ell$
  $\boldsymbol{a} := \boldsymbol{b} - G^{\frac{1}{2}} L^{\top} \backslash (L \backslash (G^{\frac{1}{2}} K \boldsymbol{b}))$
  $\boldsymbol{f}^{\text{new}} := K\boldsymbol{a} + \boldsymbol{\mu}$
until convergence

3. Problem revisited

Let's denote the logarithm of (1) as,

$$\ell_{\boldsymbol{y}}(\boldsymbol{f}) = \ell(\boldsymbol{f}) + \log \pi(\boldsymbol{f}) - \log \pi(\boldsymbol{y}) \tag{2}$$

where $\ell(\boldsymbol{f}) = \sum_{n}^{N} \ell_n(f_n)$ comprises the logarithm of the likelihood function with $\ell_n(f_n) = \log \pi(y_n|f_n)$. The last terms are the logarithm of the prior distribution and the logarithm of the normalizing constant.

When the likelihood function $\ell(\boldsymbol{f})$ is not concave, the Algorithm 3.1 presented by Rasmussen & Williams (2006) no longer works efficiently. Vanhatalo et al. (2009) and Jylänki et al. (2011) try to tackle the lack of concavity by hand-tune computer algorithms but no general solution is presented. Hartmann (2018) presented one theory based solution using the idea of natural gradient from Amari (1998). The latter approach, however, needs to compute the Fisher information matrix, which is most of the times a problem since integrals might need to be solved when the probabilistic model for the data $\boldsymbol{y}$ (the likelihood part) is not part of the exponential family. Hartmann (2018) proposed a natural gradient update to find the MAP of Equation (2). This is summarize in the Algorithm 1. Basically, this is a variant of the Algorithm 3.1 aforementioned by just changing the negative Hessian matrix of the log-likelihood function, denoted by $W = -\nabla^2 \ell$, with its expected value $G = \mathbb{E}(W)$, i.e. the Fisher information matrix. In our settings this matrix is diagonal and whose n^{th} main diagonal element is given by,

$$\begin{aligned} G_{n,n}(f_n) &= \mathbb{E}_{Y_n}\Big[-\frac{\partial^2}{\partial f_n^2} \log \pi(y_n|f_n) \Big] \\ &= \int_{\Omega} \Big[-\frac{\partial^2}{\partial f_n^2} \log \pi(y_n|f_n) \Big] \pi(y_n|f_n) \mathrm{d}y_n. \end{aligned} \tag{3}$$

Here the expectation (integration) is taken over all possible outcomes Y_n, represented by the set Ω. If Y_n is discrete, then the integral is changed to a sum. Also, it is the diagonal structure of W that makes G to be diagonal as well.

4. Approximate Natural gradient

An unbiased Fisher estimate has been presented by McLachlan & Peel (2000), Chapter 2, Section 2.15.3, page 65. In the settings of this work we denote it as

$$F = S^{\top} S - \frac{1}{N} \nabla \ell \, \nabla \ell^{\top},$$

where

$$S = \begin{bmatrix} \partial_1 \ell_1 & \cdots & \partial_N \ell_1 \\ \vdots & \ddots & \vdots \\ \partial_1 \ell_N & \cdots & \partial_N \ell_N \end{bmatrix}.$$

Hence, the matrix F can also be written as,

$$F = \sum_{n}^{N} \nabla \ell_n \nabla \ell_n^{\top} - \tfrac{1}{N} \nabla \ell \nabla \ell^{\top}.$$

Now, observe that the off-diagonal elements of S are zero, $S_{r,s} = \partial_r \ell_s = 0$ for $r \neq s$, this due to the fact that each data point y_n is tied with only one function value f_n in the likelihood function. Thus we write

$$F = D - \tfrac{1}{N} \nabla \ell \, \nabla \ell^{\top} \tag{4}$$

where

$$D = \text{diag}\left((\partial_1 \ell_1)^2, \ldots, (\partial_N \ell_N)^2 \right).$$

We make three useful observations.

Observation 1 : The approximate Fisher might be helpful for the cases when W is not positive-definite everywhere leading to possible instability of the Newton's method.

Observation 2 : There is no need to compute the Hessian matrix of the log-likelihood function using the aforementioned approximate Fisher to find the MAP of the Gaussian process posterior distribution. Moreover, the computational cost will remain theoretically unchanged when defining an approximate version of the natural gradient update.

Observation 3 : When the Newton's method is susceptible to negative curvature it may blow up at saddle points (Dauphin et al., 2014). Therefore the direction of approximating Newton's method more accurately is not reasonable. This will be common in cases where the log-likelihood function is not concave, e.g., Student-t or Cauchy.

In Algorithm 1 if instead of G we plug in the approximate Fisher F, we have what we refer to as approximate natural gradient. This substitution, however, can not be straightforward done. We need to make a suitable formulation for a stable implementation. The main reason, as we will see, is because the inverse F^{-1} can not be computed once it does not exist.

To replace G with F in the natural gradient update, use as starting point the classical Newton's update (see Rasmussen & Williams, 2006, page 43, Equation 3.18) with F instead of W. Then we have,

$$\boldsymbol{f}^{\text{new}} = (K^{-1} + F)^{-1}(F(\boldsymbol{f} - \boldsymbol{\mu}) + \nabla\ell) + \boldsymbol{\mu} \tag{5}$$

As usual, we avoid inverting the matrix K due to its numerical instability (eigenvalues possibly close to zero). By rewriting $(K^{-1} + F)^{-1}$ as

$$(K^{-1} + F)^{-1} = K - K(K + F^{-1})^{-1}K \tag{6}$$

we get F^{-1} that is also problematic. More precisely,

$$\begin{aligned} F^{-1} &= \left(D - \tfrac{1}{N}\nabla\ell\,\nabla\ell^\top\right)^{-1} \\ &= D^{-1} + \frac{D^{-1}\frac{1}{\sqrt{N}}\nabla\ell\nabla\ell^\top\frac{1}{\sqrt{N}}D^{-1}}{1 - \frac{1}{N}\left\|\nabla\ell\right\|^2_{D^{-1}}}. \end{aligned} \tag{7}$$

Observe the denominator is null. That is,

$$\begin{aligned} 1 - \tfrac{1}{N}\left\|\nabla\ell\right\|^2_{D^{-1}} &= 1 - \tfrac{1}{N}\nabla\ell^\top D^{-1}\nabla\ell \\ &= 1 - \tfrac{1}{N}\sum_{n=1}^{N}(\partial_n\ell_n)^2/(\partial_n\ell_n)^2 \\ &= 1 - \tfrac{1}{N}N = 0. \end{aligned}$$

To circumvent the above lets rewrite the matrix F^{-1} by plugging $(1-\epsilon)$ in between $\nabla\ell$ and $\nabla\ell^\top$, for $0 < \epsilon < 1$, in the Equation (7). Then we get,

$$\begin{aligned} F^{-1} &= \left(D - \tfrac{1}{N}\nabla\ell(1-\epsilon)\nabla\ell^\top\right)^{-1} \\ &= D^{-1} + \frac{D^{-1}\frac{\sqrt{(1-\epsilon)}}{\sqrt{N}}\nabla\ell\nabla\ell^\top\frac{\sqrt{(1-\epsilon)}}{\sqrt{N}}D^{-1}}{1 - \frac{(1-\epsilon)}{N}\left\|\nabla\ell\right\|^2_{D^{-1}}}. \end{aligned} \tag{8}$$

The denominator becomes,

$$\begin{aligned} 1 - \tfrac{(1-\epsilon)}{N}\left\|\nabla\ell\right\|^2_{D^{-1}} &= 1 - \tfrac{(1-\epsilon)}{N}\nabla\ell^\top D^{-1}\nabla\ell \\ &= 1 - \tfrac{(1-\epsilon)}{N}\sum_{n=1}^{N}(\partial_n\ell_n)^2/(\partial_n\ell_n)^2 \\ &= 1 - \tfrac{(1-\epsilon)}{N}N = \epsilon. \end{aligned}$$

Define $R = D^{-1}\frac{\sqrt{(1-\epsilon)}}{\sqrt{N}}\nabla\ell$ and compute the inverse matrix on the right side of (6) as the limit of ϵ approaching zero.

$$\begin{aligned} (K + F^{-1})^{-1} &= \lim_{\epsilon\to 0^+}(K + D^{-1} + R\epsilon^{-1}R^\top)^{-1} \\ &= \lim_{\epsilon\to 0^+} E - ER(\epsilon + R^\top ER)^{-1}R^\top E \\ &= \lim_{\epsilon\to 0^+} E - \frac{\frac{(1-\epsilon)}{N}ED^{-1}\nabla\ell\nabla\ell^\top D^{-1}E}{\epsilon + \frac{(1-\epsilon)}{N}\left\|\nabla\ell\right\|^2_{D^{-1}ED^{-1}}} \\ &= E - \lim_{\epsilon\to 0^+}\frac{\frac{(1-\epsilon)}{N}ED^{-1}\nabla\ell\nabla\ell^\top D^{-1}E}{\epsilon + \frac{(1-\epsilon)}{N}\left\|\nabla\ell\right\|^2_{D^{-1}ED^{-1}}} \\ &= E - \frac{ED^{-1}\nabla\ell\nabla\ell^\top D^{-1}E}{\left\|\nabla\ell\right\|^2_{D^{-1}ED^{-1}}}, \end{aligned} \tag{9}$$

where

$$E = (K + D^{-1})^{-1} = D^{\frac{1}{2}}(I + D^{\frac{1}{2}}KD^{\frac{1}{2}})^{-1}D^{\frac{1}{2}}$$

and

$$D^{\frac{1}{2}} = \operatorname{diag}(|\partial_1\ell_1|, \ldots, |\partial_N\ell_N|)).$$

Moreover, let

$$S = D^{-1}\nabla\ell = [1/(\partial_1\ell_1)\cdots 1/(\partial_N\ell_N)]^\top,$$

we simplify (9) to compute (6) as

$$(K^{-1} + F)^{-1} = K - KEK + KES(KES)^\top/\left\|S\right\|^2_E. \tag{10}$$

Observe that the matrix form of the approximate Fisher matrix in the Equation (4) is similar to that one of the Hessian matrix of the likelihood function in multi-class classification problems. See for example Williams & Barber (1998) or Rasmussen & Williams (2006). Finally, using Equation (10) into the original Newton's update and rearranging, we propose the approximate natural gradient step as follows,

$$\begin{aligned} \boldsymbol{f}^{\text{new}} &= (K^{-1} + F)^{-1}(F(\boldsymbol{f} - \boldsymbol{\mu}) + \nabla\ell) + \boldsymbol{\mu} \\ &= (K^{-1} + F)^{-1}\boldsymbol{b} + \boldsymbol{\mu} \\ &= \left[K - K\left(E - ES(ES)^\top/\left\|S\right\|^2_E\right)K\right]\boldsymbol{b} + \boldsymbol{\mu} \\ &= K\boldsymbol{b} - KEK\boldsymbol{b} + KES(KES)^\top\boldsymbol{b}/\left\|S\right\|^2_E + \boldsymbol{\mu} \\ &= K\boldsymbol{b} - K\boldsymbol{c} + KESS^\top\boldsymbol{c}/\left\|S\right\|^2_E + \boldsymbol{\mu} \\ &= K\left(\boldsymbol{b} - \boldsymbol{c} + ES\langle S, \boldsymbol{c}\rangle/\left\|S\right\|^2_E\right) + \boldsymbol{\mu} \\ &= K\left(\boldsymbol{b} - \boldsymbol{c} + ES\tfrac{\langle S,\boldsymbol{c}\rangle}{\left\|S\right\|^2_E}\right) + \boldsymbol{\mu} \\ &= K\boldsymbol{a} + \boldsymbol{\mu} \end{aligned}$$

where we had set $\boldsymbol{b} = F(\boldsymbol{f} - \boldsymbol{\mu}) + \nabla\ell$, $\boldsymbol{c} = EK\boldsymbol{b}$ and $\boldsymbol{a} = \boldsymbol{b} - \boldsymbol{c} + ES\langle S, \boldsymbol{c}\rangle / \|S\|_E^2$. The approximate natural gradient update is summarised in the Algorithm 2. The

Algorithm 2 Approximate natural gradient for finding the MAP estimate

input $\boldsymbol{y}$, K, $\boldsymbol{\mu}$, ℓ (log-likelihood function of $\boldsymbol{f}$)

$\boldsymbol{f} := \boldsymbol{\mu}$
repeat
  $D^{\frac{1}{2}} := \text{diag}(|\partial_1\ell_1|, \ldots, |\partial_N\ell_N|))$
  $S := D^{-1}\nabla\ell$
  $L := \text{cholesky}(I_N + D^{\frac{1}{2}}KD^{\frac{1}{2}})$
  $E := D^{\frac{1}{2}}L^\top \backslash (L \backslash D^{\frac{1}{2}})$
  $\boldsymbol{b} := F(\boldsymbol{f} - \boldsymbol{\mu}) + \nabla\ell$
  $\boldsymbol{c} := EK\boldsymbol{b}$
  $\boldsymbol{a} := \boldsymbol{b} - \boldsymbol{c} + ES\frac{\langle S, \boldsymbol{c}\rangle}{\|S\|_E^2}$
  $\boldsymbol{f} = K\boldsymbol{a} + \boldsymbol{\mu}$
until convergence

notation $|\cdot|$ in it stands for the absolute value function.

5. Study case

In this section we study the practical performance of the proposed Algorithms 1 (exact Fisher) and 2 (approximate Fisher). We do so by varying how strong the logarithm of likelihood function lacks being concave. The ideal candidate as a probabilistic model to study such a problem is the Student-t model. This is because we can control how strong the log-likelihood function of such a model lacks concavity. See Vanhatalo et al. (2009) and Hartmann (2018) Section 3.1 for detailed introduction for this model properties.

In the subsequent experiments we fix the data set $\boldsymbol{y}$ and the parameters of the Gaussian process model. We set $m = 0$ and choose $k(x, x') = \sigma_f^2 \exp\big(-(x - x')^2/l^2\big)$ with $\sigma_f^2 = 1$ and $l = 1$. We then consider the Student-t model for the data $\boldsymbol{y}$ and set the mean parameter of this model to follow a Gaussian process. This way the likelihood function $\pi(\boldsymbol{y}\,|\cdot) : \mathbb{R}^N \to \mathbb{R}$ is

$$\pi(\boldsymbol{y}\,|\,\boldsymbol{f}) = \prod_{n=1}^{N} \frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\sigma\sqrt{\pi\nu}\Gamma\left(\frac{\nu}{2}\right)} \left(1 + \frac{(y_n - f_n)^2}{\nu\sigma^2}\right)^{-\frac{\nu+1}{2}}. \tag{11}$$

The data set generated is obtained by drawing $y_n = f(x_n) + e_n$ where $e_n \sim \text{Student-}t(f(\boldsymbol{x}_n), \sigma^2, \nu)$ with $\sigma^2 = 0.1$ and degree of freedom $\nu = 3$, for $n = 1, \ldots, N$. The true underlying function is $f(x) = 1/4|x| + u(x)$ where u is a random draw from the $\mathcal{GP}(0, k)$ in a very dense grid of points in the interval $I = [-20, 20]$. We select $N = 150$ points randomly in the interval I. The Figure 1 depicts the data points obtained together with the true underlying function $f(x)$. Also, it shows the results of the Algorithm 1 and Algorithm 2 to find the MAP of the Equation (1) when the value of ν, in the Equation (11), is set to $5e^{-8}$.

The result of the experiment depicted in Figure 1 shows that the MAP estimate using Algorithm 1, with the exact Fisher information, was not able to recover the underlying true function. In the other hand, the MAP estimate using Algorithm 2, with an approximate Fisher, was able to provide a good estimate of $f(x)$.

Note that the parameter ν has extremely small value, thus the function (11) becomes hard to optimise. Also observe that in the works of Vanhatalo et al. (2009) and Jylänki et al. (2011) they impose the restriction $\nu \geq 1$ to avoid non-convergence issues with their approach.

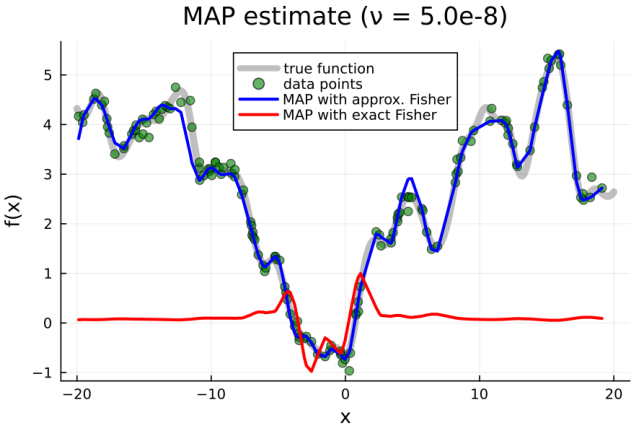

Figure 1. This figure shows the performance of Algorithms 1 and 2 in the task of finding the MAP estimate for a fixed dataset $\boldsymbol{y}$ and fixed $\nu = 5e^{-8}$. The MAP estimate using Algorithm 1 is displayed in red and for the Algorithm 2 it is in blue. Algorithm 2 was able to find a better MAP estimate than Algorithm 1.

In the next experiment, presented in Figure 2, we keep the same scenario as before, but vary only the value of ν in the Equation (11). We select 60 different values of ν equally spaced in the interval $I_\nu = [5e^{-8}, 0.5]$. For each value ν takes in I_ν we run Algorithm 1 and Algorithm 2, and report the number of steps until convergence alongside the wall-clock time required.

The results presented in Figure 2 show that the number of iterations until convergence of the Algorithms 1 and 2 are significantly different when the value of ν is small, that is the lack of concavity of the log-likelihood function is strong. The Algorithm 2 is faster in almost all values of ν in the interval given. For larger values of ν, Algorithm 1 and 2 have practically the same performance. This is expected. Since in this case the likelihood function (11) will start to approximate that of a Gaussian when ν grows.

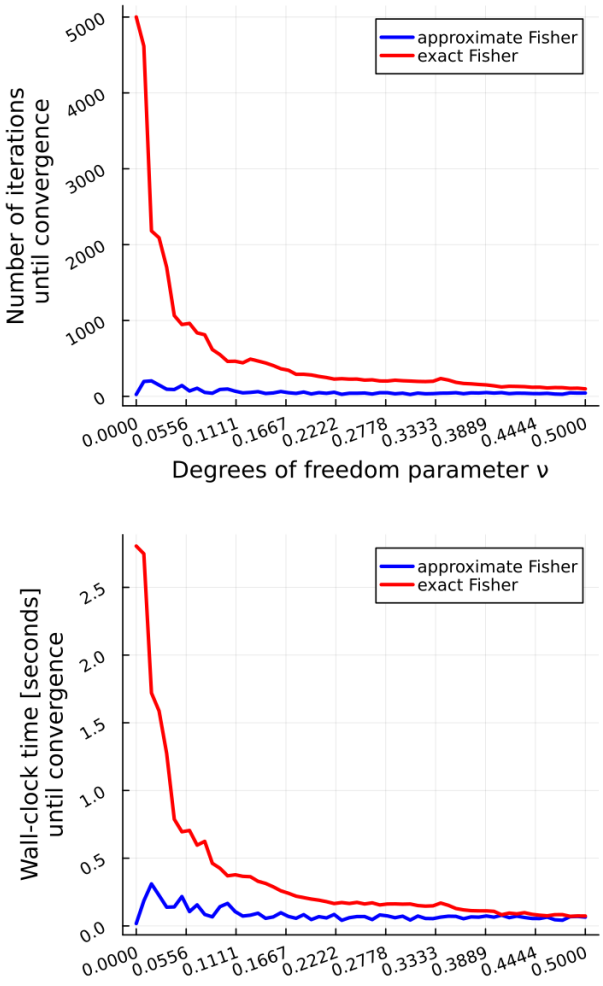

Figure 2. In the top plot, we measure the performance of Algorithm 1 and Algorithm 2 in terms of the number of iterations until convergence for the varying values of $\nu \in I_\nu$, in Equation (11). The number of iterations until convergence for Algorithm 1 is displayed in red, while for Algorithm 2 it is displayed in blue. In the bottom plot, we conduct the same experiment as in the top plot but record the wall-clock time until convergence.

In other words, we get a likelihood function that starts to lack less log-concavity in the whole domain of the posterior distribution.

Note that, in principle, the behavior observed in the previous experiments were not expected. In fact, we would expect Algorithm 1 to be faster in all cases since it uses the exact Fisher information matrix. Surprisingly, what we see is that Algorithm 2 shows better performance for smaller ν, having similar performance compared to Algorithm 1 for larger ν. This goes against the usual empirical evidence that the exact Fisher metric would improve optimisation routines and convergence.

6. Concluding remarks and discussion

Optimization methods using natural gradients have been employed by many authors in various fields. See for example Robert E. Kass (1997), Amari (1998), Taylor & Verbyla (2004), Hensman et al. (2012), Hartmann (2018), Lin et al. (2019), Martens (2020) and the references therein. In all these works, the exact form of the natural gradient has demonstrated improved convergence of optimization routines. In this work, we present a case where an approximate version of the natural gradient provides better inferential procedures compared to its exact version in a specific scenario. This finding has broader implications, as explained below.

The natural gradient can be seen as a geometry-aware optimization method, where the Fisher information describes the underlying geometry of the problem. From a differential-geometric perspective, as discussed in Do Carmo (1992), Lee (2003) and Hartmann (2019); Hartmann et al. (2022; 2023), one is free to choose the underlying geometry of the problem as long as it satisfies certain properties. The assumption that the underlying geometry is described by the Fisher information matrix is primarily motivated by its relation to the lower bound of the variance of unbiased estimators (George Casella, 2001; Lehmann, 2003) or as the second-order derivatives of the Kullback-Leibler divergence (Calin & Udrişte, 2014). However, as this work shows, the choice of the exact Fisher information matrix for the natural gradient update may not always be ideal. Furthermore, we are not aware of any work that theoretically guarantees the Fisher information matrix as the best representative of the underlying geometry of the problem.

Acknowledgements

Special thanks to Alison Pouplin for truly motivate this work and Søren Hauberg for brief discussions. MH is supported by the Research Council of Finland (RCF) grant 348952. Additionally, by the Finnish Center for Artificial Intelligence FCAI and the RCF grant 345811.

References

Amari, S. Natural Gradient Works Efficiently in Learning. *Neural Computation (communicated by Steven Nowlan and Erkki Oja)*, 10:251–276, 1998.

Calin, O. and Udrişte, C. *Geometric Modeling in Probability and Statistics*. Springer International Publishing, 1 edition, 2014.

Dauphin, Y. N., Pascanu, R., Gulcehre, C., Cho, K., Gan-

guli, S., and Bengio, Y. Identifying and attacking the saddle point problem in high-dimensional non-convex optimization. In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N., and Weinberger, K. (eds.), *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014.

Do Carmo, M. P. *Riemannian Geometry*. Mathematics. Theory & applications. Birkhäuser, 1 edition, 1992.

George Casella, R. L. B. *Statistical Inference*. Duxbury Press, 2° edition, 2001.

Hartmann, Marcelo; Vanhatalo, J. Laplace approximation and natural gradient for Gaussian process regression with heteroscedastic Student-t model. *Statistics and Computing*, 29:753–773, 2018.

Hartmann, M. *Approximate Bayesian inference in multivariate Gaussian process regression and applications to species distribution models*. PhD thesis, University of Helsinki, March 2019.

Hartmann, M., Girolami, M., and Klami, A. Lagrangian manifold Monte Carlo on Monge patches. In Camps-Valls, G., Ruiz, F. J. R., and Valera, I. (eds.), *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pp. 4764–4781. PMLR, 28–30 Mar 2022.

Hartmann, M., Williams, B., Yu, H., Girolami, M., Barp, A., and Klami, A. Warped geometric information on the optimisation of Euclidean functions, 2023. URL https://arxiv.org/abs/2308.08305.

Hensman, J., Rattray, M., and Lawrence, N. Fast variational inference in the conjugate exponential family. In Pereira, F., Burges, C., Bottou, L., and Weinberger, K. (eds.), *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012.

Jylänki, P., Vanhatalo, J., and Vehtari, A. Robust gaussian process regression with a student-¡i¿t¡/i¿ likelihood. *Journal of Machine Learning Research*, 12(99):3227–3257, 2011.

Kuss, M. *Gaussian Process Models for Robust Regression, Classification, and Reinforcement Learning*. Phd thesis, Technische Universitä Darmstadt, 2006.

Lee, J. *Introduction to Smooth Manifolds*. Graduate Texts in Mathematics. Springer, 2003.

Lehmann, G. C. *Theory of Point Estimation*. Springer texts in Statistics. Springer, 2nd ed edition, 2003.

Lin, W., Khan, M. E., and Schmidt, M. Fast and simple natural-gradient variational inference with mixture of exponential-family approximations. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 3992–4002. PMLR, 09–15 Jun 2019.

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

McLachlan, G. and Peel, D. *Finite Mixture Models*. Wiley Series in Probability and Statistics. Wiley-Interscience, 1 edition, 2000.

O'Hagan, A. Curve fitting and optimal design for prediction. *Journal of the Royal Statistical Society Series B (Methodological)*, 40:1–24, 1978. ISSN 0035-9246.

Rasmussen, C. E. and Williams, C. K. I. *Gaussian processes for machine learning*. Adaptive computation and machine learning. MIT Press, 2006.

Robert E. Kass, P. W. V. *Geometrical Foundations of Asymptotic Inference*. Probability and Statistics 125. Wiley-Interscience, 1 edition, 1997.

Taylor, J. and Verbyla, A. Joint modelling of location and scale parameters of the t distribution. *Statistical Modelling*, 4(2):91–112, 2004.

Vanhatalo, J., Jylänki, P., and Vehtari, A. Gaussian process regression with student-t likelihood. In Bengio, Y., Schuurmans, D., Lafferty, J., Williams, C., and Culotta, A. (eds.), *Advances in Neural Information Processing Systems*, volume 22. Curran Associates, Inc., 2009.

Williams, C. and Barber, D. Bayesian classification with Gaussian processes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(12):1342–1351, 1998.