Scalable Gaussian Process Computations Using Hierarchical Matrices

Christopher J. Geoga [*]

Mathematics and Computer Science Division, Argonne National Laboratory

Mihai Anitescu [†]

Mathematics and Computer Science Division, Argonne National Laboratory
Department of Statistics, University of Chicago

Michael L. Stein [‡]

Department of Statistics, University of Chicago

Abstract

We present a kernel-independent method that applies hierarchical matrices to the problem of maximum likelihood estimation for Gaussian processes. The proposed approximation provides natural and scalable stochastic estimators for its gradient and Hessian, as well as the expected Fisher information matrix, that are computable in quasilinear $O(n \log^2 n)$ complexity for a large range of models. To accomplish this, we (i) choose a specific hierarchical approximation for covariance matrices that enables the computation of their exact derivatives and (ii) use a stabilized form of the Hutchinson stochastic trace estimator. Since both the observed and expected information matrices can be computed in quasilinear complexity, covariance matrices for MLEs can also be estimated efficiently. In this study, we demonstrate the scalability of the method, show how details of its implementation effect numerical accuracy and computational effort, and validate that the resulting MLEs and confidence intervals based on the inverse Fisher information matrix faithfully approach those obtained by the exact likelihood.

Keywords: Algorithms, Numerical Linear Algebra, Spatial Analysis, Statistical Computing

[*] corresponding author: `cgeoga@anl.gov`

[†] This material was based upon work supported by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research (ASCR) under Contract DE-AC02-06CH11347. We acknowledge partial NSF funding through awards FP061151-01-PR and CNS-1545046 to MA.

[‡] This material was based upon work supported by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research (ASCR) under Contract DE-AC02-06CH11357.

1 Introduction

Many real-valued stochastic processes $Z(\boldsymbol{x})$, $\boldsymbol{x} \in \mathbb{R}^d$, are modeled as Gaussian processes, so that observing $Z(\boldsymbol{x})$ at locations $\{\boldsymbol{x}_j\}_{j=1}^n$ results in the data $\boldsymbol{y} \in \mathbb{R}^n$ following a $N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ distribution. Here the covariance matrix $\boldsymbol{\Sigma}$ is parameterized by a valid covariance function $K(\cdot, \cdot; \boldsymbol{\theta})$ that depends on parameters $\boldsymbol{\theta} \in \mathbb{R}^m$, so that

$$\begin{aligned} \boldsymbol{\Sigma}_{j,k} &= \mathrm{Cov}(Z(\boldsymbol{x}_j), Z(\boldsymbol{x}_k)) \\ &= K(\boldsymbol{x}_j, \boldsymbol{x}_k; \boldsymbol{\theta}), \quad j, k = 1, 2, \dots, n. \end{aligned}$$

In many cases in the physical sciences, estimating the parameters $\boldsymbol{\theta}$ is of great practical and scientific interest. The reference tool to estimate $\boldsymbol{\theta}$ given data $\boldsymbol{y}$ is the *maximum likelihood estimator* (MLE), which is the vector $\widehat{\boldsymbol{\theta}}$ that minimizes the negative log-likelihood, given by

$$-l(\boldsymbol{\theta}) := \frac{1}{2} \log \left|\boldsymbol{\Sigma}(\boldsymbol{\theta})\right| + \frac{1}{2}(\boldsymbol{y} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}(\boldsymbol{\theta})^{-1}(\boldsymbol{y} - \boldsymbol{\mu}) + \frac{n}{2}\log(2\pi), \tag{1}$$

where $|A|$ denotes the determinant of A. For a detailed discussion of maximum likelihood and estimation methods, see Stein (1999). In our discussions here of the log-likelihood, the mean vector $\boldsymbol{\mu}$ will be assumed to be zero, and the constant term will be suppressed. Also, for notational simplicity, the explicit dependence of $\boldsymbol{\Sigma}$ on $\boldsymbol{\theta}$ will not be indicated in the rest of this paper.

From a computational perspective, finding the minimizer of (1) can be challenging. The immediate difficulty is that the linear algebraic operations required to evaluate (1) grow with cubic time complexity and quadratic storage complexity. Thus, as the data size n increases, direct evaluation of the log-likelihood quickly becomes prohibitively slow or impossible. As a result of these difficulties, many methods have been proposed that improve both the time and memory complexity of evaluating or approximating (1) as well as finding its minimizer in indirect ways. Perhaps the oldest systematic methods for the fast approximation of the likelihood are the “composite methods,” which can loosely be thought of in this context as a block-sparse approximation of $\boldsymbol{\Sigma}$ (Vecchia 1988, Stein et al. 2004, Caragea & Smith 2007, Katzfuss 2017, Katzfuss & Guinness 2018). In a different approach, the methods of matrix tapering (Furrer et al. 2006, Kaufman et al. 2008, Castrillon-Candas et al. 2016) and Markov random fields (Rue & Held 2005, Lindgren et al. 2011) use approximations that induce sparsity in $\boldsymbol{\Sigma}$ or $\boldsymbol{\Sigma}^{-1}$, facilitating linear algebra with better scaling that way. Approximations of $\boldsymbol{\Sigma}$ as a low-rank update to a diagonal matrix have also been applied to this problem (Cressie & Johannesson 2006), although they can perform poorly in some settings (Stein 2014). In a different approach, by side-stepping the likelihood and instead solving estimating equations (Godambe 1991, Heyde 2008), one can avoid log-determinant computation and perform a smaller set of linear algebraic operations that are more easily accelerated (Anitescu et al. 2011, Stein et al. 2013, Sun & Stein 2016). The estimating equations approach involves solving a nonlinear system of equations that in the case of the score equations are given by setting

$$-\nabla l(\boldsymbol{\theta})_j := \frac{1}{2}\mathrm{tr}\left(\boldsymbol{\Sigma}^{-1}\boldsymbol{\Sigma}_j\right) - \frac{1}{2}\boldsymbol{y}^T \boldsymbol{\Sigma}^{-1}\boldsymbol{\Sigma}_j \boldsymbol{\Sigma}^{-1}\boldsymbol{y} \tag{2}$$

to 0 for all j, where here and throughout the paper $\boldsymbol{\Sigma}_j$ denotes $\frac{\partial}{\partial \theta_j}\boldsymbol{\Sigma}(\boldsymbol{\theta})$. The primary difficulty with computing these derivatives is that the trace of the matrix-matrix product $\boldsymbol{\Sigma}^{-1}\boldsymbol{\Sigma}_j$ is prohibitively expensive to compute exactly. To get around this, Anitescu et al. (2011) proposed using sample average approximation (SAA), which utilizes the unbiased stochastic trace estimator proposed by Hutchinson (1990). For (2) it is given by

$$\frac{1}{N_h}\sum_{l=1}^{N_h} \boldsymbol{u}_l^T \boldsymbol{\Sigma}^{-1}\boldsymbol{\Sigma}_j \boldsymbol{u}_l,$$

where the stochastic $\boldsymbol{u}_l$ are symmetric Bernoulli vectors (although other options are available; see Stein et al. (2013) for details). Further, if it is feasible to compute a symmetric factorization $\boldsymbol{\Sigma} = \boldsymbol{W}\boldsymbol{W}^T$, then there may be advantages to using a "symmetrized" trace estimator (Stein et al. 2013)

$$\frac{1}{N_h}\sum_{l=1}^{N_h} \boldsymbol{u}_l^T \boldsymbol{W}^{-1}\boldsymbol{\Sigma}_j \boldsymbol{W}^{-T}\boldsymbol{u}_l.$$

Specifically, writing $\mathcal{I}$ for the expected Fisher information matrix, Stein et al. (2013) shows that the covariance matrix of these estimates is bounded by $(1 + \frac{1}{N_h})\mathcal{I}$, whereas if we do not symmetrize, it is bounded by $(1 + \frac{(\kappa+1)^2}{4N_h\kappa})\mathcal{I}$, where κ is the condition number of the covariance matrix at the true parameter value. Since $\kappa \geq 1$, the bound with symmetrization is at least as small as the bound without it. Of course, this does not prove that the actual covariance matrix is smaller, but the results in Section 4 show that the improvement due to symmetrization can be large. Moreover, the symmetrized trace estimator reduces the number of linear solves from two to one if $\boldsymbol{\Sigma}^{-1}\boldsymbol{y}$ is computed as $\boldsymbol{W}^{-T}\boldsymbol{W}^{-1}\boldsymbol{y}$, saving computer time. Finally, if $\boldsymbol{\Sigma}^{-1}\boldsymbol{\Sigma}_j$ is itself positive definite, then probabilistic bounds on the error of the estimator can be controlled independently of the matrix size n (Roosta-Khorasani & Ascher 2015).

Recently, significant effort has been expended to apply the framework of hierarchical matrices (Hackbusch 1999, Grasedyck & Hackbusch 2003, Hackbusch 2015)—which utilize the low-rank block structure of certain classes of special matrices to obtain significantly better time and storage complexity for common operations—to the problem of Gaussian process computing (Börm & Garcke 2007, Ambikasaran et al. 2016, Minden et al. 2017, Litvinenko et al. 2017, Chen & Stein 2017). We follow in this vein here, extending some of these ideas by using the specific class of hierarchical matrices referred to as hierarchical off-diagonal low-rank (HODLR) matrices (Ambikasaran & Darve 2013). Unlike some of the methods described earlier, approaches to approximating l_H with hierarchical matrices have the benefit of being able to directly compute the log-determinant of $\boldsymbol{\Sigma}$ and the linear solve $\boldsymbol{\Sigma}^{-1}\boldsymbol{y}$, which would not be possible if one were to use matrix-free methods like the Fast Multipole Method (FMM) (Greengard & Rokhlin 1987) or circulant embedding (Anitescu et al. 2011). Letting $\widetilde{\boldsymbol{\Sigma}}$ denote a hierarchical approximation of $\boldsymbol{\Sigma}$ here and throughout the paper, we give the approximated log-likelihood as

$$-l_H(\boldsymbol{\theta}) := \frac{1}{2}\log\left|\widetilde{\boldsymbol{\Sigma}}\right| + \frac{1}{2}\boldsymbol{y}^T\widetilde{\boldsymbol{\Sigma}}^{-1}\boldsymbol{y}. \tag{3}$$

While the issue of how well $\widetilde{\boldsymbol{\Sigma}}$ approximates $\boldsymbol{\Sigma}$ remains, the applications of H-matrices to maximum likelihood cited above have demonstrated that such approximations for covariance matrices can yield good estimates of $\widehat{\boldsymbol{\theta}}$.

Most investigations into the use of hierarchical matrices in this area focus exclusively on the computation of $l_H(\boldsymbol{\theta})$, and not its first- and second-order derivatives. If the goal is to carry out minimization of $-l_H(\boldsymbol{\theta})$, however, access to such information would significantly reduce the number of iterations required to approximate $\widehat{\boldsymbol{\theta}}$ (Nocedal & Wright 2006). Part of the difficulty is that the matrix product $\widetilde{\boldsymbol{\Sigma}}^{-1}\widetilde{\boldsymbol{\Sigma}}_j$ is still expensive to compute for hierarchical matrices, and so the trace term in the score equations remains challenging to obtain quickly. As a result, stochastic methods for estimating the trace associated with the gradient of l_H are necessary in order to maintain good complexity. An alternative method to the Hutchinson estimator is suggested by Minden et al. (2017), who utilize the peeling algorithm of Lin et al. (2011) to assemble a hierarchical approximation of $\widetilde{\boldsymbol{\Sigma}}^{-1}\widetilde{\boldsymbol{\Sigma}}_j$ and obtain a more precise estimate for its trace. In this setting especially, however, this is done at the cost of substantially higher overhead than occurs with the Hutchinson estimator. As is demonstrated in this paper, the symmetrized Hutchinson estimator is reliable enough for the purpose of numerical optimization.

One important choice for the construction of hierarchical approximations is the method for compressing low-rank off-diagonal blocks; we refer readers to Ambikasaran et al. (2016) for a discussion. In this work, we advocate the use of the Nyström approximation (Williams & Seeger 2001, Drineas & Mahoney 2005, Chen & Stein 2017). The Nyström approximation of the block of $\boldsymbol{\Sigma}$ corresponding to indices I and J is given by

$$\widetilde{\boldsymbol{\Sigma}}_{I,J} := \boldsymbol{\Sigma}_{I,P}\boldsymbol{\Sigma}_{P,P}^{-1}\boldsymbol{\Sigma}_{P,J}, \tag{4}$$

where the indices P are for p many *landmark points* that are chosen from the dataset. As can be seen, this formulation is less natural to use in an adaptive way than, for example, a truncated singular value decomposition or other common fast approximation methods. Unlike most adaptive methods (Griewank & Walther 2008), however, it is differentiable with respect to the parameters $\boldsymbol{\theta}$, a property that is essential to the approach advocated here. As a result of this unique property, the derivatives of $\widetilde{\boldsymbol{\Sigma}}(\boldsymbol{\theta})$ are both well defined and computable in quasilinear complexity if its off-diagonal blocks are constructed with the Nyström approximation, so that the second term in the hierarchically approximated analog of (2) can be computed exactly.

In this paper, we discuss an approach to minimizing (3) that combines the HODLR matrix structure from Ambikasaran & Darve (2013), the Nyström approximation of Williams & Seeger (2001), and the sample average (SAA) trace estimation from Anitescu et al. (2011) and Stein et al. (2013) in its symmetrized form. As a result, we obtain stable and optimization-suitable stochastic estimates for the gradient and Hessian of (3) in quasilinear time and storage complexity at a comparatively low overhead. Combined with the exact derivatives of $\widetilde{\boldsymbol{\Sigma}}$, the symmetrized stochastic trace estimators are demonstrated to yield gradient and Hessian approximations with relative numerical error below 0.03% away from the MLE for a manageably small number of $\boldsymbol{u}_l$ vectors. As well as providing tools for optimization, the exact derivatives and stabilized trace estimation mean that the observed and expected Fisher information matrix may be computed in quasilinear complexity, serving as a valuable tool for estimating the covariance matrix of $\widehat{\boldsymbol{\theta}}$.

1.1 Comparison with existing methods

Like the works of Börm & Garcke (2007), Anitescu et al. (2011), Stein et al. (2013), Minden et al. (2017), Litvinenko et al. (2017), and Chen & Stein (2017), we attempt to provide an approximation of the log-likelihood that can be computed with good efficiency but whose minimizers closely resemble those of the exact likelihood. The primary distinction between our approach and other methods is that we construct our approximation with an emphasis on having mathematically well-defined and computationally feasible derivatives. By computing the exact derivative of the approximation of $\boldsymbol{\Sigma}$, given by $\widetilde{\boldsymbol{\Sigma}}_j = \frac{\partial}{\partial \theta_j}\widetilde{\boldsymbol{\Sigma}}$, instead of an independent approximation of the derivative of the exact $\boldsymbol{\Sigma}$, which might be denoted by $\widetilde{\frac{\partial}{\partial \theta_j}\boldsymbol{\Sigma}}$ to emphasize that one is approximating the derivative of the exact matrix $\boldsymbol{\Sigma}(\boldsymbol{\theta})$, we obtain a more coherent framework for thinking about both optimization and error propagation in the derivatives of l_H. As an example of the practical significance of this distinction, for a scale parameter θ_0 and covariance matrix parameterized with $\widetilde{\boldsymbol{\Sigma}} = \theta_0 \widetilde{\boldsymbol{\Sigma}}'$, the derivative $\frac{\partial}{\partial \theta_0}\widetilde{\boldsymbol{\Sigma}}$ will be numerically identical to $\widetilde{\boldsymbol{\Sigma}}'$, making $\widetilde{\boldsymbol{\Sigma}}^{-1}\widetilde{\boldsymbol{\Sigma}}_j$ an exact rescaled identity matrix. As a result, the stochastic Hutchinson trace estimator of that matrix-matrix product for scale parameters is exact to numerical precision with a single $\boldsymbol{u}_l$, which will be reflected in the numerical results section. To our knowledge, such a guarantee cannot be made if $\widetilde{\boldsymbol{\Sigma}}_j$ is constructed as an approximation of the exact derivative $\boldsymbol{\Sigma}_j$. Moreover, none of the methods mentioned above discuss computing Hessian information of any kind.

Our approach achieves quasilinear complexity both in evaluating the log-likelihood and in computing accurate and stable stochastic estimators for the gradient and Hessian of the approximated log-likelihood. This comes at the cost of abandoning a priori controllable bounds on pointwise precision of the hierarchical approximation of the exact covariance matrix, a less accurate trace estimator than has been achieved with the peeling method (Minden et al. 2017, Lin et al. 2011), and sub-optimal time and storage complexity (Chen & Stein 2017). Nevertheless, we consider this to be a worthwhile tradeoff in some applications, and we demonstrate in the numerical results section that despite the loss of control of pointwise error in the covariance, we can compute estimates for MLEs and their corresponding uncertainties from the expected Fisher matrix that agree well with exact methods. Further, by having access to a gradient and Hessian, we have many options for numerical optimization and are able to perform it reliably and efficiently.

2 HODLR matrices, derivatives, and trace estimation

In this study, we approximate $\boldsymbol{\Sigma}$ with the HODLR format (Ambikasaran & Darve 2013), which has an especially simple and tractable structure given by

$$\begin{bmatrix} \boldsymbol{A}_1 & \boldsymbol{U}\boldsymbol{V}^T \\ \boldsymbol{V}\boldsymbol{U}^T & \boldsymbol{A}_2 \end{bmatrix}, \tag{5}$$

where the matrices $\boldsymbol{U}$ and $\boldsymbol{V}$ are of dimension $n \times p$ and $\boldsymbol{A}_1$ and $\boldsymbol{A}_2$ are either dense matrices or are of the form of (5) themselves. A matrix of size $n \times n$ can be split recursively into block 2×2 matrices in this way $\lfloor \log_2(n) \rfloor$ times, although in practice it is often divided fewer times than that. The diagonal blocks of a HODLR matrix are often referred to as the *leaves*, referring to the fact that a tree is implicitly being constructed.

Symmetric positive definite HODLR matrices admit an exact symmetric factorization $\widetilde{\boldsymbol{\Sigma}} = \boldsymbol{W}\boldsymbol{W}^T$ that can be computed in $O(n \log^2 n)$ complexity if p is fixed and the level grows with $O(\log n)$ (Ambikasaran et al. 2016). For a matrix of level τ, $\boldsymbol{W}$ takes the form

$$\boldsymbol{W} = \overline{\boldsymbol{W}} \prod_{k=1}^{\tau} \left\{ \mathbb{I} + \overline{\boldsymbol{U}}_k \overline{\boldsymbol{V}}_k^T \right\},$$

where $\overline{\boldsymbol{W}}$ is a block-diagonal matrix of the symmetric factors of the leaves $\boldsymbol{L}_k$ and each $\mathbb{I} + \overline{\boldsymbol{U}}\, \overline{\boldsymbol{V}}^T$ is a block-diagonal low-rank update to the identity.

If the rank of the off-diagonal blocks is fixed at p and the level grows with $O(\log n)$, then the log-determinant and linear system computations can both be performed at $O(n \log n)$ complexity by using the symmetric factor (Ambikasaran et al. 2016). With these tools, we may evaluate the approximated Gaussian log-likelihood given in (3) exactly and directly. We note that the assembly of the matrix and its factorization are parallelizable and that many of the computations in the numerical results section are done in single-node multicore parallel. With that said, however, we also point out that effective parallel implementations of hierarchical matrix operations are challenging, and that the software suite that is companion to this paper is not sufficiently advanced that it benefits from substantially more threads or nodes than a reasonably powerful workstation would provide.

2.1 The Nyström approximation and gradient estimation

As mentioned in the Introduction, the method we advocate here for the low-rank compression of off-diagonal blocks is the Nyström approximation (Williams & Seeger 2001), a method recently applied to Gaussian process computing and hierarchical matrices (Chen & Stein 2017). Unlike the multiple common algebraic approximation methods that often construct approximations of the form $\boldsymbol{U}\boldsymbol{V}^T$ by imitating early-terminating pivoted factorization (Ambikasaran et al. 2016), for example the commonly used adaptive cross-approximation (ACA) (Bebendorf 2000, Rjasanow 2002), the Nyström approximation constructs approximations that are continuous with respect to the parameters $\boldsymbol{\theta}$ in a nonadaptive way. Another advantage of this method is that an approximation $\widetilde{\boldsymbol{\Sigma}}$ assembled with the Nyström approximation is guaranteed to be positive definite if $\boldsymbol{\Sigma}$ is (Chen & Stein 2017), avoiding the common difficulty of guaranteeing that a hierarchical approximation $\widetilde{\boldsymbol{\Sigma}}$ of positive definite $\boldsymbol{\Sigma}$ is itself positive definite (Bebendorf & Hackbusch 2007, Xia & Gu 2010, Chen & Stein 2017).

The cost of choosing the Nyström approximation for off-diagonal blocks instead of a method like the ACA is that we cannot easily adapt it locally to a prescribed accuracy. That is, constructing a factorization with a precision ε so that $||\boldsymbol{\Sigma}_{I,J} - \widetilde{\boldsymbol{\Sigma}}_{I,J}|| < \varepsilon ||\boldsymbol{\Sigma}_{I,J}||$ is not generally possible since we must choose the number and locations of the landmark points P before starting computations. Put in other terms, the rank of every off-diagonal block approximation is the same, and there are no guarantees for the pointwise quality of that fixed-rank approximation.

The primary appeal of this approximation for our purposes, however, is that its derivatives exist and are readily computable by the product rule, given by

$$\widetilde{\boldsymbol{\Sigma}}_{j,(I,J)} = \boldsymbol{\Sigma}_{j,(I,P)}\boldsymbol{\Sigma}_{P,P}^{-1}\boldsymbol{\Sigma}_{P,J} - \boldsymbol{\Sigma}_{I,P}\boldsymbol{\Sigma}_{P,P}^{-1}\boldsymbol{\Sigma}_{j,(P,P)}\boldsymbol{\Sigma}_{P,P}^{-1}\boldsymbol{\Sigma}_{P,J} + \boldsymbol{\Sigma}_{I,P}\boldsymbol{\Sigma}_{P,P}^{-1}\boldsymbol{\Sigma}_{j,(P,J)}, \qquad (6)$$

where, for example, $\boldsymbol{\Sigma}_{j,(I,J)} = \frac{\partial}{\partial \boldsymbol{\theta}_j}\boldsymbol{\Sigma}_{I,J}$, with parentheses and capitalization used to emphasize subscripts denoting block indices. Fortunately, all three terms in (6) are the product of $n \times p$ and $p \times p$ matrices. Since the diagonal blocks of $\widetilde{\boldsymbol{\Sigma}}_j$ are trivially computable as well, the exact derivative of $\widetilde{\boldsymbol{\Sigma}}$ can be represented as a HODLR matrix with the same shape as $\widetilde{\boldsymbol{\Sigma}}$ except that now the off-diagonal blocks are sums of three terms that look like truncated factorizations of the form $\boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^T$, where $\boldsymbol{S} \in \mathbb{R}^{p \times p}$. In practice, assembling and storing $\widetilde{\boldsymbol{\Sigma}}_j$ take only about twice as much time and memory as required for $\widetilde{\boldsymbol{\Sigma}}$. Thus, one can compute terms of the form $\boldsymbol{y}^T\widetilde{\boldsymbol{\Sigma}}^{-1}\widetilde{\boldsymbol{\Sigma}}_j\widetilde{\boldsymbol{\Sigma}}^{-1}\boldsymbol{y}$ exactly and in quasilinear time and storage complexity, providing an exact method for obtaining the second term in the gradient of l_H.

We now combine our results from the preceding section with the symmetrized trace estimation discussed in the Introduction and obtain a stochastic gradient approximation for $-l_H$ that can be computed in quasilinear complexity:

$$\nabla \widehat{-l_H(\boldsymbol{\theta})}_j = \frac{1}{2N_h}\sum_{l=1}^{N_h} \boldsymbol{u}_l^T \boldsymbol{W}^{-1}\widetilde{\boldsymbol{\Sigma}}_j \boldsymbol{W}^{-T}\boldsymbol{u}_l - \frac{1}{2}\boldsymbol{y}^T\widetilde{\boldsymbol{\Sigma}}^{-1}\widetilde{\boldsymbol{\Sigma}}_j\widetilde{\boldsymbol{\Sigma}}^{-1}\boldsymbol{y}.$$

Both the symmetrized trace estimator, facilitated by the symmetric factorization, and the exact derivatives, facilitated by the Nyström approximation, are important to the performance of this estimator. While the benefit of the latter is clear, the decreased variance and faster computation time are nontrivial benefits as well. The numerical results section has a brief demonstration of these benefits.

2.2 Stochastic estimation of information matrices

Being able to efficiently and effectively estimate trace terms involving the derivatives $\widetilde{\boldsymbol{\Sigma}}_j$ also facilitates accurate estimation of the expected Fisher information matrix, which has terms given by

$$\mathcal{I}_{j,k} = \frac{1}{2}\mathrm{tr}\left(\widetilde{\boldsymbol{\Sigma}}^{-1}\widetilde{\boldsymbol{\Sigma}}_j\widetilde{\boldsymbol{\Sigma}}^{-1}\widetilde{\boldsymbol{\Sigma}}_k\right).$$

Using the same symmetrization approach as above, we may compute stochastic estimates of these terms with the unbiased and fully symmetrized estimator

$$\widehat{\mathcal{I}}_{j,k} = \frac{1}{4N_h}\sum_{l=1}^{N_h} \boldsymbol{u}_l^T \boldsymbol{W}^{-1}\left(\widetilde{\boldsymbol{\Sigma}}_j + \widetilde{\boldsymbol{\Sigma}}_k\right)\widetilde{\boldsymbol{\Sigma}}^{-1}\left(\widetilde{\boldsymbol{\Sigma}}_j + \widetilde{\boldsymbol{\Sigma}}_k\right)\boldsymbol{W}^{-T}\boldsymbol{u}_l - \frac{1}{2}\widehat{\mathcal{I}}_{j,j} - \frac{1}{2}\widehat{\mathcal{I}}_{k,k},\ j \neq k \qquad (7)$$

Since the diagonal elements of $\widehat{\mathcal{I}}$ can be computed first in a simple and trivially symmetric way, this provides a fully symmetrized method for estimating $\mathcal{I}_{j,k}$. Further, computing the estimates in the form of (7) does not require any more matvec applications of derivative matrices than would the more direct estimator that computes terms as $\boldsymbol{u}_l^T \boldsymbol{W}^{-1} \widetilde{\boldsymbol{\Sigma}}_j \widetilde{\boldsymbol{\Sigma}}^{-1} \widetilde{\boldsymbol{\Sigma}}_k \boldsymbol{W}^{-T} \boldsymbol{u}_l$, since the terms in (7) look like $\boldsymbol{u}^T \boldsymbol{A} \boldsymbol{A}^T \boldsymbol{u} = ||\boldsymbol{A}^T \boldsymbol{u}||^2$, if we recall that the innermost solve $\widetilde{\boldsymbol{\Sigma}}^{-1} \boldsymbol{y}$ is computed by sequential solves $\boldsymbol{W}^{-T} \boldsymbol{W}^{-1} \boldsymbol{y}$. As a result, each term in (7) still requires only one full solve with $\widetilde{\boldsymbol{\Sigma}}$ and two derivative matrix applications. In circumstances where one expects $\widetilde{\boldsymbol{\Sigma}}^{-1} \widetilde{\boldsymbol{\Sigma}}_j \widetilde{\boldsymbol{\Sigma}}^{-1} \widetilde{\boldsymbol{\Sigma}}_k$ to itself be positive definite, then the stronger theoretical results of Roosta-Khorasani & Ascher (2015) would apply, giving a high level control over the error of the stochastic trace estimator.

Having efficient and accurate estimates for $\mathcal{I}(\boldsymbol{\theta})$ is helpful because they can be used for confidence intervals, since asymptotic theory suggests (Stein 1999) that if the smallest eigenvalue of $\mathcal{I}$ tends to infinity as the sample size increases, then we can expect that

$$\mathcal{I}(\widehat{\boldsymbol{\theta}})^{1/2}(\widehat{\boldsymbol{\theta}} - \boldsymbol{\theta}_{\text{true}}) \to_D N(0, \mathbb{I}).$$

The Hessian of $-l_H$, which requires computing second derivatives of $\widetilde{\boldsymbol{\Sigma}}$, is useful for both optimization and inference. The exact terms of the Hessian $(Hl_H)_{j,k}$ are given by

$$\frac{1}{2}\left(-\text{tr}\left(\widetilde{\boldsymbol{\Sigma}}^{-1} \widetilde{\boldsymbol{\Sigma}}_k \widetilde{\boldsymbol{\Sigma}}^{-1} \widetilde{\boldsymbol{\Sigma}}_j\right) + \text{tr}\left(\widetilde{\boldsymbol{\Sigma}}^{-1} \widetilde{\boldsymbol{\Sigma}}_{jk}\right)\right) - \frac{1}{2} \boldsymbol{y}^T \left(\frac{\partial}{\partial \theta_k} \widetilde{\boldsymbol{\Sigma}}^{-1} \widetilde{\boldsymbol{\Sigma}}_j \widetilde{\boldsymbol{\Sigma}}^{-1}\right) \boldsymbol{y}, \qquad (8)$$

where

$$\frac{\partial}{\partial \theta_k} \widetilde{\boldsymbol{\Sigma}}^{-1} \widetilde{\boldsymbol{\Sigma}}_j \widetilde{\boldsymbol{\Sigma}}^{-1} = -\widetilde{\boldsymbol{\Sigma}}^{-1} \widetilde{\boldsymbol{\Sigma}}_k \widetilde{\boldsymbol{\Sigma}}^{-1} \widetilde{\boldsymbol{\Sigma}}_j \widetilde{\boldsymbol{\Sigma}}^{-1} + \widetilde{\boldsymbol{\Sigma}}^{-1} \widetilde{\boldsymbol{\Sigma}}_{jk} \widetilde{\boldsymbol{\Sigma}}^{-1} - \widetilde{\boldsymbol{\Sigma}}^{-1} \widetilde{\boldsymbol{\Sigma}}_j \widetilde{\boldsymbol{\Sigma}}^{-1} \widetilde{\boldsymbol{\Sigma}}_k \widetilde{\boldsymbol{\Sigma}}^{-1}.$$

Continuing further with the symmetrization approach, we obtain the unbiased fully symmetrized stochastic estimator of the first two terms of (8) given by

$$\frac{1}{2N_h} \sum_{l=1}^{N_h} \boldsymbol{u}_l^T \boldsymbol{W}^{-1} \widetilde{\boldsymbol{\Sigma}}_{jk} \boldsymbol{W}^{-T} \boldsymbol{u}_l - \widehat{\mathcal{I}}_{j,k},$$

where $\widehat{\mathcal{I}}_{j,k}$ is the j, kth term of the estimated Fisher information matrix.

Since the third and fourth terms in (8) can be computed exactly, replacing the two trace terms in (8) with the stochastic trace estimator provides a stochastic approximation for the Hessian of $-l_H$. The computation of the second partial derivatives $\widetilde{\boldsymbol{\Sigma}}_{jk}$ is a straightforward continuation of Equation (6) and will again result in the sum of a small number of HODLR matrices with the same structure. The overall effort of estimating the Hessian of (3) with our approach will thus also have $O(n \log^2 n)$ complexity, providing extra tools for both optimization and covariance matrix estimation, since in some circumstances the observed information is a preferable estimator to the expected information (Efron & Hinkley 1978). We note, however, that our approximation of the expected Fisher information is guaranteed to be positive semidefinite if one uses the same $\boldsymbol{u}_l$ vectors for all components of the matrix, whereas our approximation of the Hessian may not be positive semidefinite at the MLE. Exact expressions for $\widetilde{\boldsymbol{\Sigma}}_{jk}$ are given in the Appendix.

3 Numerical results

We now present several numerical experiments to demonstrate both the accuracy and scalability of the proposed approximation to the log-likelihood for Gaussian process data. In the sections below, we demonstrate the effectiveness of this method using two parameterizations of the Matérn covariance, which is given in its most standard form by

$$K(\boldsymbol{x}, \boldsymbol{y}; \boldsymbol{\theta}, \nu) := \theta_0 \mathcal{M}_\nu \left(\frac{||\boldsymbol{x} - \boldsymbol{y}||}{\theta_1} \right). \tag{9}$$

Here, $\mathcal{M}_\nu$ is the Matérn correlation function, given by

$$\mathcal{M}_\nu(x) := \big(2^{\nu-1}\Gamma(\nu)\big)^{-1} \big(\sqrt{2\nu} x\big)^\nu \mathcal{K}_\nu\big(\sqrt{2\nu} x\big),$$

and $\mathcal{K}_\nu$ is the modified Bessel function of the second kind. The quantity θ_0 is a scale parameter, θ_1 is a range parameter, and ν is a smoothness parameter that controls the degree of differentiability if the process is differentiable or, equivalently, the high-frequency behavior of the spectral density (Stein 1999).

As well as demonstrating the similar behavior of the exact likelihood to the approximation we present, this section explores the important algorithmic choices that can be tuned or selected. These are (i) the number of block-dyadic divisions of the matrix (the *level* of the HODLR matrix) and (ii) the globally fixed *rank* of the off-diagonal blocks. These choices are of particular interest in addressing possible concerns that the resulting estimate from minimizing $-l_H$ may be sensitive to the choices of these parameters and that choosing reasonable a priori values may be difficult, although we demonstrate below that the method is relatively robust to these choices.

In all the numerical simulations and studies described below, unless otherwise stated, the fixed rank of off-diagonal blocks has been set at 72, the level at $\lfloor \log_2 n \rfloor - 8$, which results in diagonal blocks (leaves) with sizes between 256 and 512, and 35 random vectors (fixed for the duration of an optimization routine) are used for stochastic trace estimation. The ordering of the observations/spatial locations is done through the traversal of a K-D tree as in Ambikasaran et al. (2016), which is a straightforward extension to the familiar one-dimensional tree whose formalism is dimension-agnostic and resembles a multidimensional analog to sorting. By ordering our points in this way, we increase the degree to which off-diagonal blocks correspond to the covariance between groups of well-separated points, which is a critical requirement and motivation for the hierarchical approximation of $\boldsymbol{\Sigma}$.

Writing $\boldsymbol{\theta}_{-0}$ for all components of $\boldsymbol{\theta}$ other than the scale parameter θ_0, the minimizer of $-l_H(\theta_0, \boldsymbol{\theta}_{-0})$ for fixed $\boldsymbol{\theta}_{-0}$ as a function of θ_0 is $\widehat{\theta}_0(\boldsymbol{\theta}_{-0}) = n^{-1} \boldsymbol{y}^T \boldsymbol{\Sigma}(1, \boldsymbol{\theta}_{-0})^{-1} \boldsymbol{y}$. Thus, the negative log-likelihood can be minimized by instead minimizing the negative *profile log-likelihood*, given by

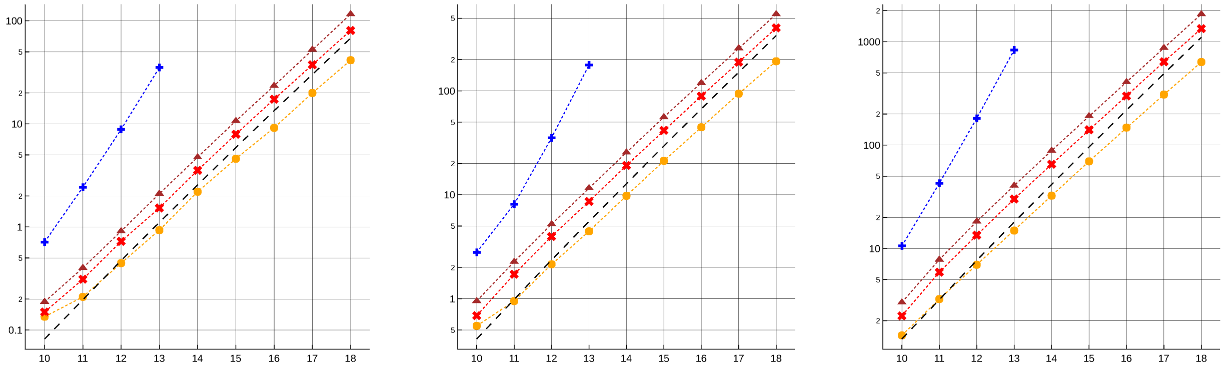
$$-l_{H,\mathrm{pr}}(\boldsymbol{\theta}_{-0}) = -l_H(\widehat{\theta}_0(\boldsymbol{\theta}_{-0}), \boldsymbol{\theta}_{-0}) = \frac{1}{2} \log |\widetilde{\boldsymbol{\Sigma}}(1, \boldsymbol{\theta}_{-0})| + \frac{n}{2} \log \left(\boldsymbol{y}^T \widetilde{\boldsymbol{\Sigma}}(1, \boldsymbol{\theta}_{-0})^{-1} \boldsymbol{y} \right),$$

which reduces the dimension of the minimization problem by one parameter. Its derivatives and trace estimators follow similarly as for the full log-likelihood.

All the computations shown in this section were performed on a standard workstation with an Intel Core i7-6700 processor with 8 threads and 40 GB of RAM, and the assembly of $\widetilde{\boldsymbol{\Sigma}}$, $\widetilde{\boldsymbol{\Sigma}}_j$, and $\widetilde{\boldsymbol{\Sigma}}_{jk}$ was done in multicore parallel with 6 threads, as was the factorization of $\widetilde{\boldsymbol{\Sigma}} = \boldsymbol{W}\boldsymbol{W}^T$. A software package written in the Julia programming language (Bezanson et al. 2017), `KernelMatrices.jl`, which provides the source code to perform all the computations described in this paper as well as reproduce the results in this section, is available at `bitbucket.org/cgeoga/kernelmatrices.jl`.

3.1 Quasilinear scaling of the log-likelihood, gradient, and Hessian

To demonstrate the scaling of the approximated log-likelihood, its stochastic gradient, and its Hessian, we show in Figure 1 the average time taken to evaluate those functions for the two-parameter (fixing ν) full Matérn log-likelihood (as opposed to profile likelihood) for sizes $n = 2^k$ for k ranging from 10 to 18, including also the time taken to call the exact functions for $k \leq 13$.



(a)

(b)

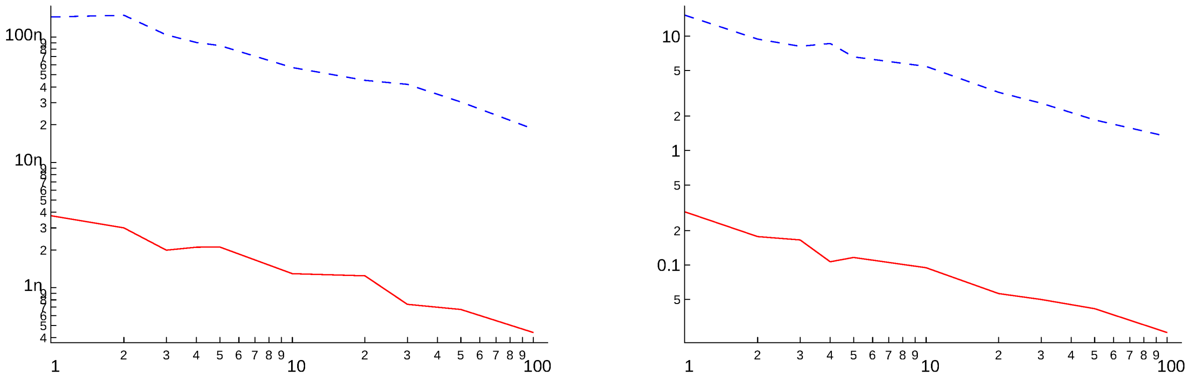
(c)

Figure 1: Times taken (in seconds) to call the likelihood (a), gradient (b), and Hessian (c) exactly (plus) and their HODLR equivalents for fixed off-diagonal rank 32 (circle), 72 (x), and 100 (triangle) for sizes 2^k with k from 10 to 18 (horizontal axis). Theoretical lines corresponding to $O(n \log^2 n)$ are added to each plot to demonstrate the scaling.

As can be seen in Figure 1, the scaling of all three operations for the approximated log-likelihood follow the expected quasilinear $O(n \log^2 n)$ growth, if not scaling even slightly better. For practical applications, the total time required to compute these values together will be slightly lower than the sum of the three times plotted here because of repeated computation of the derivative matrices $\widetilde{\boldsymbol{\Sigma}}_j$, repeated linear solves of the form $\widetilde{\boldsymbol{\Sigma}}^{-1}\boldsymbol{y}$, and other micro-optimizations that combine to save computational effort.

3.2 Numerical accuracy of symmetrized trace estimation

To demonstrate the benefit of the symmetrized stochastic trace estimation described in the Introduction, we simulate a process at random locations in the domain $[0, 100]^2$ for Matérn covariance with parameters $\theta_0 = 3$, $\theta_1 = 40$, and ν fixed at 1 and then compare the standard and symmetrized trace estimators for $\widetilde{\boldsymbol{\Sigma}}^{-1}\widetilde{\boldsymbol{\Sigma}}_j$ associated with θ_0 and θ_1. We choose a large range parameter with respect to the edge length of the domain in order to demonstrate that even for poorly conditioned $\widetilde{\boldsymbol{\Sigma}}$, the symmetrized trace estimator performs well. As the standard deviations shown in Figure 2 demonstrate, the symmetrized trace estimator reduces the standard deviation of estimates by more than a factor of 10. As remarked in the Introduction, for the scale parameter θ_0, $\widetilde{\boldsymbol{\Sigma}}^{-1}\widetilde{\boldsymbol{\Sigma}}_j$ is a multiple of the identity matrix, so there is no stochastic error even for one $\boldsymbol{u}_l$, and the errors reported here are purely numerical.



(a) (b)

Figure 2: Standard deviation of 50 stochastic trace estimates for $\widetilde{\boldsymbol{\Sigma}}^{-1}\widetilde{\boldsymbol{\Sigma}}_j$ for the scale parameter (a) and range parameter (b) of the Matérn covariance. As can be seen, nonsymmetrized estimates (dashed) have standard deviations more than one order of magnitude larger than their symmetrized counterparts (solid). In (a), we use the notation $1n = 10^{-9}$.

3.3 Effect of method parameters on likelihood surface

Using the Matérn covariance function, we simulate $n = 2^{12}$ observations of a random field with Matérn covariance function with parameters $\theta_0 = 3.0$, $\theta_1 = 5.0$, and fixed $\nu = 1$ at random locations on the box $[0, 100]^2$. In Figure 3, which shows the log-likelihood surfaces for various levels with the minimizer of each subtracted off, we see the minimal effect on the location of the minimizer caused by varying the level of the HODLR approximation with off-diagonal rank fixed at 64 for nonexact likelihoods so that the fixed rank of off-diagonal blocks does not exceed their size at level 5. Note that each graphic is on the same color scale and has the same contour levels, so that the only noticeable change between levels is an additive translation. For a similar comparison for different fixed off-diagonal ranks that yields the same interpretation, see the supplementary material.

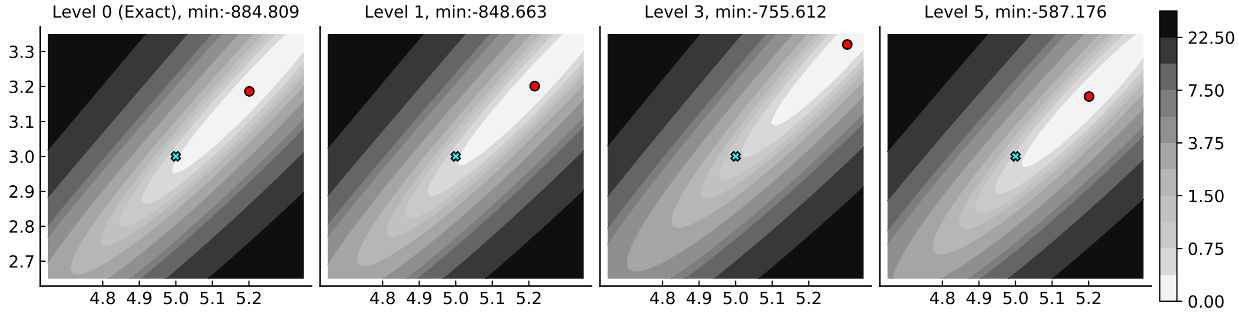


Figure 3: Centered log-likelihood surface for $n = 2^{12}$ data points randomly sampled on the box $[0, 100]^2$ with Matérn covariance with parameters $\theta_0 = 3$, $\theta_1 = 5$, and $\nu = 1$, with ν assumed known. The blue x is the "true" parameters of the simulation, and the red circle is the minimizer. The value at the minimizer is shown with the level in the title.

3.4 Numerical accuracy of stochastic derivative approximations

To explore the asymptotic behavior of the minimizers of $-l_H$, we now switch to a different parameterization of the Matérn covariance inspired by Stein (1999) and Zhang (2004), given by

$$K_s(\boldsymbol{x}, \boldsymbol{y}; \boldsymbol{\theta}, \nu) := \theta_0 \left(\frac{\theta_1}{2\sqrt{\nu}} ||\boldsymbol{x} - \boldsymbol{y}|| \right)^{\nu} \mathcal{K}_\nu \left(\frac{2\sqrt{\nu}}{\theta_1} ||\boldsymbol{x} - \boldsymbol{y}|| \right). \tag{10}$$

The advantage of this parameterization is that, unlike in (9), it clearly separates parameters that can be estimated consistently as the sample size grows on a fixed domain from those that cannot (Stein 1999, Zhang 2004, Zhang & Zimmerman 2005). Specifically, for bounded domains in 3 or fewer dimensions, results on equivalence and orthogonality of Gaussian measures suggest that both θ_0 and ν can be estimated consistently under the parameterization (10), whereas θ_0 definitely cannot be estimated consistently under (9). The range parameter θ_1 cannot be estimated consistently under either parameterization.

To demonstrate the accuracy of the stochastic gradient and Hessian estimates of $-l_H$ with this covariance function, we compare them with the exact gradient and Hessian of the approximated log-likelihood computed directly for a variety of small sizes. For the specific setting of the computations, we simulate Gaussian process data at random locations on the domain $[0, 100]^2$ with parameters $\boldsymbol{\theta} = (3, 5)$ and ν fixed at 1 to avoid the potentially confounding numerical difficulties of computing $\frac{\partial}{\partial \nu} \mathcal{K}_\nu(x)$.

To explore the numerical accuracy of the stochastic approximations, we consider both estimates at the computed MLE $\widehat{\boldsymbol{\theta}}$ and at a potential starting point for optimization, which was chosen to be $\boldsymbol{\theta}_{\text{init}} = (2, 2)$. For both cases, we consider the standard relative precision metric. We note, however, that if $-l_H$ were exactly minimized, its gradient would be exactly 0 and the relative precision would be undefined. Thus, we also consider the alternative measures of accuracy at the evaluated MLE given by

$$\eta_g := ||\widehat{\nabla l_H(\boldsymbol{\theta})} - \nabla l_H(\boldsymbol{\theta})||_{\mathcal{I}(\boldsymbol{\theta})^{-1}} \tag{11}$$

for the gradient and

$$\eta_{\mathcal{I}} := \operatorname{tr}\left\{ \left(\widehat{\mathcal{I}}(\boldsymbol{\theta}) - \mathcal{I}(\boldsymbol{\theta})\right)\left(\mathcal{I}(\boldsymbol{\theta})^{-1} - \widehat{\mathcal{I}}(\boldsymbol{\theta})^{-1}\right) \right\}^{1/2} \tag{12}$$

for the expected Fisher matrix. These measures of precision are stable at the MLE and are invariant to all linear transformations; $\eta_{\mathcal{I}}$ is a natural metric for positive definite matrices in that, for fixed positive definite $\mathcal{I}$, it tends to infinity as a sequence of positive definite $\widehat{\mathcal{I}}$ tends to a limit that is only positive semidefinite. Using ε to denote the standard relative precision, Tables 1 and 2 summarize these results.

Table 1: Average relative precision (on $\log_{10}$ scale) of the stochastic gradient and Hessian of the approximated log-likelihood for 5 simulations with $\boldsymbol{\theta} = (2, 2)$.

	$n = 2^{10}$	$n = 2^{11}$	$n = 2^{12}$	$n = 2^{13}$
$\varepsilon_{\widehat{\nabla l_H(\boldsymbol{\theta})}}$	-3.78	-3.46	-3.51	-3.60
$\varepsilon_{\widehat{Hl_H(\boldsymbol{\theta})}}$	-4.22	-3.89	-3.71	-3.79

Table 2: Average precisions (on $\log_{10}$ scale) of the stochastic gradient, expected Fisher matrix, and Hessian at $\widehat{\boldsymbol{\theta}}$. Here η_g and $\eta_{\mathcal{I}}$ are given by (11) and (12) respectively.

	$n = 2^{10}$	$n = 2^{11}$	$n = 2^{12}$	$n = 2^{13}$
$\varepsilon_{\widehat{\nabla l_H(\boldsymbol{\theta})}}$	-0.62	-1.42	-0.98	-1.75
$\varepsilon_{\widehat{Hl_H(\boldsymbol{\theta})}}$	-2.37	-1.86	-2.13	-2.04
η_g	-0.92	-0.93	-0.73	-0.96
$\eta_{\mathcal{I}}$	-1.77	-1.82	-1.86	-2.21

For the relative precision, the interpretation is clear: the estimated gradient and Hessian at $\boldsymbol{\theta} = (2, 2)$ have relative error less than 0.03%, which we believe demonstrates their suitability for numerical optimization. Near the MLE, the gradient estimate in particular can become less accurate, meaning that stopping conditions in minimization like $||\widehat{\nabla l_H(\boldsymbol{\theta})}|| < \varepsilon_{tol}$ may not be suitable, for example, and that if the gradient is sufficiently small, even the signs of the terms in the estimate may be incorrect, which can be confounding for numerical optimization. Nonetheless, in later sections we demonstrate the ability to optimize to the relative tolerance of 10^{-8} in the objective function.

3.5 Simulated data verifications

Using the alternative parameterization of the Matérn covariance function with fixed $\nu = 1$, which corresponds to a process that is just barely not mean-square differentiable, we simulate five datasets of size $n = 2^{18}$ on an even grid across the domain $[0, 100]^2$ using the

R software *RandomFields* of Schlather et al. (2015); and we then fit successively larger randomly subsampled datasets (obtaining both point estimates and approximate 95% confidence intervals via the expected Fisher information matrix) from each of these of sizes 2^k for k ranging from 11 to 17, thus working exclusively with irregularly sampled data. We do this for two range parameters of $\theta_1 = 5$ and $\theta_1 = 50$ to further demonstrate the method's flexibility, optimizing in the weak correlation case with a simple second-order trust-region method that exactly solves the subproblem as described in Nocedal & Wright (2006) and with the first-order *method of moving asymptotes* provided by the NLopt library (Johnson) in the strongly correlated case, as the second-order methods involving the Hessian did not accelerate convergence to the MLE for the strongly correlated data, which has generally been our experience. In both circumstances, the stopping condition is chosen to be a relative tolerance of 10^{-8}. For $k \leq 13$, we provide parameters fitted with the exact likelihood to the same tolerance for comparison. Figures 4 and 5 summarize the results. In Appendix B, we also provide confidence ellipses from the exact and stochastic expected Fisher matrices, providing another tool for comparing the inferential conclusions one might reach from the exact and approximated methods.

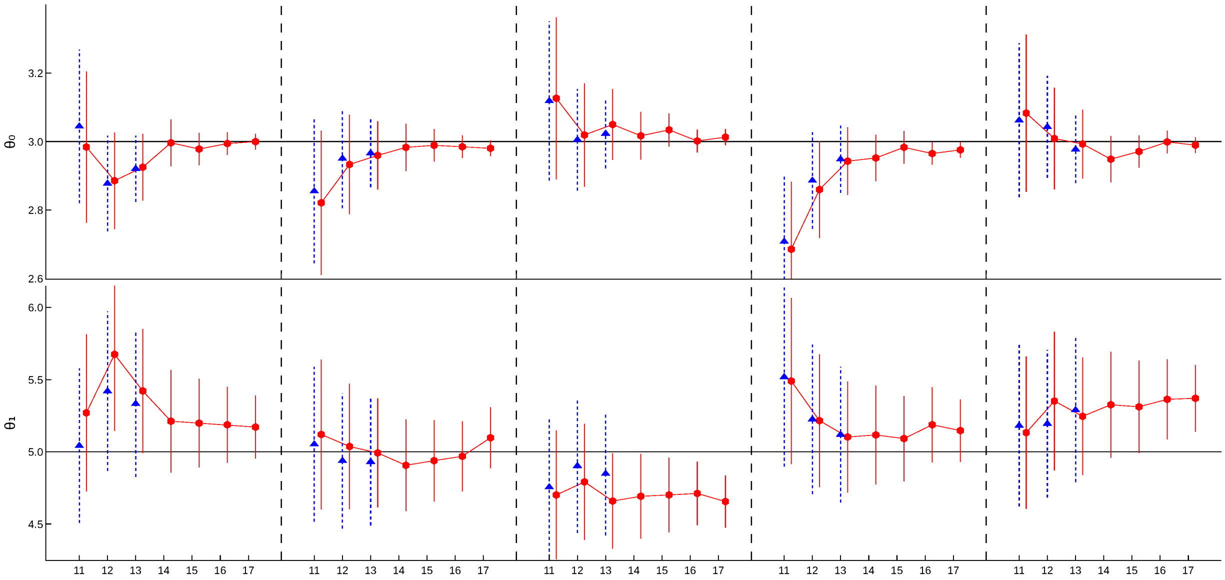


Figure 4: Estimated MLEs and confidence intervals for random subsamplings of 5 exactly simulated datasets of size $n = 2^{18}$ with covariance given by (10) and parameters $\boldsymbol{\theta} = (3, 5)$ and $\nu = 1$, with subsampled sizes 2^k with k ranging from 11 to 17 (horizontal axis). Exact estimates provided for the first three sizes with triangles.

The main conclusion from these simulations is that the minimizers of $-l_H$ closely resemble those of the exact log-likelihood and that the widths of the confidence intervals based on the approximate method are fairly close to those for the exact method. Assuming that this degree of accuracy applies to the larger sample sizes for which we were unable to do exact calculations, we see some interesting features in the behavior of the estimates as we

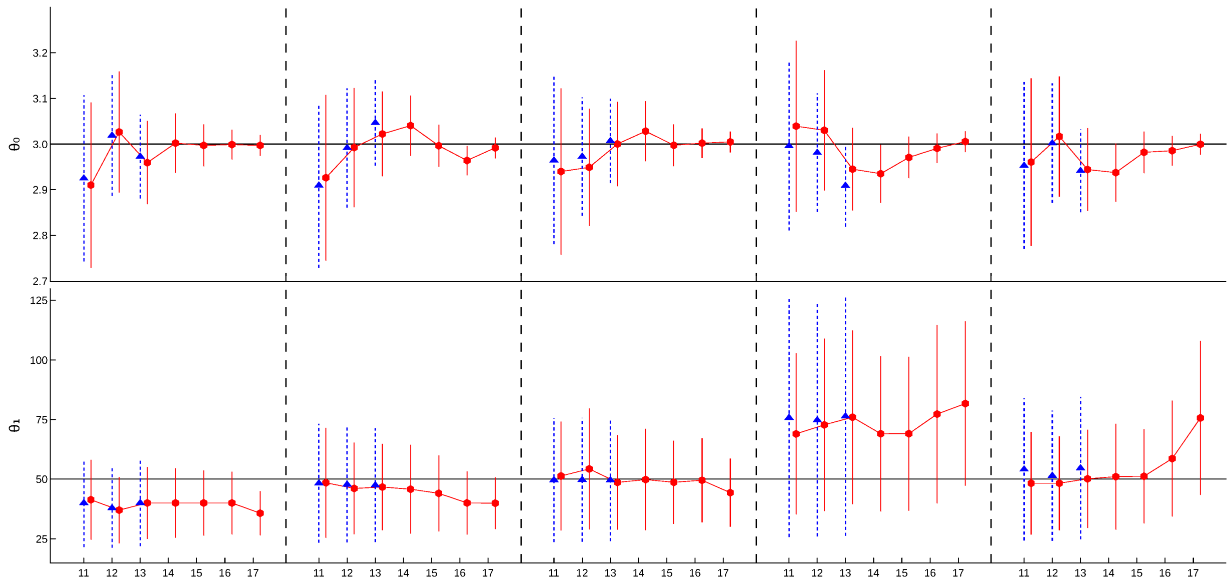


Figure 5: Same as Figure 4 except now with a large range parameter of $\theta_1 = 50$.

take larger subsamples of the initial simulations of 2^{18} gridded observations. Specifically, we see that the estimates of the scale θ_0 continue to improve as the sample size increases, whereas, especially when the true range is 50, the estimated ranges do not obviously improve at the larger sample sizes, as is correctly represented in the almost constant widths of the confidence intervals for these larger sample sizes. These results are in accord with what is known about fixed-domain asymptotics for the Matérn model (Stein 1999, Zhang 2004, Zhang & Zimmerman 2005). We note that the asymptotic results underlying the use of Fisher information for approximate confidence intervals for MLEs, which includes consistency of the point estimates, are not valid in this setting. Nevertheless, the Fisher information matrix still gives a meaningful measure of uncertainty in the parameter estimates since it corresponds to the Godambe information matrix of the score equations (Heyde 2008). Additionally, these results demonstrate that the stochastic gradient and Hessian estimates are sufficiently stable even near the MLE and that numerical optimization to a relative tolerance of 10^{-8} can be performed. It is worth noting, though, that stochastic optimization to this level of precision was more expensive in terms of the number of function calls required to reach convergence (which we found to be about twice as many), although we see no evidence to indicate that that difference will grow with n. For detailed information about the optimization performed to generate Figures 4 and 5, see the Supplementary Material.

4 Discussion

In this paper, we present an approximation of the Gaussian log-likelihood that can be computed in quasilinear time and admits both a gradient and Hessian with the same complexity. In the Numerical results section, we demonstrate that with the exact derivatives of our approximated covariance matrices and the symmetrized trace estimation we can obtain very stable and performant estimators for the gradient and Hessian of the approximated log-likelihood and the expected information matrix. Further, we demonstrate that the minimizers of our approximated likelihood are nearly identical to the minimizers of the exact likelihood for a wide variety of method parameters and that the expected information matrix and the Hessian of our approximated likelihood are relatively close to their exact values. Putting these together, we present a coherent model for computing approximations of MLEs for Gaussian processes in quasilinear time that is kernel-independent and avoids many problem-specific computing challenges (such as choosing preconditioners). Further, the approach we advocate here (and the corresponding software) is flexible, making it an attractive and fast way to do exploratory work.

In some circumstances, however, our method is potentially less useful, as utilizing the differentiability of the approximation of $\widetilde{\boldsymbol{\Sigma}}$ requires first partial derivatives of the kernel function with respect to the parameters. For simpler models as have been discussed here, this requirement is not problematic. For some covariance functions, however, these derivatives may need to be computed with the aid of automatic differentiation (AD) or even finite differencing (FD). This issue comes up even with the Matérn kernel, since, to the best of our knowledge, no usable code exists for efficiently computing the derivatives $\frac{\partial^k}{\partial \nu^k}\mathcal{K}_\nu(x)$ analytically, even though series expansions for these derivatives are available (see 10.38.2 and 10.40.8 in Olver et al. (2010)). Empirically, we have obtained reasonable estimates for $\widehat{\nu}$ using finite difference approximations for these derivatives when ν is small. The main difficulty with finite differencing, however, is that it introduces a source of error that is hard to monitor, so that it can be difficult to recognize when estimates are being materially affected by the quality of the finite difference approximations. While the algorithm will still scale with the same complexity if AD or FD is used as a substitute for exactly computed derivatives, doing so will likely introduce a serious fixed overhead cost as well as potential numerical error in the latter case. With that said, however, the overhead will be incurred during the assembly stage, which is particularly well-suited to extreme parallelization that may mitigate such performance concerns.

Another circumstance in which this method may not be the most suitable is one where very accurate trace estimation is required, such as optimizing to a very high precision. As has been discussed, the peeling method of Lin et al. (2011) may be used, but matvec actions with the derivative matrices, especially $\widetilde{\boldsymbol{\Sigma}}_{jk}$, have a very high overhead, which may make the peeling method unacceptably expensive. Parallelization would certainly also mitigate this cost, but the fact remains that performing $O(\log n)$ many matvecs with $\widetilde{\boldsymbol{\Sigma}}_{jk}$ will come at a significant price.

In some circumstances, the framework of hierarchical matrices may not be the most appropriate scientific choice. It has been shown that, at least in some cases, as the dimension of a problem increases or its geometry changes, the numerical rank of the low-rank blocks of kernel matrices will increase (Ambikasaran et al. 2016), which will affect the scaling of the algorithms that attempt to control pointwise precision. For algorithms that do not attempt to control pointwise precision, such as the one presented here, the complexity will not change, but the quality of the approximation will deteriorate. Moreover, off-diagonal blocks of kernel matrices often have low numerical rank because the corresponding kernel is smooth away from the origin (Ambikasaran et al. 2016). For covariance kernels for which this does not hold, off-diagonal blocks may not be of low numerical rank regardless of the dimension or geometry of the problem. For most standard covariance functions in spatial and space-time statistics, there is analyticity away from the origin. And most space-time processes happen in a dimension of at most four, so the problems of dimensionality may not often be encountered. But for some applications, for example in machine learning, that are done in higher dimensions, we suggest using care to be sure that the theoretical motivations for this approximation hold to a reasonable degree.

Finally, we note that many of the choices made in this method are subjective and can potentially be improved. We chose the HODLR format to approximate $\boldsymbol{\Sigma}$ due to its simplicity and transparency with regard to complexity, but there are many other options for matrix compression; see Minden et al. (2017), Chen & Stein (2017), and Wang et al. (2018) for three recent and very different examples of matrix compression. Further, the Nyström approximation for off-diagonal blocks was chosen so that the covariance matrix would be differentiable, but there are many other methods for low-rank approximation, and many of them have better theoretical properties; see Bebendorf (2000), Liberty et al. (2007) and Wang et al. (2016) for diverse examples of adaptive low-rank approximations. These particular methods are not generally differentiable with respect to kernel parameters (Griewank & Walther 2008), but if one were to prioritize pointwise approximation quality over differentiability of the approximated covariance matrix, adaptive low-rank approximation methods like those mentioned above may be reasonable choices.

Acknowledgments

This material was based upon work supported by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research (ASCR) under Contracts DE-AC02-06CH11347 and DE-AC02-06CH11357. We acknowledge partial NSF funding through awards FP061151-01-PR and CNS-1545046 to MA.

References

Ambikasaran, S. & Darve, E. (2013), ‘An O(nlogn) fast direct solver for partially hierarchically semi-separable matrices’, *Journal of Scientific Computing* **57**(3), 477–501.

Ambikasaran, S., O’Neil, M. & Singh, K. (2016), ‘Fast symmetric factorization of hierarchical matrices with applications’.
URL: *https://arxiv.org/abs/1405.0223v2*

Ambikasaran , S., Foreman-Mackey, D., Greengard, L., Hogg, D. & ONeil, M. (2016), ‘Fast direct methods for Gaussian processes’, *IEEE transactions on pattern analysis and machine intelligence* **38**(2), 252–265.

Anitescu, M., Chen, J. & Wang, L. (2011), ‘A matrix-free approach for solving the parametric Gaussian process maximum likelihood problem’, *SIAM J. Sci. Comput.* **34**(1), 240–262.

Bebendorf, M. (2000), ‘Approximation of boundary element matrices’, *Numerische Mathematik* **86**(4), 565–589.

Bebendorf, M. & Hackbusch, W. (2007), ‘Stabilized rounded addition of hierarchical matrices’, *Numerical Linear Algebra with Applications* **14**(5), 407–423.

Bezanson, J., Edelman, A., Karpinski, S. & Shah, V. (2017), ‘Julia: A fresh approach to numerical computing’, *SIAM Review* **59**(1), 65–98.

Börm, S. & Garcke, J. (2007), Approximating Gaussian processes with H2-matrices, *in* ‘European Conference on Machine Learning’, pp. 42–53.

Caragea, P. & Smith, R. (2007), ‘Asymptotic properties of computationally efficient alternative estimators for a class of multivariate normal models’, *Journal of Multivariate Analysis* **98**(7), 1417–1440.

Castrillon-Candas, J., Genton, M. & Yokota, R. (2016), ‘Multi-level restricted maximum likelihood covariance estimation and kriging for large non-gridded spatial datasets’, *Spatial Statistics* **18**, 105–124.

Chen, J. & Stein, M. (2017), ‘Linear-cost covariance functions for Gaussian random fields’.
URL: *https://arxiv.org/abs/1711.05895*

Cressie, N. & Johannesson, G. (2006), ‘Spatial prediction for massive datasets’, *Proceedings of the Australian Academy of Science, Elizabeth and Frederick White Coference* pp. 1–11.

Drineas, P. & Mahoney, M. (2005), ‘On the nyström method for approximating a gram matrix for improved kernel-based learning’, *J. Mach. Learn. Res.* **6**, 2153–2175.

Efron, B. & Hinkley, D. (1978), ‘Assessing the accuracy of the maximum likelihood estimator: Observed versus expected Fisher information’, *Biometrika* **65**.

Furrer, R., Genton, M. & Nychka, D. (2006), ‘Covariance tapering for interpolation of large spatial datasets’, *Journal of Computational and Graphical Statistics* **15**(3), 502–523.

Godambe, V. (1991), *Estimating functions*, Oxford University Press.

Grasedyck, L. & Hackbusch, W. (2003), ‘Construction and arithmetics of H-matrices’, *Computing* **70**(4), 295–334.

Greengard, L. & Rokhlin, V. (1987), ‘A fast algorithm for particle simulations’, *J. Comput. Phys.* **73**(2), 325–348.

Griewank, A. & Walther, A. (2008), *Evaluating derivatives: principles and techniques of algorithmic differentiation*, Vol. 105, SIAM.

Hackbusch, W. (1999), ‘A sparse matrix arithmetic based on H-matrices, Part I: Introduction to H-matrices’, *Computing* **62**(2), 89–108.

Hackbusch, W. (2015), *Hierarchical Matrices: Algorithms and Analysis*, Springer.

Heyde, C. (2008), *Quasi-likelihood and its application: a general approach to optimal parameter estimation*, Springer Science & Business Media.

Hutchinson, M. (1990), ‘A stochastic estimator of the trace of the influence matrix for laplacian smoothing splines’, *Communications in Statistics – Simulation and Computation* **19**(2), 433–450.

Johnson, S. (n.d.), ‘The NLopt nonlinear-optimization package’.
URL: *http://ab-initio.mit.edu/nlopt*

Katzfuss, M. (2017), ‘A multi-resolution approximation for massive spatial datasets’, *Journal of the American Statistical Association* **112**(517), 201–214.

Katzfuss, M. & Guinness, J. (2018), ‘A general framework for Vecchia approximations of Gaussian processes’.
URL: *https://arxiv.org/abs/1708.06302*

Kaufman, C., Schervish., M. & Nychka, D. (2008), ‘Covariance tapering for likelihood-based estimation in large spatial data sets’, *Journal of the American Statistical Association* **103**(484), 1545–1555.

Liberty, E., Woolfe, F., Martinsson, P.-G., Rokhlin, V. & Tygert, M. (2007), ‘Randomized algorithms for the low-rank approximation of matrices’, *Proceedings of the National Academy of Sciences* **104**(51), 2016720172.

Lin, L., Lu, J. & Ying, L. (2011), ‘Fast construction of hierarchical matrix representation from matrix-vector multiplication’, *J. Comput. Phys.* **230**, 4071–4087.

Lindgren, F., Rue, H. & Lindstrom, J. (2011), ‘An explicit link between Gaussian fields and Gaussian markov random fields: the stochastic partial differential equation approach’, *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **73**(4), 423–498.

Litvinenko, A., Sun, Y., Genton, M. & Keyes, D. (2017), ‘Likelihood approximation with hierarchical matrices for large spatial datasets’.
URL: *https://arxiv.org/abs/1709.04419*

Minden, V., Damle, A., Ho, K. & Ying, L. (2017), 'Fast spatial Gaussian process maximum likelihood estimation via skeletonization factorizations', *Multiscale Modeling & Simulation* **15**(4), 1584–1611.

Nocedal, J. & Wright, S. (2006), *Numerical Optimization*, Springer Series in Operations Research and Financial Engineering, Springer New York.

Olver, F., Lozier, D., Boisvert, R. & Clark, C. (2010), *NIST Handbook of Mathematical Functions*, Cambridge University Press.

Rjasanow, S. (2002), Adaptive cross approximation of dense matrices, *in* 'Proc Int Assoc Bound Elem Methods'.

Roosta-Khorasani, F. & Ascher, U. (2015), 'Improved bounds on sample size for implicit matrix trace estimators', *Foundations of Computational Mathematics* **15**(5), 1187–1212.

Rue, H. & Held, L. (2005), *Gaussian Markov random fields: theory and applications*, CRC press.

Schlather, M., Malinowski, A., Menck, P., Oesting, M. & Strokorb, K. (2015), 'Analysis, simulation and prediction of multivariate random fields with package RandomFields', *Journal of Statistical Software* **63**(8), 1–25.

Stein, M. (1999), *Interpolation of Spatial Data: Some Theory for Kriging*, Springer.

Stein, M. (2014), 'Limitations on low rank approximations for covariance matrices of spatial data', *Spatial Statistics* **8**.

Stein, M., Chen, J. & Anitescu, M. (2013), 'Stochastic approximation of score functions for Gaussian processes', *Ann. Appl. Stat.* **7**(2), 1162–1191.

Stein, M., Chi, Z. & Welty, L. (2004), 'Approximating likelihoods for large spatial data sets', *Journal of the Royal Statistical Society Series B* **66**, 275–296.

Sun, Y. & Stein, M. (2016), 'Statistically and computationally efficient estimating equations for large spatial datasets', *Journal of Computational and Graphical Statistics* **25**(1), 187–208.

Vecchia, A. (1988), 'Estimation and model identification for continuous spatial processes', *Journal of the Royal Statistical Society. Series B (Methodological)* **50**(2), 297–312.

Wang, R., Li, Y., Mahoney, M. & Darve, E. (2018), 'Block basis factorization for scalable kernel evaluation'.
URL: *https://arxiv.org/abs/1505.00398*

Wang, S., Luo, L. & Zhang, Z. (2016), 'SPSD matrix approximation vis column selection: Theories, algorithms, and extensions', *Journal of Machine Learning Research* **17**, 49:1–49:49.

Williams, C. & Seeger, M. (2001), Using the Nystrm method to speed up kernel machines, *in* 'Advances in Neural Information Processing Systems 13', MIT Press, pp. 682–688.

Xia, J. & Gu, M. (2010), 'Robust approximate Cholesky factorization of rank-structured symmetric positive definite matrices', *SIAM J. Matrix Anal. Appl.* **31**(5), 2899–2920.

Zhang, H. (2004), 'Inconsistent estimation and asymptotically equal interpolations in model-based geostatistics', *Journal of the American Statistical Association* **99**(465), 250–261.

Zhang, H. & Zimmerman, D. (2005), 'Towards reconciling two asymptotic frameworks in spatial statistics', *Biometrika* **92**(4), 921–936.

A Exact expressions for $\widetilde{\boldsymbol{\Sigma}}_{jk}$

As described in Section 2, derivatives of off-diagonal block approximations in the form of (4) are given by (6). The Hessian of the approximated log-likelihood requires the second derivative of $\widetilde{\boldsymbol{\Sigma}}$, which in turn requires the partial derivatives of (6). Continuing with the same notation as in Section 2, three simple product rule computations show that the kth partial derivative of (6) is given by

$$\begin{aligned}
\widetilde{\boldsymbol{\Sigma}}_{j,k,(I,J)} = {} & \boldsymbol{\Sigma}_{j,k,(I,P)} \boldsymbol{\Sigma}_{P,P}^{-1} \boldsymbol{\Sigma}_{P,J} \\
& - \boldsymbol{\Sigma}_{j,(I,P)} \boldsymbol{\Sigma}_{P,P}^{-1} \boldsymbol{\Sigma}_{k,(P,P)} \boldsymbol{\Sigma}_{P,P}^{-1} \boldsymbol{\Sigma}_{P,J} \\
& + \boldsymbol{\Sigma}_{j,(I,P)} \boldsymbol{\Sigma}_{P,P}^{-1} \boldsymbol{\Sigma}_{k,(P,J)} \\
& + \boldsymbol{\Sigma}_{k,(I,P)} \boldsymbol{\Sigma}_{(P,P)}^{-1} \boldsymbol{\Sigma}_{j,(P,P)} \boldsymbol{\Sigma}_{P,P}^{-1} \boldsymbol{\Sigma}_{P,J} \\
& - \boldsymbol{\Sigma}_{I,P} \boldsymbol{\Sigma}_{(P,P)}^{-1} \boldsymbol{\Sigma}_{k,(P,P)} \boldsymbol{\Sigma}_{(P,P)}^{-1} \boldsymbol{\Sigma}_{j,(P,P)} \boldsymbol{\Sigma}_{P,P}^{-1} \boldsymbol{\Sigma}_{P,J} \\
& + \boldsymbol{\Sigma}_{(I,P)} \boldsymbol{\Sigma}_{(P,P)}^{-1} \boldsymbol{\Sigma}_{j,k,(P,P)} \boldsymbol{\Sigma}_{P,P}^{-1} \boldsymbol{\Sigma}_{P,J} \\
& - \boldsymbol{\Sigma}_{I,P} \boldsymbol{\Sigma}_{(P,P)}^{-1} \boldsymbol{\Sigma}_{j,(P,P)} \boldsymbol{\Sigma}_{(P,P)}^{-1} \boldsymbol{\Sigma}_{k,(P,P)} \boldsymbol{\Sigma}_{P,P}^{-1} \boldsymbol{\Sigma}_{P,J} \\
& + \boldsymbol{\Sigma}_{(I,P)} \boldsymbol{\Sigma}_{(P,P)}^{-1} \boldsymbol{\Sigma}_{j,(P,P)} \boldsymbol{\Sigma}_{P,P}^{-1} \boldsymbol{\Sigma}_{k,(P,J)} \\
& + \boldsymbol{\Sigma}_{k,(I,P)} \boldsymbol{\Sigma}_{P,P}^{-1} \boldsymbol{\Sigma}_{j,(P,J)} \\
& - \boldsymbol{\Sigma}_{(I,P)} \boldsymbol{\Sigma}_{k,(P,P)}^{-1} \boldsymbol{\Sigma}_{k,(P,P)} \boldsymbol{\Sigma}_{P,P}^{-1} \boldsymbol{\Sigma}_{j,(P,J)} \\
& + \boldsymbol{\Sigma}_{(I,P)} \boldsymbol{\Sigma}_{P,P}^{-1} \boldsymbol{\Sigma}_{j,k,(P,J)}.
\end{aligned}$$

Although this expression looks unwieldy and expensive, each line is still expressible as the sum of rank p matrices that can be written as $\boldsymbol{USV}^T$, where $\boldsymbol{S} \in \mathbb{R}^{p \times p}$, meaning that a matvec operation with the block shown above will still scale with linear complexity for fixed p. While the overhead involved is undeniably substantial, the assembly and application of $\widetilde{\boldsymbol{\Sigma}}_{jk}$ is nonetheless demonstrated to scale with quasilinear complexity.